Follow-the-Sun Software Development: Controlled Experiment

Saroj Pamulapati
Divya Gaddipati

School of Computing
Blekinge Institute of Technology
SE-371 79 Karlskrona
Sweden
Follow-the-Sun Software Development: Controlled Experiment

Saroj Pamulapati
Divya Gaddipati

School of Computing
Blekinge Institute of Technology
SE-371 79 Karlskrona
Sweden
This thesis is submitted to the School of Computing at Blekinge Institute of Technology in partial fulfillment of the requirements for the degree of Master of Science in Software Engineering. The thesis is equivalent to 2 X 20 weeks of full time studies.

Contact Information:
Author(s):
Saroj Pamulapati
Address: Karlskrona, Sweden
E-mail: sarojpsmail@gmail.com

Divya Gaddipati
Address: Karlskrona, Sweden
E-mail: divya.gaddipati@gmail.com

University Advisor & Guide:
Ms. Darja Šmite
Assistant Professor

School of Computing
Blekinge Institute of Technology
SE-371 79 Karlskrona
Sweden

Internet : www.bth.se/com
Phone : +46 455 38 50 00
Fax : +46 455 38 50 57
ABSTRACT

Context: Global software development (GSD) can be defined as distributing the software development work among various teams, which are geographically distributed. Global software development is being widely used nowadays in software industries because of the numerous advantages offered. Follow-the-sun (FTS) approach is a sub-division of global software engineering where unfinished work is handed off every day from one development site to another development site, which are many time zones apart and hence development takes place round the clock instead of just eight hours per day.

Objectives: The objective of this study is to explore whether by implementing follow-the-sun approach in the development phase of the software life cycle, the time-to-market can be reduced by 50% when compared to the traditional method of software development.

Methods: In this research to fulfil the objectives, systematic literature review and an experiment were conducted. In literature review a number of papers that are related to this study were identified from the databases such as Engineering Village, IEEE Xplore, Springer Link, ISI Web of Knowledge, Science Direct, ACM Digital Library and Wiley Inter Science Journal Finder. From these databases, all the related work on follow-the-sun approach that has been done so far was collected and the selection was done through tollgate approach. An experiment was conducted at Blekinge Tekniska Högskola with the students of the computer science engineering field as subjects of the experiment. The three software development scenarios that are tested in the experiment are co-located scenario, follow-the-sun with overlap scenario and follow-the-sun without overlap scenario. The experiment compares the results of the co-located scenario with the results of the follow-the-sun with overlap scenario and follow-the-sun without overlap scenario.

Results: The results from the experiment showed that there is reduction in time-to-market by 22% when using follow-the-sun with overlap scenario and there is a reduction of 10% in the time-to-market when using follow-the-sun without scenario when compared to the co-located scenario.

Conclusions: We conclude that follow-the-sun approach has an advantage of reduction in time-to-market when compared to the traditional co-located approach. But more research needs to be done in finding out the challenges and their mitigation strategies that will be beneficial for the organizations in order to adopt this process. From the findings of the literature review we have observed that due to the communication and collaboration problems, organizations are finding it difficult to achieve the desired benefits from the follow-the-sun approach. As a part of this study we conducted an experiment and we report the various challenges that have been noticed during the execution. From the experiment results we observe that there is a difference in the results between the theory and the practical implementation. We conclude that though the time-to-market can be certainly reduced when using follow-the-sun approach, but the reduction in the development time by 50% is questionable. This may be achieved by emphasizing on the challenges and by mitigating the same thereby bringing it closer to 50%.

Keywords: Follow-the-sun, round-the-clock development, 24-hour development.
ACKNOWLEDGEMENT

Firstly, we would like to thank our advisor and guide Ms. Darja Šmite who encouraged and motivated us during the entire period of this research. Her advice and suggestion helped us in conducting the experiment and her guidance gave us the direction for the research.

Secondly, we would like to thank Professor Jürgen Börstler who provided us with the students and helped us in designing the task for the experiment. He was kind, supportive and generous to have spent his valuable time in discussing and re-scheduling the lab and class timings to fit our experiment design.

Thirdly, we are very thankful to all the students who participated in the experiment and also to our friends Veena Veerla, Maanasa Subrahmanyam, Pandiri Srinivas Reddy, Sreeramadas Veera Brahmam, M V Sarath Chandra, Ranjith Enugu, Vinood Kumar, K Sushma Joseph who helped us during the experiment execution acting as leads.

Finally we would like to thank our family members and friends for their help and support to complete the thesis successfully.
# TABLE OF CONTENTS

1 INTRODUCTION ........................................................................................................................................... 1

1.1 BACKGROUND ........................................................................................................................................... 2
1.1.1 Global Software Development ................................................................................................................ 2
1.1.2 Follow-the-sun (FTS) ............................................................................................................................... 5
1.2 RELATED WORK .......................................................................................................................................... 10
1.3 RESEARCH MOTIVATION ............................................................................................................................. 12
1.4 AIMS AND OBJECTIVES ............................................................................................................................. 13
1.5 RESEARCH QUESTIONS ............................................................................................................................... 14
1.6 RESEARCH METHODOLOGY ....................................................................................................................... 15
1.6.1 Systematic Literature Review .................................................................................................................. 15
1.6.2 Experiment .............................................................................................................................................. 16
1.7 RESEARCH OUTCOMES .............................................................................................................................. 16
1.8 THESIS STRUCTURE ................................................................................................................................. 17

2 RESEARCH METHODOLOGY ..................................................................................................................... 18

2.1 SYSTEMATIC LITERATURE REVIEW ....................................................................................................... 18
2.1.1 Planning the review ................................................................................................................................. 18
2.1.2 Conducting the Review ........................................................................................................................... 25
2.1.3 Reporting the review ............................................................................................................................... 27
2.2 EXPERIMENT ........................................................................................................................................... 30
2.2.1 Motivation .............................................................................................................................................. 30
2.2.2 Initial Experiment Definition .................................................................................................................. 31
2.2.3 Initial Experiment Planning ..................................................................................................................... 31
2.2.4 Refinements to the Initial Experiment Planning ....................................................................................... 34
2.2.5 Final Experiment Definition ................................................................................................................... 36
2.2.6 Final Experiment Planning ...................................................................................................................... 37
2.2.7 Experiment Design ................................................................................................................................ 39
2.2.8 Experiment Execution ........................................................................................................................... 42
2.2.9 Summary .............................................................................................................................................. 46

3 RESULTS AND ANALYSIS ............................................................................................................................. 48

3.1 SYSTEMATIC LITERATURE REVIEW RESULTS .................................................................................... 48
3.2 EXPERIMENT RESULTS ............................................................................................................................... 71
3.2.1 Results of the experienced students ...................................................................................................... 71
3.2.2 Results of the non-experienced students ............................................................................................... 73
3.2.3 Measuring the amount of time spent in catching up stage, reporting stage, programming stage and the communication stage ......................................................................................... 73
3.2.4 Qualitative Survey Results .................................................................................................................... 75

4 DISCUSSIONS .............................................................................................................................................. 83

4.1 EVALUATION OF THE EXPERIMENT RESULTS .................................................................................... 83
4.2 CHALLENGES .......................................................................................................................................... 84
4.2.1 Mapping the challenges of the experiment with literature ........................................................................... 86
4.2.2 How realistic are the results obtained? .................................................................................................... 89
4.2.3 Implications for the researchers .............................................................................................................. 90
4.3 VALIDITY THREATS ................................................................................................................................ 91
4.3.1 Internal Validity .................................................................................................................................... 91
4.3.2 External Validity .................................................................................................................................... 91
4.3.3 Construct Validity ................................................................................................................................ 91
4.3.4 Conclusion Validity ............................................................................................................................... 92

5 CONCLUSIONS .......................................................................................................................................... 93
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1</td>
<td>MAPPING RESEARCH QUESTIONS TO THE OUTCOME</td>
<td>93</td>
</tr>
<tr>
<td>5.2</td>
<td>FUTURE WORK</td>
<td>95</td>
</tr>
<tr>
<td>6</td>
<td>REFERENCES</td>
<td>97</td>
</tr>
<tr>
<td>7</td>
<td>APPENDIX</td>
<td>102</td>
</tr>
<tr>
<td>7.1</td>
<td>APPENDIX A</td>
<td>102</td>
</tr>
<tr>
<td>7.1.1</td>
<td>Initial Experiment Planning</td>
<td>102</td>
</tr>
<tr>
<td>7.2</td>
<td>APPENDIX B</td>
<td>106</td>
</tr>
<tr>
<td>7.2.1</td>
<td>Final Experiment Planning</td>
<td>106</td>
</tr>
<tr>
<td>7.3</td>
<td>APPENDIX C</td>
<td>124</td>
</tr>
<tr>
<td>7.3.1</td>
<td>Literature Review Tables</td>
<td>124</td>
</tr>
<tr>
<td>7.4</td>
<td>APPENDIX D</td>
<td>127</td>
</tr>
<tr>
<td>7.4.1</td>
<td>Kappa Coefficient</td>
<td>127</td>
</tr>
</tbody>
</table>
LIST OF FIGURES

Figure 1: Organizational relationships [24]................................................................................... 3
Figure 2: Phase-based development ............................................................................................ 4
Figure 3: Customization .............................................................................................................. 4
Figure 4: Modularization ............................................................................................................. 4
Figure 5: Follow-the-sun approach uses daily hand-off .............................................................. 5
Figure 6: Example of FTS with three development locations ..................................................... 6
Figure 7: Steps involved in study selection procedure ............................................................... 23
Figure 8: Overview of systematic literature review .................................................................. 28
Figure 9: Overview of the experimental process ......................................................................... 30
Figure 10: Experimental working of the co-located scenario ...................................................... 32
Figure 11: Experimental working of the follow-the-sun approach ............................................. 33
Figure 12: Working of the co-located team .............................................................................. 43
Figure 13: Working of the follow-the-sun with overlap scenario .............................................. 44
Figure 14: Working of the follow-the-sun without overlap scenario ........................................ 45
Figure 15: Distribution of development duration ........................................................................ 45
Figure 16: Effort for the three scenarios .................................................................................... 46
Figure 17: Number of included papers from each database ....................................................... 48
Figure 18: Number of published papers per year ...................................................................... 49
Figure 19: Research methods .................................................................................................... 50
Figure 20: Distribution of subjects of investigation ..................................................................... 51
Figure 21: Studies comparing the development duration of co-located and FTS scenarios 67
Figure 22: Utilized development duration .................................................................................. 71
Figure 23: Total effort utilized in each scenario .......................................................................... 72
Figure 24: Overview of initial experiment plan ......................................................................... 103
LIST OF TABLES

Table 1: Search String ............................................................................................................ 19
Table 2: Inclusion Criteria ...................................................................................................... 21
Table 3: Quality Criteria ......................................................................................................... 24
Table 4: Data Extraction Strategy ........................................................................................... 24
Table 5: Inclusion Criteria - Stage 1 ........................................................................................ 26
Table 6: Inclusion Criteria - Stage 2 ........................................................................................ 26
Table 7: Inclusion Criteria - Stage 3 ....................................................................................... 27
Table 8: Selected primary studies ........................................................................................... 29
Table 9: Assignment of subjects to different treatments ............................................................. 36
Table 10: Selection of variables ............................................................................................... 38
Table 11: Allocation of the subject groups to the treatments .................................................... 40
Table 12: Seating arrangements of the students ........................................................................ 42
Table 13: Overview of the included studies ............................................................................. 51
Table 14: Illustrates various themes of the research discussed in each of the studies .................... 54
Table 15: Challenges or obstacles ............................................................................................ 55
Table 16: Tools for enhancing the performance ....................................................................... 58
Table 17: Suggestions for improving the performance ............................................................... 60
Table 18: Comparison of follow-the-sun approach with co-located software development on various metrics ..................................................................................................................... 64
Table 19: Optimal number of development sites ..................................................................... 67
Table 20: Recommended phases of software development lifecycle to implement follow-the-sun approach ................................................................................................................................................... 69
Table 21: Percentage of time spent at each stage ..................................................................... 74
Table 22: Feedback from the experienced set of students ......................................................... 76
Table 23: Feedback from the non-experienced students ............................................................... 79
Table 24: Challenges, impact and the lessons learnt ................................................................. 84
Table 25: Mapping each of the challenges with group that has observed .................................. 87
Table 26: Time spent in each phase .......................................................................................... 119
Table 27: Calculation of actual development time .................................................................... 120
Table 28: Calculation of time spent in programming stage ....................................................... 120
Table 29: Calculation of time spent in catching up stage ............................................................ 121
Table 30: Calculation of time spent in reporting stage .............................................................. 122
Table 31: Calculation of effort ............................................................................................... 122
Table 32: Calculation of effort spent in each iteration .............................................................. 123
Table 33: Calculation of lines of code .................................................................................... 123
Table 34: Search for primary studies ....................................................................................... 124
# Abbreviations and Related Terms

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>FTS</td>
<td>Follow-the-sun approach (commonly known as round-the-clock development, 24-hour software development)</td>
</tr>
<tr>
<td>SDLC</td>
<td>Software development life cycle (The different phases of the software development)</td>
</tr>
<tr>
<td>SLR</td>
<td>Systematic literature review</td>
</tr>
</tbody>
</table>

“A process of identifying, evaluating and interpreting as much available research as possible done till now, on a particular topic or a research question or on a phenomenon of interest [8]”.

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>GSD</td>
<td>Global software development</td>
</tr>
<tr>
<td>NA</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>IM</td>
<td>Instant Messenger</td>
</tr>
</tbody>
</table>

Instant messaging (IM) is a real time communication system over the internet which is text-based. This is similar to a telephone conversation but here the conversation is text-based rather than voice based.

<table>
<thead>
<tr>
<th>Related terms</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Experiment</td>
<td>“An experiment is a formal, rigorous and a controlled investigation” [13].</td>
</tr>
<tr>
<td>Authors</td>
<td>Students who conducted this research (Saroj Pamulapati and Divya Gaddipati)</td>
</tr>
<tr>
<td>Experiment Leads</td>
<td>Friends of the authors who assisted during the experiment execution.</td>
</tr>
<tr>
<td>Subjects</td>
<td>The students who took part in the experiment. Also used as students in some context.</td>
</tr>
<tr>
<td>Time-to-market</td>
<td>Also known as speed, development time, cycle time. These are used interchangeably.</td>
</tr>
<tr>
<td>Cost efficiency</td>
<td>Effort</td>
</tr>
<tr>
<td>Day</td>
<td>Daylight hours i.e. 09:00 am – 05:00 pm</td>
</tr>
<tr>
<td>Night</td>
<td>During the night i.e. other than 09:00 am – 05:00 pm</td>
</tr>
<tr>
<td>Handover</td>
<td>Transferring of the work unit to the next subgroup that is dependent upon in order to continue [29].</td>
</tr>
</tbody>
</table>
1 INTRODUCTION

Software over the years has become an integral part of our life and almost nothing works without it. Software does not just run computers, but it does run various other machines, appliances, and systems i.e. “software is everywhere”. It is software that causes your digital alarm clock to set off in the morning, the microwave to fire up, sequences the traffic signals, operates the escalator thus helping make life more comfortable and organized. Software is today a very important commodity than it was a few decades ago. Customer today demand for software that is faster, economical and reliable.

The result of which, various software development methodologies have been proposed which are very different from the traditional form of co-located software development. Global software development is a one of those methodologies that promises reduction in development time and cost without compromising on quality. The concept of the global software development has thus become popular due to these benefits of cost advantage [10, 16, 25, 31, 33, 61], availability of large labour pool [16, 25, 31, 33, 49, 61, 63], proximity of local market [33, 48, 62, 64] etc.

Follow-the-sun approach is a way of organizing the software development globally, where development takes place during the normal working hours with development sites distributed in different time zones [30, 29]. In follow-the-sun approach, the multiple teams that are geographically distributed across time zones work on the same project [61]. Follow-the-sun approach is uniquely focused on reducing the time-to-market [6, 26, 29, 30], i.e. the time taken by the organization to develop a product i.e. from the start until the product is completed and delivered to the customer [60]. In follow-the-sun approach, as the 24-hours in a day are effectively utilized, the development time is reduced significantly. It is may be said that by practicing follow-the-sun approach the development time can be reduced by 50% if there are two sites participating and by 67% if there are three sites and so on [29]. This kind of reduction in cycle time is demanded the industry, especially were products life cycle is very short. Reduced time-to-market helps organisations to be ahead of competition and maximize the payback and be successful. Even though the concept of follow-the-sun approach sounds interesting, there are only few industrial success cases. Further academic research on follow-the-sun approach has not been substantial due to coordination difficulties [30].

The main intention and purpose of this thesis is to conduct a controlled experiment on follow-the-sun approach in the context of academia, to investigate and establish whether there is a reduction in the time-to-market by 50% as compared to the traditional method of software development known as the co-located approach. In the co-located approach, all the team members work from the same place and same time zone.

In this thesis, the follow-the-sun approach is simulated in two different scenarios. The two scenarios are: follow-the-sun with overlap and without overlap. In first scenario of follow-the-sun with overlap, the team members are split into two sub-groups, and are placed at a different locations and works on different time zone. In this scenario, both the sub-group members have an overlapping time during the working hours (also known as communication time) where the sub-group members can interact with each other through synchronous communication medium. In case of follow-the-sun without overlap scenario, the working style is almost similar to follow-the-sun with overlap scenario but, the only difference is that there is no direct communication between the sub-group members but the sub-group members can interact through asynchronous communication medium.
Also the purpose of this thesis is to provide a deeper understanding on follow-the-sun approach and evaluate the results of the experiment when implementing follow-the-sun approach in the development phase of the software development life cycle.

1.1 Background

1.1.1 Global Software Development

Global software development is a phenomenon of increasing importance; due to the stiff competition among the software firms to deliver the products fast and at low cost thus remaining profitable and competitive in the global market [31, 62]. Global software development is defined as “splitting the development of the same product or service among globally distributed sites” [36]. With the diffusion of the information and communication technologies throughout the world, has made the software development work to spread globally, which in turn has facilitated the software firms to set up globally dispersed teams coordinating across the boundaries of distance, time and culture [62]. Global software development is being extensively used in multinational companies, large corporations, joint ventures and even consulting companies because of the various advantages offered [33, 73].

1.1.1.1 Benefits of Global Software Development

Global software development offers the following benefits:

- **Reduced development cost** [10, 16, 25, 31, 33]: The main assumed advantage of global software development is reduced development cost. By having a part or the entire software developed in lower wage countries the overall development cost of the development is expected to be reduced significantly.

- **Leveraging time zone effectiveness** [27, 31, 48, 49]: Having team members from different time zones helps to maximize the number of working hours per day thereby reducing the cycle time and increasing the productivity through the follow-the-sun approach.

- **Access to large skilled labour pool** [16, 25, 31, 33, 48, 63]: Global software development allows choosing the best ones, i.e. the cream from the top universities around the world, from the huge pool of people available. Hence, one can have the best ones working on the project.

- **Closer proximity to market and customer** [33, 49, 63]: By having a development site at the customer’s location, helps to have a better understanding of the clients requirements. This is because of the direct interaction with the client; one tends to understand the trends and the expectations of the local market much better and thereby helping the company to develop a product that is of high demand. And also creates a job opportunity which helps in increasing the good will of the company and thereby increasing the chances of bagging in the new contracts [34, 31].

- **Innovation and shared best practice** [10, 33]: As people from different backgrounds, cultures, and organizations come together to work, it can be taken as an opportunity to learn and improve oneself by imbibing the best (such as the practices, techniques etc.) from each of them.

The benefits of the global software development that have been discussed above are only the main advantages. There are many other advantages such as improved task modularization, formal record of communication, improved documentation, clearly defined process, improved resource allocation, increased team autonomy etc that might be achieved through global software development [31].
1.1.1.2 Coordination of Software Development Activities in global software development

Global software development is the norm today for the technology companies [50] and every company that engages in global software development wants to achieve the benefits by developing the software at a cheaper, faster and with best quality [24]. For this reason organizations have started to engage in different kinds of sourcing strategies [43] such as offshoring, outsourcing, offshore outsourcing, nearshoring and onshoring. The different kinds of organizational relationships that exists are represented in the below figure 1.

<table>
<thead>
<tr>
<th>Same Country</th>
<th>National insourcing</th>
<th>Outsourcing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Different Country</td>
<td>Offshore insourcing</td>
<td>Offshore outsourcing</td>
</tr>
<tr>
<td>Same organization</td>
<td>Different organization</td>
<td></td>
</tr>
</tbody>
</table>

Figure 1: Organizational relationships [24]

The term sourcing is defined as “the act through which work is contracted or delegated to an external or internal entity that could be physically located anywhere [43]”. Outsourcing is defined as “contracting with the third party service provider for the management and completion of certain amount of work, for a specified length of time, cost and level of service [43]”. Offshore outsourcing is defined as “subcontracting a process, such as a product development, to a third party company that is located in a different country [24]”. Offshore insourcing is defined as “relocation of work by a company from one country to another [24]”. And finally national insourcing is termed as relocation of work by a company within the country.

Now by achieving a clear idea about the different organizational relationships that exists in global software development, the next step is to know, how the tasks are distributed among the organizations. Before knowing about the different task allocation strategies, it should be noted that global software development is not about only outsourcing [24], it is complex concept that can be understood through the different types of organizational relationships and task distribution methods [24]. The different ways of organizing global software development projects are
• **Phase-based development:** A particular phases of the software development cycle is relocated, for example testing.

![Phase-based development](image)

• **Customization:** Here the software is developed in one site and later customized by each site based on their local market requirements.

![Customization](image)

• **Modularization:** In this scenario, modules are relocated to different sites thereby causing the development to take place in parallel fashion.

![Modularization](image)

• **Follow-the-sun approach:** In this scenario, software is developed sequentially, where software activities are transferred on a daily basis between development sites which are many time zones apart. The rest of the thesis is based on this approach hence this concept will be explained in detail in the subsequent sections.
Phased-based, module-based and follow-the-sun approaches are global distributed configurations that decompose tasks, so as to allocate them to multiple sites, so that there is very little dependency between the sites [27, 28, 43]. The difference between follow-the-sun approach and phased-based as well as module-based approach is related to the concept handover. Follow-the-sun approach is built based on daily handover while phased-based as well as module-based approaches are based on the opposite concept by trying to minimize the handover as much as possible leaving the integration to the end of the project. This ultimately means that the risk is pushed to the end of the project in case of module-based and phase-based approach which may jeopardize the project, while in the case of follow-the-sun approach the iterative development can help to uncover the problems earlier.

As it is said everything in this world has a good side as well as a bad side, global software development is no exception. Even though global software development offers a number of benefits it does have a number of disadvantages also associated with it due to geographical, temporal and socio-cultural differences [41]. Global software development no longer allows the team members to have a rich interaction, real-time collaboration, face-to-face meetings, informal talks during lunch or coffee breaks [1], corridor talks [1], share a common organisational culture which helps in building coordination and control among the team members. Even though global software development has these obvious challenges, these are often overlooked because of the numerous assumed advantages, causing many organisations to take up global software development.

### 1.1.2 Follow-the-sun (FTS)

Follow-the-sun (also known as 24-hour development, round-the-clock development) software development is a particular direction of global software development. The concept of follow-the-sun is very simple, it reminds of an old saying that “Sun never sets on the British Empire” meaning that the British Empire was so vast that the sun was up on at least one of its numerous territories, which no longer holds true because of the disintegration of the territory. But the saying can be altered to fit into the follow-the-sun context by rephrasing as “Sun never sets in 24-hour knowledge factory” [2] which is very much true as the underlining concept of the follow-the-sun approach is that, the development takes place continuously for 24-hours a day by having development sites that have huge variations in the time zones.

For better understanding, take for example three development sites that are placed at three completely different time zones (see Figure 6), where one site is located in San Francisco, USA (marked as A in the figure 6), the other in London, UK (marked as B in the figure 6) while the third site in Darwin, Australia (marked as C in the figure 6). In the below figure 6, it can be seen that all the three locations are separated, where each development site is located in a different continent. The three sites are separated in the time zone by almost 8 hours. Let’s look at a normal working day closely when follow-the-sun approach is
implemented with these three sites. When it is 5 PM in Darwin, Australia where people are about to leave their work place, it is 8:30 AM in London, UK where people are just about to start their work. Hence the work that has been done till then by Darwin, Australia colleagues can be handed over to the colleagues in London, UK who are just about to start their day and by continuing to work from the point where the Darwin, Australia colleagues have stopped working on the project. Similarly, when it is 5 PM in London, UK it is 9 AM in San Francisco, California, USA. Hence the so far completed work by the London, UK team can be handed over to the team in San Francisco, California, USA who has just started their work and later the work is handed over to the team in Darwin, Australia. In this fashion the work can be handed over from one site to the other, allowing the development to take place continuously throughout the day by effectively utilizing the 24-hours a day without any halt.

This kind of round-the-clock development is also termed as Sequential Collaborative Software Engineering (SCSE) [6] as development takes place in a sequential manner. So in this follow-the-sun approach, instead of just working on the project for eight hours a day, the whole 24-hours in a day are effectively utilized which helps to reduce the development cycle time and thereby reducing the time-to-market by helping the organizations to be at a higher competitive edge over the others. It is said that the development duration reduces by 50% when there are two development sites, by 67% when there are three development sites [31].

The main motivation for the software development organizations to implement follow-the-sun approach is reduced time-to-market, which is a crucial factor in order to survive in this competitive world.

1.1.2.1 Definition of follow-the-sun

Carmel [29] who is a well-known researcher in the field of global software development coined the term follow-the-sun approach as

“A type of global knowledge workflow, which is designed in order to reduce the development time, in which the knowledge product is owned and evolved by a production site and is handed over daily to the next production site that is many times zones apart to continue with the so far completed work” [29].

Amar Gupta et. al [2, 3, 7] who is also into research on 24-hour development defines the term follow-the-sun approach as
“A global model in which the members of a globally distributed team work around the clock, where each member of the team works for the normal workday hours and then after their working hours pass on the work to their fellow team members who are located in a different time zone” [2].

Some successful cases of follow-the-sun approach (for example as seen in paper [71]) have been noted during the last decade. But when investigated closely, it was observed that in these cases where success was noticed when using follow-the-sun approach, there was no daily handover [28]. But daily handover is the underlining principle of follow-the-sun approach where without daily hand over it cannot be termed as follow-the-sun approach. It has been observed by the authors in [28] that this kind of situation prevails even in industries too, where the follow-the-sun approach is being used loosely with no or very little concept of follow-the-sun approach. For example: Indian offshore firms do a little follow-the-sun approach [15, 28]. Hence success when using follow-the-sun approach by following all the underline principles of follow-the-sun approach is very rarely if at all it has been noted also. And more often follow-the-sun approach is confused with other approaches such as global knowledge work, 24-hour business process, 24-hour manufacturing, co-located multi-shift approach [28]. Let’s have a deeper look at these concepts and see how they differ when compared to follow-the-sun approach so that there is no confusion.

- **Global knowledge work** [28, 25]: Here little task dependency exists causing the team members from different geographical locations to collaborate. The handover is practiced but not with an intention to reduce the development duration. Follow-the-sun approach is subset of global knowledge work but not vice versa.
- **24-hour business process** [45]: Here work is directed to workers who are on an active shift that is usually in daylight hours, anywhere in the world. There is little task dependency and handover is not intended to reduce the duration so, it is not practised on a daily bases. For example call centers, helpdesks practice 24-hour business process to provide continuous service.
- **24-hour manufacturing** [45]: Here workers assemble the products till the end of their shift. Shifts here are used to take advantage of expensive resources.
- **Co-located multi-shifts** [45]: Here a lower waged location is selected and several 8 hour shifts of the software developers are done to take advantage of low cost, and sometimes overlap persists between the shifts to allow face-to-face handover coordination.

So in order to avoid to this confusion on the concept of follow-the-sun approach, it is important to note that the following criteria’s are to be met while using follow-the-sun approach:

- There should be more than one development sites participating in the process which works during its daylight hours as one of the sub-team.
- Each sub-team should consist of one or more members.
- The sites that are participating should be many time zones apart.
- There should be only one development site that works on the project at any given point of time.
- Hand-over has to be carried out on a daily basis.
- The main goal of using follow-the-sun should be to reduce the development duration.
- The handover between sites can be empty sometimes in case of national holidays or emergencies.
- There should be a common digital repository for all the sites to commit the code at the end of the work day.
1.1.2.2 Advantages of the follow-the-sun approach

All the advantages that are achieved while using global software development are applicable even when using follow-the-sun approach. Follow-the-sun approach offers an additional advantage that is reduced “time-to-market”. Apart from time-to-market another benefit of follow-the-sun approach is that follow-the-sun approach avoids the need to work during the odd hours [4]. In global distributed development, the developers need to work during odd hour i.e. late in the night. But in case of follow-the-sun approach as there are three developments sites involved on the same task, there is less pressure on the developers to work during the night.

1.1.2.3 Challenges in the follow-the-sun approach

Among the different global software development scenarios such as phase-based, module-based etc, follow-the-sun approach is the least emphasized one. This is because of the various difficulties faced, where some of them are specific to follow-the-sun context while others are caused by global software development environment in general. According to Carmel [27], global software teams are pulled apart mainly due to the five centrifugal forces namely, geographical distribution, coordination breakdown, cultural differences, loss of teamness and loss of communication richness.

- **Geographic distance:** In global software development the development sites are geographically dispersed because of which the team members seldom get to meet each other. Due to the lack of face-to-face interactions among the members there is a lack of teamness resulting in lack of trust among the members which has a significant impact on the team’s performance.

- **Coordination breakdown:** It is important to have proper coordination in order to integrate the work that is distributed geographically to control, solve and discuss issues that arise during the software development process. But in global software development due to lack of communication richness, time zone difference, absence of collaboration etc coordination becomes a challenging task causing misunderstandings, losing control over the project milestones, causes delays in the schedule etc.

- **Cultural differences:** Culture here involves organisational culture [41], national culture [41], language [41], politics [41], work ethics [41], attitude towards hierarchy [38, 47], time sense [38, 47], communication style [38, 47], use of terminologies [55], understanding of a problem [55] etc. Global software development involves people from various cultural backgrounds where each culture has its own unwritten rules and etiquettes [2] causing misunderstanding among the team members which has a profound impact on the final output.

- **Loss of teamness:** Due to the heterogeneous nature of the teams and the distance among them, building of trust becomes difficult there by hampering the spirit of teamness among the members.

- **Loss of communication richness:** It is important to have a proper communication while developing software, especially during its initial phases [17, 47] for developing a product which is of high quality. Communication just does not imply conveying the message but it also involves the expression of emotion [64]. But using global software development one can observe a reduction in communication and also the little communication that takes place is also not effective because of temporal, language, culture and geographical distance. Due to this, there are many challenges that are faced such as lack of information about the skill set of a person [47], no information regarding the person responsible for a particular task [47], misalignment and rework [1, 4], lack of project status update [47], increase in project related issue [47] etc.
Apart from the five centrifugal forces described by Carmel another important factor is the “Handover”. Handover is one of the important aspects in follow-the-sun approach as work has to be appropriately delivered to the distributed team. There are many problems that arise due to this handover scenario. Teams should properly document the things that they have done during their working hours so that the other member who is on a different time zone does not waste much time in understanding the task. If the task is not clearly described and the distributed team can stuck up in between, which causes the team to wait until the other team returns to work the next day. If there are any such kind of problems, instead of the promised reduction in the development time there will be an increase in the development time and cost, which is eventually a great loss to the company.

Other issues such as decisions regarding division of work among the sites participating, organizations resistance to practise global software development [56], knowledge, project and process management issues [47], technical issues (such as the network bridging the globally distributed teams may be slow and unreliable), incompatible data format, difference in the version of a tool [47] etc make global software development difficult to practice.

1.1.2.4 Solutions for the follow-the-sun approach

As every problem has a possible solution to mitigate the effect that caused the problem, here in this section, the solutions for the efficient utilization of the follow-the-sun approach are discussed.

- **Measures to overcome geographic distance**: To overcome problems caused by the geographic distance there needs to be a frequent face-to-face communication [41, 75], where travelling is the only way to achieve this. By doing so, it increases the feeling of teamness and trust among the team members. But keeping mind the cost and time constraints that are involved, this mitigation strategy can be applied for particular phases which are considered to be crucial in the development process such as during integration [41] etc. Good use of conference calls, audio and video conversation tools can act as alternatives to face-to-face meetings. Use of team website can also help minimize the geographical distance [41], where information related to individuals such as photo of each employ is maintained so that one actually envisions that, there exists a person on the other end thereby inculcating a feeling of team spirit [41, 58]. Tasks related information can also be maintained to keep the members updated with the status, the person responsible for a particular task and so on [41, 58].

- **Minimizing coordination breakdown**: Coordination can be achieved by using collaborative tools that support analysis, design and development to keep track of the activities and manage dependencies, notify, and implement corrective measures. Use of models such as the simulation model that has been examined in paper [55], which calculates the impact of coordination efficiency and its effect on productivity and whereas in paper [9] an Ariadne tool is presented that identifies dependencies in a project can be used to determine any coordination problems that exists through a visual environment.

- **Measures for reducing cultural differences**: Problems related to language can be diluted by using formal asynchronous communication rather than informal synchronous communication [41]. By using asynchronous communication people with not good communication skills or language proficiency can take time to understand what the other member is trying to convey and reply accordingly. Another possible solution is to use a buddy system which has worked well in companies like Fidelity [41]. But the best solution is travelling [41]. By travelling one gets to know about the other culture, their way of looking at things, their working style, the way situations are handled, can understand the subtle difference
which pave paths to serious misunderstandings etc. The use of liaison helps to understand the cultural issues of the other development site thereby alleviating misunderstanding between the teams.

- **Steps to be taken to incorporate a feeling of teamness [64]**: This can be done by decreasing the team size thereby decreasing number of communication links in the team. Secondly, by organizing storming sessions at the very beginning so that team members get to know each other enriching the feeling of teamness. Besides, by adopting a common practise helps inculcate a feeling of oneness.

- **Mitigation strategies for the loss of communication richness**: Communication problems can be eradicated by clearly defining requirements which eliminates the need for communication. Use of good communication tools in which facial expressions and social cues are evident are to be used to avoid confusions and misunderstandings that crop up between the members. Another solution can be to use a common repository between the development sites.

### 1.2 Related Work

The first well-documented case of follow-the-sun approach was at IBM in the mid 1990’s [27, 28] with 5 development sites. But the team was not successful in implementing follow-the-sun approach as it was abnormal to move the software artefacts between sites on a daily basis, for this reason the use of follow-the-sun approach has been given up. It was decided that the effort put into the daily handover has to be stopped thereby decreasing the collaboration between the sites.

The first researchers to investigate follow-the-sun approach were Hawryszkiewycz, Gorton and their peers during mid-1990’s. They set up a number of small controlled experiments [28, 39] but did not continue their research. Another researcher named Cameron [14, 28] noted limited success while using follow-the-sun approach at global American firm EDS (now known as HP) but did not continue his line of research either. Promises of 24-hour knowledge factory have been noted extensively in paper [40] by Gupta [28, 40].

A controlled experiment was conducted at IBM to evaluate the difference in the performance between the co-located and the distributed team. The follow-the-sun team was geographically dispersed in India and United States of America (USA). The team was closely monitored for 52 weeks and it was noticed that the distributed team was good in terms of documentation and knowledge repository when compared to the co-located team. It was evident that the turnaround time taken to resolve the tasks in the case of follow-the-sun team was shorter [7]. There were obstacles that were faced by the team members such as loss of informal communication between the developers while using follow-the-sun approach. It was evident from the experiment that, it is not required to be geographically close for achieving high productivity but by using appropriate technology and work procedures the barriers could be mitigated.

Carmel et al. in [29] conducted a quasi-experiment with experienced computer science and electrical engineering students for fourteen weeks to investigate follow-the-sun approach in an agile environment to determine if follow-the-sun approach actually reduces time-to-market as promised in theory. During this experimental observation, it was noted that there is a 10% reduction in the development time rather than the theoretical 50% for the follow-the-sun team when compared to the co-located team. From the experiment it was concluded that developing small pieces of code, then testing the developed code and storing it in a common repository with daily handover will have profound effect while practicing follow-the-sun approach. It was also observed that the distance and trust are the most influencing factors with 30% impact on the project duration while culture and language have negligible impact on the project duration. It was finally concluded that follow-the-sun approach suits best for
testing phase in many companies rather than in the other phases of the software development lifecycle thereby reducing the overall project duration by 12.5% rather than the ideal 50%.

Monica Yap in her paper [56] presents the experiences such as challenges, lessons learned and mitigation strategies while implementing follow-the-sun approach. The team members were placed in United States of America, United Kingdom and Singapore to form a 24*5 around-the-clock extreme programming team. After a year of practicing extreme programming the team was considered to be successful. It was noted that around-the-clock development works in a distributed environment with a shared codebase.

Siri-on Setamanit et al. in their paper [67] discuss a proposed hybrid computer simulation model for the software development processes to examine various allocation strategies such as phased-based, module-based and follow-the-sun approach. It was follow-the-sun that took the longest time among phased-based, module-based and single-site configuration. In paper [67] when the model was run for 30 replications for each configuration, it was observed that there was increase in effort by 50%, while the duration increased by 50% and the quality of the end product was worse in case of follow-the-sun approach when compared to the single site strategy. Hence it was concluded [67] that the use of follow-the-sun approach has to be avoided as it requires communication and coordination between sites resulting in higher effort and longer duration. It was also found that it is extremely difficult to adapt follow-the-sun approach with two development sites, but preferable to use with three development sites to achieve the benefit of reduced development time.

Rini van Solingen and Menno Valkema in [65] conducted a control experiment to find out the impact of number of sites on the individual’s working speed and the accuracy. The experiment was designed based on the experiment conducted by Carmel et al. [29]. From the experiment it was observed that there is a decrease of 20% in the working speed of the individual sites with the addition of a site and also increase in number of sites had a small impact on the working accuracy of the individuals. It was finally concluded in the experiment that it is better to concentrate on maximizing the working speed and the accuracy of the individual sites participating instead of focusing on addition of a site.

Christian Visser and Rini van Solingen [12] in their paper “Selecting Locations for the Follow-the-sun Development” constructed a routing model based on an algorithm to determine the optimal geographical locations that are best suited to implement follow-the-sun development. An algorithm was used in designing the routing model and it was executed on a website, so that anyone who wishes to select the optimal locations can input the data and get the information about the feasibility of the selected locations. From the experiment the authors concluded that with the increase in the overlapping hours, the coordination cost was decreased but there is an increase in the vulnerability cost. It was said that there was no need for the overlapping hours to be large, but the communication between the locations has to be frequent. And finally the authors in their paper concluded that the optimal time zone difference depends on the transfer time for handing over the project to the other distributed locations and the ease of communication depends on the culture and language skills [12].

Supraja Doma et al. [66] in the paper “Resource Allocation Optimization for the GSD Projects” proposed a model for the allocation of the work to the development site which contains efficient resources without affecting the project duration. The entire model was build based on two main assumptions, one is that there is no delay in the project and another one is that, during the allocation of the tasks to the development sites the resources where randomly picked which may lead to higher development cost. Authors stated that even though time and resource are equally important, in the paper time has been given more priority than resource because for getting the best resource one has to wait for a longer time resulting in increase of the development time of the product.
Amar Gupta et al. [3] in the paper “Use of Collaborative Technologies and Knowledge Sharing in Co-located and Distributed teams” the authors conducted a quasi-experiment at IBM to discover the difference in the technology use, knowledge sharing practices and the performance between the 24-hour knowledge factory team and the co-located team. From the case study and the experiment, it was noted that the distributed team and the co-located team did not differ in terms of quality and efficiency but rather differed in terms of technology use and knowledge sharing processes. And it was observed that by introducing the gratification strategies such as face-to-face meeting and informal phone calls one can level off the performance between the co-located and the distributed team.

Amar Gupta et al. [2] in the paper “24-hour Knowledge Factory: Using Internet Technology to Leverage Spatial and Temporal Separations” described an internet based prototype to optimize the data collection and the storing of the design rationale from the various stakeholders and workers. Authors have stated that the 24-hour knowledge factory increased the efficiency when compared to a single-site team because there is a lot of overnight progress in case of distributed team. The authors also quoted that by having the work in a semi-structured manner helps reduce the time-to-market for the products and services which in turn reduces the operating and development costs.

Amar Gupta et al. [7] in the paper “Leveraging Temporal and Spatial Separations” conducted an experiment at IBM and found that the distributed team outperformed the co-located team in terms of turnaround time, documentation and knowledge repository. In this paper the authors discussed various tools for the efficient usage of the round-the-clock development. Some of the tools that were discussed are CPro, MultiMind and EchoEdit that helps to keep track of the reason behind each decision along with an explanation of the code. These tools generate a schedule for achieving maximum performance and help the teams to collaborate in an optimal manner.

Pankanj Jolote and Gourav Jain [61] in their paper “Assigning Tasks in a 24-hour Software Development Model” a model was developed based on an algorithm for the distributed team and the model has been applied to synthetic projects and two real time projects to prove the reduction in the development time and to improve the resource utilization in case of the 24-hour development team. When the developed model was applied to two real time projects, it was observed that there is reduction in the development time by 10-20% in case of non-overlapping hours.

In summary, follow-the-sun approach sounds great if it worked as promised, but from previous studies it can be noted that it does not work in reality as promised in theory. Despite the failures of using follow-the-sun approach at various instances, the desire or the thirst to enjoy the numerous advantages provided by follow-the-sun approach has lead us to design an experiment that tests follow-the-sun approach in two different sequences (follow-the-sun approach with overlap scenario and v without overlap scenario) to determine which scenario suits better for follow-the-sun approach. As the property of any engineering discipline is to match the theoretical speculations with reality [59], so the research seems interesting, to make the follow-the-sun approach process more beneficial in real time implementation. So to bridge the gap between theory and real time scenario a controlled experiment has been conducted in a simulated environment.

### 1.3 Research Motivation

From the above related work section, it can be concluded that follow-the-sun approach is not a new concept as the organizations have started implementing the approach from mid 1990’s [27, 28, 39]. During 1990’s when follow-the-sun approach was implemented in the organizations the amount of failure rate [14, 27, 28, 40] was high. For this reason...
organizations have stopped practicing follow-the-sun approach. Even though the organizations were not successful in implementing follow-the-sun approach [27], the thirst to enjoy the numerous benefits offered by follow-the-sun approach, have not stopped the researchers in figuring out the different ways for the effective implementation of follow-the-sun approach. Slowly, the researchers started experimenting by conducting small controlled experiments [3, 29, 65] and developing mathematical models [12, 61, 66, 67] to gain a deeper understanding of the follow-the-sun approach. By doing so, it was observed that development time can be reduced 10-20% [29, 61, 62] when compared to the co-located approach.

In most of the research papers where a similar research is conducted [29] on follow-the-sun approach, it is being said that follow-the-sun approach suits better either in testing phase [29, 41], prototyping phase [29], maintenance phase [31], defect resolution and support [41] and for bug fixing or call center activity [45]. There is only one paper where the research is carried out in software development phase of the follow-the-sun approach [29]. In this paper [29], follow-the-sun approach without overlap scenario was compared with the co-located scenario. This means that to the best of the author’s knowledge, there is no research that has been carried out till now in comparing follow-the-sun with overlap scenario and follow-the-sun without overlap scenario with the co-located scenario. As there is only one paper [29] that discusses regarding the implementation of follow-the-sun approach in the development phase of the software development life cycle, the experiment design of this thesis was based on the experiment that was conducted by Carmel [29]. As we know that improved understanding is the basis for changing and improving the way we work [8], the authors of this thesis are conducting an experiment for comparing follow-the-sun with overlap scenario and follow-the-sun without overlap scenario with the co-located scenario, to investigate whether there is a reduction in the development duration by 50% for follow-the-sun approach when compared to the development duration of the co-located approach.

1.4 Aims and Objectives

Aim: To determine whether the assumed benefit of reduced time-to-market by 50% can be achieved by using follow-the-sun with overlap scenario and follow-the-sun without overlap scenario when compared to the co-located scenario. In order to achieve the aim of the thesis, the authors will gather the research evidence on follow-the-sun approach by conducting a systematic literature review and then empirically test the results of the literature review by conducting an experiment.

The following objectives are defined to meet the aim of this thesis:

- Identify the problems and challenges faced while implementing the follow-the-sun approach.
- Identify the tools to be used for the better performance of the follow-the-sun approach.
- Identify the optimal number of site for implementation of follow-the-sun approach.
- Identify the suggestions for improved performance of follow-the-sun approach from the literature.
- Identify the phases that are suitable for implementing follow-the-sun approach.
- Compare follow-the-sun approach with co-located approach based on various parameters.
- Measure how long does it take for the co-located approach in the experiment to complete the given task when compared to the follow-the-sun approach.
- Analyse whether the time-to-market reduces by 50% in the experiment when using follow-the-sun with overlap scenario when compared to the co-located scenario.
• Analyse whether the cost reduces by 50% in the experiment when using follow-the-sun without overlap scenario when compared to the co-located scenario.
• Compare the results obtained from the experiment of follow-the-sun approach applied on with overlap and without overlap scenario.
• Evaluate the cost efficiency of the follow-the-sun scenario and co-located scenario.

1.5 Research Questions

In this section, the research questions of the systematic literature review and the experiment are discussed. Here answers to the RQ1 are addressed from the results of the systematic literature review and the answers to the research question from RQ2-RQ4 are addressed from the results of the experiment.

RQ 1 What are the best suitable factors for the implementation of follow-the-sun approach?

RQ 1.1 What are the challenges reported in the literature that are faced while using follow-the-sun approach?
RQ 1.2 Which tools can be used to enhance the performance of follow-the-sun approach?
RQ 1.3 What suggestions are cited in the literature for improved performance when using follow-the-sun approach?
RQ 1.4 What is the effect of follow-the-sun approach on various parameters in comparison with co-located approach?
RQ 1.5 What is the optimal number of development sites for implementation of follow-the-sun approach?
RQ 1.6 Which phases of the software development life cycle are reported to be suitable for follow-the-sun approach?

RQ 2 Can we deliver faster using follow-the-sun approach when comparing to co-located scenario?

RQ 2.1 What is the effect of follow-the-sun with overlap scenario on the time-to-market in comparison with the co-located scenario?
RQ 2.2 What is the effect of follow-the-sun without overlap scenario on the time-to-market in comparison with the co-located scenario?
RQ 2.3 What is the effect of follow-the-sun with overlap scenario on the time-to-market in comparison with the follow-the-sun without overlap scenario?

RQ 3 Can we be cost-efficient when using follow-the-sun approach when comparing to co-located scenario?

RQ 3.1 What is the effect of follow-the-sun with overlap scenario on cost efficiency in comparison with the co-located scenario?
RQ 3.2 What is the effect of follow-the-sun without overlap scenario on cost efficiency in comparison with the co-located scenario?
RQ 3.3 What is the effect of follow-the-sun with overlap scenario on cost efficiency in comparison with the follow-the-sun without overlap scenario?

RQ 4 What are the challenges faced by the subjects of the follow-the-sun scenario during the experiment and the recommended solutions to overcome the challenges?
1.6 Research Methodology

In this thesis we are following a mixed methodology i.e. systematic literature review and experiment. Systematic literature review is used to find out the challenges faced, tools to be used, favourable conditions, other software development methodologies and the software development phases that are suitable for implementation of follow-the-sun approach. Experiment helps in determining the time-to-market, cost efficiency and the challenges that are faced by the subjects during the experiment execution.

1.6.1 Systematic Literature Review

In this study, systematic literature review is performed according to the guidelines given by Kitchenham et al [8]. A systematic literature review according to the guidelines defined by Kitchenham is the process of identifying, evaluating and interpreting as much available research as possible done till now, on a particular topic or a research question or on a phenomenon of interest [8]. As any kind of research kick-starts with some sort of a literature review [8], so it is important that a literature review is done thoroughly without any bias to have a detailed scientific knowledge on the area of research. The intention behind the authors of this thesis for choosing systematic literature review is that, it helps to analyse the research that has been done so far with minimal bias and also as the systematic literature review is a formally planned process which is executed based on a predefined procedure at every instance.

In this thesis the systematic literature review is performed in order to find out the relevant research that has been done till now on follow-the-sun approach. The results of the systematic literature review will helps gaining an overview on the characteristics of the included studies and for the deeper analysis of the studies we have chosen narrative synthesis for synthesizing the findings of the various studies that are included in our research.

Narrative synthesis is a synthesis method in which findings of various studies that are included, are summarized and explained, primarily with the help of words and text [52]. Narrative synthesis is a form of storytelling approach that relies on the use of words and text to summarize and explain the findings of the studies included in the review [52]. It consists of four main steps. They are [52]:

- Developing a theory that details on how the intervention works, why and for whom.
- Preliminary synthesis of the findings of the studies included in the systematic literature review.
- Investigating relationships between the findings.
- Evaluating the robustness of the conducted synthesis.

The main reasons for the authors to choose narrative synthesis are:

- Applicable for studies covering a wide range of research questions [52].
- Applicable when the evidences are diverse in nature [19].
- Is flexible [19].
- Applicable to the reviews of qualitative and/or quantitative research [18].

The detailed description of the systematic literature review process and its stages are explained in the Chapter 2 of this thesis.
1.6.2 Experiment

Based on the purpose of evaluation and depending on conditions of the empirical investigation, the different types of empirical research are case studies, surveys, experiments etc [13]. Case studies and surveys are not suitable for this thesis as case studies are used for monitoring the projects, activities or assignments and as they are mainly observational studies which are used to monitor on-going project [13], whereas surveys are carried out by taking a sample which is representation from the population to be studies [13]. Experiment is the best suited empirical methodology for this thesis as the effect of development duration on the follow-the-sun approach is compared with the co-located approach which requires the variables to be manipulated by having control on all other variables [13] and as this is only possible while conducting an experiment. In this thesis to address the research questions an experiment is conducted involving the bachelor students of Blekinge Tekniska Högskola.

According to Claes Wohlin [13], experiment is defined as

“A formal, rigorous and a controlled investigation and the experimentation is a process of systematic, disciplined, quantifiable and controlled way of evaluating the human based activities [13].”

From the characteristics of the experiment stated by Claes Wohlin [13], the authors of this thesis think that experiment is the best suited research methodology for this study because of the following characteristics [13]:

- Experiments are used to test the existing theories,
- Experiments are used to test people’s conception,
- Experiments are used to explore the relationships,
- Experiments are used to evaluate the accuracy of the models,
- Experiments are used to validate the measures.

The step by step processes that are involved in carrying out the experiment are [13]:

- Stage 1: Experiment definition
- Stage 2: Experiment planning
- Stage 3: Experiment operation
- Stage 4: Experiment analysis and interpretation

The stages 1-3 are discussed in the chapter 2 and the stage 4 is discussed in the chapter 3 of this thesis. In this study, experiment is conducted in an offline mode i.e. the experiment is conducted in a laboratory setting with limited students as subjects for the experiment and in a controlled manner. The results from the experiment will be a contribution towards the research in the field of global software engineering especially for follow-the-sun software development.

1.7 Research Outcomes

- **Problem scope and current research gaps:** The results of the literature review (RQ 1) provide the background information related to follow-the-sun approach such as the challenges that are faced, tools to be used, favourable conditions for follow-the-sun implementation, comparison with co-located approach based on various parameters and the optimal number of development sites for the implementation of follow-the-sun approach.
• **Empirically tested time-to-market in three different settings:** The results of the experiment helps to determine if follow-the-sun approach decreases or increases the development duration when compared to the co-located scenario (RQ 2). The output of the experiment also helps to compare the time-to-market of follow-the-sun with overlap scenario and follow-the-sun without overlap scenario and see if follow-the-sun approach works the same for both scenarios. Here the development duration between the co-located scenario and the follow-the-sun scenario is calculated in terms of amount of time spent on catching up activity, amount of time spent on programming the task, time spent on reporting the task to the experiment leads and the total amount of time spent on communication.

• **Empirically tested cost efficiency of work in three different settings:** The results of the co-located scenario and the follow-the-sun scenario (RQ 3) are compared to determine the amount of effort that was required by the co-located scenario and follow-the-sun scenario in completion of the task.

• **Empirically collected challenges and improvement suggestions:** Based on the authors observation during the experiment and based on the feedback collected from the subjects, the problems and challenges that are faced by the follow-the-sun team (RQ 4) such as the cultural differences, communication and coordination issues, technical skills of the students, timeliness, satisfying the requirements etc are noted down and also the subjects are asked for the suggestions to mitigate the problems that they came across during the experiment execution. The students are asked to fill in a feedback form at the end of each iteration for noting down the amount of work done and the experience of the students.

1.8 **Thesis Structure**

Chapter 1 (Introduction): This chapter gives a brief overview of the research i.e. discusses regarding the motivation for conducting the research, aims and objectives and the research questions that are addressed in this paper.

Chapter 2 (Research Methodology): This chapter discusses about the research methodology that is adapted in this thesis, i.e. systematic literature review and experiment. The systematic literature review part details the step by step procedure that is followed for the selection of the research papers and the information that is extracted from the papers. The experimental part describes the details from defining the experiment definition till the experiment execution plan.

Chapter 3 (Results and Analysis): The results and analysis part explains the results of the systematic literature review and the experiment.

Chapter 4 (Discussions): This chapter talks about the findings of the experiment, lessons learnt and the validity threats of this thesis.

Chapter 5 (Conclusion): This chapter consists of discussion regarding the experimental outcome, conclusion of the research and suggestions for future work

Chapter 6 (References)

Chapter 7 (Appendix)
2 RESEARCH METHODOLOGY

2.1 Systematic literature review

The systematic literature review followed in this thesis is based on the guidelines given by Kitchenham [8]. The main motivation that drew the authors to opt for this systematic literature review procedure is the well-defined search strategy. By following a well-defined process, it becomes easier to find out the reliable set of representative research from existing literature, identify the gap in the research and finally can come with a new theory or can add more empirical evidence to existing literature by contributing informative knowledge to the field of research.

In systematic literature review the review process is more rigorous and well-defined [8] than the traditional literature review. The set of predefined and repeatable steps in the systematic literature review are:

- 2.1 Planning the review
- 2.2 Conducting the review
- 2.3 Reporting the review

2.1.1 Planning the review

Planning the review process is the first and the foremost step that is to be carried in systematic literature review, which is performed to confirm the need for a review process [8]. Planning the review process helps in determining the path by making things easier before one starts conducting the actual review process.

2.1.1.1 Need for systematic literature review

The driving force for performing the systematic literature review is to add scientific value to the literature review [8]. In this research the main aim for conducting a systematic literature review is to identify, evaluate and summarize the evidences till date on follow-the-sun approach in a thorough, structured and unbiased way, helping the authors to draw more concrete solutions. The results of the experiment are compared with the results in the literature, for determining the similarities and differences in the obtained results and also to analyse the reasons for the deviation in the experimental results.

2.1.1.2 Defining Research Questions

Formulating the research questions is one of the important activity that is to be carried out during the planning of the review process. The following are the research questions that are addressed using the systematic literature review.

RQ 1 What are the best suitable factors for the implementation of follow-the-sun approach?

 RQ 1.1 What are the challenges reported in the literature that are faced while using follow-the-sun approach?
 RQ 1.2 Which tools can be used to enhance the performance of follow-the-sun approach?
**RQ 1.3** What suggestions are cited in the literature for improved performance when using follow-the-sun approach?

**RQ 1.4** What is the effect of follow-the-sun approach on various parameters in comparison with co-located approach?

**RQ 1.5** What is the optimal number of development sites for implementation of the follow-the-sun approach?

**RQ 1.6** Which phases of the software development life cycle are reported to be suitable for follow-the-sun approach?

### 2.1.1.3 Review protocol development

The development of a review protocol specifies the methods that are to be used in the development of the review process [8]. For finding out the research done till now on follow-the-sun approach, a systematic pre-defined review process is performed.

#### 2.1.1.3.1 Search for primary studies

#### 2.1.1.3.1.1 Formation of search string

For the selection of the primary studies, the first step is to determine the search string for performing electronic search in the databases and to determine the research articles that are related to follow-the-sun approach. The formation of the search string takes place in three sub-stages, where in the first stage, the keywords of the research are determined. In the second stage, for finding more number of relevant research articles, the synonyms for the keywords are listed down. In the final stage of the search string formation, the search string is formed using the Boolean operator “AND” and “OR”.

**Step1: Determining Keywords**

- “Global software engineering”
- “Follow-the-sun”

**Step2: Synonyms and Alternative Spellings for the Keywords**

<table>
<thead>
<tr>
<th>&quot;Global software engineering&quot;</th>
<th>“Follow-the-sun”</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;global software development&quot;</td>
<td>“follow-the-sun”</td>
</tr>
<tr>
<td>&quot;distributed software engineering&quot;</td>
<td>“round-the-clock development”</td>
</tr>
<tr>
<td>&quot;distributed software development&quot;</td>
<td>“24-hour development”</td>
</tr>
<tr>
<td>&quot;multi-site software development&quot;</td>
<td>“24-hour knowledge factory”</td>
</tr>
<tr>
<td>&quot;multisite software development&quot;</td>
<td>“24/7”</td>
</tr>
<tr>
<td>&quot;multisite development&quot; AND &quot;software&quot;</td>
<td>“24/365”</td>
</tr>
<tr>
<td>&quot;multi-site development&quot; AND &quot;software&quot;</td>
<td></td>
</tr>
</tbody>
</table>

---

**Table 1: Search String**
Step 3: Search string is constructed by joining different keywords using AND operator and the synonyms of each keyword using OR operator.

("global software engineering" OR "global software development" OR "distributed software engineering" OR "distributed software development" OR "multi-site software development" OR "multi-site software development" OR "multi-site development" AND "software") OR "collaborative software development" OR "collaborative software engineering" OR ("globally distributed work" AND "software") OR "geographically distributed software development" OR ("distributed development" AND "software") OR "cross continent development" AND "software" OR ("cross-continent development" AND "software") OR "cross-site software development" OR ("cross-site development" AND "software") AND ("24-hour development" OR "24-hour knowledge factory" OR "24/7" OR "24/365")

2.1.1.3.1.2 Databases searched

The electronic databases that are selected to perform the search for primary studies are:

- Engineering Village
- IEEE Xplore
- Springer Link
- ISI Web of Knowledge
- ScienceDirect
- ACM Digital Library
- Wiley Inter Science Journal Finder

It is important in software engineering that as many as possible different electronic databases are to be searched so that all the studies are covered without missing upon much of the studies as the present software engineering search engines do not support systematic literature review [8]. In particular, the above databases have been selected as they would cover most of the research that has been done till now and also consists of only those studies that have been published adding scientific value to the research that is being conducted. The main reason of selecting these databases is to find out the relevant and only the peer-reviewed articles that are relevant to the software engineering field.

Follow-the-sun approach is increasing its widespread usage in the current market due to various benefits attained by following this approach. Even though there are various benefits,
organizations are facing difficulties in achieving the desired results. Researchers have been experimenting in this field in finding out the correct way of practicing this approach. As very little research has been done in this field and to provide a complete insight of follow-the-sun approach, an extensive search is formed in the above cited databases.

Papers published till the time of the search (i.e. April 2011) are considered in order not to miss any research in this field.

2.1.1.3.2 Study Selection Criteria

The study selection criterion is used for justifying the selected search strategy that is appropriate for the research questions [9]. Based on the inclusion and exclusion criteria, studies relevant to the research are selected from the search resulted after applying the search string. Inclusion and exclusion criteria are performed so that the data is interpreted and classified correctly [9].

- **Inclusion Criteria**
  - The article/study answers any one of the research questions mentioned above.
  - The article/study is accessible through Blekinge Tekniska Högskola online library service.
  - Studies published in English only.

Apart from the above mentioned inclusion criteria for the selection of the primary study the general selection criteria form that is followed in selection of the research articles are described below in table.

**Table 2: Inclusion Criteria**

<table>
<thead>
<tr>
<th>Stage</th>
<th>Relevance</th>
<th>Selection Criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>By search</td>
<td>- Contains the search string</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Published only in English</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Publication date: till present</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Availability of the full text</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Remove duplicates</td>
</tr>
<tr>
<td>2</td>
<td>By title and abstract</td>
<td>- Remove (editorials, prefaces, discussions, comments, summary of tutorials)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Related to global software development</td>
</tr>
<tr>
<td>3</td>
<td>By full text</td>
<td>- Focus mainly on follow-the-sun approach in global software development</td>
</tr>
</tbody>
</table>
Exclusion Criteria

The exclusion criterion determines the restriction of the search strategy. The following is the exclusion criterion that is applied in this literature review:

- The grey literature is not included in the study as it is mentioned previously that to add scientific evidence only the peer-reviewed articles are considered in the study.
- Secondary study papers are not included.
- Anecdotal studies.
- Do not relate to software engineering.
- Do not relate to follow-the-sun approach.

2.1.1.3.3 Study Selection Procedure

The study selection procedure followed is similar to the Toolgate method [74] (see also figure 7)

- **Stage 1:** A search is performed by both authors together in the seven databases mentioned based on the pre-defined search string. The resulting papers are further refined in the stages 2, 3 and 4.
- **Stage 2:** For the list of papers obtained from the stage 1 the authors apply the inclusion criteria, where the papers are selected based on the availability of full text, containing the search string and the papers published in English. Once the list of papers is selected, the duplicates in the papers are removed.
- **Stage 3:** With the list of papers obtained from the stage 2, each author separately selects those studies that the author feels is relevant to the research field after skimming through the title and abstract of the papers, resulting in a new list of papers from each of the authors. The resulting new lists of the selected papers by both authors are compared with each other. In case of any difference in the selected lists of both authors, the authors discussed among them for finding out the reasons for the variation in the selected list or consult a specialist and finally come up with a single list.
- **Stage 4:** From the list of papers obtained from stage 3, both authors will go through the full text of the research articles individually and finally come up with a final list of included primary studies of the research.

In order to measure the argument between the authors for the stage 3 and stage 4 the kappa co-efficient is calculated.
2.1.1.3.4 Study Quality Assessment

The quality of the selected study is determined in this step. Here the authors determine if the study addresses the following questions or not. Here the quality criterion is used to only determine the quality of the included studies but not exclude the selected primary study.
Table 3: Quality Criteria

<table>
<thead>
<tr>
<th>Quality Criterion</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
</tr>
<tr>
<td>Does the paper relate to the aim of our thesis?</td>
</tr>
<tr>
<td>Is the paper related to global software engineering field?</td>
</tr>
<tr>
<td>Is the paper relevant to follow-the-sun approach?</td>
</tr>
<tr>
<td>Does the paper discuss the challenges or mitigation strategies in relation to follow-the-sun approach?</td>
</tr>
</tbody>
</table>

2.1.1.3.5 Data Extraction

Data extraction process is used to record the accurate information obtained from the primary studies with minimum bias [9]. The data extraction form is used to collect all the necessary information and address the research questions and the study quality criteria [9]. The data extraction form used in this thesis is similar to the one mentioned in the paper by Smite et al [23] but, the form is modified as per our thesis design. The answers to the data extraction strategy are addressed in the chapter 3, in reporting the results section, where all the data is collected in a excel file. The following data extraction form is used in this research.

Table 4: Data Extraction Strategy

<table>
<thead>
<tr>
<th>Information regarding research Study/Article</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title of the article</td>
</tr>
<tr>
<td>Name of the author</td>
</tr>
</tbody>
</table>
| Publication | ▪ Journal article  
▪ Conference paper  
▪ Book  
▪ Article in press |
| Database | ▪ Engineering Village  
▪ IEEE Xplore  
▪ Springer Link  
▪ ISI Web of Knowledge  
▪ Science Direct  
▪ ACM Digital Library  
▪ Wiley Inter Science Journal Finder |
| Prime focus of the paper | ▪ GSD based  
▪ FTS based |
| Research method | ▪ Experiment  
▪ Survey  
▪ Interviews  
▪ Case study  
▪ Model proposed  
▪ Experience report |
2.1.2 Conducting the Review

2.1.2.1 Identification of research

After the search string has been formed and verified with the supervisor, to avoid publication bias and to perform the literature review in an unbiased way, Librarian at Blekinge Tekniska Högskola is consulted regarding the usage and access to the online databases. Zotero reference management software is used for managing the reference set obtained through the primary studies.

2.1.2.2 Selection of primary studies

The selection of the primary studies, explains in detail on how the steps that are mentioned in the planning of the review process i.e. in the sections 2.1.3.1.1 and 2.1.3.1.2 are carried out. A table that details the search string that is used for each of the database and also the number of papers included under each database is presented because each database has different way of performing the search strategy and also as the same search string was not possible to be used in all the databases, so the search string was modified according to the database. Please refer to table 33 in the appendix-C. From the table, it is clearly evident that the search string used in Springer Link and the ACM database are different from those used in the other databases. The reason for using a different search string in the Springer Link is that, there is a restriction in the usage of the word count in the search string. So to avoid this situation, we have modified the search string so that it covers the important terminology related to follow-the-sun approach and global software engineering. Where as in case of ACM database, whatever the search string might be the result displayed was: the papers on the following search string are not found and displayed all the papers that are available in the ACM database. So we have consulted the Librarian regarding this issue, the reply was as
follows, “there is some problem with the ACM database currently and it is showing wrong results”. And the librarian recommendation was to emphasize more on Engineering Village database where the papers from all the databases are listed there. But out of our curiosity we have tried different ways of entering the search string in the ACM database. At last we have figured out that the database is taking only one word as input and the combination of the search strings is not possible. As our study is mainly focused on follow-the-sun approach we have searched in the database for the papers that are relevant to follow-the-sun approach.

2.1.2.3 Finalized papers from primary study

In this section, finalized papers from the primary studies, a detailed description of study selection criteria is discussed. In the table 5, the results after applying the first stage of the inclusion criteria are shown.

Table 5: Inclusion Criteria - Stage 1

<table>
<thead>
<tr>
<th>Stage</th>
<th>Relevance</th>
<th>Selection Criteria</th>
<th>Papers Included</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>By search</td>
<td>- Contains the search string</td>
<td>141</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Only in English</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Publication date: till present</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Availability of the full text</td>
<td>88</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Remove duplicates</td>
<td>80</td>
</tr>
</tbody>
</table>

In the first stage of the selection criteria, both authors have done the search together. As the search string is pre-defined for all the databases as mentioned in table, so whoever enters the search string the result would be the same. So both the authors have done selection criteria in all the seven databases (Engineering village, IEEE Xplore, Springer Link, ISI web of knowledge, Science Direct, ACM and Wiley Inter Science Journal Finder). The papers are selected by applying the selection criteria as mentioned in the inclusion criteria table. The papers in stage 1 are refined by selecting only those papers that contain the search string, papers written in English, publication date: till 2011, contains the full text and finally removing the duplicates of the papers. By applying this selection criteria a total of 88 papers were found. From these 88 papers, after removing the duplicates a total of 80 papers were found. Reference management software Zotero was used to maintain the reference list of all the papers.

The second stage of the inclusion criteria discusses about how the papers are selected by the relevance of title and abstract. Table 6 below summarizes the second stage of the inclusion criteria.

Table 6: Inclusion Criteria - Stage 2

<table>
<thead>
<tr>
<th>Stage</th>
<th>Relevance</th>
<th>Selection Criteria</th>
<th>Papers Included</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>By title and abstract</td>
<td>- Remove (editorials, prefaces, discussions, comments, summary of tutorials)</td>
<td>44</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Related to global software development</td>
<td></td>
</tr>
</tbody>
</table>
In the second stage of the inclusion criteria, from the total of 80 papers, a further refinement is done by both authors individually and a total 44 papers are selected by reading through the title and the abstract of the papers. In this stage, the papers that are related to only global software engineering are selected as in the first stage the papers found may not be confined to the software engineering domain as the databases contain papers from all the fields of research. By reading the papers individually and discussing with each other, we have found 44 papers that are relevant to our study.

In the third stage of the inclusion criteria details about the selection criteria of papers that resulted after reading the full text is discussed. In the table 7 below, the synopsis of the stage 3 is detailed.

<table>
<thead>
<tr>
<th>Stage</th>
<th>Relevance</th>
<th>Selection Criteria</th>
<th>Papers Included</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>By full text</td>
<td>Focus mainly on FTS in GSD</td>
<td>16</td>
</tr>
</tbody>
</table>

In the final stage of selection of papers from the primary studies, by reading through the full text and based upon the selection criteria, a total of 16 papers were included in this study.

2.1.3 Reporting the review

The results of the systematic literature review are presented in the chapter 3.

2.1.3.1 Summary of systematic literature review

This section describes the overview of the entire process of the systematic literature review. Zotero reference management software was used for collection and maintenance of all the papers. The steps involved in the systematic literature review are:

- For the selection of the primary studies, the seven electronic databases searched are Engineering Village, IEEE Xplore, Springer Link, ISI Web of Knowledge, ScienceDirect, ACM digital library and Wiley Inter Science Journal Finder. The papers selected are the ones that are published till the time of the search (including 2011). As the search string is pre-defined for each of the database, the search for the papers in the databases was performed together by both authors. When the pre-defined search string was entered in their respective database a total of 141 papers were collected using the Zotero software.
- From the total of 141 papers, a total of 53 papers were removed as the access to the full text is not available. Now from the full text of 88 papers, 8 papers were duplicated. So after removing the duplicates a total of 80 papers are present. These 80 papers are selected by applying the inclusion criteria and the selection process is done together by both authors. All these papers are collected and maintained in the Zotero to help manage the referencing.
- In the next stage, the papers are selected based on the selection criteria of removing the editorials, prefaces, comments and summary of tutorials and also the papers that are not related to global software development are removed. Now while screening the papers based on title and abstract and by applying the selection criteria a total of 44 papers are found. This selection process was carried out individually by both authors. Author 1 has selected a total of 43 papers, whereas author selected a total of
42 papers. After discussing with each other about the difference in paper selection, a total of 44 papers are selected by both authors for the next stage of refinement process.

- In the final stage of refinement process, papers are selected that focus on the areas of follow-the-sun and global software development. From the list of 44 papers selected from the previous stage, both authors read these 44 papers individually and selected a total of 16 papers. In this stage, author 1 found that 24 papers are relevant to the research. Whereas author 2 finalized 23 papers that are relevant to the research. To remove the difference between the selections of articles, both authors discussed about difference in papers and decided that 16 papers out of 44 papers are relevant to the research area.

In selection of the primary studies in the stage 3 and stage 4 both authors have reviewed the papers individually and the kappa coefficient is calculated. Please refer Appendix-D for the calculation of the kappa coefficient. For easier and better understanding of how the review process is conducted, a pictorial representation of the review process is shown in the figure 8 below.
Table 8: Selected primary studies

<table>
<thead>
<tr>
<th>Paper ID</th>
<th>Selected primary studies</th>
</tr>
</thead>
<tbody>
<tr>
<td>[P1]</td>
<td>A. Gupta and S. Seshasai, “24-hour knowledge factory: Using Internet technology to leverage spatial and temporal separations,”</td>
</tr>
<tr>
<td>[P2]</td>
<td>A. Gupta, E. Mattarelli, S. Seshasai, and J. Broschak, “Use of collaborative technologies and knowledge sharing in co-located and distributed teams: Towards the 24-h knowledge factory,”</td>
</tr>
<tr>
<td>[P4]</td>
<td>Amar Gupta, Igor Crk and Rajdeep Bondade, &quot;Leveraging temporal and spatial separations with 24-hour knowledge factory paradigm”</td>
</tr>
</tbody>
</table>
2.2 Experiment

2.2.1 Motivation

Software development is a complex activity that takes lots of time, consists of many activities and documents that have to be written before the product is delivered, and involves many people. Many companies are trying to come up with new approaches to develop software in a systematic and disciplined way by improving the software process to deliver a product with better quality, lower development cost etc. In this process, researchers are coming up with new proposals that are supposed to provide many more advantages compared to the existing ones. It is important that the proposals are evaluated first to determine the actual effect, whether it works as promised or not, rather than using it directly. For this reason we have chosen to conduct an experiment to evaluate the proposal called follow-the-sun approach which promises of reducing the development time by 50% when compared to the traditional co-located approach.

Experiment aims at evaluating human-based approaches in a “systematic, disciplined, quantifiable and controlled way” [13]. An experiment allows the researcher to have control over the execution, measurement and ease of replication is high with an experiment. So in order to have full control over the execution process a controlled experiment is conducted and also due to cost, time and other constraints authors conducted the experiment in the university instead of selecting software professionals as subjects of the experiment. In our thesis we are comparing two software development approaches that is traditional co-located approach and follow-the-sun approach. Experiment is the best suitable option when there is more than one treatment to be compared. Hence the results obtained from the controlled experiment provide better knowledge for the organizations when implementing follow-the-sun approach in real time projects.

Experiment is not a one shot process but consists of a number of activities. The various stages in our experimental process are: initial experiment definition, initial experiment planning, refinement to the initial planning, final experiment definition, final experiment planning, experiment design and experiment execution. Each of these stages is detailed in the following sections to give a clear idea of the whole experimental process.

![Figure 9: Overview of the experimental process](image)

During the starting stages of this thesis, the author’s intension of conducting the experiment was modified to a greater extent due to the various constraints that the authors had come across. For this reason, in this chapter the authors explain in detail from the initial proposed plan till the final experiment that was executed in this thesis. Here in this chapter the term initial refers to what the authors have intended to perform in the thesis proposal and the term final refers to what the authors have actually carried out in this thesis.
2.2.2 Initial Experiment Definition

- **Object of the study**: The objects that were intended to be studied were: co-located scenario and follow-the-sun approach under two scenarios that is development-development scenario and development-testing scenario.
- **Purpose**: The purpose was to evaluate the development time of the three scenarios i.e. the co-located scenario, development-development scenario and development-testing scenario.
- **Quality focus**: The quality focus was the software development time of the three development scenarios.
- **Perspective**: The perspective was from the researcher’s point of view.
- **Context**: The experiment intended to be implemented using the Bachelor students of Blekinge Tekniska Högskola with Computer Science background, who were about to start their thesis work.

2.2.3 Initial Experiment Planning

- **Context selection**: The experiment was supposed to be conducted using students of Blekinge Tekniska Högskola. Keeping in mind with the cost and time constraints that would have incurred by having software professionals as subjects, the authors rather opted for conducting the experiment with students of Blekinge Tekniska Högskola as subjects of the experiment.
- **Hypothesis formulation**: The hypothesis for the experiment was:
  - **Null hypothesis, H₀**: Development time of co-located scenario is less than follow-the-sun approach with development-development and development-testing scenario. Also the development time of the follow-the-sun approach with development-development and development-testing scenario is not the same.
  - **Alternative hypothesis, H₁**: Development time of follow-the-sun approach with development-development and development-testing scenario is less than the co-located scenario. Also the development time of follow-the-sun approach with development-development and development-testing scenario is the same.
- **Variables selection**
  - **Independent variables**: The independent variables for the experiment were the three development scenarios (co-located scenario, follow-the-sun with development-development scenario and follow-the-sun with development-testing scenario), expertise of the participants and programming language.
  - **Dependent variables**: The dependent variable for the experiment was the development time or the speed of each of the three scenarios.
- **Selection of subjects**: The Bachelor students of computer science background who are about to do their thesis work at Blekinge Tekniska Högskola would have been selected
as the subjects for the experiment. The selection of the subjects would have been done through the pre-evaluation survey which would have tested their programming knowledge (on Java, Html, My SQL, and Tomcat- As the experiment is based on these programming languages) and experience of the subjects (if any).

2.2.3.1 Experimental setting

2.2.3.1.1 Experimental design

The subjects would have been selected and grouped into teams with 4 students in each team. The selected subjects would have been assigned to work under each of the three scenarios in such a way that the number of subjects is balanced among the three scenarios. And also the subject of the all the scenarios would have been given the same functionality to develop.

2.2.3.1.2 Experiment setup

- **Co-located scenario**: The co-located scenario would consist of four members, who would have worked for three hours a day. Where all the team members would have worked together at the same time and at the same place. The deadline for the co-located scenario would have been two days and after that they would have to deliver the project. The deadline of the task would have been extended, if in case they were unable to complete the task within the specified deadline. Extending the deadline of the task, would have enabled us to calculate the total amount of time taken by the students in completing the task.

  ![Figure 10: Experimental working of the co-located scenario](image)

- **Follow-the-sun scenario**: In follow-the-sun approach, each team would have consisted of 4 members, where the 4 members would have been divided into sub-team A and sub-team B. The sub-teams A and B would have consisted of two students each working on the same task. The follow-the-sun team would have been expected to complete the task using the same effort in man-hours as that of the co-located scenario, but the calendar duration has to be 1 day instead of the 2 days as that of the co-located scenario. The sub-team A and sub-team B would have been located in a different place, without allowing any type of synchronous communication between the both follow-the-sun sub-teams in
order to simulate follow-the-sun environment. The follow-the-sun sub-team A would have worked for three hours and then would have handed over the task to the sub-team B. The sub team B then would have started working for the next three hours from the point when the sub-team A has stopped working. The task would have been communicated between the sub-teams through the use of asynchronous communication tools. We have planned to artificially impose the deadline for the follow-the-sun team as 1 day which is exactly the half the duration when compared to the co-located scenario, which would have worked for two days, where the logic being that follow-the-sun approach reduces the development duration by 50%. The deadline of the task would have been extended if in case the follow-the-sun team was unable to complete the task within the specified deadline for determining by how much time the follow-the-sun team would have been faster or slower when compared with the co-located scenario.

Sub team A

3 hours

Sub team B

3 hours

3 hours

3 hours

1 Day Task

Follow-the-sun team would have worked under two different scenarios. The two scenarios are detailed below:

- **Development-Development Scenario:**

  In case of development-development scenario, the task was to be started by the follow-the-sun sub-team A and at the end of their time period the work was to be handed over to the follow-the-sun sub-team B who were present in a different place and in different time zone. The sub-team B would have continued the work from the point where the sub-team A has stopped working. This cycle would have continued until the fixed deadline of the task is met. Here both the follow-the-sun sub-teams would have worked continuously by handing over the work to each other after their working hours. The sub-teams would have been given a detailed requirements specification about how the task has to be carried out. Based on the requirement specifications the students would have been expected to develop the given application and also the developed code was to be unit tested by both the sub-team members in order to check if there were any errors in the developed code.

- **Development-Testing Scenario:**

  In case of development-testing scenario, the sub-team A would have been assigned the responsibility of coding whereas the sub-team B would have tested the code developed by
the sub-team A. In this scenario, follow-the-sun sub-team A would have developed the code and at the end of their working hours, the so far completed task would have been handed over to the follow-the-sun sub-team B for validation (testing). The responsibility of the follow-the-sun sub-team B would have been to verify whether the sub-team A was doing the task as per the designed requirements and then design the test cases for the developed code and finally check for the bugs based on the test cases designed. This cycle would have been continued until the fixed deadline of the task is met.

2.2.3.2 Instrumentation

- **Object:** We have designed a task (Please refer appendix A) for the teams to work on.
- **Tools:** The communication between the follow-the-sun team members would have been through HFS (HTTP File Server) where at the end of their working hours (time), the follow-the-sun sub-team A would have posted the so far completed task onto the HFS and follow-the-sun sub-team B would have downloaded the documents that were uploaded by sub-team A and continued to work on the task from the point where sub-team A have stopped working. The changes in the documents would have been traced by using CVS (Concurrent Versions System). Access rights to HFS and CVS would have been given accordingly to the students such that they wouldn’t have permission to access the files after their respective working hours.
- **Guidelines:** Guidelines with sample expected outcome would have been shown to the participants to give them a clear idea about the experiment design.

2.2.4 Refinements to the Initial Experiment Planning

While searching for the final year Bachelor students at Blekinge Tekniska Högskola, we contacted academic staff at Blekinge Tekniska Högskola to ask if they could help us in providing students for the experiment. In the process when the authors approached Associate professor Jürgen Börstler, he was kind and was willing to provide with the first year bachelor students who were doing Object Oriented Design course under his guidance. The authors had no other option at that point of time than to have these first year students as subjects for our experiment. So, the authors ended up having first year bachelor students as the subjects for the experiment based on the convince sampling instead of final year bachelor students.

Associate professor Börstler promised to dedicate an entire lab session for the experiment. When the authors detailed the aims and objectives of the experiment, Associate professor Börstler was kind enough to modify the existing lab session and designed the task to suit with the aims and objectives of the experiment. So, the authors had to drop out the plans of conducting the experimental task that the authors have designed. The authors felt that the experiment should be executed with only those students who were willing to participate in the experiment to achieve accurate results. When this thought was put forward to Assoc. prof. Börstler, he felt that if this choice is given to the students then there is a risk that not many would show up. Therefore experimental task was made a part of the course curriculum and all the students were expected to participate in the experiment.

There were certain modifications that had to be made to the initial experiment plan and design because of the following factors. Firstly, the students were not familiar with testing and have not done testing before. Secondly, the lab session was already scheduled for only four hours and the schedule could not be altered because of various issues such as resources, conflict with other courses etc. Thirdly, the use of CVS software was not possible due to lack of resources.
Keeping in mind of all these factors, the authors have redesigned the experiment accordingly. Firstly, the idea of measuring the development duration when using follow-the-sun approach under development-testing scenario had to be dropped out. So now development time of follow-the-sun approach using development-development scenario is only to be measured. For measuring the development duration in only development-development scenario, the authors have redesigned the experiment to have three different software development scenarios namely, follow-the-sun with overlap scenario, follow-the-sun without overlap scenario and co-located scenario. Here overlap and without overlap concept have been added to see whether development duration with overlap and without overlap scenarios would really make a difference. To come up with accurate results the authors decided to simulate two days with the available four hours rather than having just one iteration (one day transfer of task between the follow-the-sun sun teams). Assoc.prof. Börstler helped us in designing the experimental task in such a way that the students could come up with the required output within the deadline of four hours. During the initial experiment planning we thought of having CVS software so that it would be easier for the students to trace changes that have been done. But after discussing about the implementation of CVS software with the Assoc.prof. Börstler, the authors learned that computers in the lab where not high end system and are not compatible to install CVS software. So the authors had to drop out from the idea of tracking the student’s performance and decided to use pen drives to communicate the task between the follow-the-sun teams.

To analyse the background of the students and to determine if it there would be any factor that would have an impact on the experiment results, a pre-experiment survey form (attached in the appendix B) was posted onto Blekinge Tekniska Högskola It’s learning course page for the students to fill in. The students were also asked to form teams consisting of four members by themselves. Students were given one week time to respond to the pre-experiment survey and to form the groups for the experiment. Because of the low response rate the deadline for the survey was extended until the day before the experiment execution. Out of 70 students in the course only 40 students responded to the survey and from those 40 students only six teams (24 students) were formed team as only these students sent an email of their group names to the Assoc.prof. Börstler.

After analysing the results of the pre-experiment survey, the authors noticed that the teams could be grouped into two sets. There were three teams that had students who did Java course before while the other three teams had students who had done C programming language course earlier. The authors felt that the difference in the student’s background knowledge would have an impact on the final results because the experiment task is to be developed in Java programming language. When Assoc.prof. Börstler was asked about this, he analysed the performance of the previous lab session and informed that the students who have done Java course did perform a lot better when compared to the students who have done C programming language course. After Assoc.prof. Börstler confirmed the assumption of the impact on the results, the authors thought that it would be better to have only those students who have done Java programming course to participate in the experiment in order draw better conclusions. Keeping this in mind the authors decided to group the teams into two sets, namely, experienced and non-experienced students. Experienced students were those who have done java course while non-experienced students were those who have done C programming language course.

As the response rate to the pre-experiment survey was low, that is as only 6 student groups responded, among the 6 student groups 3 groups were categorized as the experienced set of students and the remaining three teams as the non-experience set of students. With the available students in hand we had to change the plans of assigning 2-3 teams to each of the three scenarios and finally decided to assign each set of experienced and non-experienced team to each of the three development scenarios i.e. follow-the-sun with overlap, follow-the-
sun with overlap and the co-located scenario. The assignment of the students to the different treatments is a random process.

Table 9: Assignment of subjects to different treatments

<table>
<thead>
<tr>
<th>Group Name</th>
<th>Background</th>
<th>Treatment name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group P</td>
<td>Experienced</td>
<td>Co-located scenario</td>
</tr>
<tr>
<td>Group Q</td>
<td>Experienced</td>
<td>Follow-the-sun with overlap scenario</td>
</tr>
<tr>
<td>Group R</td>
<td>Experienced</td>
<td>Follow-the-sun without overlap scenario</td>
</tr>
<tr>
<td>Group S</td>
<td>Non-experienced</td>
<td>Co-located scenario</td>
</tr>
<tr>
<td>Group T</td>
<td>Non-experienced</td>
<td>Follow-the-sun with overlap scenario</td>
</tr>
<tr>
<td>Group U</td>
<td>Non-experienced</td>
<td>Follow-the-sun without overlap scenario</td>
</tr>
</tbody>
</table>

2.2.5 Final Experiment Definition

- **Object of study**: The objects of the study are: co-located scenario, follow-the-sun with overlap scenario and follow-the-sun without overlap scenario are taken into account.
- **Purpose**: The purpose is to evaluate the three software development scenarios.
- **Quality focus**: The quality focus is the development time or the time-to-market and cost efficiency of the three scenarios.
- **Perspective**: The perspective of this experiment is from the researcher’s and developer’s point of view.
- **Context**: The experiment is conducted using Computer Science undergraduate’s students of Blekinge Tekniska Högskola. The experiment is conducted as a single object study.

2.2.5.1 Summary of the definition

Analyse the development duration of follow-the-sun and co-located approach for the purpose of evaluation with respect to the time-to-market and cost efficiency from the point of view of the researcher’s and the developer’s in the context of students in academia developing the code in three different scenarios, i.e. follow-the-sun with overlap, follow-the-sun without overlap and co-located scenario.
2.2.6 Final Experiment Planning

2.2.6.1 Context Selection

The subjects of the experiment were the undergraduate students of computer science background doing object oriented design course at Blekinge Tekniska Högskola and the selection of the students is specific as the students are from computer science background.

2.2.6.2 Hypothesis formulation

- **Null Hypothesis, \( H_0 \):** Time-to-market of co-located scenario is less by 50% when compared to follow-the-sun approach with overlap scenario and follow-the-sun approach without overlap scenario. And the development time of follow-the-sun approach with overlap scenario and follow-the-sun approach without overlap scenario is not the same.

- **Alternative hypothesis, \( H_1 \):** Time-to-market of follow-the-sun approach with overlap scenario and follow-the-sun approach without overlap scenario is less by 50% when compared to the co-located scenario. And the development time of follow-the-sun approach with overlap scenario and follow-the-sun approach without overlap scenario is the same.

2.2.6.3 Selection of variables

- **Independent variable:** Experience level of the students i.e. experienced set of students and the non-experienced set of students.

- **Dependent variable:** Time-to-market and cost efficiency.

Where,

**Time-to-market (T):** Is defined as the time taken by the subjects from the time of reading the experimental task until the subjects complete the task with the desired output.

**Cost efficiency (C):** It is defined as the amount of effort (number of students multiplied with development duration) put in by the subjects for completion of the task.

In case of follow-the-sun without overlap and co-located scenario the development duration of the subjects is calculated as:

\[
\text{Development duration (T_D)} = \text{Catching up time (T_{CT})} + \text{Programming time (T_{PT})} + \text{Reporting time (T_{RT})}
\]

In case of follow-the-sun with overlap scenario the development duration is calculated as:

\[
\text{Development duration (T_D)} = \text{Catching up time (T_{CT})} + \text{Programming time (T_{PT})} + \text{Communication time (T_{COT})} + \text{Reporting time (T_{RT})}
\]
Where,

**Catching up time** ($T_{CT}$): Catching up time is the total time spent by the students for analyzing the given task. That is the time spent by the students; from point where the student first starts reading the task until the student starts with the programming or coding. This time is termed as catching up time.

**Programming time** ($T_{PT}$): The total time spent by the students on programming the task is termed as programming time.

**Communication time** ($T_{COT}$): In case of follow-the-sun with overlap scenario, there is an overlap time of 4 minutes for the students to communicate with the other sub-group members using synchronous communication tools (such as use of cellular calls, voice calls, video calls etc. Here communication time is the amount of time utilized by the students for communication in the given 4 minutes.

**Reporting time** ($T_{RT}$): The total time spent by the students from the time of stopping with the programming part till the time when the so far completed task is handed over to the experiment leads (friends of the authors who were responsible for assisting the subjects and also for recording the development duration during the experiment execution, where each team is allocated with one experiment lead) in a pen drive. This usually consists of activities such as zipping all the files in folder etc.

Table 10: Selection of variables

<table>
<thead>
<tr>
<th>Variable Type</th>
<th>Variable Name</th>
<th>Value</th>
<th>Scale</th>
</tr>
</thead>
<tbody>
<tr>
<td>Independent Variable</td>
<td>Experience level of the students</td>
<td>Experienced students</td>
<td>Nominal Scale</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Non-experienced students</td>
<td></td>
</tr>
<tr>
<td>Dependent Variable</td>
<td>Time-to-market</td>
<td>$T$</td>
<td>Interval scale</td>
</tr>
<tr>
<td></td>
<td>Cost efficiency</td>
<td>$C$</td>
<td></td>
</tr>
<tr>
<td>Direct Measure</td>
<td>The amount of time spent in catching up ($T_{CT}$)</td>
<td>In minutes</td>
<td>Interval scale</td>
</tr>
<tr>
<td></td>
<td>The amount of time spent in programming the task ($T_{PT}$)</td>
<td>In minutes</td>
<td></td>
</tr>
<tr>
<td></td>
<td>The amount time spent in communication ($T_{COT}$)</td>
<td>In minutes</td>
<td></td>
</tr>
<tr>
<td></td>
<td>The amount of time spent in reporting the task ($T_{RT}$)</td>
<td>In minutes</td>
<td></td>
</tr>
<tr>
<td></td>
<td>The total development duration ($T_{D}$)</td>
<td>In minutes</td>
<td></td>
</tr>
</tbody>
</table>
2.2.6.4 Selection of the subjects

As mentioned in the section 2.2.4, the selection of the subjects was based on a pre-experiment survey which was posted in the student portal (It’s Learning) where the students were asked questions regarding their background knowledge, experience in Java programming and also to assess themselves on their Java programming skills. Please refer to the appendix A for the pre-experiment survey form. The students were asked to answer to the questions by ranking themselves among the four given options: mostly agree, mostly disagree, fully agree and fully disagree. The result of the pre-experiment survey showed that among 70 students in the class only 40 students responded to the pre-experiment survey. Among these 40 students, when the students are asked to rate themselves about their experienced in Java programming, 50% of the students answered as mostly agree, 13% of the students as fully agree and 37% of the students as mostly disagree. From these results, it can be clearly evident that there are two sets of students, one set consists of experienced students and another set consists of non-experienced students. The results of the experiment also revealed the students sustainability as, all the students from the Object Oriented Design course are aged between 19-20 years and all the students are Swedish (Nationality) students and also all of them are males. So, based on the number of the available students for the experiment and also based on the background knowledge of the students, the subjects are selected for the experiment.

The students are motivated well to participate in the experiment, by giving a short presentation about follow-the-sun approach and explaining in detail of how it is being practiced in the present organizations. Students are motivated by saying that by participating in such kind of experiments; they will have an exposure of working in real time and also that there will be small party for the participated students arranged by the authors. And also as the experiment task is part of their curriculum and as the output of the experiment reflects their grade, the students were motivated to participate in the experiment.

2.2.7 Experiment Design

According to Wohlin [13], there are three kinds of design principles that are applicable while designing an experiment. The three design principles are: randomization, blocking and balancing. The combinations of all the three design principles were used in this experiment.

- **Blocking:** In order to eliminate the undesirable effects on the experiment [13], the authors have looked into the background experience of the students prior to the experiment execution and divided the students into two groups, the experienced group and the non-experienced group and then assigned them randomly to the different treatments thereby blocking the effect of experience level of the students on the results of the experiment.

- **Randomization:** Only those students who responded to the pre-experiment survey and sent in their team names participated in the experiment, meaning that only interested students participated in the experiment rather than following any kind of selection criteria. Even while assigning to treatments, the experienced and non-experienced groups were assigned randomly. There was no criterion that was followed for assigning a treatment to a particular team of the experienced set and the non-experience set of students.

- **Balancing:** Subjects were equally assigned to all the treatments thereby balancing the experienced and the non-experienced set of students in all the three software development scenarios.
Table 11: Allocation of the subject groups to the treatments

<table>
<thead>
<tr>
<th>Factor</th>
<th>Group Name</th>
<th>Treatments</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Co-located scenario</td>
</tr>
<tr>
<td>Experienced Students</td>
<td>Group P</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>Group Q</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Group R</td>
<td></td>
</tr>
<tr>
<td>Non-Experienced</td>
<td>Group S</td>
<td></td>
</tr>
<tr>
<td>Students</td>
<td>Group T</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Group U</td>
<td></td>
</tr>
</tbody>
</table>

In this experiment, “Two factors with two treatments” design type is used. In the experiment there are two factors, one is experienced set of students and another is non-experienced set of students. The 12 subjects from each set experienced and non-experienced set of students are balanced and are assigned randomly to the three treatments.

Once the design type for the experiment is decided, next step is to decide what type of descriptive statistics tests that is to be performed on the data. But in the experiment, as there is only one control group for each treatment it is not possible to perform descriptive statistics test with the available data. For this reason, qualitative and quantitative analysis of the data is performed and the results of the experiment are represented by means of tables and figures.

2.2.7.1 Instrumentation

- **Objects**: The objects in the experiment were: recording the programming time, reporting time, catching up time and the communication time spent by each of the student groups and the task to be developed (kindly refer to appendix B). These objects of instruments were noted manually by the experiment leads. This data was used later on for calculating the time-to-market and the cost efficiency of the all the three treatments.

- **Guidelines**: All the student groups were provided with a different set of guidelines which they had to follow depending upon on their treatment. The guidelines were presented to the students during a presentation given to the students regarding the experiment design and the experiment execution. And also the guidelines were posted on the It’s learning, so that the students would have a clear idea of how the experiment is going to be conducted and also at the start of the experiment the guidelines were given to the students in the form of pamphlets. In the experiment task description, students were provided with a checklist of the deliverables that the students have to deliver to the experiment leads by the end of the experiment (kindly refer to appendix B).

- **Measurements**: Manual data forms were designed to collect the experimental data from the subjects and use the recorded data during experiment analysis (kindly refer to appendix B). A different type of data form was designed for each of the three scenarios.
i.e. follow-the-sun approach with overlap, follow-the-sun approach without overlap and the co-located approach which were handed over to the student to fill in at the end of their each day (Iteration). At the end of the experiment a feedback form was given to the students to brief about the overall experience of working in the experiment. All these templates are included in the appendix B.

2.2.7.2 Validity Evaluation

- **Internal Validity**

  In the experiment design, the threats to the internal validity are concerned with the design of the data collection forms. Data collection forms are used to get feedback from the subjects, which are later helpful during the experiment analysis. So the data collection forms were designed carefully by reading the literature and by consulting with the supervisor. Based on the feedback of the supervisor the forms were refined. Another internal validity threat is maturation. The students who participated in the experiment may lack concentration or enthusiasm which might affect the experiment results. To avoid this interval validity threat, the experiment is carried out with only interested subjects.

- **External Validity**

  In this thesis, the external validity is concerned regarding the generalizations of the experiment results. In this thesis, the main intention of the authors is to investigate on follow-the-sun approach but to not show the results of the experiment as a representation of the real world. Due to this reasons, the threat to external validity is reduced in this thesis. But as the experiment is conducted with less number of subjects there might be a threat that more challenges regarding the follow-the-sun approach might not captured.

- **Construct Validity**

  In this experiment, construct validity deals with the performance of the students on the outcome of the experiment. To overcome this problem, the goal of the experiment is clearly stated to the students and only the interested students were selected as the subjects of the experiment. Another construct validity threat is hypothesis guessing as the subjects of the experiment might try to figure out the purpose and result of the experiment in a negative way. To avoid this, the subjects were given a presentation for about a half an hour (refer appendix B) detailing the experimental setup and the expected output on the day before execution of the experiment. And also the subjects were also given time to interact with the authors in order to clarify their doubts.

  External disturbance is also one of the validity threats that might have an impact on the outcome of the experiment. To avoid this problem the students were placed in their regular labs and also the whole lab was booked only for the students of the object oriented design course.

- **Conclusion Validity**

  The conclusion validity threat in this experiment is related to data collected by the experiment leads. To avoid this problem, during the execution of the experiment the experiment leads were assisted by the authors in recording the details of the experiment. And also in order to mitigate this threat only the students (experiment leads) who have attended global software engineering course were selected as they will have a basic knowledge on the
background of the experiment. And also the day before the experiment the authors clearly explained about the execution of the experiment.

2.2.8 Experiment Execution

For each of the three scenarios: co-located scenario, follow-the-sun with overlap scenario and follow-the-sun without overlap scenario, we had one batch of experienced students and another batch of non-experienced students. Co-located team worked in a typical student manner with all the four team members working at the same place and at the same time. In case follow-the-sun approach each team consisting of four members were split two subgroups, (that is sub-group A and sub-group B) consisting of two members in each sub-group. Now each of the groups Q, R, T and U consists of sub-group A and sub-group B. In order to stimulate a distributed environment, the subjects of the sub-group A and sub-group B are placed in two different labs by working at different times. That is, all the sub-group A students were placed in the lab G, whereas all the sub-group B students were placed in lab H. It should be noted that sequential task distribution takes place between the sub-group A and sub-group B members in case of follow-the-sun with overlap scenario and follow-the-sun without overlap scenario. Here sequential distribution refers to, working on the same task from the point where the previous sub-group member has stopped working on it i.e. if sub-group A starts working on the task first, after the working hours of the sub-group A they handover the task to the sub-group B students. Now the sub-group B students work on the task from the point where the sub-group A has stopped working. This cycle continues till the fixed deadline of the task is met. It is crucial that the sub group members handover the work that has been done till then to the next sub group members to continue on the work. If they do not, then the other sub group members would be clueless and would be lost as they would not have anything to work on. We made sure during the experiment that the sub group members hand over the work to next sub group members by reminding them time and again during the last 15 minutes. As each sub group had an experiment lead, they made sure that the task is handed over to the next sub group without failure. The seating arrangements of the students are shown in the table below.

Table 12: Seating arrangements of the students

<table>
<thead>
<tr>
<th>Factor</th>
<th>Group Name</th>
<th>Sub-groups</th>
<th>Lab arrangements</th>
<th>Treatments</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Co-located scenario</td>
</tr>
<tr>
<td>Experienced</td>
<td>Group P</td>
<td></td>
<td>Lab G</td>
<td>X</td>
</tr>
<tr>
<td>Students</td>
<td>Group Q</td>
<td>Sub-group QA</td>
<td>Lab G</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>Sub-group QB</td>
<td>Lab H</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Group R</td>
<td>Sub-group RA</td>
<td>Lab G</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>Sub-group RB</td>
<td>Lab H</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Non-Experienced</td>
<td>Group S</td>
<td></td>
<td>Lab H</td>
<td>X</td>
</tr>
<tr>
<td>Students</td>
<td>Group T</td>
<td>Sub-group TA</td>
<td>Lab G</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>Sub-group TB</td>
<td>Lab H</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Group U</td>
<td>Sub-group UA</td>
<td>Lab G</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>Sub-group UB</td>
<td>Lab H</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
All the instruments for the experiment execution were made ready and the subjects of the experiment were also selected before the execution of the experiment. The selected student group names, timings and the seating arrangements of the selected students were posted in the It’s learning (student portal) along with the guidelines and the design of the experiment. The students were informed to look into the posted details and contact the authors through email if they have any doubts regarding the experiment before the start of the experiment execution.

Central repository for the students was the pen drive. Students after their working hours copied all the task related files onto the pen drive with a naming convention (their batch name and their working hours) that was followed and handed over it to the experiment leads.

2.2.8.1 Experimental setup

The three software development scenarios or treatments in this experiment are: co-located scenario, follow-the-sun with overlap scenario and follow-the-sun without overlap scenario. All the team members of the three scenarios were given the same functionality to develop. In the experiment, the development time required to complete the given task and the cost efficiency of each scenario is measured. The experiment is designed in such a way that two days were simulated in just four hours where sub-group A works during the day while sub-group B works during the night in case of follow-the-sun approach. The experimental set up for each of the three scenarios is detailed below.

- **Co-located scenario:** This team worked in a typical student manner i.e. working only during the day. That is the working hours for this scenario are: day one from 8:00 am to 9:03 am and on day two from 10:06 am to 11:09 am. The groups P and S worked in the co-located scenario, where group P was placed in lab G and the group S was placed in lab H.


- **FTS with overlap:** In this scenario the four members in the team were split into two sub-groups: sub-group A and sub-group B where each sub-group consists of two members. On the day one, sub-group A starts working on the task first during the day from 8:00 am and 9:00 am, where from 8:00 am – 8:56 am student’s work on the given task and in last 4 minutes that is from 8:56 am – 9:00 am students can communicate with the sub-group B students. Now the sub-group B works during the night on the day one from 9:00 am till 9:56 am and after their working hours from 9:56 am till 10:00 am they handover the task to the sub-group A and communicate with them. On day two, the working hours for the sub-group A during the day are from 10:00 am till 10:56 am and from 10:56 am till 11:00 am they can communicate with the sub-group B students. And during the night, the working hours for the sub-group B are from 11:00 am and 12:00 pm where in the last iteration students were expected to deliver the task. Both the sub-group members were placed in different rooms to simulate the geographical distribution between the team members and synchronous communications such as phones, instant messenger etc were allowed to use only during the overlap time. Here the follow-the-sun sub-groups work continuously by handing over the work to each other after their working hours.
Sub-group A  
(Lab G)  
8:00 am - 8:56 am  
10:00 am-
10:56 am  
4 minutes overlap 4 minutes overlap 4 minutes overlap

Sub-group B  
(Lab H)  
9:00 am - 9:56 am  
11:00 am-
12:00 pm

Figure 13: Working of the follow-the-sun with overlap scenario

- **FTS without overlap:** In this scenario the four members in the team were split into two sub teams: sub-group A and sub-group B where each sub-group consists of two members. On day one, sub-group A works during the day from 8:00 am and 9:03 am, while the sub-group B members work during the night on day one from 9:03 am till 10:06 am. On day two, the working hours for the sub-group A during the day are from 10:06 am till 11:09 am. And during the night, the working hours for the sub-group B are from 11:09 am and 12:12 pm, where in the last iteration students are expected to deliver the task. Both the sub-group members were placed in different rooms to simulate the geographical distribution between the team members where the sub-groups are not allowed to use any kind of synchronous communication such as face-to-face meeting, phone, instant messenger etc.

From the above experimental setup it can be observed that the working of the follow-the-sun without overlap scenario is same as follow-the-sun with overlap scenario, but the only difference is that in case follow-the-sun without overlap there is no direct communication between the sub-group members. But the sub-group members of the follow-the-sun without overlap scenario can use asynchronous communication medium such as email, where they can send mail to the other sub-group members regarding the progress of the task, clarification regarding the code that has been developed by the other sub-group members etc. It such be noted that the response to the mail would only come when the sub-group members are back to work on the next day as it is night for the other sub-group members when they have sent an email.
The subjects of the follow-the-sun with overlap scenario and follow-the-sun without overlap scenario were required to perform the task using same effort (in man-hours) in one day (that is between 8:00 am to 10:00 am in case of follow-the-sun with overlap scenario and between 8:00 am and 10:06 am in case of follow-the-sun without overlap scenario) instead of the two days that the co-located team takes (that is 8:00 am to 9:03 am on day one and 10:06 am to 11:09 am on the second day). This 50% reduction in development time when using follow-the-sun approach is evident only in theory. There are only few empirical studies that have noticed reduction in the development time at the first place and also reduction was not 50%, but rather less than that. Keeping this in mind we have simulated equal number of days for all the three scenarios to note if there is a reduction in development time using follow-the-sun approach and if so, by how much percent the reduction in the development time is achieved.

From the design of the experimental setup, it can be observed that the duration (length of the day) was different for follow-the-sun with overlap scenario when compared to co-located scenario and follow-the-sun without overlap scenario as the reason being that the effort is maintained to be constant for the all the three development scenarios. The calculation of the effort is explained in detail in the experiment results section of the chapter 3.

At the end of the experiment a survey is conducted with the help of the feedback forms given to the students for determining the challenges that the students faced during the experiment execution.

2.2.8.1.1 Distribution of development time for the three scenarios

Here, the distribution of the development time for the above discussed co-located scenario, follow-the-sun with overlap scenario and follow-the-sun without overlap scenario is discussed. The figure 15 below, represents the distribution of the development time per day in minutes, where 126 minutes represents 1 day i.e. 8 hours in a day is represented as 63 minutes. The distribution of the development time is based on the factor of effort, where the amount of the effort put into the experiment by the three scenarios is the same.

<table>
<thead>
<tr>
<th></th>
<th>1\textsuperscript{st} Day</th>
<th>2\textsuperscript{nd} Day</th>
</tr>
</thead>
<tbody>
<tr>
<td>Co-located scenario</td>
<td>63 Minutes</td>
<td>63 Minutes</td>
</tr>
<tr>
<td>FTS with overlap scenario</td>
<td>56 Minutes 4</td>
<td>4  56 Minutes 4</td>
</tr>
<tr>
<td>FTS without overlap scenario</td>
<td>63 Minutes</td>
<td>63 Minutes</td>
</tr>
</tbody>
</table>

Figure 15: Distribution of development duration
In the figure 15, the red line represents the total duration given to the students for performing the given task. The line in green represents the overlapping time between the sub-group members where, the subjects communicate with each other. And the numbers in-between the overlapping lines represents the given overlapping minutes i.e. 4 minutes of overlapping time for each iteration.

In case of co-located scenario, the development duration is 63 minutes per each day, as the subjects of the co-located scenario work only for 8 hours a day, which is half the amount of development duration when compared to the follow-the-sun scenario. The effort of the co-located scenario is (63 minutes * 4 students) + (63 minutes * 4 students) = 504 minutes.

In case of follow-the-sun with overlap scenario, the development duration is 60 minutes, where during the first 56 minutes students work on the given task and during their overlapping time of 4 minutes, they handover the task to the next sub-group members. From the figure 15, it is evident that the subjects are not given the complete 24-hours to work on the task as the reason being that, the effort is maintained as constant for all of the three scenarios. In case of follow-the-sun with overlap scenario, as there is an overlapping time of 4 minutes both sub-group members interact with each other. So the effort during the overlapping time is 4 minutes * 2 students (these 2 students are from sub-group A) + 4 minutes * 2 students (these 2 students are from sub-group B) = 16 minutes. So the total effort of the follow-the-sun with overlap scenario is [(56 minutes * 2 students + 4 minutes * 4 students) + (56 minutes * 2 students + 4 minutes * 4 students) + (56 minutes * 2 students + 4 minutes * 4 students) + (60 minutes * 2 students) = 504 minutes, which is the same as the effort of the co-located scenario.

Similarly in case of follow-the-sun without overlap scenario, the subjects utilize the complete in 24-hours in a day by working in 8-hours shift basis. The effort of the follow-the-sun without overlap scenario is [(63 minutes * 2 students) + (63 minutes * 2 students) + (63 minutes * 2 students) + (63 minutes * 2 students)] = 504 minutes.

<table>
<thead>
<tr>
<th></th>
<th>1st Day</th>
<th>2nd Day</th>
<th>Effort</th>
</tr>
</thead>
<tbody>
<tr>
<td>Co-located scenario</td>
<td>63 Min</td>
<td>63 Min</td>
<td>(63 minutes<em>4 students) + (63 minutes</em>4 students) = 504 minutes</td>
</tr>
<tr>
<td>FTS with overlap scenario</td>
<td>56 Min</td>
<td>56 Min</td>
<td>[(56 minutes<em>2 students)+4 minutes</em>4 students] + [(56 minutes<em>2 students)+4 minutes</em>4 students] = 504 minutes</td>
</tr>
<tr>
<td>FTS without overlap scenario</td>
<td>63 Min</td>
<td>63 Min</td>
<td>(63 minutes<em>2 students) + (63 minutes</em>2 students) + (63 minutes*2 students) = 504 minutes</td>
</tr>
</tbody>
</table>

Figure 16: Effort for the three scenarios

2.2.9 Summary

After the refinement to the initial experiment planning, the final experiment was designed and defined as

Analyse the development duration of follow-the-sun and co-located approach for the purpose of evaluation
with respect to the *time-to-market and cost efficiency*
from the point of view of the researcher’s and the developer’s
in the context of students in academia developing the code in three different scenarios, i.e. *follow-the-sun with overlap, follow-the-sun without overlap and co-located scenario.*

Based on the definition of the experiment, the variables for the experiment are selected. The experience level of the students is the only independent variable in the experiment with three different treatments i.e. co-located scenario, follow-the-sun with overlap scenario and follow-the-sun without overlap scenario. Independent variable affects the dependent variables i.e. the time-to-market and the cost efficiency including the direct and the indirect measures. The direct measures in the experiment are the time-to-market and cost efficiency which are calculated based on the indirect measures. The goal of the experiment is designed based on the outcome of the dependent variables.

The final experiment is executed with the students of Blekinge Tekniska Högskola, who are currently pursuing first year of their bachelor program in computer science and engineering field. A total of 24 students are selected randomly for the experiment. Based on the background of the students, the students are classified into two groups i.e. the experienced set of students and the non-experienced set of students.

The instruments for the experiment includes the task specifications, guidelines of the experiment and the various data collections forms that are given to the students during the experiment execution as well as after the execution of the experiment.

The experiment can be termed as a type of balanced with one factor, (experience level of the students) with three treatments (co-located scenario, follow-the-sun with overlap scenario and follow-the-sun without overlap scenario) study. The students are balanced and randomly assigned to all the three treatments.

The experiment is executed in two different labs for four hours in a single day. The experiment is executed in a similar manner in both the laboratories respective of their treatments. During the working hours, the students performed the task as they were intended to do and after their working hours the task was handed over to the experiment leads in the pen drive.

The experiment planning and execution is completed and the experimental data is collected as per the plan of the experiment. The results of the collected experimental data are presented in the next chapter 3.
3 RESULTS AND ANALYSIS

Results and analysis section is divided into two sections, literature review results and results of the empirical research. In literature review results section, the characteristics of the included papers in this thesis and the results of the narrative synthesis are discussed. In results of the empirical research section, the results obtained from the experiment are discussed.

3.1 Systematic Literature Review Results

3.1.1 Included Studies

Seven databases: Engineering Village, IEEE Xplore, Springer Link, ISI Web of Knowledge, Wiley Inter Science Journal Finder, ScienceDirect, ACM Digital Library were searched by the authors, 141 papers were found but finally 16 papers were included based on the relevance of our research. The figure 17 below represents the count of the number of papers found in each of the selected database.

![Included papers from each database](image)

Figure 17: Number of included papers from each database

The figures represent the actual count of the included papers that are obtained in each of the database, without removing the duplicates. During the process of selecting the primary studies it was observed that same papers were obtained in different database. So for the better understanding of the results the authors of the papers kept the count of the included papers undistributed.

3.1.1.2 Year wise distribution of papers

In the figure 18 below, year wise distribution of the included papers is represented. A total of 16 papers that are published till the time of search i.e. till April 2011, that are found relevant to the research field are represented. Among the 8 papers i.e. 50% papers completely address the implementation of follow-the-sun approach, whereas the remaining 50% of the papers discuss about the global software development by emphasizing on follow-the-sun approach i.e. partially discussing about follow-the-sun approach. From the table below, it is clearly...
evident that not much research is done in this field before 2003, which illustrates the fact that global software development is a 21st century trend [23]. It can be observed from Figure 16 that, from 2003 there is gradual increase in the research field that is 81% of published papers are between the years 2006-2010. Note that the statistics for 2011 can be incomplete since the search was conducted in April 2011 and the most recent articles could have not yet been added to the databases for that reason, maybe we could not find any papers that are published in 2011.

Figure 18: Number of published papers per year

3.1.1.3 Distribution of research methods

The various research methods we have found in the selected studies are case studies, experiments, interviews and models proposed. Among the various research methods, the most common one is the models proposed. From the figure below it can be noted that in 37% of the papers models were proposed in order to find out the optimal favourable conditions for the implementation of follow-the-sun approach. The authors of other studies reported results from case studies, findings based on their experience in industry or obtained through surveys or interviews.

Among the 16 papers, 21% of the papers are based on experiments. There are various kinds of experiments such as controlled experiments, laboratory experiments and industrial experiments. And some of the experiments are based on the model proposed. As the researchers started finding it difficult to find out the optimal conditions for the implementation of follow-the-sun strategy, they started with mathematical models [P5] [P13]. The remaining 42% of the papers are distributed among the case studies (16%) and studies based on interviews (21%). And there was one paper which is based on experience report.
3.1.1.4 Subjects of investigation

A subject of investigation refers to whether the research is carried out in an industry or in an academic setting. From the figure below it can be noticed that among the 16 papers that are included for the study, there is almost equal distribution of the papers in both industrial and academic setting. 7 papers reported studies that are based on academic background, which can be concluded that the researchers are putting their efforts in finding out the different scenarios with the goal of reducing the development time. The main purpose of the follow-the-sun approach is to reduce the development time for the organization, so even though researchers experiment in finding out the different scenarios ultimately the organizations have to adapt to the new process. From the results it was observed that the organizations are facing lot of challenges in implementing distributed development and also as it requires a huge capital for implementing a new process, not many organizations are opting for distributed development. From the figure below, it can be noticed from the 16 papers, only 9 of the papers research was carried out in industrial background. This lack of implementation of distributed development in industries might be because of various reasons such as difficulty in finding out the right locations, coordination, communication and control challenges faced etc. So for this reason, the researchers are conducting various experiments [P5], [P6] and proposing mathematical models [P5], [P13] to find out the ideal scenario with the goal of reducing the development time.
3.1.1.5 Narrative synthesis results

Follow-the-sun is a global software development method, in which handover of work takes place every day from one development site to another development site that are many time zones apart. There by reducing the development time by 50% if there are two sites involved, by 67% if there are three sites involved and soon on. Resulting in reduced time to market thereby achieving higher competitive edge over the others especially for industries where there is a sense of urgency as products get out-dated quickly. This reduction in development time is rather perceived as an assumed benefit because of availability of only few documented evidences available of achieving the reduction in development time. So here we review the literature to determine if follow-the-sun approach works or not, determine the challenges involved while using follow-the-sun approach, when is it best to use, to achieve the benefit of reduced development time? Etc. Table 13 gives a brief overview of the included studies.

Table 13: Overview of the included studies

<table>
<thead>
<tr>
<th>Paper no</th>
<th>Methodology</th>
<th>Context</th>
<th>The main contribution of the paper</th>
</tr>
</thead>
<tbody>
<tr>
<td>[P1]</td>
<td>Model proposed</td>
<td>Academia</td>
<td>-Models and implements an internet-based prototype system that uses Web-based interaction approach along with a unique data model that collects and stores design rationale and history from stakeholders and workers. -The model also reflects some of the important characteristics of FTS.</td>
</tr>
<tr>
<td>[P2]</td>
<td>Experiment</td>
<td>Industry</td>
<td>-It has been noticed that both FTS and co-located teams differ in the use of technology and knowledge sharing processes but not in efficiency and quality.</td>
</tr>
<tr>
<td>Page</td>
<td>Section</td>
<td>Source</td>
<td>Text</td>
</tr>
<tr>
<td>------</td>
<td>---------</td>
<td>--------</td>
<td>------</td>
</tr>
<tr>
<td>P3</td>
<td>Interview</td>
<td>Industry</td>
<td>-It has been found that benefit of FTS can be achieved with high process maturity.</td>
</tr>
<tr>
<td>P4</td>
<td>Model proposed</td>
<td>Academia</td>
<td>-Data representation models for FTS such as composite persona model, CPro process, MultiMind and EchoEdit have been discussed. -These models keep track of reason for each decision along with an explanation of the code and generate a schedule that helps in achieving maximum performance when using FTS.</td>
</tr>
<tr>
<td>P5</td>
<td>Model proposed</td>
<td>Academia</td>
<td>A routing model that for FTS that calculates optimal deployment routes. Where different routes are prioritized based on a set of parameters.</td>
</tr>
<tr>
<td>P6</td>
<td>Experiment</td>
<td>Academia</td>
<td>-Both FTS and co-located teams completed the work in the same time. -Developing small pieces of code, testing, and storing into a common repository is important for handing-off the work in FTS</td>
</tr>
<tr>
<td>P7</td>
<td>Interview</td>
<td>Industry</td>
<td>-Benefits of GSD have been discussed. -Time saving using FTS has been discussed where company realizes the benefit while other company dismisses the benefit.</td>
</tr>
<tr>
<td>P8</td>
<td>Case study</td>
<td>Industry</td>
<td>-Benefits of GSD and the extent of their realization have been discussed. -Investigation of time saving through FTS has been concluded to be a mythical benefit.</td>
</tr>
<tr>
<td>P9</td>
<td>Interview</td>
<td>Industry</td>
<td>-Coping strategies to overcome the challenges associated with GSD have been determined. Where suggestions for improved performance while using FTS approach been discussed.</td>
</tr>
<tr>
<td>P10</td>
<td>Case study</td>
<td>Industry</td>
<td>-GSD challenges and solution to mitigate the challenges have been discussed. -While investigating solutions to overcome temporal distance, it has been found that FTS is suitable for support and defect resolution activities but not for development activities.</td>
</tr>
<tr>
<td>P11</td>
<td>Interview, model proposed</td>
<td>Industry, Academia</td>
<td>-A model based on communication theory is proposed. -The model has been applied for FTS and hence challenges and suggestions have been drawn from the results of the model.</td>
</tr>
<tr>
<td>[P12]</td>
<td>Experience report</td>
<td>Industry</td>
<td>-Challenges, solutions to overcome, lessons learnt and issues while using FTS has been discussed.</td>
</tr>
<tr>
<td>[P13]</td>
<td>Experiment</td>
<td>Industry and academia</td>
<td>-Task allocation algorithm has been proposed. -Applying the algorithm it has been observed that project duration is reduced and an improvement in resource utilization is observed when using FTS.</td>
</tr>
<tr>
<td>[P14]</td>
<td>Experiment</td>
<td>Academia</td>
<td>-Participant’s perception of working speed and accuracy using FTS is worse when compared to the actual working speed and accuracy.</td>
</tr>
<tr>
<td>[P15]</td>
<td>Model proposed</td>
<td>Academia</td>
<td>-A GSD model that allocates tasks that can be used with FTS for selecting the optimal site based on availability of efficient resources thereby improving the performance and also handling uncertainties has been proposed.</td>
</tr>
<tr>
<td>[P16]</td>
<td>Model proposed</td>
<td>Academia</td>
<td>-A hybrid simulation GSD model that is used to exam GSD projects has been discussed. -By using the model it is observed that it is better to avoid using FTS as it requires communication and coordination between sites resulting in higher effort and longer duration. -FTS works best with three development sites.</td>
</tr>
</tbody>
</table>

After studying each of the articles in detail we identify six themes that have been discussed often in most of the articles. The six themes are:

- Challenges
- Tools for enhancing the performance when using follow-the-sun approach
- Suggestions for improved performance
- Comparison of follow-the-sun with co-located software development
- Optimal number of sites when using follow-the-sun approach
- Phases of software development lifecycle that are best suitable while using follow-the-sun approach
Table 14: Illustrates various themes of the research discussed in each of the studies

<table>
<thead>
<tr>
<th>Paper No</th>
<th>Challenges</th>
<th>Tools for enhancing the performance</th>
<th>Suggestions for improved performance</th>
<th>Comparisons of FTS with colocated software development</th>
<th>Optimal number of sites</th>
<th>Phases of software development lifecycle that are best suitable while using FTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>[P1]</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[P2]</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[P3]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[P4]</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[P5]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[P6]</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>[P7]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>[P8]</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>[P9]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>[P10]</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>[P11]</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>[P12]</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[P13]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>[P14]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>[P15]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>[P16]</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

Now, let’s look at each of the themes individually and learn what each of the article, that has discussed the theme, has to say about the theme.

3.1.1.5.1 Challenges or obstacles

The challenges that arise while using follow-the-sun approach are identified in the studies. The table gives the brief overview of the challenges that have been discussed and certain details about the studies have been mentioned to help reader understand under what circumstances that particular challenges identified.
From the above table 15 it is evident that, of the sixteen studies that have been included in the literature review eight studies have discussed about the challenges when using follow-
the-sun approach. Various studies have identified various challenges. We now discuss the various challenges mentioned in each of the studies in detail.

In paper [P6] it has been said that handing-off the work in progress makes follow-the-sun difficult to implement. As work done by one team may not be understood by another team and requires further clarification [P6] due to the lack of understanding, in which case an entire day may be wasted as the site that has developed that particular part of the code has already gone home when clarification is needed. In other cases re-work [P6] needs to be done because of misunderstandings resulting in wastage of time also termed as vulnerability costs. Where as in paper [P16] it has been said that follow-the-sun strategy requires high effort, results in poor quality of the outcome, increased distribution overhead and declined productivity causing follow-the-sun approach not to be used by many of them. It has been noticed that the primary reason for the high effort is because of hand-off [P16] of the task that takes place between the development sites due to the distribution of the work among more than one site. Based on a case study conducted in three companies [P8] it has been concluded that one of the main disadvantages with follow-the-sun approach is that it is not suitable for all the phases of the software development lifecycle such as for the development phase [P10]. Paper [P8] which is also based on a case study in industrial setting has found that follow-the-sun approach leads to many problems due to temporal dislocation. The fact that it is not suitable for the all the phases of the software development and due to the delayed response time [P8] makes follow-the-sun approach not so popular.

Paper [P12] which is an experience report has reported a number of obstacles such as communication issues, cultural difference, temporal difference, difference in technical background and technical platform between the development sites involved which causes a number of problems as a result hindering the use of follow-the-sun approach. It has been observed that even though English was a common language for all the team members who were from different countries, there was much more to language which caused many misunderstandings due to communication problems [P12], also issues arose due to unknown differences or clashes in social morays and cultural values causing frustration and conflicts among the team members from different regions [P12]. In addition there was a minimum overlap in working hours between the team members in study, causing the team members to communicate after their working hours affecting the family commitments and personal schedules. It has been also noted that the team members from different regions had different programming background that is they were proficient in a certain programming language while the members from different region where not good in that particular language and also programming styles differed causing mismatch among the team members. The technical environment while using follow-the-sun approach is quite different when compared to the conventional co-located approach. Follow-the-sun approach demands that the sites involved have a shared source control system, shared test servers, shared development environment and shared technical platform hence company needs to build the new, matured technical infrastructure [P12]. While using follow-the-sun approach the development sites have to share the ownership of code with other team members from other regions and understand that even the other team has the right to make changes, which is new and takes time for team members to get used. It has been noted in the study that people even left the company because of the fact that ownership of the code has to be shared with other team members. It is needed that the follow-the-sun teams follow a formal development process that is an uninterrupted weekly plan needs to developed and followed, which created chaotic situation among most of them, as not many companies are used to follow a formal development process rather developing software in an ad-hoc fashion.

It has been reported in paper [P11] that continuous use of follow-the-sun approach is difficult because of the high dependencies and the communication and co-ordination that are demanded. It has been said that vulnerability costs that incurs as a result of clarification and rework [P11] hampers the use of follow-the-sun approach. Paper [P2] which is based on a
quasi-experiment conducted in IBM for a period of one year with the developers and leads with 5-20 years of experience it has been observed that problems arise with follow-the-sun approach mainly due to the frequent hand-off that is being carried out. Paper [P1] which is based on an experiment has observed that there are number of challenges while using follow-the-sun approach. Since people from different countries work together on a project, issues such as cultural mismatches arise as each culture has some rules that are not written but are implicit. This problem could be minimized by employing a person who reconciles dissimilar rules on a continuous basis [P1] to overcome this problem but this increases the overhead of employing new employees. It has been observed that follow-the-sun environment requires that the global managers possess skill sets for distributed management [P1] of globally dispersed teams that are not taught even in prestigious degree programs. Hence, finding a global manager with the required skill set is a challenge in itself. Many countries restrict on the type of data and intellectual property that is exported to other country [P1] which is a challenge faced.

From the above it is clear that various studies have mentioned different challenges. This variation may be because of the various reasons such as the context in which they have been studied, the methodology adapted, the duration of the study, participants involved and experience level of the participants or because of the difference in study setting of each of the study. Of the seven studies that have discussed the challenges four studies [P8], [P10], [P2], [P12] have been studied in an industrial setting while the other three studies [P1], [P11], [P16] have drawn their conclusions while doing research in an academia setting while one study [P6] does not mention how it has identified the challenges even though the particular study discusses about an experiment it does not mention anywhere that the challenges mentioned where observed while conducting the experiment. Papers [P11], [P16] presents the challenges based on global software development model proposed in each of them. [P11] proposes a model based on coordination theory it draws conclusions based on randomly taken inputs so one cannot say that the results can be generalized for all situations. [P16] discusses about a hybrid simulation model of the software development process, however the author does not discuss about the validity of the result hence one cannot determine how valid the results are [P1] is based on an experiment conducted with students and faculty of three universities Wroclaw University of Technology in Poland, University of Arizona, Tucson and University of Technology, Sydney, Australia but nothing much is told about the experiment setting, the task on which the participants worked, or the validity of the results. While paper [P2] draws its conclusions based on an experiment that is conducted at IBM for one year with experienced employees, with experience ranging from 5 to 20 years it has been said the main drawback of the experiment is that the results are drawn based on single organization study, the small sample size and the experiment did not focus on issues such as external knowledge sharing, sub team dynamics, conflict and trust which could have had a great influence on the experiment result, which does not allow the results to be generalized. Papers [P8], [P10] have drawn conclusions based on case studies which have been conducted in industries. Paper [P10] is based on the interviews conducted with project manager at Fidelity, manager at HP, project leader at Intel all though the validity of the results haven’t been discussed. Whereas [P8] is based on the feedback that is collected from site managers, project managers, a project architect, team leads, software engineers and technical support staff of International Semiconductor, Global Investments Inc., and Digital solutions the validity of the results have not been discussed, whereas [P12] has made conclusions based on the experience of WDS Global who has incorporated 24 x 5 around-the-clock extreme programming for one year. There may be certain challenges that may not have been identified or some addition challenges that have been mentioned because of the various reasons discussed above. If not all the challenges some of the challenges may be observed when using follow-the-sun approach depending upon the context and the extent to which follow-the-sun approach is applied.
Summary

Here research question RQ 1.1 is addressed. A number of challenges or obstacles are faced while using follow-the-sun approach. The most cited obstacles are the hand-off [P2], [P6], clarification cost [P6], [P11], not suitable for all the phases [P8], [P10], delayed response [P6], [P8], [P11], time zone difference [P8], [P12], communication issues [P1], [P11], [P12], cultural difference [P1], [P12] and re-work cost [P6], [P11]. While other obstacles are misunderstandings that arise [P6], high effort [P16], poor quality [P16], increased overhead distribution [P16], decreased productivity [P16], technical background difference [P12], demands matured infrastructure [P12] and formal development process [P12], high dependencies [P12], co-ordination [P11], demands employees to poses different skill set [P1] and restriction on the data exported [P1].

3.1.1.5.2 Tools for enhancing the performance

A number of tools have been cited in the literature that promise to make follow-the-sun approach more convenient and easy to use while at the same time promises to improve the performance. The table 16 gives a summary of the tools that have been discussed in the literature.

<table>
<thead>
<tr>
<th>Paper no</th>
<th>Tools</th>
<th>Methodology</th>
</tr>
</thead>
</table>
| [P1]     | - Collaboration software  
- Wiki  
- Web-day  
- Version control  
- Extreme programming  
- Internet security  
- Information system tools such as UMEA  
- Internet-based prototype proposed | Model proposed |
| [P4]     | - Data representation tools such as composite persona and CPro  
- Efficient hand-off tools such as MultiMind and EchoEdit | Model proposed |
| [P5]     | - Model for selecting development location is proposed | Model proposed |
| [P13]    | - Task scheduling algorithm proposed | Experiment |
| [P15]    | - Resource allocation model proposed | Model proposed |

Five studies have discussed tools to be used while using follow-the-sun approach that could help improve the performance and make follow-the-sun approach easy to implement. We now discuss what each study has mentioned in detail.

In paper [P5], a routing model has been discussed that helps in making strategic decision of where to establish development locations in follow-the-sun setting. The model calculates optimal deployment routes by taking as input the number of locations, current locations, amount of overlap in the office hours, whether overlap needs to be exact or not and language
that is to be spoken by all the team members, and by accessing demographic data about the location which is stored in database. The model then generates a number of possible routes that are prioritized based on a set of parameters such as, based on the percentage of match with requested overlapping hours and total number of developers present in that particular combination of locations. The author discusses about the validity of the model, where he says there are a number of limitation with the model such as the database containing the demographic information about the location is not so accurate, the time zones contained in the database are based on the setting for 2008 but these may differ from year to year and also for countries with daylight saving the mean of summer and winter times is taken, the figures for number of people speaking a particular language are inaccurate, number of developers present in a particular country keeps on changing and never fixed this may affect the results, the model assumes that all the countries follow a standard static nine hours a day from Monday to Friday nor does it consider the breaks such as lunch breaks etc. These limitations may have profound impact on the final output. But the model can be used to at least initially to get an idea of which sites to be used. The result obtained could be further validated.

While in paper [P15], a resource allocation model that efficiently minimizes the time using follow-the-sun approach has been proposed. The model allocates work to the site containing efficient resources by analysing the constraints and its impact on the product development. The model allocates work in such a way that it does not affect the estimated time or product release. The model also helps improve the performance by allocating tasks to the appropriate optimal resource available, helping one to meet the deadlines without any slippages. The estimated deadline takes care of the problems that are a cause of the slippage. The model has been tested on two examples, it was seen that there was almost 15 % to 40 % improvement. It has been said that there is a need to conduct a simulation experiment to further determine the efficiency of the model.

A task scheduling algorithm that minimizes the completion time of a project has been proposed in paper [P13]. The model assumes that there would be three follow-the-sun teams working from three different time zones. The model takes as input a task graph and resource table and generates a project schedule that is of minimal length for a given project. The model has been applied to few synthetic projects and on two real life projects. For the two real time projects the schedule while using follow-the-sun approach has been generated using the model, it has been noted that there was a substantial reduction of about 10% to 20% in completion time of a project and also an improvement in resource utilization of the project.

In paper [P1] technologies such as collaboration software, wiki, WEB-DAV, version control, extreme programming have been discussed that help make follow-the-sun approach much easier to implement. Collaboration software helps share information to be transferred easily rather than having to store the information locally and then transfer throw e-mail. While wiki helps make changes and reflects the changes instantly on the online version as well making things simpler. WEB-DAV is a tool for web authorizing which is apt for distributed environment. On the other hand, version control systems such as RSYNC, CVS, and Subversion reflects changes made at one site to be shown at the other site as well, which is very necessary while using follow-the-sun approach. While extreme programming allows to practice shared ownership that is allowing changes to be made by all the team members spread across the global which is very much the case for follow-the-sun approach and internet security helps the data to transmit among different location in a secured manner. It is said that information system tools and methodologies such as UMEA [P1] to be used with follow-the-sun approach. These tools automatically organize data objects that are related to a particular task into collections so that knowledge that is spread over multiple locations is easily accessible to all the team members, at any point in time and from anywhere across the globe. As a result time is not wasted, to go through the database and also helps locate task specific information on a daily basis. Here in this paper [P1] the author also discusses about
a new framework which uses a web-based interactive approach along with a unique data model that collects and stores the design rationale and history from the stakeholders and workers. It primarily consists of two prototype system that uses internet technology as the key enabler. The two prototypes are KNOWFACT and MULTIMIND. KNOWFACT prototype is one that automatically captures design rationale and history by using web-based form. Which helps in sharing structured knowledge using the internet with stakeholders who are spread across the world. Whereas MULTIMIND prototype shares tasks with composite personae that is team members using internet.

It has been said in paper [P4] that knowledge management and transfer are critical for the success of 24 hour knowledge factory model and discusses about a number of data representation tools. In paper [P4] the author discusses about data representation tools such as composite persona and CPro that can be used for efficient knowledge management and transfer, while it is said that tools like MultiMind and EchoEdit to be used for validating design principles and for efficient hand off. There by reducing the communication overhead and increasing productivity. It is said that these tools help maintain a database to store the reasons behind each decision, providing an explanation of the code artefacts and helps develop schedule for maximizing the performance.

It is that evident number of tools have been mentioned and proposed in the literature that can be used while using follow-the-sun approach to increase the performance as well as help the work to be carried out on a continuous fashion. But the validity of tools for all kinds of situations is questionable. For instance the model proposed in paper [P13] does not consider the communication overhead or the overlapping working hours hence making it not applicable for situations where overlap in working hours persists. Whereas the validity of the tools and models that has been discussed in paper [P1], [P4] is not discussed making it difficult to comment on how far the tools are applicable and help achieve the promised advantages. But literature proposes number of tools which can help the practitioners willing to take up follow-the-sun approach to use the tools to achieve better performance and motivate the others to take up follow-the-sun approach. Tools should be selected based on the applicability of the tool for that particular situation.

**Summary**

Here research question RQ 1.2 is addressed. Various tools have been suggested to be used while using FTS approach for improved performance. The tools that have been discussed in literature are resource allocation model [P15], [P5], task scheduling algorithm [P13], collaboration software [P1], wiki [P1], web-day [P1], version control [P1], extreme programming [P1], internet security [P1], information system tools [P1], internet-based prototype has been proposed [P1], data representation tools [P4], and hand-off tools [P4].

3.1.1.5.3 Suggestions for improvement performance

Various suggestions based on the experiences and from the lessons learnt have been discussed in literature. Table 17 summarizes the main findings.

<table>
<thead>
<tr>
<th>Paper no</th>
<th>Suggestions</th>
<th>Methodology</th>
<th>Context</th>
</tr>
</thead>
</table>
| [P1]     | - Tasks have to be semi-structured  
- Required to increase knowledge transfer and communication between the team members | Experiment, unclear and model proposed | Academia |
<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>P2</strong></td>
<td>- Face to face meeting</td>
<td>Experiment</td>
</tr>
<tr>
<td><strong>P3</strong></td>
<td>- High process maturity</td>
<td>Interview</td>
</tr>
</tbody>
</table>
| **P4** | - Sharing and transferring knowledge efficiently by using efficient hand-off tools  
- Making information accessible by structuring  
- Using reliable and user friendly technology for transferring data  
- By reusing the design the development time can be reduced  
- Minimize interaction by dividing the tasks into smaller components | Model proposed | Academia |
| **P6** | - Training software engineers on communication, cooperation and collaboration skills.  
- Using agile approaches | Experiment | Academia |
| **P8** | - By having only two development sites participating time zone issues can be avoided | Case study | Industry |
| **P9** | - Using simple, unambiguous, well defined tasks | Interview | Industry |
| **P11** | - Handing off at the right time  
- By having clearly defined requirements  
- Using good communication tools | Interview, model proposed | Industry |
| **P12** | - Hire a coach  
- Boot camp  
- By using rotating guru concept  
- Remote pairing  
- By having shared common environment | Experience report | Industry |
| **P14** | - By having fewer sites with good performance decrease development time and increase accuracy of the results. | Experiment | Academia |

From the above table 17 it is clear that there are number of studies that have suggested techniques for improved performance while using follow-the-sun approach, based on their experience with follow-the-sun approach. Let’s look into to each of the suggestions in detail to understand what works well while using follow-the-sun approach.

The authors [P6] based on their experience of using follow-the-sun approach have proposed certain suggestions that could help improve the performance while using follow-the-sun approach. It is said that if software engineers are given training on cooperation and collaboration skills [P6] then the miscommunication among the team members would be mitigated. Issues such as reward balancing [P6] between individuals and team have to be looked into. The authors even suggest that agile approaches [P6] to be used along with follow-the-sun approach rather than the waterfall model, iterative methods (e.g. RUP) etc.
The reasons being that in each and every agile iteration all the software development activities (i.e. define, design, code, test, integration) take place for small scope. Testing these small portions leads to a duration reduction of at least 12.5% and more [P6]. Agile even insists on having the integration green (i.e. all the test pass) at the end of the day there by supporting daily hand-offs [P6] which is the bases of follow-the-sun approach.

It was noted from the experiment results that fastest two site workflow is same as the working speed of slowest four sites [P14], meaning that one has to consider maximizing the performance in terms of working speed and working accuracy for less number of sites rather than having clutter of sites. Semiconductor Company feels it is better not to distribute work across more than two time zones to pacify the effect of time zone separation [P6].

While in paper [P12] which is based on one year of experience of working in follow-the-sun environment have proposed a number of techniques. The authors say by hiring a coach [P12], who is an expert in a particular field would train the team members in that particular field so that, the not so proficient members can cope up with the other team members located at a different time zone. It has been mentioned that constant communication and synchronization among the coaches of different locations [P12] help to address and resolve cross regional issues or issues in general. So it has been suggested that the coaches interact on a daily basis after the daily handover, meet on weekly bases via teleconference and face to face meeting every quarter for two weeks to resolve complex issues and planning for the quarter. In order to build trust among team members who are located at different locations who do not get to meet, the authors proposes boot camp [P12]. Where the team members meet physically and spend time in a boot camp fashion so that members have a chance to see the others work there by building trust relationships and common work practices. Another technique that has been suggested was rotating guru [P12]. Where a senior team member is sent to a remote location for setting up infrastructure, for initial training, mentoring and also identify the reasons for the difficulties such as long delays faced in the remote location. It has been said that by hiring an educator [P12] who would help prepare and train both, the team and the business to use this new programming technique which is a dramatic change from the existing process. It has been said that there will be a need to change the infrastructure [P12] as follow-the-sun approach requires different kind of infrastructure when compared to traditional approaches. It requires that the network bandwidth between regions to be high in order to allow large amount of data and traffic to be passed through the network to avoid long delays and to continuously integrate servers located at various locations around the world. Another approach that has been proposed is remote pairing [P12]. Using colocated approach one can take advantage of the informal discussions during coffee or lunch breaks which is not possible using follow-the-sun approach. Remote pairing facilitates pairing sessions where two developers from different regions would be paired using VNC, teleconferencing and common IDE to collaborate on ideas and share experiences and mitigate misunderstandings. Yet another technique is round the world program [P12]. In this technique each member from a location is sent to work at a different site in the follow-the-sun approach for a few weeks so it would help to understand the background, culture and to build trust among the members in that region. High level process changes without the consultation of other team would create confusion and frustration affecting the collaboration and trust between the teams. In order to overcome this process change [P12] processes could be used. Process changes have to be agreed upon by all the coaches. Processes have to be flexible in order to fit the cultures and climates of the different sites. It has been suggested to use extreme programming long with follow-the-sun approach, with a shared codebase [P12]. Extreme programming requires that there is frequent communication among the team members. Extreme programming helps maintain common values, knowledge and also inspires one for open communication and collaboration thereby increasing communication across sites. One can improve the performance of follow-the-sun approach having shared environment [P12]. Using extreme programming, pair machines are setup by using the same tools and environmental setup across regions’, pairing machines in such a way that they are
same. Helping one to minimize the configuration time and adjust accordingly when
developers rotate pairs to different machines. One could replace face-to-face communication
with video conferencing during the daily handover and during customer requirement
gathering sessions [P12] in order to achieve the same effective communication that is
achieved when using face to face communication.

One of the recommendations is to have very high process maturity [P3]. In an interview
conducted with software professionals it has been advised that teams must coordinate
properly [P11]. It has been said tasks have to be handed off a bit before overlap time [P11] in
order to minimize delay costs that would incur otherwise. It is also essential that the tasks are
requested at the right time to avoid delay [P11]. It has been said in paper [P11] that,
requirements to be specified in a clear fashion and the communication to be clear in order to
reduce the clarification cost especially during the overlapping hours and rework cost.

It has been mentioned during the interview that it is important to have at least one face to
face [P2] meeting so that team members can introduce each other and incorporate a social
component to the relationship and also advocated the use of phone calls so that team
members can discuss issues in a better way.

While in study [P9], which is based on an interview it has been found that follow-the-sun
approach worked for tasks with little dependency or when task dependencies can be
managed by easily programming and automating in the direction of the workflow. It is said
that by having task which are simple, unambiguous and well defined tasks [P9] one can
achieve the benefits of follow-the-sun approach.

In has been mentioned that is good to have work is semi structured while using follow-the-
sun approach [P1]. Follow-the-sun approach uses vertical decomposition of tasks that means
multiple workers operating on the task in a sequential manner, which requires high cohesion
mode collaboration that means more knowledge transfer and more communication between
the collaborating parties. So it is necessary to have information in a structured way [P1], so
that the team can grasp easily the work done by the other team. It is good to have that the
technology that is used for transferring information to be user friendly, reliable over network
irregularities and provide a framework for the user to structure the information in a
convenient way [P1]. Design reuse [P1] is a good option while using follow-the-sun
approach so that time is spent on producing core information rather than spending on
documenting the decisions. Follow-the-sun approach would achieve better results if used for
activities that can be reduced to components [P1], where knowledge is digitized there by
allowing different individuals to work on the components with minimum interaction with the
other team located many time zones apart and the work in progress be handed off from one
development site to another with high speed, high efficiency, low overhead and low cost. In
paper [P4] the author confesses that knowledge has to be transferred and shared efficiently
for improved follow-the-sun performance.

Summary

Here research question RQ 1.3 is being discussed. A number of suggestions have been given
for improved performance while using follow-the-sun approach. They are training software
ingeniers on communication, cooperation and collaboration skills [P6]; reward balancing
[P6]; using agile approaches [P6]; having fewer sites with good performance [P8], [P14];
hiring a coach [P12]; boot camping [P12]; by having rotating guru [P12]; by practicing
remote pairing [P12]; by having shared common environment [P12]; by having high process
maturity [P3]; coordinating properly [P11]; handning off at the right time [P11]; by having
clearly stated requirements [P11]; by using good communication tools [P11]; by having
face-to-face meeting [P2]; with simple, unambiguous, well defined [P9] and semi-structured
[P1] tasks; by increasing knowledge transfer and communication [P1]; by sharing and
transferring knowledge efficiently [P4]; by having information in a structured format [P4];
by using reliable and user friendly technology for transferring of data [P4]; by reusing
design [P4]; minimizing interaction by dividing the tasks into smaller components [P4]. It is
possible that by following the above suggestions if not all, at least few based on the context
and applicability for that particular situation; it is a possibility that there is an increase in
performance using follow-the-sun approach.

3.1.1.5.4 Comparison of follow-the-sun approach with the co-located approach

A number of studies have compared follow-the-sun approach with co-located approach. The
table 18 below gives a brief overview of each of the findings.

Table 18: Comparison of follow-the-sun approach with co-located software development on
various metrics

<table>
<thead>
<tr>
<th>Paper no.</th>
<th>Metric</th>
<th>Results</th>
<th>Task</th>
<th>Context</th>
<th>Methodology</th>
</tr>
</thead>
<tbody>
<tr>
<td>[P2]</td>
<td>- Development duration</td>
<td>-Development duration and quality was same.</td>
<td>Software project for IBM</td>
<td>Industry</td>
<td>Experiment</td>
</tr>
<tr>
<td></td>
<td>- Quality</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Technology use</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Knowledge sharing</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Communication pattern</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[P6]</td>
<td>- Development duration</td>
<td>-10% reduction using FTS</td>
<td>To develop simulator of</td>
<td>Academia</td>
<td>Experiment</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>processes and threads</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>scheduler in Unix</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[P13]</td>
<td>- Development duration</td>
<td>-10%-20% reduction in development duration using FTS</td>
<td>Two software development</td>
<td>Industry</td>
<td>Experiment</td>
</tr>
<tr>
<td></td>
<td>- Resource utilization</td>
<td></td>
<td>projects of Infosys</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Technologies limited</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>and some synthetic</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>projects</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[P16]</td>
<td>- Development duration</td>
<td>-50% longer using FTS</td>
<td>NA</td>
<td>Academia</td>
<td>Model proposed</td>
</tr>
<tr>
<td></td>
<td>- Effort</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Quality</td>
<td>-70% higher effort using FTS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>-Quality much worse using FTS</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Follow-the-sun approach has been compared with co-located approach in number of studies. Let’s look into each of the study to determine the metrics on which the comparison was made, how accurate the obtained results are etc.

Paper [P6] is based on an experiment which took place at a university for almost fourteen weeks with fifteen experienced computer science and electrical engineering students aged between 20 to 30 years, where the time taken to develop a simulator of processes and threads scheduler in UNIX operating system by follow-the-sun team working in an agile environment and co-located team were compared. The experiment took place as part of practicum for a course in software engineering and the performance in this reflected on the final grading of the course hence participants were motivated to perform well. Follow-the-sun team was forced to follow strict communication rules in order to simulate the time zone difference. The follow-the-sun team in the middle of the experiment requested a liaison so that communication and co-ordination between the sub-teams would be improved. It has been observed that there was an approximate 10% reduction in the development duration when using follow-the-sun approach as opposed to co-located approach. The author feels that this reduction in development time may be because of the use of the agile methods. The experiment had a number of limitations such as small sample, the measures were imperfect, students were not select but rather self-selected, and agile had not been fully implemented.

In paper [P16] a hybrid simulation model of the software development process that is used to examine global software development projects has been discussed. The model has been run for thirty times, for follow-the-sun setting as well as co-located setting. It has been observed that follow-the-sun strategy performed much worse than single site on all the three performance measures. The effort was 70% higher, duration was 50% longer and the quality was worse when using follow-the-sun approach using two development sites when compared to co-located approach. It has been said that the increase in effort and duration for follow-the-sun approach is because of the high communication and coordination that is demanded between the sites that are spread across the globe.

In paper [P13], the development time and the resource utilization of follow-the-sun and co-located approaches by using the task allocation algorithm have been generated and compared. The model has been applied for few synthetic projects and two real time project of Infosys technology limited. It has been observed that there is 10% to 20% reduction in development duration and 54% to 78% improvement in resource utilization when using follow-the-sun approach as opposed to co-located approach.

Paper [P2] is based on a quasi-experiment that has been conducted for a period of one year in IBM with its employees with 5 to 20 years of experience. Both co-located and follow-the-sun teams differ in the use of technology and knowledge sharing processes but not in quality and efficiency or the speed. Both the teams used written communication for group discussion but follow-the-sun members used it to share broad information while co-located team used it for informal interactions. Both the teams differed in the way they shared knowledge. Follow-the-sun team used formal source codification process based on documenting decisions where the history could be re-used when needed and used formal systems for knowledge capture. While co-located teams used personalization strategy based on informal communication to share knowledge and did not document the decisions that were made, as a result more than one person worked on each code element. The distributed teams used meetings for short term issues and to assign tasks while co-located team used it for status update of the coding process and to discuss about the future direction that has to be taken while the short term issues and task assignments issues were discussed during informal face to face meet, where the decisions were not documented. It has been said that there are a number of limitations in the study such as the results were based on an experiment that was conducted in a single organization and the number of respondents.
From the above discussion it is clear that number of studies have compared follow-the-sun approach with co-located approach on various metrics. But is observed that one metric has been primarily studied by all the studies that is, the development duration. If given a glance at the result there is a huge variation in the results, certain studies have noticed a reduction [P6], [P13] in development time as in theory, but the reduction is not as much as 50% as mentioned in theory while some have noticed an increase in the development duration [P16] while the other study has noticed the development time of follow-the-sun approach as well as the co-located approach remains the same [P2]. This variation in results may be accounted due to various reasons such as because of the participants involved, experience level of the participants, study setting etc. Of the studies, [P13] and [P6] have noticed a reduction in development duration of 10 % and 10% -20% respectively this reduction may be because of the tools that have been used by the studies. [P6] has used agile method which may be a reason for the positive result while [P13] uses a task allocation algorithm which may have helped allocate tasks in an efficient fashion helping to achieve a reduction in development time when compared to co-located approach. While [P16] which is based on a hybrid simulation model has noticed an increase in development duration when using follow-the-sun approach the validity of the results are questionable as the model has been tested on a random example and not on a real project hence it is difficult to generalize the results. While [P2] has concluded that both follow-the-sun approach and co-located approach take same duration while the study is single organization study, has very less number of participants making it difficult to make valid conclusion. The other metrics is quality that has been studied [P2], [P16] extensively. [P2] and [P16], both the studies have noticed varying results [P16] has noticed a deterioration in quality while [P2] says the quality is same for both co-located and follow-the-sun approach this may be because of the fact that the participants in this particular study had 5-20 years of experience which may have helped to maintain the quality.

Summary

Here research question RQ 1.4 is answered. A number of studies have compared FTS and co-located approach. Both approaches differ on a number of parameters such as development duration, effort, quality, resource utilization, technology use and knowledge sharing and communication pattern. It has been noted in [P6], [P13] that the development duration reduces by almost by 10-20% using follow-the-sun approach while in paper [P16] noted an increase in project duration by 50% using follow-the-sun approach while another paper [P2] has noted a completely different result, that both co-located and follow-the-sun approaches take the same duration. With this kind of contradictory conclusions in hand, it is difficult to come up with a conclusion if follow-the-sun approach reduces or increases the development duration when compared to co-located approach. Follow-the-sun approach increases effort and resource utilization by 70 % [P16] and 54% -78% [P2] respectively when compared to the co-located approach. Quality has been studied by [P2] and [P16], paper [P16] has concluded that quality decreases with follow-the-sun approach while [P2] has noted that quality remains same for both the approaches. Both the teams differed in the communication pattern, knowledge sharing, technology use, resource utilization [P2] etc. Written communication is used by follow-the-sun members to share broad information while co-located team used it for informal interactions [P2]. Follow-the-sun team uses formal source codification process while co-located team uses personalization strategy for knowledge sharing [P2]. Follow-the-sun teams discussed about short term issues and to assign tasks while co-located team used formal meetings for status update and to discuss the future direction to be taken while short term and task assignments issues where discussed during informal meetings [P2].
3.1.1.5.5 Optimal number of development sites

A number of studies have commented on the optimal number of sites while using follow-the-sun approach based on various parameters such as development time, quality, accuracy etc. The table 19 below summarizes the details of the findings along with the details of the studies.

Table 19: Optimal number of development sites

<table>
<thead>
<tr>
<th>Paper no</th>
<th>Investigated no. of sites</th>
<th>Recommended no. of sites</th>
<th>Parameter</th>
<th>Methodology</th>
<th>Context</th>
</tr>
</thead>
<tbody>
<tr>
<td>[P8]</td>
<td>_</td>
<td>2</td>
<td>- Temporal distance manageable</td>
<td>Case study</td>
<td>Industry</td>
</tr>
<tr>
<td>[P10]</td>
<td>_</td>
<td>2</td>
<td>- Temporal distance manageable</td>
<td>Case study</td>
<td>Industry</td>
</tr>
<tr>
<td>[P14]</td>
<td>2,3,4</td>
<td>2</td>
<td>- Development time - Accuracy</td>
<td>Experiment</td>
<td>Academia</td>
</tr>
<tr>
<td>[P16]</td>
<td>2,3</td>
<td>3</td>
<td>- Development time</td>
<td>Model proposed</td>
<td>Academia</td>
</tr>
</tbody>
</table>

The optimal number of sites while using follow-the-sun approach has been discussed in a number of studies. Most of the studies in our thesis recommend two sites as the optimal number while it was noticed that one study recommended that three sites to be the ideal number while using follow-the-sun approach. Let’s look at each of the study in detail to determine on what bases the studies have stated that optimal number of sites.

Paper [P14] draws conclusions based on controlled experiment which is conducted for three days with computer science graduates. In this study the effect of number of sites on the working speed and accuracy are determined by varying the number of sites between two to four. It has been observed in the experiment that it is better to have two sites with good performance in terms of working speed and working accuracy instead of going through the clutter of adding sites in different time zones.

In paper [P16] using a hybrid simulation model, follow-the-sun approach has been tested on the bases of effort required, development duration and quality by having two as well as three development sites. It has been noted that follow-the-sun approach performed much worse on all the three measurements when compared to co-located approach but it has been noted follow-the-sun approach with two sites performed much worse than follow-the-sun approach.
with three sites. Hence it has been concluded that if the priority is to reduce the cycle time, follow-the-sun with three sites may be preferable rather than follow-the-sun approach with two sites. It is extremely difficult, if not impossible for a global software development project to be successful using follow-the-sun approach with only two development sites.

Paper [P10] draws conclusions based on a case study conducted with industrial practitioners. A Project manager at Intel mentions that temporal distance can be made manageable by dividing the work between no more than two sites, while paper [P8] which is also based on a case study conducted in industry. It has been specified by an employee at semiconductor that work should not be distributed across more than two time zones to pacify the effect of time zone separation.

Almost all the papers [P8], [P10], [P14] have stated two sites as the optimal number of sites while using follow-the-sun approach to enjoy the advantage of improved working speed [P14], working accuracy [P14], and to make things manageable [P8], [P10]. While study [P16] states follow-the-sun with three sites would help achieve the reduction in the cycle time. The variation may be due to number of reasons. Paper [P8] and [P10] have stated two sites as optimal for managing time zone difference while paper [P14] and [P16] both draw conclusion for improved speed but both have made contradictory conclusions. [P14] states three sites while [P16] feels by having two sites the cycle time can be reduced. This variation may be due to the methodological difference. In [P16] conclusions are drawn based on the results generated from the hybrid simulation model by taking a random example. The example taken may not represent a real world situation and there is a possibility that results may not be accurate or the hybrid model may itself not be accurate because author have not tested its accuracy. While [P14] makes conclusions based on an experiment which has a number of limitations such as full day was simulated in just four minutes and the results have been drawn based on the task of manipulating or developing maps and not based on the development of a real programming task hence it is difficult to say if the results could be valid even for software development tasks. So it is difficult to decide if the reduction in development duration is possible with two sites or by using three sites.

Summary

Here research question RQ 1.5 is addressed. If the objective is to makes time zone separation manageable or to improve the working accuracy then two sites is preferable [P8], [P10], [P14]. Instead, if working speed is the main objective then two sites [P14] or three sites [P16] should be considered.

3.1.1.5.6 Recommended phases of software development lifecycle that are best suitable while using follow-the-sun approach

It has been noted that follow-the-sun approach does not work for all the phases but works only for few phases. Number of studies have drawn conclusion on this aspect. The table 20 below summarizes the findings of the various studies.
Table 20: Recommended phases of software development lifecycle to implement follow-the-sun approach

<table>
<thead>
<tr>
<th>Paper no</th>
<th>Recommended Phases</th>
<th>Not Recommended Phases</th>
<th>Methodology</th>
<th>Context</th>
</tr>
</thead>
<tbody>
<tr>
<td>[P6]</td>
<td>- Testing phase</td>
<td>_</td>
<td>Not clear</td>
<td>Academia</td>
</tr>
<tr>
<td></td>
<td>- Prototyping phase</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[P7]</td>
<td>- Maintenance phase</td>
<td>_</td>
<td>Interview</td>
<td>Industry</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[P8]</td>
<td>- Testing phase</td>
<td>_</td>
<td>Case study</td>
<td>Industry</td>
</tr>
<tr>
<td></td>
<td>- Defect resolution</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Maintenance phase</td>
<td>_</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[P10]</td>
<td>- Defect resolution</td>
<td>- Development phase</td>
<td>Case study</td>
<td>Industry</td>
</tr>
<tr>
<td></td>
<td>- Maintenance phase</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[P11]</td>
<td>- Testing phase</td>
<td>_</td>
<td>Interview and model proposed</td>
<td>Industry</td>
</tr>
<tr>
<td></td>
<td>- Defect resolution</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

In paper [P6] it is mentioned that follow-the-sun approach works well for certain phases in particular such as testing phase and prototyping phase. In testing phase one team searches for bugs, documents them. While the other team accesses the documents and works on it. The reason given by the authors for testing to work well using follow-the-sun approach is because in testing phase hand-off is structured, granular and does not lead to miscommunication in the presence of trained staff.

In paper [P10], while interviewing a project manager at Fidelity mentioned that follow-the-sun approach does not work for the development phase but works for defect resolution and support. It has been said in paper [P8] that International Semiconductor Company based on its experience has found that hand-off is possible during the testing phase while Global Investments Inc, company feels that it can be best used for defect resolution and support tasks.

In paper [P7] it has been said that benefit of follow-the-sun approach can be achieved during the maintenance phase of the project. Here the tasks are comparatively small and can be handed to one person at a time. From the interview it has been evident that follow-the-sun approach is more suitable for low granular tasks such as bug fixing or call-centre activity (e.g. technical support) [P11]. Based on the coordination model it has been said that less equivocal tasks (structured tasks such as testing and error fixing) may be better suited for follow-the-sun approach [P11].

**Summary**

Here research question RQ 1.6 is being addressed. From the above it is clear that all the studies have recommended follow-the-sun approach for less equivocal phases of the software development such as for testing [P8], [P11], [P6], defect resolution [P8], [P10], [P11] and in maintenance phase [P8], [P10], [P7] in particular. While another phase that looks promising is prototyping [P6].
3.1.1.6 Literature review summary

Follow-the-sun approach is not recommended for development phase even though theory promises of a reduction in development duration by 50% when using follow-the-sun approach. After reading the included papers in our literature review we understood that follow-the-sun approach is suitable for low granular task such as testing, maintenance etc, as follow-the-sun approach is built on the concept of hand-off hence not suitable for high granular task such as development, as such tasks demand a high communication and coordination which is difficult when using follow-the-sun approach hence may be not recommended for development. But there was only one study that experimented follow-the-sun approach in the development phase. Even we where curious to find if it is really true that follow-the-sun does not work well for development phase so we designed the experiment taking development related tasks to test while using follow-the-sun approach.
3.2 Experiment Results

In this section the results obtained from the experiment are presented. In this discussion, the time-to-market and the cost efficiency of the experienced students and non-experienced students are addressed separately as the non-experienced students did not complete the given task within the specified deadline in all the three development scenarios. So with this results in hand as it is difficult to measure time-to-market and the cost efficiency of the experienced students in comparison to the non-experienced students, so the analysis for the experienced and non-experienced students is performed separately.

3.2.1 Results of the experienced students

3.2.1.1 Measuring the time-to-market of the software development scenarios

As defined in the section 2.2.6.3, time-to-market is defined as the total time taken by the subjects i.e. from the time of reading the experimental task until the subjects completed the task with the desired output. As per the experiment design, the students were expected to deliver two modules within the specified deadline. So the time-to-market is calculated from the time the students started their work till the students executed the two modules. Based on the given development time for the students, as in figure 15, the time utilized by the students is represented in the below figure 22. The figures below indicate the total time in minutes utilized by the subjects, but for the actual duration during each day in minutes i.e. the exact start and the end time, please refer to the table 27 in the appendix B.

![Figure 22: Utilized development duration](image)

In the figure 22, the line in blue or the numbers represented in blue represents the unutilized time by the students from the given development time. In case of co-located and follow-the-sun with overlap scenario, during the 1st iteration (morning session) the number 10 denotes as, the students were late to the lab and started the experiment late by 10 minutes. Similarly, in the other situations the blue line represents as the students handed over the task to the experiment leads ahead of the given duration. And the red line indicates the given duration whereas the green line represents the overlapping time.
In comparing the results of the co-located scenario and the follow-the-sun with overlap scenario, it can be observed that the co-located members as well as the follow-the-sun with overlap members took more than the given amount of duration for completion of the task. The co-located members were expected to complete the task within 2 days and the follow-the-sun members within 1 day, as the reason being that follow-the-sun approach takes half the amount of duration when compared to the co-located scenario. Even though the subjects of both scenarios took extra amount of time for completion of the given task, it can be observed that the follow-the-sun with overlap subjects completed the task faster when compared to co-located members. The follow-the-sun with overlap member took $t+1.83$ days for completing the task, whereas the co-located members took $t+2.31$ days for completing the task. Here $t$ represents the time i.e. duration in days, which is initially zero. These results show that the follow-the-sun with overlap scenario completed the task by $t+0.48$ days i.e. 20.7% faster when compared to the co-located scenario.

Similarly, when comparing the results of the co-located scenario and follow-the-sun without overlap scenario, it was noted that the subjects of the follow-the-sun without overlap scenario completed the task faster when compared to the co-located scenario. The subjects of the follow-the-sun without overlap scenario completed the task in $t+1.97$ days whereas the subjects of the co-located scenario completed the task in $t+2.31$ days. Therefore, it can be concluded that the follow-the-sun without overlap scenario completed the task faster by $t+0.34$ days, i.e. by 14.7% when compared to the co-located scenario.

And finally, when investing whether the difference in the overlap time has an impact on the results, it was observed that the follow-the-sun with overlap scenario compared the task faster by $t+0.14$ days i.e. by 7% when compared to the follow-the-sun without overlap scenario.

So, when comparing the development duration utilized by the subjects of the three development scenarios, it is observed that the follow-the-sun with overlap scenario completed the task faster when compared to follow-the-sun without overlap scenario and the co-located scenario.

3.2.1.2 Measuring the cost effectiveness of the software development scenarios

As discussed in the section 2.2.6.3, cost efficiency is the defined as the amount of effort utilized by the subjects of the three scenarios. The below figure 23, represents the effort utilized by the subjects for the completion of the task.

![Figure 23: Total effort utilized in each scenario](image-url)
Here in the experiment, each scenario consists of 4 students. In case of co-located scenario as all the team member’s work from the same place and at the same time, the development duration is multiplied by 4 i.e. number of students working at a given point of time. In case of follow-the-sun with overlap and follow-the-sun without overlap scenario, as the 4 students in a group are split up into 2 students each, who will be working from a different place and time zone the development duration is multiplied by 2 i.e. as 2 students are working at a given point of time.

From the above figure 23, when comparing the effort of the co-located scenario and follow-the-sun with overlap scenario, it can be observed that the co-located scenario’s effort is more when compared to the follow-the-sun with overlap scenario. These results are surprising, as the authors thought the amount of effort for the co-located scenario will be less when compared to follow-the-sun with overlap scenario as the students of the co-located scenario work from the same place and time zone. For this reason the authors expected the effort to be less in case of co-located scenario and also as there is a direct communication between all the team members. But the results are not as expected. The reason behind this is discussed in the section 3.2.4.1 below where it was evident by observing the students during the experiment execution.

Similarly in the case of follow-the-sun without overlap and co-located scenario, the effort of the co-located scenario was high and the difference is almost equal to the difference in the effort between the follow-the-sun with overlap scenario and follow-the-sun without overlap scenario. The reasons behind the difference in the effort are discussed in the section 3.2.4.1.

From these results we can conclude that follow-the-sun approach with and without overlap scenario is more cost efficient when compared to the co-located scenario. And also from the section 3.2.1.1, it was observed that the time-market was also faster in case of follow-the-sun approach when compared to the co-located approach. So, we finally conclude that follow-the-sun approach is the most cost efficient one when compared to the co-located scenario.

### 3.2.2 Results of the non-experienced students

In this section, the results of the non-experienced students are presented. As the non-experienced students are not familiar in the area of the programming task, all the three groups S, T, U were not able to complete the task within the deadline. As the non-experienced students were not complete the task within the given deadline, neither the time-to-market nor the effort of the results could be calculated. For this reason, the authors of the thesis were not able to compare the results of the non-experienced students with the experienced students and draw conclusions from the results as they are not measurable. The development time spent by the non-experiment students is represented in the table 27 in the appendix B.

### 3.2.3 Measuring the amount of time spent in catching up stage, reporting stage, programming stage and the communication stage

As discussed in the section 2.2.6.3, the development time \( T_D \) is defined as the sum of catching up time \( T_{CT} \), programming time \( T_{PT} \) communication time \( T_{COT} \) and reporting time \( T_{RT} \). Here in this section, the amount of time spent by the students in each of the four stages is reported. These results can form as one of the basis for analysing the reasons behind the difference in the development time of the three development scenarios. Table 21 below represents the amount of time spent by the students of the experienced and the non-experienced set of students at each stage of the three software development scenarios.
Table 21: Percentage of time spent at each stage

<table>
<thead>
<tr>
<th>Treatment Name</th>
<th>Group Name</th>
<th>Background Experience</th>
<th>Catching up time (in minutes)</th>
<th>Programming time (in minutes)</th>
<th>Reporting time (in minutes)</th>
<th>Communication time (in minutes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Co-located</td>
<td>Group P</td>
<td>Experienced</td>
<td>12</td>
<td>122</td>
<td>2</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Group S</td>
<td>Non-experienced</td>
<td>8</td>
<td>114</td>
<td>2</td>
<td>-</td>
</tr>
<tr>
<td>FTS with overlap</td>
<td>Group Q</td>
<td>Experienced</td>
<td>21</td>
<td>168</td>
<td>6</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>Group R</td>
<td>Non-experienced</td>
<td>34</td>
<td>168</td>
<td>7</td>
<td>6</td>
</tr>
<tr>
<td>FTS without overlap</td>
<td>Group T</td>
<td>Experienced</td>
<td>20</td>
<td>217</td>
<td>9</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Group U</td>
<td>Non-experienced</td>
<td>30</td>
<td>200</td>
<td>17</td>
<td>-</td>
</tr>
</tbody>
</table>

### 3.2.3.1 Catching up time

Catching up time as discussed in the section 2.2.6.3 is defined as the total time spent by the students from the time of reading and analyzing of the given experimental task till the start of the programming (coding) the task is termed as catching up time. The catching up time spent by the students in each of the iteration was noted by the experiment leads during the experiment execution. From the above table we can notice that the subjects spent on an average 20 minutes of their total time in understanding the results, where the experienced set of students spent on an average of 17 minutes and the non-experienced students spent on an average of 24 minutes in understanding the work. And also by observing the students during the experiment execution we have noticed that the students spent more time in understanding the task during their first iteration and from the next iteration the time spent on the catching up time was gradually reduced. When comparing the amount of time spent by the subjects of the all three scenarios in the catching activity, it is observed that the subjects of the co-located scenario spent less time when compared to the subjects of the follow-the-sun with and without overlap scenario.

### 3.2.3.2 Programming time

Programming time is defined as the total time taken by the students from the start of the coding part until they stop doing the coding. From the above table we can notice that the subjects spent their maximum time of 164 minutes in the programming stage only. The above table also shows that the subjects of the co-located scenario spent maximum time on the programming than the subjects of the follow-the-sun scenario. And also we can notice that the subjects of the follow-the-sun with overlap scenario utilized the less amount of time.
and by utilizing the communication time effectively they were able to complete the task faster than the other two development scenarios.

3.2.3.3 Reporting time

Reporting time is defined as the total time spent by the students from the time of stopping with the programming part till zipping all the files in folder and handing over the so far completed task in a pen drive to the experiment leads. From the above table we can notice that the subjects spent least amount of time in the reporting stage. On an average, the subjects took around 1-2 minutes in reporting the task to the experiment leads.

3.2.3.4 Communication time

Communication time is defined as the amount of time spent by the students in communication with the other sub-group students by the use of synchronous communication medium in case of follow-the-sun with overlap scenario. From observation it was noted that the subjects tried to utilize maximum amount of time from the given four minutes during each iteration. The figures in the above table represents that the experienced students effectively utilized their four minutes of communication time and completed the task faster when compared to the other two software development scenarios.

3.2.4 Qualitative Survey Results

The qualitative survey is conducted to find out the experience of the subjects, working on the experimental task from the perspective of the subjects. Here, the feedback results from the subjects of the co-located scenario, follow-the-sun with overlap scenario and follow-the-sun without overlap scenario are presented (refer to appendix A, for the feedback forms). From the results of the follow-the-sun with overlap scenario and follow-the-sun without overlap scenario, the authors expect to receive the answer for the research question 4 (RQ 4). The designed feedback forms were mainly based on open-ended questions, to get feedback from the experiment subjects. The feedback forms are designed carefully by verifying with the supervisor. On completion of the experiment, each student was asked to fill in the feedback form, based on their experience on working in the experiment. As there were two sets of students i.e. the experienced students and the non-experienced students, the results of the experienced and the non-experienced students are presented separately in order to have a closer look at the challenges faced by each set of students. The data presented in the below table 21, represent the summary of the most reported answers from data collected from each student in the group. Here the authors mentioned as the most reported answers because as there are only 4 members in each scenario, almost all the members reported the same answer to the addressed question.

3.2.4.1 Feedback from the experienced set of students

Experienced students were the ones that had previous experience of working on the Java platform. From the experiment results we have noticed that the experienced students completed the task in the three software development scenarios i.e. co-located scenario, follow-the-sun with overlap scenario and follow-the-sun without overlap scenario. Among the 12 experienced students who participated in the experiment, 11 students responded to the feedback and among the 11 students, 4 students are from co-located scenario, 4 students are
follow-the-sun with overlap scenario and the remaining 3 students are from follow-the-sun without overlap scenario. The feedback from the experienced set of students is represented in the table 22 below.

Table 22: Feedback from the experienced set of students

<table>
<thead>
<tr>
<th>Feedback questions</th>
<th>Co-located scenario</th>
<th>Follow-the-sun with overlap scenario</th>
<th>Follow-the-sun without overlap scenario</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rate the overall experience in working in follow-the-sun environment on a scale of 5 where 1= unhappy, 2= Ok, 3= happy, 4=good, 5= excellent)</td>
<td>4</td>
<td>4. Good if it was on a bigger project</td>
<td>4. Interesting to see how to work without communication and a very useful experience.</td>
</tr>
<tr>
<td>Difficulties faced while performing in the experiment?</td>
<td>Starting problem and integration problem but fixed them easily.</td>
<td>Starting problem and require some more time at the beginning.</td>
<td>- Communication problem.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- It was difficult to understand intentions and choice of implementation.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- Documentation leads to misunderstanding.</td>
</tr>
<tr>
<td>Is the task completed within the deadline?</td>
<td>Yes, last minute integration and missing code problem.</td>
<td>Yes</td>
<td>- Yes with some minor flaws.</td>
</tr>
<tr>
<td>Did you trust your sub team members?</td>
<td>Yes, good group everyone was co-operating.</td>
<td>Yes</td>
<td>- Yes, we all worked together against the same design and implementation.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- They did their best to complete the task.</td>
</tr>
<tr>
<td>Did you feel that you required synchronous communication medium? (In case of follow-the-sun without overlap scenario)</td>
<td>-</td>
<td>-</td>
<td>- Yes, to clarify doubts and design structure.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- Some structured way of documentation is required.</td>
</tr>
<tr>
<td>Did you face any kind of environmental</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Question</td>
<td>Yes/No</td>
<td>Notes</td>
<td></td>
</tr>
<tr>
<td>-------------------------------------------------------------------------</td>
<td>--------</td>
<td>-----------------------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>Is the overlapping time sufficient? (In case of follow-the-sun with overlap scenario)</td>
<td>-</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Are there any doubts that were unclear until next iteration? (In case of follow-the-sun with overlap and follow-the-sun without overlap scenario)</td>
<td>-</td>
<td>No, understood the code after analyzing.</td>
<td></td>
</tr>
<tr>
<td>Any suggestion for the improvement of the process?</td>
<td>-</td>
<td>Time was too short. Require some time before the start of the task.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Need more time for development. 56 minutes for the development was not enough.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Kickoff meeting is required to discuss regarding the work structure and communication</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>It is good concept in follow-the-sun approach.</td>
<td></td>
</tr>
</tbody>
</table>

### 3.2.4.1.1 Discussions on the results of the experienced set of students

In the table 22, the results of the co-located scenario, follow-the-sun with overlap scenario and follow-the-sun without overlap scenario of the experienced students are tabulated separately. In this discussion section, the results are analysed based on the feedback and the suggestions given by the students, and also from the results of observing the students and talking to the students personally during the experiment execution by authors of this thesis.

#### 3.2.4.1.1.1 Co-located Scenario

From the feedback of the co-located scenario, we can observe that the subjects of the co-located scenario reported that they had some starting problems and some integration problems. So, the authors made conclusions that as the subjects faced integration problems they took another 20 minutes of extra time for the completion of the task.

By observing the students and talking to the students personally, the authors have noted that the students initially misunderstood the task and by discussing with each other the students divided the task among themselves in such a way that two students work on a module. As the students worked independently, there was no communication between the group members. The students reported that even though one of the group members completed the task earlier, he had to wait until the rest of the members completed the task.

Based on the observation and from the feedback given by the students of the co-located scenario, it can be interpreted as the students of the co-located scenario followed module-based software development. The reasons for the starting problem was due to misunderstanding the task initially, so during the day one, students were not able to do much of the productive work will resulted in increase in development time in the case of co-located scenario. As the students worked independently and did not communicate with each other during the implementation of the task, the students faced missing code and integration
problems during the last minute. From these discussions, we can conclude that, if the students had initial kick off meeting before the implementation of the task, then the students would have discussed the task structure and would be likely to have completed the task within the deadline. And also we can conclude that as the students did not communicate among themselves during the experiment execution, resulted in the increase time-to-market and also the effort in completion of the task.

3.2.4.1.1.2 Follow-the-sun with overlap scenario

In case of follow-the-sun with overlap scenario, the students reported that they had some starting problems and required more time for development. And they also reported that, implementation on follow-the-sun approach on a larger scale would be better. Apart from these problems, students did not face any other problems and the students were happy with the implementation of the work.

From the feedback given by the students and based on observing the students during the experiment execution, it can be interpreted as, as the students knew each other earlier, they were able to communicate their intentions freely and clearly to the other sub-group members. And as there are only 4 minutes for the students to communicate with the other sub-group members, students felt that they required more time for communication. In practical scenario as it is difficult to deliver the project to other sub-group members in four minutes and convey them regarding the progress of the task, students felt that they required more time for communication. But as the experiment is to be conducted within four hours it was difficult for the authors of the thesis to simulate two days with more overlapping time. So it should be noted when implementing follow-the-sun strategy in future there need to more time for communication between the sub-group members and also that sub-groups require enough time for working on the task by themselves before they handover the task to the sub-group members. So, it can be concluded that even though there was an overlapping time of 4 minutes between the sub-group members, this overlapping time played a vital role in reducing the time-to-market and the effort when compared to the co-located scenario, where the students of the co-located scenario even though they were at the same place and time zone did not communicate with each other during the task implementation, which eventually increased the time-to-market and effort when compared to the follow-the-sun with overlap scenario.

3.2.4.1.1.3 Follow-the-sun without overlap scenario

The students of the follow-the-sun without overlap scenario reported that, they really liked working in follow-the-sun environment and it was interesting for the students to see on how to collaborate and work without any communication between the sub-group members. Apart from this, students felt that they had some trouble in communicating with the sub-group members especially in understanding their intentions on task. As a result of this students felt that they require some communication time to discuss about the structure of the task. Students suggested that in order to improve the process of follow-the-sun without overlap scenario students required initial kick-off meeting and standardized documentation format to avoid any kind of misunderstandings.

By observing the students during the experiment execution, the authors noted that the students know each other from a long time and felt that the task was easy to implement. The students communicated with each other through emails and sent email to the other sub-group members on what they have done so far after each iteration, where the communication among the students was mostly in Swedish. While talking to the students personally, students reported that they had problems in communicating their intentions as the thinking level of
both sub-groups was different. And finally students suggested that overlap is the good way to communicate between the sub-group members.

So, based on the feedback and after talking to the students personally, the authors felt that, in case of follow-the-sun without overlap approach students require structured documentation and initial kick-off meeting to discuss regarding the structure of the task implementation. Before the start of the experiment students were provided with guidelines stating that they have to follow a structured design and documentation, but it was from the perspective of the authors. From these results the authors have learnt that the subjects should have common understanding between them and define what a structured documentation is and not from the perspective of the authors. And also from the feedback from the students it was noted that the developers require initial kick-off meeting so that they can work effectively towards the directed goal. So finally it can be concluded that follow-the-sun strategy without overlap can also reduce the development duration if the developers have an initial face-face meeting before the start of the development and discuss regarding the implementation strategy. From these it can also be concluded that the difference in the time-to-market and effort between the follow-the-sun with overlap and without overlap scenario is due to documentation which created misunderstanding between the students. When comparing the experiment results of the follow-the-sun without overlap and the co-located scenario, it can be noted that even though the students did not have direct communication between them, the communication they had through mails made them to deliver the task faster when compared to the co-located scenario.

3.2.4.1.1.4 Summary of the experienced set of students

Based on the feedback from the students and by observing and talking to the students personally during the experiment execution, the authors noted that the main challenges faced by the students are: starting problem, communication problem, understanding the documentation, and while integrating the task. And from the challenges faced by the students, the students suggested that, they required initial kick-off meeting and some communication time in case of follow-the-sun scenario. From these results, the authors learned that, initial kick-off meeting is the driving force for the students to communicate better and work effectively under a directed goal i.e. follow-the-sun approach works better when having a joint plan.

3.2.4.2 Feedback from the non-experienced set of students

Non-experienced students are the ones who do not have any previous experience of working on the Java development platform. From the results of the experiment we have noticed that the non-experienced students were not able to complete the task within the given deadline in the three software development scenarios i.e. co-located scenario, follow-the-sun with overlap scenario and follow-the-sun without overlap scenario. The results in the table 23, below are based on the feedback given by 3 students from the follow-the-sun with overlap scenario and another set of 3 students from the follow-the-sun without overlap scenario. The feedback from the non-experienced set of students is tabulated in the table 23 below.

Table 23: Feedback from the non-experienced students

<table>
<thead>
<tr>
<th>Feedback questions</th>
<th>Follow-the-sun with overlap scenario</th>
<th>Follow-the-sun without overlap scenario</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rate the overall experience in working in follow-the-sun environment on a scale of 5 where 1 = unhappy, 2 = Ok, 3 = happy,</td>
<td>3</td>
<td>3. Powerful system since you have very little time for development.</td>
</tr>
</tbody>
</table>
Difficulties faced while performing in the experiment?

- Slow computers
- Restricted communication
- Starting problem in communication.

Is the task completed within the deadline?

- No

Did you trust your sub team members?

- Yes, but little progress from the other sub team members.
- No, did not understand what the other sub team members were doing.
- Even though the other sub team members were trying to understand, we were not just clear enough.

Did you feel that you required synchronous communication medium? (In case of follow-the-sun without overlap scenario)

- Yes, by phone it would have helped.

Did you face any kind of environmental disturbances?

- No

Is the overlapping time sufficient? (In case of follow-the-sun with overlap scenario)

- If we had a plan than the time would have been enough.

Are there any doubts that were unclear until next iteration? (In case of follow-the-sun with overlap and follow-the-sun without overlap scenario)

- No

Any suggestion for the improvement of the process?

- Faster computers
- At least one member in a have to communicate
- Kick off meeting is essential to discuss regarding the task structure

### 3.2.4.2.1 Discussions on the results of the non-experienced set of students

The results of the co-located scenario, follow-the-sun with overlap scenario and follow-the-sun without overlap scenario are tabulated separately. In this discussion section, the results are analysed based on the feedback and the suggestions given by the students, and also from the results of observing the students and talking to the students personally during the experiment execution by authors of this thesis.

#### 3.2.4.2.1.1 Co-located scenario
In the table above we can notice that there is no data that is reported in case of co-located scenario, the reason being that as in case of co-located scenario there was only a team that participated in the experiment and none of the team members did not turn up during the feedback session. So the feedback from the co-located team members was not noted.

But from the feedback that was collected from the students during the experiment execution it was noted that the students have started programming in the wrong order and the students also reported that as they were new to Java programming they require more time for development. And the experiment leads who were monitoring the group reported that the students were not enthusiastic while participating in the experiment where only 2 members participated actively during the development duration.

From these observations, we can conclude that due to lack of active participation by all the team members in the group, the students were not able to complete the task within the deadline. And also, as only 2 members actively participated in the group students felt that they require more time for development and it might not be because as they are not familiar in Java platform. This is just the author’s perception but might not be the real fact also. Because when have talked to the Assoc.prof. Börstler, he said that even though the students are not familiar in Java language earlier, he gave some introductory classes, so that the students can perform the task.

3.2.4.2.1.2 Follow-the-sun with overlap scenario

In case of follow-the-sun with overlap scenario, the students reported that they had some starting problems and required more time for development. And they also reported that, implementation on follow-the-sun approach on a larger scale would be better. Apart from these problems, students did not face any other problems and the students were happy with the implementation of the work.

From the feedback given by the students and based on observing the students during the experiment execution, it can be interpreted as, as the students knew each other earlier, they were able to communicate their intensions freely and clearly to the other sub-group members during the overlapping hours hence effectively using the overlapping time. During the overlap they primarily discussed issues related to work that has been done in the current iteration, the problems faced during the implementation of the code etc. As there are only 4 minutes for the students to communicate and deliver the work done to the other sub group members, students felt that they required more time for communication and also require more time to work individually on the task before they hand over to the other sub group members. But as the experiment is to be conducted within four hours it was difficult for the authors of the thesis to simulate two days with more overlapping time. So it should be noted when implementing follow-the-sun strategy in future there need to more time for communication between the sub-group members and also that sub-groups require enough time for working on the task by themselves before they handover the task to the sub-group members. So, it can be concluded that even though there was an overlapping time of 4 minutes between the sub-group members, this overlapping time played a vital role in reducing the time-to-market and the effort when compared to the co-located scenario, where the students of the co-located scenario even though they were at the same place and time zone did not communicate with each other during the task implementation, which eventually increased the time-to-market and effort when compared to the follow-the-sun with overlap scenario.

3.2.4.2.1.3 Follow-the-sun without overlap scenario
In case of follow-the-sun without overlap scenario, the students during feedback session termed the follow-the-sun approach as a powerful system and reported that they have come across many problems during the experiment execution. The main problem that the students faced was that the sub-group members were not able to follow what the other team members were doing, so until the second iteration they did not have a correct plan for implementation of the task. Team members felt that more time was wasted in explaining about the structure of the task to the other sub-group members and they felt that if both sub-group members had a kick-off meeting before the experiment execution the development time would have been reduced to a greater extent.

By observing the students during the experiment execution, the experiment leads reported that even though the students knew each very well, they have not worked with each other earlier. The students communicated with each other by exchanging emails after every iteration in explaining clearly about the progress of the task of the current iteration.

From these results it can be conclude that, even though the students felt they communicated well by sending an email to the other sub-group members regarding the progress of the task, the other sub-group member (receiver) found it difficult in understanding the task. This resulted in lack of proper planning till the second iteration which eventually was the reason behind not completing the task within the given deadline.

3.2.4.2.1.4 Summary of the non-experienced set of students

The feedback from the non-experienced students who were not able to complete the task and from observing the students during the experiment execution resulted in analysing the reasons of the non-experienced students for not completing the task within the given deadline. Among the various challenges that are reported by the students of the three different scenarios, the most common one is the communication problem. As in follow-the-sun approach where sequential task distribution takes place, students found it hard in understanding the work of the other sub-group members with minimal communication between them. So in order to mitigate this challenge students reported that they required initial kick-off meeting to discuss regarding the structure and the design of the implementation strategy.
4 DISCUSSIONS

4.1 Evaluation of the experiment results

In this section, the results of the experiment are discussed. In this thesis, a controlled experiment was conducted to investigate whether the development time can be reduced to 50% when using follow-the-sun approach when compared to the co-located approach. As there are very few success cases [P6, P12, P13] that are documented in case of follow-the-sun approach, this became as the major driving force for the authors of this thesis to investigate deeply on follow-the-sun approach for achieving the full benefits of it through its effective implementation. The experiment was conducted with two different sets of students i.e. the experienced and non-experienced students. In order to mitigate the impact of their background knowledge on the experiment results, the students were differentiated before the execution of the experiment. The assumption that the authors made regarding the impact on the experiment results based on the subject’s background knowledge was observed to be true as the results of the experienced and non-experienced students are completely different. The experiment results showed that the experienced students completed the task within the deadline where as the non-experienced students were not able to complete the task within the given deadline. As the non-experienced students did not complete the task, there is no data in hand to determine the speed and accuracy of the experiment. So the results obtained from the experienced set of students are evaluated and discussed below and these results showed that experienced students can have higher advantage while using follow-the-sun approach when compared to the non-experienced students.

From the experiment results, it was observed that development time of the experienced set of students was reduced by 22.1% in case of follow-the-sun with overlap scenario when compared to the co-located approach. As there is no research that has been done till now in comparing follow-the-sun with overlap scenario with the co-located scenario, this result of 22.1% reduction in development time compared to co-located approach adds as a new dimension to the research field. This result is the major finding of this thesis.

When comparing the results of the experienced set of students of follow-the-sun without overlap scenario with the students of the co-located scenario, it was observed that the development time is reduced by 10.7%. This result is in line with the results of the experiment conducted by Carmel [P6], where he states that there is reduction in development time of 10% rather than theoretical 50% when comparing follow-the-sun without overlap scenario with the co-located scenario [P6]. Apart from Carmel there are other researchers who stated that there is reduction in the development time to 10-20% in case of non-overlapping hours [P13].

When comparing the results of the follow-the-sun with overlap scenario and follow-the-sun without overlap scenario, it is observed that students of the follow-the-sun with overlap scenario were able to complete the task faster by 11.4%. From the results of the measurement instruments, where the students of the of the follow-the-sun without overlap scenario reported that they took more time in understanding each other’s work or sometimes redo the work due to misunderstanding in the work done by the other sub-team. Resulting in wastage of development time for unproductive work. While this kind of killing of the development time could be overcome in follow-the-sun with overlap scenario. Where the overlap time was used by the team members to discuss about what has been done and what is to be done. The students of the follow-the-sun with overlap scenario reported that they would have performed even better if they had more overlapping minutes. From these observations
the authors of this thesis concludes that the main reason for the difference in the time-to-market between the follow-the-sun with overlap scenario and follow-the-sun without overlap scenarios is that in case of follow-the-sun with overlap scenario, the sub-group members communicated with each other effectively during their overlapping time which resulted in earlier time-to-market when compared to follow-the-sun without overlap scenario. It can also be concluded that with good communication between the sub-group members and by having some amount of overlap in working hours there may be reduction in time-to-market when using follow-the-sun even though sometime is wasted in communication but that is worth it to avoid time the time put in understanding the work done by the others or time spent for rework which is sometimes many times more than the time spent on communication.

And finally, when comparing the experimental outcome of the co-located scenario with follow-the-sun with overlap scenario and follow-the-sun without overlap scenario, it is observed that follow-the-sun with overlap scenario completed the task faster among all the three software development scenarios. The authors think that even though it is not possible to completely reduce the development duration by 50% as in theory, but there will be a reduction in the development time by the use of follow-the-sun approach. Our findings indicate that by addressing the known problems of the follow-the-sun approach, it is possible to reduce the development time when compared to the co-located scenario.

### 4.2 Challenges

From the results of the experiment, we have noticed that even though there is a reduction in the development time by the use of follow-the-sun approach, it does not reduce by exactly 50% as in theory. The main reason is being number of challenges that are faced when implementing the follow-the-sun scenario. Here we discuss some of the challenges that we have come across during the implementation stage and the lesson that we have learnt from the challenges. The authors of the paper think that by emphasizing more on the lesson learnt from the experiment, the development time can be reduced to a greater extent if the experiment is conducted again. In the table below “(+)” denotes as advantages and “(-)” denotes as disadvantages.

<table>
<thead>
<tr>
<th>Challenges</th>
<th>Impact on the experiment</th>
<th>Lessons learnt</th>
</tr>
</thead>
<tbody>
<tr>
<td>Background of the subjects</td>
<td>The difference in the experience level in the programming had a huge impact on the results of the experiment. (+) Experienced students (-) Inexperienced students</td>
<td>When selecting people involved in FTS development, make sure that they have a strong grip on the area of software development. (+) By selecting the subjects of the related platform, development time can be reduced to 50%. (-) If the subjects are not familiar on the development platform, results in increased development time which finally results in increase in cost of the software development.</td>
</tr>
<tr>
<td>Communication</td>
<td>There is a difference in the results of the co-located scenario, follow-the-sun with overlapping time of four</td>
<td>(+) In case of follow-the-sun with overlap as there is an overlapping time of four</td>
</tr>
</tbody>
</table>
overlap scenario and follow-the-sun without overlap scenario, the main reason for difference in results in all the three scenarios is the communication gap.

(+) Increase in overlapping hours.
(-) Decrease in overlapping hours

(minutes, they were able to complete the task faster when compared to follow-the-sun without overlap scenario.

(-) If there is less time for communication between the teams, it takes more time in understanding the work of the other sub team members, which finally results in increase in development time.

(+ ) From the experiment we have noticed that, if there is less time for communication between the sub-groups then if the work is structured and a defined process with a good communication link between the sub teams than the desired results can be achieved.

| Face-to-face meeting | (-) Increase in development time. | As there is no direct communication between all team members before the start of the experiment, it took a lot of time for the students to understand each other’s work. |
| | (-) Increase in time to understand the task. | (+) In case of distributed development, it requires an initial kick-off meeting between the sub team members to discuss the task and the process of implementing it. |
| | (-) Co-ordination problems | (+) The subjects of the experiment suggested that they required a defined process that has to be followed and both sub teams have to stick to the defined rules and implement the task. |

| Geographical distance | (+) Familiarity among the team members reduced the problems of geographical distance. | (+) As the subjects were asked to form the groups by themselves, familiarity among the subjects helped the subjects in knowing the efficiency of their friends which resulted in doing the task with a good ease and helped in reducing the coordination problems. |
| | (-) Loss of teamness. | (-) In case of inexperienced students, students faced a lot difficulty in understanding each other’s work which resulted in increased developed time. |

| Infrastructure | (+) As the experiment is conducted in the premises of BTH, both laboratories had a | (+) The problem of difference in the infrastructural background having an impact |
similar infrastructure.
(-) Slow computer.

on the performance of any one of the sub team was reduced.
(+ ) Students reported that computer were very slow which resulted in increase in the development time.

<table>
<thead>
<tr>
<th>Documentation</th>
<th>(-) Difficulty in understanding (-) Resulted in loss of feeling of oneness.</th>
</tr>
</thead>
<tbody>
<tr>
<td>(-) In case of follow-the-sun without overlap, as there is no direct communication between the team members, students exchanged emails to communicate with the other sub team members. As the structure of the implementation is not discussed among them, the sub team members felt difficult in understanding the way they are implementing the task.</td>
<td></td>
</tr>
</tbody>
</table>

4.2.1 Mapping the challenges of the experiment with literature

The challenges that are cited in the literature as well as in the experiment are:

- Clarification cost [P6], [P11],
- Delayed response [P6], [P8], [P11],
- Time zone difference [P8], [P12],
- Communication issues [P1], [P11], [P12],
- Re-work cost [P6], [P11],
- Misunderstandings that arise [P6],
- Technical background difference [P12],
- Demands matured infrastructure [P12],
- High dependencies [P12],
- Co-ordination [P11],

A number of challenges that have been found in the literature have been noted also in our experiment. The obstacles that hinder the use of follow-the-sun approach, which where commonly noted both in literature and in our experiment are lack of communication, co-ordination problem, misunderstandings, high dependency, delayed response, time zone difference, clarification cost, technical background difference etc. Let’s look at each of the issues and discuss how the authors of this thesis identified and concluded it as challenge for follow-the-sun team in our experiment.
Table 25: Mapping each of the challenges with group that has observed

<table>
<thead>
<tr>
<th>Challenges</th>
<th>Experienced</th>
<th>Non-experienced</th>
<th>Author’s conclusion</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FTS with overlap scenario</td>
<td>FTS without overlap scenario</td>
<td>FTS with overlap scenario</td>
</tr>
<tr>
<td>Clarification cost</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Delayed response</td>
<td>✓</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Time zone difference</td>
<td>✓</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Communication issues</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Re-work cost</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Misunderstandings that arise</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Technical background difference</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Demands matured infrastructure</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>High dependencies</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Co-ordination</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

In the experiment the authors noted a difference in the results of follow-the-sun with overlap and follow-the-sun without overlap scenario primarily due to communication issues. The follow-the-sun with overlap could complete the task before follow-the-sun without overlap team because of the overlap of four minutes that persists in the follow-the-sun with overlap team where they could clarify and resolve issues and plan the future course of action. But follow-the-sun without overlap team could only communicate through asynchronous communication resulting in misunderstandings and co-ordination problems due to time zone difference and due to high dependency. As a of result delay in response when clarification is needed, the teams had to re do the work resulting in re-work cost as observed in literature. These challenges are in line with the challenges seen in the literature that is communication issues [P1], [P11], [P12], misunderstandings [P6], co-ordination problem, time zone difference[P8], [P12], high dependencies [P12], delayed response[P6], [P8], [P11], clarification cost [P6], [P11], re-work cost[P6], [P11] as found in the respective studies.

Technical background difference also was noticed very clearly in our experiment as seen in literature [P12]. There has been noted a dramatic difference in the outcome with experience set of participants (java students) and with non- experience set of participants (non java students). From the results it is clear that it is necessary that experienced people are involved while using follow-the-sun approach to enjoy the benefit of reduction in the development time. Which may be a hurdle for a company to employee only experienced set of workers rather than fresher.
Demand for matured infrastructure was a challenge for our experiment as well. The systems that were used by the students in the experiment where not high-end systems and could not install CVS due to capacity of the processor. So for this reason the data could not be shared using a shared common repository and hence we had to use pen drives to hand over the work between the sites. From above it is clear that there is a need for matured infrastructure while using follow-the-sun approach, this challenge is in line with challenge that has been cited in literature [P12]. It may require companies to rebuild the whole infrastructure to work in this new environment.

Challenges such as cultural difference, suitability to all the phases, poor quality and restriction on the data exported have not been observed due to various reasons such as because of the participants involved, study setting etc.

The participants in the experiment felt the need for a formal development process to develop a plan of action and act accordingly which would help for improved performance. This was contradictory to the findings of literature where it has been felt that follow-the-sun approach requires that a formal uninterrupted one week plan [P12] which created a situation of chaos between the employees as a result was observed as a challenge. If this kind of formal development was enforced in the experiment then the subjects would have perceived it as an obstacle for their progress.

In our study, the subjects where all from Sweden with Swedish being their mother tongue. Communication between the participants both, written communication that is through mails between the sub-groups and oral communication among the co-located team members took place in Swedish hence there was no misunderstanding that emerged due to culture difference, clashes due to social morays etc as seen in literature [P1], [P12]. By having all the team members from the same cultural background we could not observe challenges that would arise due to cultural differences. Suitability of follow-the-sun approach for all the phases of the software development cycle could not be observed as a challenge in our experiment as seen in literature [P8], [P10] because of the simplicity of the given task and the duration for which the experiment was conducted. The task on which the participants worked was toy task (pay station application described in appendix A) with only a small set of features to be developed. As the task was small and did not require to go through all the phases of the software lifecycle such as designing, testing, maintenance etc and because the duration was also less that is just four hours it is difficult to go through all the phases of the lifecycle in just four hours where as a real software project with much larger functionalities to be developed and the demand to go through the phases such as designing, testing, maintenance taking a few months cannot be compared with our task. Hence the challenge of not suitable for all the phases of the software development cycle could not be studied in our experiment.

In our experiment there may be difference in quality in the code developed by co-located team, follow-the-sun with overlap team and follow-the-sun without overlap team. But determining the quality [P16] of the code was outside the scope of our experiment. Restriction on the data that is exported to other country [P1] was not applicable for our experiment design because we just simulated the environment with time zone difference but did not have the experiment run with team members from countries participating. If we had the experiment run with a real project and with team member belonging to different countries and if the data has to be exchanged between countries than this kind of challenged would have been noticed.
4.2.2 How realistic are the results obtained?

It has been observed in the experiment that when using follow-the-sun with overlap scenario, the development time was reduced by 21% when compared to the co-located scenario. This reduction of 21% in the development time when implementing follow-the-sun with overlap scenario in the real world scenario is questionable (except in special cases that will be discussed in the following paragraphs), as there are certain drawbacks in the experiment design that does not reflect the exact real world scenario. Let’s look at the drawback in the experiment design that makes the results not so realistic and accurate.

In the experiment the simulated days (duration in minutes) varied between the three scenarios. In case of co-located and follow-the-sun without overlap scenario, 126 minutes was considered as one day (that is, 63 minutes represent as day time while the other 63 minutes represents as night time), whereas for the follow-the-sun with overlap scenario the simulated day was for 120 minutes. The experiment has been designed in this way because, here in this experiment the authors focused on maintaining the same effort for all the three scenarios (that is follow-the-sun with overlap, follow-the-sun without overlap and co-located scenarios), which was inspired from the paper [P6]. In paper [P6], it is said that it is important for the follow-the-sun team to have the same effort as that of the co-located team. For this reason the authors designed the experiment in such a way that same effort per day is maintained for all the three scenarios for which reason the duration per day for follow-the-sun with overlap varied when compared to the other scenarios keep in mind the extra effort that incurs during the overlap. And the other drawback in the experiment is that, in case of follow-the-sun with overlap scenario, the overlap at the end of the first day where sub-group B after there working hours communicated with sub-group A members of the 2nd day, which is not possible in the real world (but only in exceptional cases, that will be discussed in the following paragraph). May be because of this extra overlap time where the subjects communicated with each other, we might have achieved 21% reduction in development time where as in literature the highest reduction that has been cited is 20% [P6]. So next time when conducting a similar experiment attention has to be paid on simulating the days more realistically rather than on balancing the effort and the extra overlap time that has been given in the experiment has to be avoided in order to get more realistic and accurate results.

Example: Special case where the extra overlap is possible in real world is when there are three development sites and when one of the development site works for 10 hours a day rather than the usual 8 hours a day. The cost incurred by having extra 2 hours could be minimized by having the site in lower wage countries such as in Asia.

The authors feel that the results of follow-the-sun without overlap scenario and co-located scenario are more or less realistic and such kind of reduction of 15% in development using follow-the-sun without overlap scenario is possible as the design looks close to real world scenario.

The other constraints that also have to be considered are professionalism, no cultural difference, no organisational difference, familiarity, task complexity. Each of the constraint and the effect on the results for each of it is detailed.

- **Professionalism**: This may have a significant effect, however, the expected benefits would be higher than using students.
- **No cultural and organisational differences**: In real life the results would be more complicated therefore the same benefits would not be achieved. This means that the achieved results are relevant for follow-the-sun teams with no cultural differences.
• Familarity: Familarity among the team member may have helped achieve such kind of reduction in development time as there were no issues in communication and coordination to familiarity which would not be the case in real world scenario.
• The main difference in relation to the real world is the task. This experiment does not allow to predict what would happen if the work was more complicated.

The experiment should be viewed as a trial that demonstrated promising results, and shall trigger further investigation in the large scale. The link that emerged through the experiment, which is expected to repeat in the real life is that follow-the-sun with overlap is more successful than without overlap. To be able to generalize this conclusion, more data points are necessary, however this relation does not seem to be influenced by the experience or complexity of the task.

4.2.2.1 Generalizing the results to industrial practitioners

Yes, the same experiment can be repeated using industrial practitioners. The difference in the subjects would vary the results. Such as if industrial practitioners are the subjects of the experiment then the professionalism may have a significant effect, however, the expected benefits would be higher than using students. Now, as the experiment has been conducted with students as subjects there was no cultural and organisational differences that has been experienced by the subjects. But with industrial practitioners the results would be more complicated therefore the same benefits would not be achieved. The experiment should be viewed as a trial that demonstrated promising results, and shall trigger further investigation in the large scale. The link that emerged through the experiment, which is expected to repeat in the real life is that follow-the-sun with overlap is more successful than without overlap.

4.2.2.2 Difference in the efficiency of the co-located team members

From the results of the experiment it has been observed that the co-located team is less efficient when compared to the follow-the-sun team. It maybe because of inability to split the work among too many people (there was a difference in the setup 4 students and 2+2 students incase of follow-the-sun team). Based on the internal dynamics of the group the results may have been affected and the conclusions can be made are:
• More data points would have provided perhaps a better explanation of the reasons for this difference.
• If equal number of students were involved, perhaps the results would have been different.

Since the experiment was inspired by related work performed by other researchers, the contribution to this findings is a suggestion for future research.

4.2.3 Implications for the researchers

The lessons learnt from the above table 25 can act as good implications for the researchers in the field of follow-the-sun approach. From the results of the experiment and based on the personal observation by the authors, the important implication is that by varying the team size for co-located approach and follow-the-sun approach that is by having more number of people in follow-the-sun team the experiment would have had different results that are important if a company would not mind investing more cost for being first in the market. The cost that incurs can also be minimized by having a team in lower wage countries. This dimension of research is important for some companies.
4.3  Validity Threats

In this section, the threats to validity concerning the research methodology of this thesis are discussed. In this thesis we have conducted a systematic literature review and experiment to investigate the results of the follow-the-sun approach. According to Wohlin [13], there are four different types of validity threats.

4.3.1  Internal Validity


Publication bias in the systematic literature review is one of the threats related to internal validity. Publication bias refers to where the researcher of that particular paper discusses only the positive results whereas the negative results are not presented properly. To mitigate this threat, the authors of this thesis during the study selection criteria selected the papers individually and discussed with each other if there is any difference in the selection criteria of the authors. Further in order to check for the argument level between the authors kappa coefficient is calculated. The results of the kappa coefficient indicate that there is negligible level difference of argument between the authors.

Selecting only few included studies by rejecting large number of papers might be one of the threats to internal validity as there might be any missing relevant papers. To mitigate this threat the authors had read all the papers individually and then discussed with each about the papers to be included.

Threat to mortality was reduced in this thesis, as the selected subjects for the experiment are related to field of software development.

The threat to instrumentation was reduced as the data collection forms are designed carefully by reading the literature thoroughly and also by consulting the supervisor the forms are updated so that the students do not misunderstand the questions.

4.3.2  External Validity


In this thesis, the external validity is concerned regarding the generalizations of the experiment results. In this thesis, the main intension of the authors is to investigate on follow-the-sun approach but to not show the results of the experiment as a representation of the real world. Due to this reasons, the threat to external validity is reduced in this thesis. As the experiment is conducted with less number of subjects there might be a threat that more challenges regarding the follow-the-sun approach are not covered.

4.3.3  Construct Validity

Wohlin [13] defined construct validity as the validity which is concerned with the relationship between theory and observation.

Missing of the relevant papers due to inappropriate search string is one of the construct validity threats in this thesis. To avoid this threat, the formed search string was discussed with the supervisor and with the suggestion of the supervisor the search string was updated.
And also the librarian at Blekinge Tekniska Högskola was consulted regarding the search operation in the online databases.

One of the construct validity is simulation of day that varied between the three development scenarios and the extra overlap that is given to follow-the-sun with overlap that would have had profound impact on the results that have been generated.

Another threat to construct validity in the experiment is due to evaluation apprehension. As the students are informed clearly that the performance of the students in the experiment will not have any impact on the grades on the students and also only the interested students were asked to participate in the experiment the threat to evaluation apprehension is reduced.

### 4.3.4 Conclusion Validity

Conclusion validity is defined by Wohlin [13] as the threat that is concerned with the relationship between the treatment and outcome.

In this thesis the conclusion validity threat is concerned with the violation of analysis of the experiment results. As the results of the experiment are crucial for this thesis, researchers in the field of the global software engineering were consulted. And also by discussing with the supervisor, the results were verified so that there are no mistakes in the analysis part of the experiment results.

Another threat to validity is concerned with the reliability of the measures. In this experiment, there might be a threat the students have misunderstood the questions in the data collection forms. To mitigate this threat, the students were guided by the experiment leads during the feedback sessions.

Another conclusion validity threat is related to data collected by the experiment leads. During the execution of the experiment as the experiment leads assisted in recording the duration of the experiment execution there is a chance that they have misunderstood the concept and noted wrong results. In order to mitigate this threat the students that have attended global software engineering course were selected, as these students will have a basic knowledge on the background of the experiment. And also the day before the experiment the authors clearly explained about the execution of the experiment and also the authors assisted the experiment leads during the experiment execution.
5 CONCLUSIONS

In this thesis, an experiment is conducted on follow-the-sun approach to investigate whether there is a reduction in the development time by 50% when compared to the co-located approach. In order to investigate these result, first systematic literature was conducted to find out about the research that has been done till now. After that an experiment is conducted to determine the development time of the follow-the-sun approach and also to find out the challenges that are faced by the subjects of the follow-the-sun approach.

The results of the experiment showed that there is gap between the results in theory and in practice in case of follow-the-sun approach. The experiment results also showed there is a difference in the development time between the subjects of experienced students and the non-experienced students. In order to find out the challenges faced by the student’s different data collection forms were used. The results of the survey helped in determining the various factors that caused hindrance to students in achieving the desired results. The feedback and the suggestions given by the students were very helpful for deeper understanding on the implementation of the follow-the-sun approach.

This research adds a new dimension stating that by implementing follow-the-sun approach with overlap there will be a further reduction in the development time when compared to follow-the-sun without overlap scenario, but this reduction is not as much as promised. The authors conclude that by focusing more on the challenges and by reducing the problems faced, the reduction in development time can be closer to the promise reduction in development time.

5.1 Mapping research questions to the outcome

The results of the research questions that are addressed in the section 1.3 are discussed in brief here

RQ 1 What are the best suitable factors for the implementation of follow-the-sun approach?

The findings of the literature can be divided into six main findings. They are challenges faced, tools to be used, suggestions for improved performance, optimal number of sites, suitable phases when using follow-the-sun and comparison of co-located approach with follow-the-sun approach. The findings for each of the aspects are:

- **Challenges:** The most common obstacles are the hand-off [P2], [P6], clarification cost [P6], [P11], not suitable for all the phases [P8], [P10], delayed response [P6], [P8], [P11], time zone difference [P8], [P12], communication issues [P1], [P11], [P12], cultural difference [P1], [P12] and re-work cost[P6], [P11]. The others are misunderstandings that arise [P6], high effort [P16], poor quality [P16], increased overhead distribution [P16], decreased productivity [P16], technical background difference [P12], demands matured infrastructure [P12] and formal development process [P12], high dependencies [P12], co-ordination [P11], demands employees to poses different skill set [P1] and restriction on the data exported [P1].

- **Tools:** The tools that could be used to improve performance while using follow-the-sun approach are resource allocation model [P5], [P15], task scheduling algorithm [P13], collaboration software[P1], wiki[P1], web-day[P1], version control[P1], extreme programming[P1], internet security [P1], information system tools [P1],
internet-based prototype has been proposed [P1], data representation tools [P4], and hand-off tools [P4].

- **Comparison of follow-the-sun approach with co-located approach:** Follow-the-sun approach takes 70% higher effort [P16]; resource utilization increases by 54%-78% [P13]; use codification process while co-located uses personalization strategy for knowledge sharing [P2] and differed in technology use [P2] and communication pattern [P2] while conclusions on quality and development duration cannot be made because different studies have different conclusions. It has been noted in [P6], [P13] that the development duration reduces by almost by 10-20% using follow-the-sun approach while in paper [P16] there was noted an increase in project duration by 50% using follow-the-sun approach while another study [P2] has noted that both co-located and follow-the-sun approaches take the same duration. The same way quality also had contradictory conclusion, it has been noted quality decreases with follow-the-sun approach while [P2] has noted that quality remains same for both the approaches.

- **Optimal number of sites while using follow-the-sun approach:** If the objective is to makes time zone separation manageable or to improve the working accuracy then two sites is preferable [P8], [P10], [P14]. While if working speed is the main objective then two sites [P14] or three sites [P16] should be considered.

- **Recommended phases to implement follow-the-sun approach:** Follow-the-sun approach is recommended for less equivocal phases of the software development such as for testing [P6], [P8], [P11], defect resolution [P10], [P8], [P11], maintenance phase [P10], [P8], [P7] and prototyping [P6].

- **Suggestions for improvement when using follow-the-sun approach:** By training software engineers on communication, cooperation and collaboration skills [P6]; reward balancing [P6]; using agile approaches [P6]; having fewer sites with good performance [P14], [P8]; hiring a coach [P12]; boot camping [P12]; by having rotating guru [P12]; by practicing remote pairing [P12]; by having shared common environment [P12]; by having high process maturity [P3]; coordinating properly [P11]; handing off at the right time [P11]; by having clearly stated requirements [P11]; by using good communication tools [P11]; by having face to face meeting [P2]; with simple, unambiguous, well defined [P9] and semi-structured [P1] tasks; by increasing knowledge transfer and communication [P1]; by sharing and transferring knowledge efficiently [P4]; by having information in a structured format [P4]; by using reliable and user friendly technology for transferring of data [P4]; by reusing design [P4]; minimizing interaction by dividing the tasks into smaller components [P4]. By adapting the above mentioned suggestions one can achieve better results using follow-the-sun approach.

**RQ 2** Can we deliver faster using follow-the-sun approach when comparing to co-located scenario?

The answer is yes. In section 4, the results of the experimental outcome are discussed where the results of the co-located scenario and follow-the-sun scenario are obtained by conducting an experiment. From these results, it is evident that follow-the-sun with overlap scenario was able to complete the task faster when compared to the co-located scenario and follow-the-sun without overlap scenario. And when comparing the results of the follow-the-sun with overlap scenario with the co-located scenario, it is observed that the follow-the-sun with overlap
scenario completed the task faster by 20.7%, and while comparing follow-the-sun without overlap scenario with the co-located scenario it is observed that the follow-the-sun without overlap scenario completed the task faster by 14.7% when compared to the co-located scenario. It is evident that follow-the-sun approach reduces development time compared to co-located approach the reduction is even further when using follow-the-sun approach with overlap.

**RQ 3** Can we be cost-efficient when using follow-the-sun approach when comparing to co-located scenario?

The answer is yes one can be cost-efficient while using follow-the-sun approach this is evident from the experiment results. It has been seen that follow-the-sun approach took less effort when compared to co-located scenario in both the scenarios that is follow-the-sun with overlap and follow-the-sun without overlap scenario. While using follow-the-sun with overlap the cost could be cut down even further when compared to follow-the-sun without overlap.

**RQ 4** What are the challenges faced by the subjects of the follow-the-sun scenario during the experiment?

During the experiment execution, we have noticed that the subjects faced some challenges while performing the task as designed. The challenges faced by the subjects are communication problem, time zone difference etc. The positive and the negative impact of these challenges on the experiment are discussed in the section 5 in detail.

### 5.2 Future Work

In this section, the future research in the field of follow-the-sun approach is discussed here. Future work that can be performed on the controlled experiment and in field of follow-the-sun approach is discussed below.

**Future work on the controlled experiment**

The conducted controlled experiment can be replicated in the future to get confidence in the results. Next time one can conduct the same experiment by having varied team size for follow-the-sun approach and co-located approach. That is by having almost doubled the number of people working for follow-the-sun approach when compared to co-located approach (that is by having the effort approximately doubled for follow-the-sun approach when compared to co-located approach). In this way the authors hope the development time would reduce even further even though the cost would increase for having extra people working, but this increase in cost has to be put in if the benefit of releasing the product is much more when weighed with the cost that is incurred. Experiment execution can be refined even further by installing the CVS software as the central repository for the subjects of the follow-the-sun approach as it becomes easier for the students to trace back the changes made in the code. By doing so, the author’s hope the amount of time spent by the students in understanding the task can be reduced thereby reducing the total development time of the task. And also as discussed in the validity threats of the experiment that very small sample of subjects participated in the experiment i.e. one group of four students for each of the three development scenarios. So, when replicating the experiment it is advised to have a larger sample so that there is chance that more number of challenges on follow-the-sun approach can be addressed.
Future work in the field of follow-the-sun approach

As from the experiment results it can be observed that there is a reduction in development time when following sequential development in the development of phase of the software development life cycle, in case of follow-the-sun approach. As the experiment results showed a new dimension in case follow-the-sun with overlap scenario, by focusing more on the challenges, the researchers can further experiment to achieve the full benefits of follow-the-sun approach.
6 References


D. Šmite, Lecture slide- Scenarios, Global Software Engineering (PA2411), University of Blekinge Tekniska Hogskola at Karlskrona, Sweden, Ht10, LP1, 2010.


distance,” in *International Conference on Global Software Engineering (ICGSE’06)* October 2006.


Systematic Literature Review References


7 APPENDIX

7.1 Appendix A

7.1.1 Initial Experiment Planning

7.1.1.1 Experimental Task Description

**Task Title:** Help Center Request Management Tool (HCRMT)

**Number of Students:** Total 18 students (10 students for co-located team and 4+4=8 for distributed team).

**Task time period:** 16 hours (2 days, 8 hours each day, 10 students*8 hours each day= 80 hours i.e. Total amount of time spent on the task) for co-located team and 10 hours (2 days, 5 hours each day, 2 hours and thirty minutes and for each sub team, 8 students*5 hours each day= 40 hours i.e. Half the amount of time spent than that of the co-located team) for distributed team. And another important factor to consider is the recess time for each team. For the co-located if the leisure time is 1 hour a day, then the total amount of time spent on doing the task is 78 hours (1 hour/day * 2 days= 2 hours, 80 hours-2 hours= 78 hours). Similarly if the distributed teams are given 15 minutes of break in between their work time, then the total amount of time spent completely on performing the task is 39 hours (15 minutes/day * 2 days= 1 hour, 40 hours-1 hour= 39 hours i.e. exactly half the amount of time that of the co-located team).

**Programming languages:** Java (Java Servlets), HTML, My SQL server.

**Software Requirements:** Java, My SQL, Tomcat, CVS (Concurrent Version System), HFS (HTTP File Server).

**Introduction:** HCRMT is a multi-user web application that uses a database as persistent store to hold data pertaining to users’ requests. Each user has an account and the user has to login with the given username and password so as to access the tool. The tool will be available round the clock. The tool caters to two kinds of customers namely, user’s (who lodge requests) and IT service personnel (who resolve requests). HCRMT is an online tool that helps the users at the university to lodge their requests to the service personnel and view the feedback given by the service personnel, where the service personal is responsible to provide a solution according to the user request.

**Project description provided to the students:** You need to come up with a HELP CENTER REQUEST MANAGEMENT TOOL for BTH. The description of the task is given below.

**Project Description:** The online help center at the universities can be very supportive when asked to resolve user’s problems. The purpose of this online request management tool is to provide solutions to the users and make their life easier. At BTH the students have to meet the IT service personnel directly or in- person and have to discuss about their problem. By providing an online management tool the user’s can post their queries at any given point of time, with their given university login username and password.

There are many difficulties while discussing the issue manually with the IT service personnel, such as: **Waste of time and effort:** The user’s have to check the timings for the availability of the IT service personnel and raise their problem, wait until their turn if there are many students already waiting and wait till the problem is fixed. This takes much more time and effort than actually required in solving the problem.
The goal of the teams is to design a mechanism that helps the IT service personnel in solving the problems in an easy manner by already providing the solutions for frequently asked questions related to the category of the problem chosen by service personnel. This reduces time required by the service personnel in solving the problems of similar kind. If a new kind of problem is lodged, the service personnel should be given privilege to provide with a new solution.

The HCRMT operates in two modes namely, user mode and IT service personnel mode

- **User mode**: In the user mode of HCRMT, the students are given the privileges of lodging a request and viewing the feedback.
  - Lodge request: In the lodge request event, the user submits the details of his request, while the IT service personnel searches for past resolution of the same category and populates it to the user to view.
  - View request: In the view request event, the user can view the feedback of the query provided by the IT service personnel.

- **IT service personal mode**: HCRMT offers view and update request privilege to the IT service personnel where he can login to his account, views the request and then provides with a solution according to the user request

The teams have to build four events.
1. User Lodge Request Event.
2. User View Event
4. Service personal edits complaints Event.

The team is free to produce the software, in Java, HTML, and other tools as languages, as they see that fit into the context. For the better understanding of the task the team is provided with the sample of screen shots of how the final outcome should look like.

**Learning objective**: The aim of designing and assigning this task to the students is for the introducing on-line help center management system and also if the students are interested in working on this task, they can add more functionalities to the existing work, which gives them to obtain good credits for their course work also for their better future.

**Pre requisites for the students**: We planned to conduct a survey for the students, by asking them question on the task related topics such as Java, My SQL. Based on the results of the survey and their performance in the course work; we will select the suitable students for the project.
7.1.1.2 Expected Experimental Outcome

**Screen1:** User Login Form

![User Login Form Image]

**Screen2:** User Lodge request form

![User Lodge Request Form Image]

**Screen3:** User selects the category of the problem

![User Selecting Category Image]
**Screen4:** After giving the description of the problem along with the student id and phone number, student sends request to the help center

**Screen5:** Service personal views the complaint and sends the reply to the user
7.2 Appendix B

7.2.1 Final Experiment Planning

7.2.1.1 Pre-experiment online survey form

<table>
<thead>
<tr>
<th>Question</th>
<th>Fully agree</th>
<th>Mostly agree</th>
<th>Mostly disagree</th>
<th>Fully disagree</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Student Name:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2. Please state your age in years</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3. Please enter your gender</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4. Please state your nationality</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5. How many months of professional experience do you have as a software</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>developer/programmer?</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6. Do you have a skype account?</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7. Do you have a gmail account?</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8. Please provide your details, to which extent you agree with the following statements</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1. I am a complete novice in Java
2. I am an experienced programmer
3. I am comfortable working in teams

7.2.1.2 Experimental Task

1. **Background**
The new pay station you developed for K-Town is so successful that other towns also want to set up similar pay stations. Your new customers, however, need support for different currencies and also for different fee structures.

2. **Use Cases / User Stories**
The basic use cases are the same as in the previous version (see Laboration 3 – Pay Station). In addition there are two new features: (i) the pay station shall accept further currencies and (ii) the pay station should support different fee structures.

(i) Besides US Dollars, the pay station shall accept Swedish Kronor and Euros:
   - **US$**
     - Valid coins: 5c, 10c, 25c, 50c
     - Standard rate: 5c for 2 minutes of parking
   - **SEK**
     - Valid coins: 1kr, 5kr, 10kr
     - Standard rate: 1SEK for 5 minutes of parking
   - **Euro**
     - Valid coins: 1c, 2c, 5c, 10c, 20c, 50c, 1€
Standard rate: 5c for 3 minutes of parking

(ii) The pay station shall support different ways to calculate parking time:
   • A linear rate, according to the standard rate for the respective currency (see above).
   • A progressive rate where parking becomes more expensive the longer you park.
     - The first 60 minutes are according to the standard rate, e.g., 5c for 2 minutes of parking in US$ (and accordingly for the other currencies)
     - The second 60 minutes are 5c for 1.5 minutes of parking in US$
     - From the third hour onwards, the rate is twice the standard rate, e.g., 5c for 1 minute of parking in US$
   • A progressive rate where parking becomes cheaper the longer you park.
     - The first 60 minutes are according to the standard rate, e.g., 5c for 2 minutes of parking in US$ (and accordingly for the other currencies)
     - The second 60 minutes are 5c for 3 minutes of parking in US
     - From the third hour onwards, the rate is half the standard rate, e.g., 5c for 4 minutes of parking in US$

3. Other Requirements
   • You have to start from the design described in Laboration 3. You might extend it, if necessary, but not change any of the classes. You can reuse your solution from Laboration 3, but you don’t need to.
   • Implementation has to be done in Java.
   • A simple GUI, mainly for testing purposes has to be implemented as well. The GUI should enable switching currencies and rates.
   • For showing parking tickets and the IllegalCoin exceptions, you can use a pop-up window.
   • The GUI should be separated from the rest of the code.
   • You should use consistent coding guidelines.
   • Switching currencies and rate calculations should be possible at runtime, i.e. without stopping and restarting the application. However, during a transaction the currency and rate calculation must be the same (i.e. you can for example not by a ticket using 10c plus 1kr).
   • You have to work in teams of four. Make sure to note all names on your submission.
   • Your solution should be easy to understand and easy extend with further currencies or other ways of calculating rates.

5. Deliverables
   • A written report containing at least the following
     • A header page with (at least) course name, exercise name and your name(s)
     • An introduction describing the main idea(s) behind your solution
     • A UML Class Diagram of your solution
     • A description of how your solution can be extended
     • A reflection about your work: how you shared the work, which part(s) you perceived as particularly difficult
     • A self-evaluation: what do you think you did well and not so well
   • Documented source code (the .java files), preferably as a zip-archive
   • A working solution, preferably as a .jar file

*Please do not submit complete projects which we have to download and compile first. This consumes a lot of time and space and both are limited.*
1 By *easy to extend* we mean that existing code should preferably be left unchanged when new functionality is added.

7.2.1.3 Expected Outcome

![Image of software development interface](image)

7.2.1.4 Presentation given to the students

Slide 1

Experiment on follow-the-sun software development

By,
Divya Gaddipati
Saroj Pamulapati

Slide 2

Agenda

- Introduction
- Experiment
- Clarification
Global Software Development

- Global software development is defined as the distributing the software development among various teams, who are geographically distributed.
- Terminology:
  - Co-located
  - Follow-the-sun

Co-located

- Co-located: Where the students work together at the same time and at the same place.
- Example: Employees belonging to a particular project in Ericsson, Karlskrona working at same time and place

Follow-the-sun

- Follow-the-sun is a sub division of global software development, where the unfinished work is handed off daily from one development site to another development site
- Work is done on the project for 24 hours instead of 8 hours a day
- Overlapping (Texas-Sweden)
- Non overlapping (California-India)
Experiment

Experiment details

- Scenario’s and team size:
  - Co-located team (4 students - lab G)
  - Follow-the-sun without overlap team
    - Sub-team A: 2 students (lab G)
    - Sub-team B: 2 students (lab H)
  - Follow-the-sun with overlap team
    - Sub-team A: 2 students (lab G)
    - Sub-team B: 2 students (lab H)

Co-located team

- Your task is simple, all the four students are expected to sit together and work on the given lab exercise and submit the deliverables within the deadline.
- Work structure:

<table>
<thead>
<tr>
<th>Time</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>9:00 am</td>
<td>Work (8:00 am - 9:03 am)</td>
</tr>
<tr>
<td>10:06 am</td>
<td>Work (10:06 am - 11:09 am)</td>
</tr>
</tbody>
</table>
Follow the sun without overlap

- Here both the team work at completely different place and time, without any overlap in working hours
- Work structure:
  - Lab G
    - Sub team A (8:00am-9:03am)
    - Sub team B (9:03am-10:06am)
  - Lab H
    - Sub team A (10:06am-11:09am)
    - Sub team B (11:09am-12:12pm)

Follow the sun with overlap

- Here both the team work at completely different place and time, with small overlap in working hours. Overlap of **4 minutes**.
- Work structure:
  - Lab G
    - FTS sub team A (8 am - 8:56 am)
    - FTS sub team B (9 am - 9:56 am)
  - Lab H
    - FTS sub team A (10 am - 10:56am)
    - FTS sub team B (11 am - 12 pm)

Rules

- Lab starts sharp by 8 am
- Cannot leave the lab
- Follow the sun without overlap
  - No SMS, No audio/video call, No chats
- Follow the sun with overlap
  - Can communicate only during the overlapping time using chats, audio/video conferencing,
Can reach us at: divya.gaddipati@gmail.com
saroipsmail@gmail.com
+46734784546
+46734784547

Thank you

Questions?
7.2.1.5 Experiment Instruments

7.2.1.5.1 Guidelines given to the students before the experiment

1. If something is unclear about the program that you are supposed to develop during the lab, please do talk with your team members or ask the supervisor.
2. Lab will start sharp by 8:00 am.
3. You are not supposed to leave the lab temporarily during the lab hours that is from 8:00 am till 12:00 pm except to use the washroom facilities or the reason of illness.
4. Goal of the experiment: Try completing the task within the deadline. That is fine even if you do not finish the work but please note that the communication rules and deadlines have to be followed strictly. Communication rules are specific for each of the arrangement (that is the co-located team, FTS with overlap team and FTS with no overlap team). So please check the rules based on the arrangement (that is co-located team or FTS with overlap or FTS with no overlap) that you are supposed to follow.
5. The lab assignment is divided into a number of modules. So please inform the experiment leads when you are done with first part of the task, before starting the next module we would note the time.
6. If you have any doubts before the lab session regarding the way the experiment will be carried out or implemented do not hesitate to contact us through mail divya.gaddipati@gmail ,sarojpamulapati@gmail, or through phone : +46734784546 , +46734784547.
7. During the experiment we will be there in the lab if you have any issues do ask us.

7.2.1.5.2 Guidelines for co-located team

1. The team works in a typical manner without any particular communication rules to be followed.
2. Please inform the experiment leads when you are done with each module before starting the next module as they would note done the time.
7.2.1.5.3 Guidelines for follow-the-sun with overlap team

1. Remember that the other sub-team working in the other lab is also part of your team and the final grade for the lab is same for each of the team members and as the assignment is not graded individually so do work in a team by helping each other.

2. Note that you are working in a sequential task distribution manner, that is if sub-team A starts the work during their lab hours and handovers the task to the sub-team B after the lab hours, then sub-team B will continue to work on the task from the point sub-team A has stopped and finally deliver it to sub-team A again, this way the work progresses. So, you will probably would like to ensure that the program is readable, easy to understand, following the coding conventions and well-commented so that it becomes easier for the other sub-team who is working in the other lab to understand the deliverables without wasting much time in trying to figure out what you have done so far.

3. You may have to organize your work as follows: work efficiently for the first 45-50 minutes and in the last 10-15 minutes zip the documents that you have developed till then and mail them to the other sub-team. Make sure that your mail is sent before the deadline. You can also describe your progress in the email body for the other sub-team, where you explain the details of what you have done.

4. You may want to organize your work as follows: work effectively for 30-35 minutes and in the last 10-15 minutes zip the documents that you have developed till then and mail them to the other team. You have 15 minutes of time overlap with the other sub-team where you can communicate with the other team. Try to make the most of this time to update the other sub-team on the progress and clarify if there are any issues.

5. Both the sub team members can communicate with each other through phone calls, SMS, IM, e-mails and through audio/video conferencing, but ONLY during the overlap time that is 15 minutes. Strictly not after or before the overlap time. Please do not delete the e-mails, chats, SMS and call logs as we will need them for our experiment analysis. Note that when you use e-mails to communicate please put a CC to experiment leads. Please do not violate this rule. Our e-mail ID’s 1. Diga10@student.bth.se, 2. Sapc10@student.bth.se.

6. In case of change in the code developed by the other sub team members comment the code stating the reason. Doing so would be easier for you as well as the other sub-team members to keep track of the changes.

7. Even in the case, where you get stuck up while writing the code or need any help from the other sub-team members, please do not communicate with other sub-team members after the permitted overlapping hours.
Guidelines for follow-the-sun without overlap team

1. Remember that the other sub-team working in the other lab is also part of your team and the final grade for the lab is same for each of the team members and as the assignment is not graded individually so do work in a team by helping each other.

2. Note that you are working in a sequential task distribution manner, that is if sub-team A starts the work during their lab hours and handovers the task to the sub-team B after the lab hours, then sub-team B will continue to work on the task from the point sub-team A has stopped and finally deliver it to sub-team A again, this way the work progresses. So, you will probably would like to ensure that the program is readable, easy to understand, following the coding conventions and well-commented so that it becomes easier for the other sub-team who is working in the other lab to understand the deliverables without wasting much time in trying to figure out what you have done so far.

3. You may have to organize your work as follows: work efficiently for the first 45-50 minutes and in the last 10-15 minutes zip the documents that you have developed till then and mail them to the other sub-team members. Make sure that your mail is sent before the deadline. You can also describe your progress in the email body for the other sub-team, where you explain the details of what you have done.

4. Communication between the sub-team members is allowed ONLY through e-mails. It is strictly prohibited to use IM, SMS, phone calls or face-to-face communication, for the clarity of the experiment. Please do not delete the e-mails as we will need them for our experiment analysis.

5. In case of change in the code developed by the other sub team members comment the code stating the reason. Doing so would be easier for you as well as the other sub-team members to keep track of the changes.

6. Kindly follow the guidelines without any kind of communication, except with the supervisor or the experiment leads.
7.2.1.5.5 Feedback form for the Co-located and the follow-the-sun without overlap students after each iteration

This form is to be filled in by one person in a group, by discussing with your other team members. The intention of this form is to find out your experience in working in a co-located environment and note down if there are any problems that you encountered during the experiment. Kindly fill in the form. Your valuable feedback will help us to do good thesis!

1. What did you do in this current session?

2. What was blocking your progress of implementing the task? (if any)

3. Were there any problems related to:
   a. Collaboration (collaboration: the act of working together):
   b. Communication (communication: the exchange of thoughts, messages, information through signals, writing, speech or behavior):
   c. Coordination (co-ordination: synchronization and integration of activities, responsibilities) problems?

4. Did you trust your team members who were working along with you in the group?

Thank you for your participation and cooperation.
7.2.1.5.6 Feedback form for the Follow-the-sun with overlap scenario after the first iteration

As you have completed your first session in follow the sun with overlap scenario, we want to note down your experience. So kindly fill in the form. Your valuable feedback will help us to do a good thesis!

1. What did you do in this session?

2. What do you expect the other sub-team members to complete before you receive the work again? Have you expressed your expectations to the other sub-team?

3. Was the communication you had with the other sub-team members sufficient for you? Would you require any other means of communication for the successful completion of the task? (For overlapping scenario members)

4. Are there any impediments while performing the task?

5. Number of compile time errors in the code that were not resolved before sending to the other team? (In the code that is sent to the other team)

Thank you for your participating and cooperation
Feedback form for the follow-the-sun without overlap team from the second iteration onwards

The intension of this form is to analyze your progress of reaching the goal. Kindly fill in the form. Your valuable feedback will help us to do a good thesis!

1. What did you do in this session? (In terms of the task-task progress)

2. What did you do in the last session?

3. Are you satisfied with the amount of work done by the other sub-team members?

4. Where you able to understand the code developed by the other sub-team members?

5. What are the challenges/impediments you faced while working in the FTS environment?

6. What do you expect the other sub-team members to complete before you receive the work again? Have you expressed your expectations to the other sub-team members?

7. Was the code developed by the other team meeting your expectations?

8. Did you face any integration problems? (Here integration refers to understanding each other’s code and finally developing the code)

9. Was the communication you had with the other sub-team members sufficient for you? (For overlapping and no overlapping members) Would you require any other means of communication for the successful completion of the task?

10. Were the documents handed over to you by the other sub-team members understandable/easy to understand?

11. Which answer is more suitable to characterize the work division in your team:
   a. We split the modules among the team members – this way we integrated modules developed individually at the end.
   b. We worked together in the co-located sub-team, but we don’t know how the other sub-team organized the work.

12. Did you find the work done by other sub-team members useful to you for the progress of your task? If no, what are the reasons?

13. Are there any impediments for performing the task?

14. Number of compile time errors in the code that were not resolved before sending to the other team? (in the code that is sent to the other team)

Thank you for your participation and cooperation
### 7.2.1.5.8 Self progress form after the experiment

This form has to be filled in individually. The intention of this form is to get feedback from each individual who has taken part in the experiment. Kindly fill in the form. Your valuable feedback will help us to do good thesis!

1. Describe your overall experience in working in FTS environment? In a scale of 1 to 5 rate your experience. (1= didn’t like working/unhappy, 2: OK, 3: satisfied/happy, 4: good, 5: Excellent)

2. What kind of difficulties did you come across while performing the task in the experimental setting?

3. Did you complete the given task in the specified deadline? If no, what were the reasons for not completion of the task?

4. Did you trust the other sub-team members of your group? (trust: Did you feel that other sub-team members in the other lab were cooperating with you)

5. Did you feel that you required any kind of synchronous communication (chat, SMS and phone calls) for discussing with your other team members while performing the task? (For non-overlapping team)

6. As the experiment is conducted in the lab, along other classmates, did you face any distractions while working? (Such as noise etc which hindered your progress)? If yes, what kind of disturbances did you face while working?

7. Was the overlapping time sufficient enough to communicate with the team? (For Overlapping Team). If no, how much more do you require for successful completion of the task?

8. Were there any doubts that where unclear and not resolved until the next iteration? (For Non Overlapping team)

9. Your suggestions for improvement of the process?

**Thank you for your participation and co operation**

### 7.2.1.5.9 Time spent by the students in each phase

Table 26: Time spent in each phase

<table>
<thead>
<tr>
<th>Time (in minutes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Catching up</td>
</tr>
<tr>
<td>Development time</td>
</tr>
<tr>
<td>Communication time</td>
</tr>
<tr>
<td>Reporting time</td>
</tr>
</tbody>
</table>
7.2.1.6 Measurement tables

7.2.1.6.1 Calculation of the actual development time ($T_{\text{actual}}$)

Table 27: Calculation of actual development time

<table>
<thead>
<tr>
<th>Treatment Name</th>
<th>Group Name</th>
<th>Background Experience</th>
<th>Iteration (Time)</th>
<th>Total development time (in minutes)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>8:00 am - 9:00 am</td>
<td>9:00am - 10:00 am</td>
</tr>
<tr>
<td>Co-located</td>
<td>Group P</td>
<td>Experienced</td>
<td>8:10 am – 9:03 am</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Group S</td>
<td>Non-experienced</td>
<td>8:13 am – 9:16 am</td>
<td>-</td>
</tr>
<tr>
<td>FTS with overlap</td>
<td>Group Q</td>
<td>Experienced</td>
<td>8:10 am – 8:56 am, 8:56 am - 9:00 am</td>
<td>9:00 am – 9:52 am, 9:56 am – 10:00 am</td>
</tr>
<tr>
<td></td>
<td>Group T</td>
<td>Non-experienced</td>
<td>8:07 am – 8:52 am</td>
<td>8:55 am – 9:48 am, 9:49 am – 9:52 am</td>
</tr>
<tr>
<td>FTS without overlap</td>
<td>Group R</td>
<td>Experienced</td>
<td>8:00 am – 9:01 am</td>
<td>9:03 am – 10:04 am</td>
</tr>
<tr>
<td></td>
<td>Group U</td>
<td>Non-experienced</td>
<td>8:02 am – 9:03 am</td>
<td>9:03 am – 10:04 am</td>
</tr>
</tbody>
</table>

7.2.1.6.2 Calculation of time spent in the programming stage ($T_{\text{PT}}$)

Table 28: Calculation of time spent in programming stage

<table>
<thead>
<tr>
<th>Treatment Name</th>
<th>Group Name</th>
<th>Background Experience</th>
<th>Iteration</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>8:00 am - 9:00 am</td>
</tr>
<tr>
<td>Co-located</td>
<td>Group P</td>
<td>Experienced</td>
<td>8:20am – 9:02 am</td>
</tr>
<tr>
<td></td>
<td>Group S</td>
<td>Non-experienced</td>
<td>8:20am – 9:15 am</td>
</tr>
<tr>
<td>FTS with overlap</td>
<td>Group Q</td>
<td>Experienced</td>
<td>8:20 am – 8:55 am</td>
</tr>
<tr>
<td></td>
<td>Group T</td>
<td>Non-experienced</td>
<td>8:20 am – 8:50am</td>
</tr>
</tbody>
</table>
7.2.1.6.3 Calculation of catching up time ($T_{CT}$)

Table 29: Calculation of time spent in catching up stage

<table>
<thead>
<tr>
<th>Treatment Name</th>
<th>Group Name</th>
<th>Background Experience</th>
<th>Iteration</th>
<th>Total catching up time</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>8:00 am - 9:00 am</td>
<td>9:00 am - 10:00 am</td>
</tr>
<tr>
<td>Co-located</td>
<td>Group P</td>
<td>Experienced</td>
<td>8:10 am – 8:20 am</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Group S</td>
<td>Non-experienced</td>
<td>8:13 am – 8:20 am</td>
<td>-</td>
</tr>
<tr>
<td>FTS with overlap</td>
<td>Group Q</td>
<td>Experienced</td>
<td>8:10 am – 8:20 am</td>
<td>9:00am – 9:05am</td>
</tr>
<tr>
<td></td>
<td>Group T</td>
<td>Non-experienced</td>
<td>8:07 am – 8:20 am</td>
<td>8:55am – 9:09am</td>
</tr>
<tr>
<td>FTS without overlap</td>
<td>Group R</td>
<td>Experienced</td>
<td>8:00 am – 8:10 am</td>
<td>9:01 am – 9:05am</td>
</tr>
<tr>
<td></td>
<td>Group U</td>
<td>Non-experienced</td>
<td>8:02 am – 8:16 am</td>
<td>9:03 am – 9:08 am</td>
</tr>
</tbody>
</table>
### 7.2.1.6.4 Calculation of reporting time ($T_{RT}$)

Table 30: Calculation of time spent in reporting stage

<table>
<thead>
<tr>
<th>Treatment Name</th>
<th>Group Name</th>
<th>Background Experience</th>
<th>Iteration</th>
<th>Total reporting time</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>8:00am - 9:00 am</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>9:00am - 10:00 am</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>10:00 am - 11:00 am</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>11:00 am - 12:00 pm</td>
<td></td>
</tr>
<tr>
<td>Co-located</td>
<td>Group P</td>
<td>Experienced</td>
<td>9:02am – 9:03am</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Group S</td>
<td>Non-experienced</td>
<td>9:15am – 9:16 am</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>10:06am – 11:29am</td>
<td></td>
</tr>
<tr>
<td>FTS with overlap</td>
<td>Group Q</td>
<td>Experienced</td>
<td>8:55am – 8:56am</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Group T</td>
<td>Non-experienced</td>
<td>8:50am – 8:52am</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>10:52am – 11:50am</td>
<td></td>
</tr>
<tr>
<td>FTS without overlap</td>
<td>Group R</td>
<td>Experienced</td>
<td>8:58am – 9:01am</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Group U</td>
<td>Non-experienced</td>
<td>9:00am – 9:03 am</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>10:05am – 12:09am</td>
<td></td>
</tr>
</tbody>
</table>

### 7.2.1.6.5 Calculation of Effort

Table 31: Calculation of effort

<table>
<thead>
<tr>
<th>Treatment Name</th>
<th>Group Name</th>
<th>Background Experience</th>
<th>Iteration</th>
<th>Total effort</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>8:00am - 9:00 am</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>9:00am - 10:00 am</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>10:00am - 11:00am</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>11:00am - 12:00 pm</td>
<td></td>
</tr>
<tr>
<td>Co-located</td>
<td>Group P</td>
<td>Experienced</td>
<td>8:10am – 9:03 am</td>
<td>544</td>
</tr>
<tr>
<td></td>
<td>Group S</td>
<td>Non-experienced</td>
<td>8:13am – 9:16 am</td>
<td>504</td>
</tr>
<tr>
<td>FTS with overlap</td>
<td>Group Q</td>
<td>Experienced</td>
<td>8:10am – 8:56am</td>
<td>434</td>
</tr>
<tr>
<td></td>
<td>Group T</td>
<td>Non-experienced</td>
<td>8:07am – 8:52am</td>
<td>444</td>
</tr>
</tbody>
</table>
### 7.2.1.6.6 Calculation of effort at each iteration

Table 32: Calculation of effort spent in each iteration

<table>
<thead>
<tr>
<th>Treatment Name</th>
<th>Group Name</th>
<th>Background Experience</th>
<th>8:00am-9:00 am</th>
<th>9:00 am-10:00 am</th>
<th>10:00 am-11:00 am</th>
<th>11:00 am-12:00 am</th>
</tr>
</thead>
<tbody>
<tr>
<td>Co-located</td>
<td>Group P</td>
<td>Experienced</td>
<td>212</td>
<td>-</td>
<td>332</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Group S</td>
<td>Non-experienced</td>
<td>252</td>
<td>-</td>
<td>252</td>
<td>-</td>
</tr>
<tr>
<td>FTS with overlap</td>
<td>Group Q</td>
<td>Experienced</td>
<td>108</td>
<td>120</td>
<td>120</td>
<td>86</td>
</tr>
<tr>
<td></td>
<td>Group T</td>
<td>Non-experienced</td>
<td>90</td>
<td>118</td>
<td>134</td>
<td>102</td>
</tr>
<tr>
<td>FTS without overlap</td>
<td>Group R</td>
<td>Experienced</td>
<td>122</td>
<td>122</td>
<td>126</td>
<td>120</td>
</tr>
<tr>
<td></td>
<td>Group U</td>
<td>Non-experienced</td>
<td>122</td>
<td>122</td>
<td>116</td>
<td>134</td>
</tr>
</tbody>
</table>

### 7.2.1.6.7 Calculation of Lines of code for each iteration

Table 33: Calculation of lines of code

<table>
<thead>
<tr>
<th>Treatment Name</th>
<th>Group Name</th>
<th>Background Experience</th>
<th>Iteration</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>8:00 am-9:00 am</td>
</tr>
<tr>
<td>Co-located</td>
<td>Group P</td>
<td>Experienced</td>
<td>254</td>
</tr>
<tr>
<td></td>
<td>Group S</td>
<td>Non-experienced</td>
<td>306</td>
</tr>
<tr>
<td>FTS with overlap</td>
<td>Group Q</td>
<td>Experienced</td>
<td>328</td>
</tr>
<tr>
<td></td>
<td>Group T</td>
<td>Non-experienced</td>
<td>329</td>
</tr>
<tr>
<td>FTS without overlap</td>
<td>Group R</td>
<td>Experienced</td>
<td>454</td>
</tr>
<tr>
<td></td>
<td>Group U</td>
<td>Non-experienced</td>
<td>402</td>
</tr>
</tbody>
</table>
# Appendix C

## 7.3.1 Literature Review Tables

### 7.3.1.1 Search for primary studies

Table 34: Search for primary studies

<table>
<thead>
<tr>
<th>S. No</th>
<th>Database</th>
<th>Search String</th>
<th>Total number of papers found</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Engineering village</td>
<td>(&quot;global software engineering&quot; OR &quot;global software development&quot; OR &quot;distributed software engineering&quot; OR &quot;distributed software development&quot; OR &quot;multi-site software development&quot; OR &quot;multisite software development&quot; OR (&quot;multisite development&quot; AND &quot;software&quot;) OR (&quot;multi-site development&quot; AND &quot;software&quot;) OR &quot;collaborative software development&quot; OR &quot;collaborative software engineering&quot; OR (&quot;globally distributed work&quot; AND &quot;software&quot;) OR &quot;geographically distributed software development&quot; OR (&quot;distributed development&quot; AND &quot;software&quot;) OR (&quot;cross continent development&quot; AND &quot;software&quot;) OR &quot;cross-site software development&quot; OR (&quot;cross-site development&quot; AND &quot;software&quot;) AND (&quot;Follow-the-sun&quot; OR &quot;follow the sun&quot; OR &quot;round the clock development&quot; OR &quot;24-hour development&quot; OR &quot;24-hour knowledge factory&quot; OR &quot;24/7&quot; OR &quot;24/365&quot;) ) Refine with keywords:&quot;Follow-the-sun&quot;</td>
<td>28</td>
</tr>
<tr>
<td>2.</td>
<td>IEEE Xplore</td>
<td>(&quot;global software engineering&quot; OR &quot;global software development&quot; OR &quot;distributed software engineering&quot; OR &quot;distributed software development&quot; OR &quot;multi-site software development&quot; OR &quot;multisite software development&quot; OR (&quot;multisite development&quot; AND &quot;software&quot;) OR (&quot;multi-site development&quot; AND &quot;software&quot;) OR &quot;collaborative software development&quot; OR &quot;collaborative software engineering&quot; OR (&quot;globally distributed work&quot; AND &quot;software&quot;) OR &quot;geographically distributed software development&quot; OR (&quot;distributed development&quot; AND &quot;software&quot;) OR (&quot;cross continent development&quot; AND &quot;software&quot;) OR &quot;cross-site software development&quot; OR (&quot;cross-site development&quot; AND &quot;software&quot;) AND (&quot;Follow-the-sun&quot; OR &quot;follow the sun&quot; OR &quot;round the clock development&quot; OR &quot;24-hour development&quot; OR &quot;24-hour knowledge factory&quot; OR &quot;24/7&quot; OR &quot;24/365&quot;)</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>3.</td>
<td>Springer Link</td>
<td>(“global software development” OR “distributed software development” OR “collaborative software development”) AND (“follow the sun” OR “round the clock development” OR “24-hour development” OR “24-hour knowledge factory”)</td>
<td>27</td>
</tr>
<tr>
<td>4.</td>
<td>ISI Web of Science</td>
<td>(“global software engineering” OR &quot;global software development&quot; OR &quot;distributed software engineering&quot; OR &quot;distributed software development&quot; OR &quot;multi-site software development&quot; OR &quot;multisite software development&quot; OR &quot;multisite development&quot; AND &quot;software&quot;) OR (&quot;multi-site development&quot; AND &quot;software&quot;) OR &quot;collaborative software development&quot; OR &quot;collaborative software engineering&quot; OR (&quot;globally distributed work&quot; AND &quot;software&quot;) OR &quot;geographically distributed software development&quot; OR (&quot;distributed development&quot; AND &quot;software&quot;) OR (&quot;cross continent development&quot; AND &quot;software&quot;) OR &quot;cross-site software development&quot; OR (&quot;cross-site development&quot; AND &quot;software&quot;) AND (&quot;Follow-the-sun&quot; OR &quot;follow the sun&quot; OR &quot;round the clock development&quot; OR &quot;24-hour development&quot; OR &quot;24-hour knowledge factory&quot; OR &quot;24/7&quot; OR &quot;24/365&quot;)</td>
<td>7</td>
</tr>
<tr>
<td>5.</td>
<td>Science Direct</td>
<td>(“global software engineering” OR &quot;global software development&quot; OR &quot;distributed software engineering&quot; OR &quot;distributed software development&quot; OR &quot;multi-site software development&quot; OR &quot;multisite software development&quot; OR (&quot;multisite development&quot; AND &quot;software&quot;) OR (&quot;multi-site development&quot; AND &quot;software&quot;) OR &quot;collaborative software development&quot; OR &quot;collaborative software engineering&quot; OR (&quot;globally distributed work&quot; AND &quot;software&quot;) OR &quot;geographically distributed software development&quot; OR (&quot;distributed development&quot; AND &quot;software&quot;) OR (&quot;cross continent development&quot; AND &quot;software&quot;) OR &quot;cross-site software development&quot; OR (&quot;cross-site development&quot; AND &quot;software&quot;) AND (&quot;Follow-the-sun&quot; OR &quot;follow the sun&quot; OR &quot;round the clock development&quot; OR &quot;24-hour development&quot; OR &quot;24-hour knowledge factory&quot; OR &quot;24/7&quot; OR &quot;24/365&quot;)</td>
<td>24</td>
</tr>
</tbody>
</table>
| 6. | ACM Digital library | (“global software development”)
Refine with keyword: “Follow-the-sun” | 23 |
| 7. | Wiley inter science journal finder | ("global software engineering" OR "global software development" OR "distributed software engineering" OR "distributed software development" OR "multi-site software development" OR "multisite software development" OR ("multi site development" | 25 |
| database | Total number of papers found in all the databases | 141 |
7.4 Appendix D

7.4.1 Kappa Coefficient

“It is a statistical measure to evaluate the nominal scale agreement between the fixed pair of rater [59].”

As we know that the thinking process various from person to person, researchers are becoming aware that the observer (rater) as an important source of measurement error [50]. So in order to find out the observers variability [52] kappa coefficient is calculated. In this thesis, kappa coefficient is calculated for the stage 3 and stage 4 of the study selection criteria, as both authors have evaluated the papers separately.

The kappa coefficient is calculated as,

\[ k = \frac{\bar{p} - \bar{p}_e}{1 - \bar{p}_e} \]

Where,

\( \bar{p} - \bar{p}_e \) represents the degree of argument attained in excess of change and the \( 1 - \bar{p}_e \) represents the degree of argument attainable over and above what would be predicted by chance [52]. The value kappa coefficient value that represents the strength of agreement level between the observers given by Koch and Landis [52] is represented in the table below.

<table>
<thead>
<tr>
<th>Kappa statistics</th>
<th>Strength of agreement</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;0.00</td>
<td>Poor</td>
</tr>
<tr>
<td>0.00-0.20</td>
<td>Slight</td>
</tr>
<tr>
<td>0.21-0.40</td>
<td>Fair</td>
</tr>
<tr>
<td>0.41-0.60</td>
<td>Moderate</td>
</tr>
<tr>
<td>0.61-0.80</td>
<td>Substantial</td>
</tr>
<tr>
<td>0.81-1.00</td>
<td>Almost perfect</td>
</tr>
</tbody>
</table>

The kappa coefficient i.e. the inter observer variation is based on the difference between how much agreement is actually present is compared to how much agreement is expected to be present by chance [5]. The steps involved in calculation of k, is represented below.

<table>
<thead>
<tr>
<th>Author 1</th>
<th>Author 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>a</td>
</tr>
<tr>
<td>No</td>
<td>c</td>
</tr>
<tr>
<td>Total</td>
<td>n_1</td>
</tr>
<tr>
<td>Yes</td>
<td>b</td>
</tr>
<tr>
<td>No</td>
<td>d</td>
</tr>
<tr>
<td>Total</td>
<td>n_0</td>
</tr>
<tr>
<td></td>
<td>n</td>
</tr>
</tbody>
</table>

Where, the (a) represents, the number of times the two observers agree, (d) represents the number of time the two observers disagrees, (c) represents the number of times observer 1 agrees while observer 2 disagrees and finally (b) represents the number of times observer 1 disagrees while observer 2 agrees.
The $\overline{P}$ is calculated as $\frac{1}{n}[a + d]$

The $\overline{p}_e$ is calculated as $\left[\left(\frac{n_1}{n} \times \frac{m_1}{n}\right) + \left(\frac{n_0}{n} \times \frac{m_0}{n}\right)\right]$

7.4.1.1 Calculation of kappa coefficient for the stage 3 of the study selection criteria

In the stage 3 of the study selection criteria papers are selected by reading through the title and the abstract. In this stage, a total of 44 papers were selected by the authors, where author 1 selected 47 papers where as author 2 selected 46 papers that are relevant to the research field according to their perspective. From the total of 80 papers included in the stage 1, 44 papers were included in this stage. As there is a difference in the results of the included papers by both authors, the kappa coefficient is calculated by comparing the included papers of both authors. A matrix is formed for comparing the similarities and differences in the results of both authors.

<table>
<thead>
<tr>
<th>Author 2</th>
<th>Author 1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>Yes</td>
<td>44</td>
</tr>
<tr>
<td>No</td>
<td>2</td>
</tr>
<tr>
<td>Total</td>
<td>46</td>
</tr>
</tbody>
</table>

$k = \frac{0.962 - 0.508}{1 - 0.508} = 0.9227$

7.4.1.2 Calculation of kappa coefficient for the stage 4 of the study selection criteria

In the stage 4 of the study selection criteria papers are selected by reading through the full text. In this stage, a total of 16 papers were selected by the authors, where author 1 selected 16 papers where as author 2 selected 19 papers that are relevant to the research field according to their perspective. From the total of 44 papers included in the stage 3, 16 papers were included in this stage. As there is a difference in the results of the included papers by both authors, the kappa coefficient is calculated by comparing the included papers of both authors.

<table>
<thead>
<tr>
<th>Author 2</th>
<th>Author 1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>Yes</td>
<td>14</td>
</tr>
<tr>
<td>No</td>
<td>2</td>
</tr>
<tr>
<td>Total</td>
<td>16</td>
</tr>
</tbody>
</table>

$k = \frac{0.847 - 0.524}{1 - 0.524} = 0.678$
This thesis is submitted to the School of Computing at Blekinge Institute of Technology in partial fulfillment of the requirements for the degree of Master of Science in Software Engineering. The thesis is equivalent to 2 X 20 weeks of full time studies.

Contact Information:
Author(s):
Saroj Pamulapati
Address: Karlskrona, Sweden
E-mail: sarojpsmail@gmail.com

Divya Gaddipati
Address: Karlskrona, Sweden
E-mail: divya.gaddipati@gmail.com

University Advisor & Guide:
Ms. Darja Šmite
Assistant Professor

School of Computing
Blekinge Institute of Technology
SE-371 79 Karlskrona
Sweden

Internet : www.bth.se/com
Phone : +46 455 38 50 00
Fax : +46 455 38 50 57
1.1.1.2 Coordination of Software Development Activities in global software development

Global software development is the norm today for the technology companies [50] and every company that engages in global software development wants to achieve the benefits by developing the software at a cheaper, faster and with best quality [24]. For this reason organizations have started to engage in different kinds of sourcing strategies [43] such as offshoring, outsourcing, offshore outsourcing, nearshoring and onshoring. The different kinds of organizational relationships that exists are represented in the below figure 1.

<table>
<thead>
<tr>
<th>Same Country</th>
<th>National insourcing</th>
<th>Outsourcing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Different Country</td>
<td>Offshore insourcing</td>
<td>Offshore outsourcing</td>
</tr>
<tr>
<td>Same organization</td>
<td>Different organization</td>
<td></td>
</tr>
</tbody>
</table>

Figure 1: Organizational relationships [24]

The term sourcing is defined as “the act through which work is contracted or delegated to an external or internal entity that could be physically located anywhere [43]”. Outsourcing is defined as “contracting with the third party service provider for the management and completion of certain amount of work, for a specified length of time, cost and level of service [43]”. Offshore outsourcing is defined as “subcontracting a process, such as a product development, to a third party company that is located in a different country [24]”. Offshore insourcing is defined as “relocation of work by a company from one country to another [24]”. And finally national insourcing is termed as relocation of work by a company within the country.

Now by achieving a clear idea about the different organizational relationships that exists in global software development, the next step is to know, how the tasks are distributed among the organizations. Before knowing about the different task allocation strategies, it should be noted that global software development is not about only outsourcing [24], it is complex concept that can be understood through the different types of organizational relationships and task distribution methods [24]. The different ways of organizing global software development projects are
• **Phase-based development:** A particular phases of the software development cycle is relocated, for example testing.

![Phase-based development](image)

Figure 2: Phase-based development

• **Customization:** Here the software is developed in one site and later customized by each site based on their local market requirements.

![Customization](image)

Figure 3: Customization

• **Modularization:** In this scenario, modules are relocated to different sites thereby causing the development to take place in parallel fashion.

![Modularization](image)

Figure 4: Modularization

• **Follow-the-sun approach:** In this scenario, software is developed sequentially, where software activities are transferred on a daily basis between development sites which are many time zones apart. The rest of the thesis is based on this approach hence this concept will be explained in detail in the subsequent sections.
implemented with these three sites. When it is 5 PM in Darwin, Australia where people are about to leave their work place, it is 8:30 AM in London, UK where people are just about to start their work. Hence the work that has been done till then by Darwin, Australia colleagues can be handed over to the colleagues in London, UK who are just about to start their day and by continuing to work from the point where the Darwin, Australia colleagues have stopped working on the project. Similarly, when it is 5 PM in London, UK it is 9 AM in San Francisco, California, USA. Hence the so far completed work by the London, UK team can be handed over to the team in San Francisco, California, USA who has just started their work and later the work is handed over to the team in Darwin, Australia. In this fashion the work can be handed over from one site to the other, allowing the development to take place continuously throughout the day by effectively utilizing the 24-hours a day without any halt.

This kind of round-the-clock development is also termed as Sequential Collaborative Software Engineering (SCSE) [6] as development takes place in a sequential manner. So in this follow-the-sun approach, instead of just working on the project for eight hours a day, the whole 24-hours in a day are effectively utilized which helps to reduce the development cycle time and thereby reducing the time-to-market by helping the organizations to be at a higher competitive edge over the others. It is said that the development duration reduces by 50% when there are two development sites, by 67% when there are three development sites [31]. The main motivation for the software development organizations to implement follow-the-sun approach is reduced time-to-market, which is a crucial factor in order to survive in this competitive world.

1.1.2.1 Definition of follow-the-sun

Carmel [29] who is a well-known researcher in the field of global software development coined the term follow-the-sun approach as

“A type of global knowledge workflow, which is designed in order to reduce the development time, in which the knowledge product is owned and evolved by a production site and is handed over daily to the next production site that is many times zones apart to continue with the so far completed work” [29].

Amar Gupta et. al [2, 3, 7] who is also into research on 24-hour development defines the term follow-the-sun approach as
2.1.1.3.4 Study Quality Assessment

The quality of the selected study is determined in this step. Here the authors determine if the study addresses the following questions or not. Here the quality criterion is used to only determine the quality of the included studies but not exclude the selected primary study.
2.2 Experiment

2.2.1 Motivation

Software development is a complex activity that takes lots of time, consists of many activities and documents that have to be written before the product is delivered, and involves many people. Many companies are trying to come up with new approaches to develop software in a systematic and disciplined way by improving the software process to deliver a product with better quality, lower development cost etc. In this process, researchers are coming up with new proposals that are supposed to provide many more advantages compared to the existing ones. It is important that the proposals are evaluated first to determine the actual effect, whether it works as promised or not, rather than using it directly. For this reason we have chosen to conduct an experiment to evaluate the proposal called follow-the-sun approach which promises of reducing the development time by 50% when compared to the traditional co-located approach.

Experiment aims at evaluating human-based approaches in a “systematic, disciplined, quantifiable and controlled way” [13]. An experiment allows the researcher to have control over the execution, measurement and ease of replication is high with an experiment. So in order to have full control over the execution process a controlled experiment is conducted and also due to cost, time and other constraints authors conducted the experiment in the university instead of selecting software professionals as subjects of the experiment. In our thesis we are comparing two software development approaches that is traditional co-located approach and follow-the-sun approach. Experiment is the best suitable option when there is more than one treatment to be compared. Hence the results obtained from the controlled experiment provide better knowledge for the organizations when implementing follow-the-sun approach in real time projects.

Experiment is not a one shot process but consists of a number of activities. The various stages in our experimental process are: initial experiment definition, initial experiment planning, refinement to the initial planning, final experiment definition, final experiment planning, experiment design and experiment execution. Each of these stages is detailed in the following sections to give a clear idea of the whole experimental process.

![Figure 9: Overview of the experimental process](image)

During the starting stages of this thesis, the author’s intension of conducting the experiment was modified to a greater extent due to the various constraints that the authors had come across. For this reason, in this chapter the authors explain in detail from the initial proposed plan till the final experiment that was executed in this thesis. Here in this chapter the term initial refers to what the authors have intended to perform in the thesis proposal and the term final refers to what the authors have actually carried out in this thesis.
as the subjects for the experiment. The selection of the subjects would have been done through the pre-evaluation survey which would have tested their programming knowledge (on Java, Html, My SQL, and Tomcat- As the experiment is based on these programming languages) and experience of the subjects (if any).

2.2.3.1 Experimental setting

2.2.3.1.1 Experimental design

The subjects would have been selected and grouped into teams with 4 students in each team. The selected subjects would have been assigned to work under each of the three scenarios in such a way that the number of subjects is balanced among the three scenarios. And also the subject of the all the scenarios would have been given the same functionality to develop.

2.2.3.1.2 Experiment setup

- **Co-located scenario:** The co-located scenario would consist of four members, who would have worked for three hours a day. Where all the team members would have worked together at the same time and at the same place. The deadline for the co-located scenario would have been two days and after that they would have to deliver the project. The deadline of the task would have been extended, if in case they were unable to complete the task within the specified deadline. Extending the deadline of the task, would have enabled us to calculate the total amount of time taken by the students in completing the task.

- **Follow-the-sun scenario:** In follow-the-sun approach, each team would have consisted of 4 members, where the 4 members would have been divided into sub-team A and sub-team B. The sub-teams A and B would have consisted of two students each working on the same task. The follow-the-sun team would have been expected to complete the task using the same effort in man-hours as that of the co-located scenario, but the calendar duration has to be 1 day instead of the 2 days as that of the co-located scenario. The sub-team A and sub-team B would have been located in a different place, without allowing any type of synchronous communication between the both follow-the-sun sub-teams in

![Diagram](image-url)
order to simulate follow-the-sun environment. The follow-the-sun sub-team A would have worked for three hours and then would have handed over the task to the sub-team B. The sub team B then would have started working for the next three hours from the point when the sub-team A has stopped working. The task would have been communicated between the sub-teams through the use of asynchronous communication tools. We have planned to artificially impose the deadline for the follow-the-sun team as 1 day which is exactly the half the duration when compared to the co-located scenario, which would have worked for two days, where the logic being that follow-the-sun approach reduces the development duration by 50%. The deadline of the task would have been extended if in case the follow-the-sun team was unable to complete the task within the specified deadline for determining by how much time the follow-the-sun team would have been faster or slower when compared with the co-located scenario.

Follow-the-sun team would have worked under two different scenarios. The two scenarios are detailed below:

- **Development-Development Scenario:**

  In case of development-development scenario, the task was to be started by the follow-the-sun sub-team A and at the end of their time period the work was to be handed over to the follow-the-sun sub-team B who were present in a different place and in different time zone. The sub-team B would have continued the work from the point where the sub-team A has stopped working. This cycle would have continued until the fixed deadline of the task is met. Here both the follow-the-sun sub-teams would have worked continuously by handing over the work to each other after their working hours. The sub-teams would have been given a detailed requirements specification about how the task has to be carried out. Based on the requirement specifications the students would have been expected to develop the given application and also the developed code was to be unit tested by both the sub-team members in order to check if there were any errors in the developed code.

- **Development-Testing Scenario:**

  In case of development-testing scenario, the sub-team A would have been assigned the responsibility of coding whereas the sub-team B would have tested the code developed by
All the instruments for the experiment execution were made ready and the subjects of the experiment were also selected before the execution of the experiment. The selected student group names, timings and the seating arrangements of the selected students were posted in the It’s learning (student portal) along with the guidelines and the design of the experiment. The students were informed to look into the posted details and contact the authors through email if they have any doubts regarding the experiment before the start of the experiment execution.

Central repository for the students was the pen drive. Students after their working hours copied all the task related files onto the pen drive with a naming convention (their batch name and their working hours) that was followed and handed over it to the experiment leads.

2.2.8.1 Experimental setup

The three software development scenarios or treatments in this experiment are: co-located scenario, follow-the-sun with overlap scenario and follow-the-sun without overlap scenario. All the team members of the three scenarios were given the same functionality to develop. In the experiment, the development time required to complete the given task and the cost efficiency of each scenario is measured. The experiment is designed in such a way that two days were simulated in just four hours where sub-group A works during the day while sub-group B works during the night in case of follow-the-sun approach. The experimental set up for each of the three scenarios is detailed below.

- **Co-located scenario**: This team worked in a typical student manner i.e. working only during the day. That is the working hours for this scenario are: day one from 8:00 am to 9:03 am and on day two from 10:06 am to 11:09 am. The groups P and S worked in the co-located scenario, where group P was placed in lab G and the group S was placed in lab H.

- **FTS with overlap**: In this scenario the four members in the team were split into two sub-groups: sub-group A and sub-group B where each sub-group consists of two members. On the day one, sub-group A starts working on the task first during the day from 8:00 am and 9:00 am, where from 8:00 am – 8:56 am student’s work on the given task and in last 4 minutes that is from 8:56 am – 9:00 am students can communicate with the sub-group B students. Now the sub-group B works during the night on the day one from 9:00 am till 9:56 am and after their working hours from 9:56 am till 10:00 am they handover the task to the sub-group A and communicate with them. On day two, the working hours for the sub-group A during the day are from 10:00 am till 10:56 am and from 10:56 am till 11:00 am they can communicate with the sub-group B students. And during the night, the working hours for the sub-group B are from 11:00 am and12:00 pm where in the last iteration students were expected to deliver the task. Both the sub-group members were placed in different rooms to simulate the geographical distribution between the team members and synchronous communications such as phones, instant messenger etc were allowed to use only during the overlap time. Here the follow-the-sun sub-groups work continuously by handing over the work to each other after their working hours.
Sub-group A
(Lab G)

Sub-group B
(Lab H)

Figure 13: Working of the follow-the-sun with overlap scenario

- **FTS without overlap:** In this scenario the four members in the team were split into two sub teams: sub-group A and sub-group B where each sub-group consists of two members. On day one, sub-group A works during the day from 8:00 am and 9:03 am, while the sub-group B members work during the night on day one from 9:03 am till 10:06 am. On day two, the working hours for the sub-group A during the day are from 10:06 am till 11:09 am. And during the night, the working hours for the sub-group B are from 11:09 am and 12:12 pm, where in the last iteration students are expected to deliver the task. Both the sub-group members were placed in different rooms to simulate the geographical distribution between the team members where the sub-groups are not allowed to use any kind of synchronous communication such as face-to-face meeting, phone, instant messenger etc.

From the above experimental setup it can be observed that the working of the follow-the-sun without overlap scenario is same as follow-the-sun with overlap scenario, but the only difference is that in case follow-the-sun without overlap there is no direct communication between the sub-group members. But the sub-group members of the follow-the-sun without overlap scenario can use asynchronous communication medium such as email, where they can send mail to the other sub-group members regarding the progress of the task, clarification regarding the code that has been developed by the other sub-group members etc. It such be noted that the response to the mail would only come when the sub-group members are back to work on the next day as it is night for the other sub-group members when they have sent an email.
3 Results and Analysis

Results and analysis section is divided into two sections, literature review results and results of the empirical research. In literature review results section, the characteristics of the included papers in this thesis and the results of the narrative synthesis are discussed. In results of the empirical research section, the results obtained from the experiment are discussed.

3.1 Systematic Literature Review Results

3.1.1 Included Studies

Seven databases: Engineering Village, IEEE Xplore, Springer Link, ISI Web of Knowledge, Wiley Inter Science Journal Finder, ScienceDirect, ACM Digital Library were searched by the authors, 141 papers were found but finally 16 papers were included based on the relevance of our research. The figure 17 below represents the count of the number of papers found in each of the selected database.

![Included papers from each database](image)

Figure 17: Number of included papers from each database

The figures represent the actual count of the included papers that are obtained in each of the database, without removing the duplicates. During the process of selecting the primary studies it was observed that same papers were obtained in different database. So for the better understanding of the results the authors of the papers kept the count of the included papers undistributed.

3.1.1.2 Year wise distribution of papers

In the figure 18 below, year wise distribution of the included papers is represented. A total of 16 papers that are published till the time of search i.e. till April 2011, that are found relevant to the research field are represented. Among the 8 papers i.e. 50% papers completely address the implementation of follow-the-sun approach, whereas the remaining 50% of the papers discuss about the global software development by emphasizing on follow-the-sun approach i.e. partially discussing about follow-the-sun approach. From the table below, it is clearly
evident that not much research is done in this field before 2003, which illustrates the fact that
global software development is a 21st century trend [23]. It can be observed from Figure 16
that, from 2003 there is gradual increase in the research field that is 81% of published papers
are between the years 2006-2010. Note that the statistics for 2011 can be incomplete since
the search was conducted in April 2011 and the most recent articles could have not yet been
added to the databases for that reason, maybe we could not find any papers that are published
in 2011.

![Number of papers published per year](image)

Figure 18: Number of published papers per year

3.1.1.3 Distribution of research methods

The various research methods we have found in the selected studies are case studies,
experiments, interviews and models proposed. Among the various research methods, the
most common one is the models proposed. From the figure below it can be noted that in 37% 
of the papers models were proposed in order to find out the optimal favourable conditions for
the implementation of follow-the-sun approach. The authors of other studies reported results 
from case studies, findings based on their experience in industry or obtained through surveys 
or interviews.

Among the 16 papers, 21% of the papers are based on experiments. There are various kinds of
experiments such as controlled experiments, laboratory experiments and industrial
experiments. And some of the experiments are based on the model proposed. As the
researchers started finding it difficult to find out the optimal conditions for the
implementation of follow-the-sun strategy, they started with mathematical models [P5] 
[P13]. The remaining 42% of the papers are distributed among the case studies (16%) and
studies based on interviews (21%). And there was one paper which is based on experience
report.
3.1.1.4 Subjects of investigation

A subject of investigation refers to whether the research is carried out in an industry or in an academic setting. From the figure below it can be noticed that among the 16 papers that are included for the study, there is almost equal distribution of the papers in both industrial and academic setting. 7 papers reported studies that are based on academic background, which can be concluded that the researchers are putting their efforts in finding out the different scenarios with the goal of reducing the development time. The main purpose of the follow-the-sun approach is to reduce the development time for the organization, so even though researchers experiment in finding out the different scenarios ultimately the organizations have to adapt to the new process. From the results it was observed that the organizations are facing lot of challenges in implementing distributed development and also as it requires a huge capital for implementing a new process, not many organizations are opting for distributed development. From the figure below, it can be noticed from the 16 papers, only 9 of the papers research was carried out in industrial background. This lack of implementation of distributed development in industries might be because of various reasons such as difficulty in finding out the right locations, coordination, communication and control challenges faced etc. So for this reason, the researchers are conducting various experiments [P5], [P6] and proposing mathematical models [P5], [P13] to find out the ideal scenario with the goal of reducing the development time.
3.1.1.5 Narrative synthesis results

Follow-the-sun is a global software development method, in which handover of work takes place every day from one development site to another development site that are many time zones apart. There by reducing the development time by 50% if there are two sites involved, by 67% if there are three sites involved and soon on. Resulting in reduced time to market thereby achieving higher competitive edge over the others especially for industries where there is a sense of urgency as products get out-dated quickly. This reduction in development time is rather perceived as an assumed benefit because of availability of only few documented evidences available of achieving the reduction in development time. So here we review the literature to determine if follow-the-sun approach works or not, determine the challenges involved while using follow-the-sun approach, when is it best to use, to achieve the benefit of reduced development time? Etc. Table 13 gives a brief overview of the included studies.

Table 13: Overview of the included studies

<table>
<thead>
<tr>
<th>Paper no</th>
<th>Methodology</th>
<th>Context</th>
<th>The main contribution of the paper</th>
</tr>
</thead>
<tbody>
<tr>
<td>[P1]</td>
<td>Model proposed</td>
<td>Academia</td>
<td>-Models and implements an internet - based prototype system that uses Web-based interaction approach along with a unique data model that collects and stores design rationale and history from stakeholders and workers. -The model also reflects some of the important characteristics of FTS.</td>
</tr>
<tr>
<td>[P2]</td>
<td>Experiment</td>
<td>Industry</td>
<td>-It has been noticed that both FTS and co-located teams differ in the use of technology and knowledge sharing processes but not in efficiency and quality.</td>
</tr>
</tbody>
</table>
The goal of the teams is to design a mechanism that helps the IT service personnel in solving the problems in an easy manner by already providing the solutions for frequently asked questions related to the category of the problem chosen by service personnel. This reduces time required by the service personnel in solving the problems of similar kind. If a new kind of problem is lodged, the service personnel should be given privilege to provide with a new solution.

The HCRMT operates in two modes namely, user mode and IT service personnel mode

- **User mode**: In the user mode of HCRMT, the students are given the privileges of lodging a request and viewing the feedback.
  - Lodge request: In the lodge request event, the user submits the details of his request, while the IT service personnel searches for past resolution of the same category and populates it to the user to view.
  - View request: In the view request event, the user can view the feedback of the query provided by the IT service personnel.

- **IT service personal mode**: HCRMT offers view and update request privilege to the IT service personnel where he can login to his account, views the request and then provides with a solution according to the user request

The teams have to build four events.
1. User Lodge Request Event.
2. User View Event
4. Service personal edits complaints Event.

![Diagram showing HCRMT system](image)

**Figure 24: Overview of initial experiment plan**

The team is free to produce the software, in Java, HTML, and other tools as languages, as they see that fit into the context. For the better understanding of the task the team is provided with the sample of screen shots of how the final outcome should look like.

**Learning objective**: The aim of designing and assigning this task to the students is for the introducing on-line help center management system and also if the students are interested in working on this task, they can add more functionalities to the existing work, which gives them to obtain good credits for their course work also for their better future.

**Pre requisites for the students**: We planned to conduct a survey for the students, by asking them question on the task related topics such as Java, My SQL. Based on the results of the survey and their performance in the course work; we will select the suitable students for the project.
7.2 Appendix B

7.2.1 Final Experiment Planning

7.2.1.1 Pre-experiment online survey form

1. Student Name:
2. Please state your age in years
3. Please enter your gender
4. Please state your nationality
5. How many months of professional experience do you have as a software developer/programmer?
6. Do you have a skype account?
7. Do you have a gmail account?
8. Please provide your details, to which extent you agree with the following statements

<table>
<thead>
<tr>
<th></th>
<th>Fully agree</th>
<th>Mostly agree</th>
<th>Mostly disagree</th>
<th>Fully disagree</th>
</tr>
</thead>
<tbody>
<tr>
<td>I am a complete novice in Java</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I am an experienced programmer</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I am comfortable working in teams</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

7.2.1.2 Experimental Task

1. **Background**
The new pay station you developed for K-Town is so successful that other towns also want to set up similar pay stations. Your new customers, however, need support for different currencies and also for different fee structures.

2. **Use Cases / User Stories**
The basic use cases are the same as in the previous version (see Laboration 3 – Pay Station). In addition there are two new features: (i) the pay station shall accept further currencies and (ii) the pay station should support different fee structures.

(i) Besides US Dollars, the pay station shall accept Swedish Kronor and Euros:
   - **US$**
     - Valid coins: 5c, 10c, 25c, 50c
     - Standard rate: 5c for 2 minutes of parking
   - **SEK**
     - Valid coins: 1kr, 5kr, 10kr
     - Standard rate: 1SEK for 5 minutes of parking
   - **Euro**
     - Valid coins: 1c, 2c, 5c, 10c, 20c, 50c, 1€
Slide 9

**Follow the sun without overlap**

- Here both the team work at completely different place and time, without any overlap in working hours.
- Work structure:
  - Lab G
    - Sub team A (9:00 am - 11:00 am)
  - Lab H
    - Sub team B (9:00 am - 11:00 am)

Slide 10

**Follow the sun with overlap**

- Here both the team work at completely different place and time, with small overlap in working hours. Overlap of 4 minutes.
- Work structure:
  - Lab G
    - FTS sub team A (8 am - 8:56 am)
    - FTS sub team A (10 am - 10:56 am)
  - Lab H
    - FTS sub team B (8 am - 8:56 am)
    - FTS sub team B (11 am - 12 pm)

Slide 11

**Rules**

- Lab starts sharp by 8 am
- Cannot leave the lab
- Follow the sun without overlap
  - No SMS, No audio/video call, No chats
- Follow the sun with overlap
  - Can communicate only during the overlapping time using chats, audio/video conferencing,
7.2.1.5 Experiment Instruments

7.2.1.5.1 Guidelines given to the students before the experiment

1. If something is unclear about the program that you are supposed to develop during the lab, please do talk with your team members or ask the supervisor.
2. Lab will start sharp by 8:00 am.
3. You are not supposed to leave the lab temporarily during the lab hours that is from 8:00 am till 12:00 pm except to use the washroom facilities or the reason of illness.
4. Goal of the experiment: Try completing the task within the deadline. That is fine even if you do not finish the work but please note that the communication rules and deadlines have to be followed strictly. Communication rules are specific for each of the arrangement (that is the co-located team, FTS with overlap team and FTS with no overlap team). So please check the rules based on the arrangement (that is co-located team or FTS with overlap or FTS with no overlap) that you are supposed to follow.
5. The lab assignment is divided into a number of modules. So please inform the experiment leads when you are done with first part of the task, before starting the next module we would note the time.
6. If you have any doubts before the lab session regarding the way the experiment will be carried out or implemented do not hesitate to contact us through mail divya.gaddipati@gmail, sarojpamulapati@gmail, or through phone: +46734784546, +46734784547.
7. During the experiment we will be there in the lab if you have any issues do ask us.

7.2.1.5.2 Guidelines for co-located team

1. The team works in a typical manner without any particular communication rules to be followed.
2. Please inform the experiment leads when you are done with each module before starting the next module as they would note done the time.
7.2.1.5.3 Guidelines for follow-the-sun with overlap team

1. Remember that the other sub-team working in the other lab is also part of your team and the final grade for the lab is same for each of the team members and as the assignment is not graded individually so do work in a team by helping each other.

2. Note that you are working in a sequential task distribution manner, that is if sub-team A starts the work during their lab hours and handovers the task to the sub-team B after the lab hours, then sub-team B will continue to work on the task from the point sub-team A has stopped and finally deliver it to sub-team A again, this way the work progresses. So, you will probably would like to ensure that the program is readable, easy to understand, following the coding conventions and well-commented so that it becomes easier for the other sub-team who is working in the other lab to understand the deliverables without wasting much time in trying to figure out what you have done so far.

3. You may have to organize your work as follows: work efficiently for the first 45-50 minutes and in the last 10-15 minutes zip the documents that you have developed till then and mail them to the other sub-team. Make sure that your mail is sent before the deadline. You can also describe your progress in the email body for the other sub-team, where you explain the details of what you have done.

4. You may want to organize your work as follows: work effectively for 30-35 minutes and in the last 10-15 minutes zip the documents that you have developed till then and mail them to the other team. You have 15 minutes of time overlap with the other sub-team where you can communicate with the other team. Try to make the most of this time to update the other sub-team on the progress and clarify if there are any issues.

5. Both the sub team members can communicate with each other through phone calls, SMS, IM, e-mails and through audio/video conferencing, but ONLY during the overlap time that is 15 minutes. Strictly not after or before the overlap time. Please do not delete the e-mails, chats, SMS and call logs as we will need them for our experiment analysis. Note that when you use e-mails to communicate please put a CC to experiment leads. Please do not violate this rule. Our e-mail ID’s 1. Diga10@student.bth.se, 2. Sapc10@student.bth.se.

6. In case of change in the code developed by the other sub team members comment the code stating the reason. Doing so would be easier for you as well as the other sub-team members to keep track of the changes.

7. Even in the case, where you get stuck up while writing the code or need any help from the other sub-team members, please do not communicate with other sub-team members after the permitted overlapping hours.
7.2.1.5.4 Guidelines for follow-the-sun without overlap team

1. Remember that the other sub-team working in the other lab is also part of your team and the final grade for the lab is same for each of the team members and as the assignment is not graded individually so do work in a team by helping each other.

2. Note that you are working in a sequential task distribution manner, that is if sub-team A starts the work during their lab hours and handovers the task to the sub-team B after the lab hours, then sub-team B will continue to work on the task from the point sub-team A has stopped and finally deliver it to sub-team A again, this way the work progresses. So, you will probably would like to ensure that the program is readable, easy to understand, following the coding conventions and well-commented so that it becomes easier for the other sub-team who is working in the other lab to understand the deliverables without wasting much time in trying to figure out what you have done so far.

3. You may have to organize your work as follows: work efficiently for the first 45-50 minutes and in the last 10-15 minutes zip the documents that you have developed till then and mail them to the other sub-team members. Make sure that your mail is sent before the deadline. You can also describe your progress in the email body for the other sub-team, where you explain the details of what you have done.

4. Communication between the sub-team members is allowed ONLY through e-mails. It is strictly prohibited to use IM, SMS, phone calls or face-to-face communication, for the clarity of the experiment. Please do not delete the e-mails as we will need them for our experiment analysis.

5. In case of change in the code developed by the other sub team members comment the code stating the reason. Doing so would be easier for you as well as the other sub-team members to keep track of the changes.

6. Kindly follow the guidelines without any kind of communication, except with the supervisor or the experiment leads.
7.2.1.5.5 Feedback form for the Co-located and the follow-the-sun without overlap students after each iteration

This form is to be filled in by one person in a group, by discussing with your other team members. The intention of this form is to find out your experience in working in a co-located environment and note down if there are any problems that you encountered during the experiment. Kindly fill in the form. Your valuable feedback will help us to do good thesis!

1. What did you do in this current session?

2. What was blocking your progress of implementing the task? (if any)

3. Were there any problems related to:
   a. Collaboration (collaboration: the act of working together):
   b. Communication (communication: the exchange of thoughts, messages, information through signals, writing, speech or behavior):
   c. Coordination (co-ordination: synchronization and integration of activities, responsibilities) problems?

4. Did you trust your team members who were working along with you in the group?

    Thank you for your participation and cooperation
7.2.1.5.6 Feedback form for the Follow-the-sun with overlap scenario after the first iteration

As you have completed your first session in follow the sun with overlap scenario, we want to note down your experience. So kindly fill in the form. Your valuable feedback will help us to do good thesis!

1. What did you do in this session?

2. What do you expect the other sub-team members to complete before you receive the work again? Have you expressed your expectations to the other sub-team?

3. Was the communication you had with the other sub-team members sufficient for you? Would you require any other means of communication for the successful completion of the task? (For overlapping scenario members)

4. Are there any impediments while performing the task?

5. Number of compile time errors in the code that were not resolved before sending to the other team? (In the code that is sent to the other team)

Thank you for your participating and cooperation
7.2.1.5.7 Feedback form for the follow-the-sun without overlap team from the second iteration onwards

<table>
<thead>
<tr>
<th>Question</th>
</tr>
</thead>
<tbody>
<tr>
<td>The intension of this form is to analyze your progress of reaching the goal. Kindly fill in</td>
</tr>
<tr>
<td>the form. Your valuable feedback will helps us to do good thesis!</td>
</tr>
<tr>
<td>1. What did you do in this session? (In terms of the task-task progress)</td>
</tr>
<tr>
<td>2. What did you do in the last session?</td>
</tr>
<tr>
<td>3. Are you satisfied with the amount of work done by the other sub-team members?</td>
</tr>
<tr>
<td>4. Where you able to understand the code developed by the other sub-team members?</td>
</tr>
<tr>
<td>5. What are the challenges/ impediments you faced while working in the FTS environment?</td>
</tr>
<tr>
<td>6. What do you expect the other sub-team members to complete before you receive the work</td>
</tr>
<tr>
<td>again? Have you expressed your expectations to the other sub-team members?</td>
</tr>
<tr>
<td>7. Was the code developed by the other team meeting your expectations?</td>
</tr>
<tr>
<td>8. Did you face any integration problems ?(here integration refers to understanding each other’s code and finally developing the code)</td>
</tr>
<tr>
<td>9. Was the communication you had with the other sub-team members sufficient for you? (For overlapping and no overlapping members) Would you require any other means of communication for the successful completion of the task?</td>
</tr>
<tr>
<td>10. Were the documents handed over to you by the other sub-team members understandable/ easy</td>
</tr>
<tr>
<td>to understand?</td>
</tr>
<tr>
<td>11. Which answer is more suitable to characterize the work division in your team:</td>
</tr>
<tr>
<td>a. We split the modules among the team members – this way we integrated modules</td>
</tr>
<tr>
<td>developed individually at the end.</td>
</tr>
<tr>
<td>b. We worked together in the co-located sub-team, but we don’t know how the other sub-</td>
</tr>
<tr>
<td>team organized the work.</td>
</tr>
<tr>
<td>12. Did you find the work done by other sub team members useful to you for the progress of</td>
</tr>
<tr>
<td>your task? If no, what are the reasons?</td>
</tr>
<tr>
<td>13. Are there any impediments for performing the task?</td>
</tr>
<tr>
<td>14. Number of compile time errors in the code that were not resolved before sending to the</td>
</tr>
<tr>
<td>other team? (in the code that is sent to the other team)</td>
</tr>
</tbody>
</table>

Thank you for your participation and cooperation
7.2.1.5.8 Self progress form after the experiment

This form has to be filled in individually. The intension of this form is to get feedback from each individual who has taken part in the experiment. Kindly fill in the form. Your valuable feedback will helps us to do good thesis!

1. Describe your overall experience in working in FTS environment? In a scale of 1 to 5 rate your experience. (1= didn’t like working/unhappy, 2: OK, 3: satisfied/happy, 4:good, 5: Excellent)

2. What kind of difficulties did you come across while performing the task in the experimental setting?

3. Did you complete the given task in the specified deadline? If no, what were the reasons for not completion of the task?

4. Did you trust the other sub-team members of your group? (trust: Did you feel that other sub-team members in the other lab were cooperating with you)

5. Did you feel that you required any kind of synchronous communication (chat, SMS and phone calls) for discussing with your other team members while performing the task? (For non-overlapping team)

6. As the experiment is conducted in the lab, along other classmates, did you face any distractions while working? (Such as noise etc which hindered your progress)? If yes, what kind of disturbances did you face while working?

7. Was the overlapping time sufficient enough to communicate with the team? (For Overlapping Team). If no, how much more do you require for successful completion of the task?

8. Were there any doubts that where unclear and not resolved until the next iteration?(For Non Overlapping team)

9. Your suggestions for improvement of the process?

Thank you for your participation and cooperation

7.2.1.5.9 Time spent by the students in each phase

Table 26: Time spent in each phase

<table>
<thead>
<tr>
<th>Time (in minutes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Catching up</td>
</tr>
<tr>
<td>Development time</td>
</tr>
<tr>
<td>Communication time</td>
</tr>
<tr>
<td>Reporting time</td>
</tr>
</tbody>
</table>
7.4  Appendix D

7.4.1  Kappa Coefficient

“It is a statistical measure to evaluate the nominal scale agreement between the fixed pair of rater [59].”

As we know that the thinking process various from person to person, researchers are becoming aware that the observer (rater) as an important source of measurement error [50]. So in order to find out the observers variability [52] kappa coefficient is calculated. In this thesis, kappa coefficient is calculated for the stage 3 and stage 4 of the study selection criteria, as both authors have evaluated the papers separately. The kappa coefficient is calculated as,

\[ k = \frac{\bar{p} - \bar{p}_e}{1 - \bar{p}_e} \]

Where,

\( \bar{p} - \bar{p}_e \) represents the degree of argument attained in excess of change and the \( 1 - \bar{p}_e \) represents the degree of argument attainable over and above what would be predicted by chance [52]. The value kappa coefficient value that represents the strength of agreement level between the observers given by Koch and Landis [52] is represented in the table below.

<table>
<thead>
<tr>
<th>Kappa statistics</th>
<th>Strength of agreement</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;0.00</td>
<td>Poor</td>
</tr>
<tr>
<td>0.00-0.20</td>
<td>Slight</td>
</tr>
<tr>
<td>0.21-0.40</td>
<td>Fair</td>
</tr>
<tr>
<td>0.41-0.60</td>
<td>Moderate</td>
</tr>
<tr>
<td>0.61-0.80</td>
<td>Substantial</td>
</tr>
<tr>
<td>0.81-1.00</td>
<td>Almost perfect</td>
</tr>
</tbody>
</table>

The kappa coefficient i.e. the inter observer variation is based on the difference between how much agreement is actually present is compared to how much agreement is expected to be present by chance [5]. The steps involved in calculation of k, is represented below.

<table>
<thead>
<tr>
<th>Author 2</th>
<th>Author 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Yes</td>
<td>a</td>
</tr>
<tr>
<td>No</td>
<td>c</td>
</tr>
<tr>
<td>Total</td>
<td>n_1</td>
</tr>
</tbody>
</table>

Where, the (a) represents, the number of times the two observers agree, (d) represents the number of time the two observers disagrees, (c) represents the number of times observer 1 agrees while observer 2 disagrees and finally (b) represents the number of times observer 1 disagrees while observer 2 agrees.