Master Thesis
Electrical Engineering
April 2014

The Impact of Waiting Time Distributions on QoE of Task-Based Web Browsing Sessions

Nazrul Islam and Vijaya John David Elepe

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 Electrical Engineering. The thesis is equivalent to 20 weeks of full time studies.

This master thesis is typeset using LaTeX

Contact Information:

Author(1):
Nazrul Islam
Address: Minervavägen 20 LGH 1114
371 41 Karlskrona
E-mail: nazrul.mbstu@gmail.com

Author(2):
Vijaya John David Elepe
Address: Valhallavagen 2 flats 1301
371 41 Karlskrona
E-mail: john77david@gmail.com

University advisor:
Junaid Shaikh
School of Computing

University Examiner:
Dr. Patrik Arlos
School of Computing

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

There has been an enormous growth in the Internet usage in recent years, fueled by the increasing number of multimedia applications and widespread availability of World Wide Web (WWW). The end-user generally accesses these applications through web browsing activities. These time-critical services often suffer from the delays ranging from small chunks to long peaks which can have severe implications on the Quality of Experience (QoE). Hence, it is worthwhile to identify the impact of different variations of delay on the end-user QoE.

This research focused on the end-user QoE for three different distributions of delays occurring during an e-commerce shopping experiment. By keeping the overall waiting time of every sessions same, the study shows that the end-user QoE is different for different variety of delays. And the research also concludes that, the users prefer small frequently occurring delays as compared to the long rarely occurring delays within a task-driven web browsing session.

Keywords: Web Browsing-QoE, Waiting Time Perception, Temporal QoE, Network delays, Network Emulator.

Acknowledgments

We would like to thank and express our deepest appreciation and sincere gratitude to our supervisor Juniad Shaikh, who inspired us to take pride in our research; his enthusiasm for research efforts will have a significant effect on our future research. His advice throughout this process kept us focused on the right course. It was an experience of a life time. We are really thankful to him for sharing his ideas and insights.

We are grateful to Prof. Markus Fiedler for his encouragements.

In the end, we would like to thank our friends, especially Golam Furkani Quraishy, Pangkaj Chandra Paul, Shafiqul Islam, Sadh Ibna Rahmat, Monir Hossain, Abdur Razzak and Vinod Kumar. They supported us tremendously with their reviews and comments throughout the work.

Nazrul Islam
Vijaya John David Elepe

April 2014

Preface

This master thesis is outlined based on the results obtained from the laboratory experiment. This is carried out in the Department of Electrical Engineering with emphasis on Telecommunication Systems, School of Computing, at Blekinge Institute of Technology in Karlskrona, Sweden.

This thesis includes six chapters which are briefed as follows:

Chapter-1
Chapter 1 provides a detailed discussion on the importance of the work that has been done and why the current topic is selected for Master Thesis. It gives the demonstration of the research question and the methodology to answer these questions. Also, it gives the information about the contribution of several researches in this field.

Chapter-2
Chapter 2 discusses about the technical prerequisites relevant to understand the results.

Chapter-3
Chapter 3 discusses about the experiment and how is it implemented to capture users perceived QoE.

Chapter-4
Chapter 4 gives the information about how data is collected and calculated for further analysis.

Chapter-5
Chapter 5 provides a detailed discussion based on the results obtained.

Chapter-6
Chapter 6 provides conclusions to the discussions based on the results and proves ideas for future scope.

Appendix
Appendix covers information about every relevant technical work that has been done.

Contents

List of Figures

List of Tables

List of Abbreviations

ACR	Absolute Category Rating
BIND	Berkley Internet Naming Daemon
CPU	Central Processing Unit
CSS	Cascading Style Sheets
DNS	Domain Name System
DOM	Document Object Model
GPU	Graphics Processing Unit
HTML	HyperText Markup Language
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
IP	Internet Protocol
ISPs	Internet Service Providers
ITU-T	ITU Telecommunication Standardization Sector
JSON	JavaScript Object Notation
MOS	Mean Opinion Score

OS	Operating System
PHP	Hypertext Preprocessor
CPU	Central Processing Unit
PLT	Page Load Time
QoE	Quality of Experience
QoS	Quality of Service
RDBMS	Relational Database Management System
SSH	Secure Shell
TCP	Transmission Control Protocol
TLS	Transport Layer Security
URL	Uniform Resource Locator
WinHTTP	Microsoft Windows HTTP Services
WinINET	Windows Internet
XML	Extensible Markup Language

Chapter 1

Introduction

People are relying heavily for day to day needs on the Internet and its wide domain of applications and services. These applications and services are mainly accessed through the World Wide Web (WWW). The dominant protocol of the WWW is Hypertext Transfer Protocol (HTTP) which is used for delivering these applications and services from the server to the end-users. HTTP generally uses the Transmission Control Protocol (TCP) as the transport protocol. For this reason, a major amount of applications and services use the client-server model. In this model, the client sends a request to the server and the server responds with a reply to that particular request. These requests-responses are all carried out by the TCP protocol. Though the end-users are guaranteed to be served because of the TCP protocol, there might be some waiting time before service consumption [1]. The waiting time is defined as the time between a client sending a request to the server and the response to that request which is fully visible to the client.

User can face waiting time for many reasons. Let us consider a simple web page transfer. When an end-user selects a simple hyperlink on a web page, the browser immediately sends the request to the appropriate server. The server then invokes the appropriate software for preparing the response. This response is sent to the client through the transfer medium between the server and client. The client's browser receives the response, which renders and displays the content on the web page to the end-user. There are lots of parties involved in delivering a simple web page. The reason for waiting time may occur because of unavailability or inefficiency of any of these parties during the transfer. Usually, the dominant reason for waiting time is the unavailability of the transfer medium between the server and the client. Then the data has to be delivered than the resources available in the medium, data get lost. As TCP guarantees the delivery of the data packet, eventually every packet gets transferred but in between the end-user faces waiting time [2]. Network impairments are very common phenomena in the Internet as well as the waiting time.

The waiting time can be measured objectively by monitoring the conventional network performance parameters. However, the subjective perception of the waiting time depends on the individual user and overall underlying timing systems in the human brain [3]. The perception of time is processed by the human brain by transforming the physical signals into electrical signals in the nervous system. For web-based services, mainly visual stimuli are in action which is processed less efficiently than the auditory stimuli. In order to measure the effect of visual stimuli on users' perception, the perceived duration is compared to a tolerance threshold [4] serving as a reference. If the perceived duration is shorter than the tolerance threshold, the user interprets that as fast and decent. Conversely, if the duration is perceived longer than the tolerance threshold, the user interprets the duration as slow and insufficient. The value of this tolerance threshold varies and is influenced by the browsing context, personal factors, past experiences etc.

Depending on the visual stimuli established the end-user waiting time as the key determinant of QoE in the domain of web browsing [5, 6]. The users are more dissatisfied with the service if they wait for the web page transactions to complete. It is very essential to reduce the waiting time for any applications or services to succeed. This is crucial to examine how the end-user reacts to the waiting time if occurred during any service consumption. Moreover, it is required to investigate the random appearances of such network disturbances when network setting varies within a single web- service. Also, when these disturbances are longer than expected and around the threshold level [4] occurring in an unpredictable pattern. Thus, studies are needed to investigate the impact of different delays around the threshold level on QoE over time. It would be interesting to evaluate, if the end-users perceive different delay sessions randomly, how the end-user perception changes to a website and for assessing its QoE.

This research focuses on the impact of different chunks of delay on the users, while the overall waiting time for any session is same. This study is based on task-based user subjective tests done in the context of web browsing on an e-commerce website. The e-commerce website consists of five pages for each session and placed on an Apache web server. A DNS server is deployed on the same machine for resolving multiple DNS requests. A network emulator is used to shape all the network traffic in the network, without re-ordering the data packets. The shaper applies the delay based on the direction of data flow being specified. We developed an automated tool to manage the entire experiment. The end-user can have a continuous flow of a real life web browsing experience using this tool in the experiment. A network protocol analyzer tool is used to capture packets on the network-level. A web debugging tool is used to capture the application-level data. The experimental results showed that, end-user QoE is changed for different set of delays. Various sequences of session appearance also influence the end-user QoE.

1.1 Motivation

Study of end-user opinion about the service is a powerful tool to measure the quality of any services in the field of communication and to build understanding about it. This concept of relating end-user perception with the quality of the service has gained an increasing attention by practitioners over the last couple of years. Quality is a significant aspect of any kind of service. The Internet service providers (ISPs) are always eager to offer a better quality to the client. Users are accessing the web based service via a web browser. Even a tiny increment of a fraction of second (s) in the waiting time, significantly affects the overall browsing experience of the user surfing the web page [7]. When there are frequent small disturbances in the network, it results in longer delays on the web. Hence, it is worthwhile to identify the impact of the same amount of delay with different variations on QoE. The random appearances of network disturbance are the result of sudden degradation or raise of the network impairments that have a significant impact on user satisfaction and loyalty. Therefore, it needs to take into account the QoE evaluation of web-based services.

1.2 Aims and Objectives

The central goal of this research is to analyze the impact of different distribution of network delay on end-user QoE in a task-driven web browsing experiment. Also, to find out how user experience changes for different set of delays.

- Investigate how the smallest change of network delay occurring continuously effect end-user QoE in a web browsing sessions.
- To find out how the sudden rise of network delay on one of the web pages affects overall end-user QoE during a web browsing session.
- Find out the relationship between the occurring frequency and duration of network delays and end-user QoE in a web browsing session.

1.3 Research Question

1. What is the effect of low intensity disturbances occurring continuously for a long period of time in a web browsing session?
2. What is the effect of high intensity disturbances going for a small period of time in a web browsing session?
3. How the overall user QoE changes in multiple web browsing session span over time?
4. How the users previous experience of a network disturbance affects the present perception in web browsing QoE?

1.4 Research Methodology

A literature review is performed to find out the details about web transfers and common network delay patterns in the web browsing sessions. Based on the literature study the required software tools and hardware equipment are selected to develop a web browsing experiment. Then all the software tools used in the experiment had been studied and tested separately. Then a preliminary experiment setup is designed which include the hardware setup and software configurations. The experimental setup had a client, a server and a traffic shaper. The traffic shaper was placed between the server and the client to generate a desired pattern of network delays. All communications pass through the traffic shaper. An automated controller had been developed to control every machine used in the experiment in order to keep the whole experiment process automated. Then the delay patterns had been formulated keeping the total amount of delay approximately equal for each session. After implementing the primary design the delay patterns had been verified with sufficient amount of dry runs. The setup had been modified and tested rigorously to ensure the desired browsing environment for the end-user.

The whole web browsing experiment for a single end-user were divided into three different web browsing sessions. The user had to go though the five web pages of an e-commerce web site in each web session. The used e-commerce website in the experiment had been developed emulating popular real world e-commerce web sites. Different distribution of delays had been applied on each session within the five pages. Despite the difference in the delay distribution, the total waiting time of every shopping session was approximately same. In the first session, user perceived smaller change of network delay in all five pages. In the second session user perceived higher change of network delay on the second and fourth page. At last, in the third session, user perceived highest delay only at the third page with other pages having a very small amount of delay. Additionally, we randomized the order in which the above mentioned delay sessions appeared to each of the participants in the experiment.

At the end of each web session users were asked to answer two simple questions to grade the service using ITU-T recommended 5 points ACR scale [8], 5 point quality scale. A training session was conducted for each user before starting the experiment. The users browsed the same web site in three different delay sessions one by one. The experimental data is collected properly and carefully from network, application and user level. Scripts are used to extract required information from the data. After analyzing the data, appropriate graphs are plotted. A careful study of the plots provides a quantitative measure of the effect of different delay patterns.

1.5 Background and Related Work

Web usage includes a wide range of applications and services which generate enormous amount of data traffic over the Internet. In this hugely competitive environment, quality and availability of data are the two major factors for any service to succeed. Service providers always want to provide a better service to their users. In order to do that, they usually monitor the services from the network performance parameters, namely Quality of Service (QoS). But the end-user perception of any service cannot be fully measured only from QoS as the relationship between these two are not linear, i.e. Bandwidth does not linearly transform into page load time. Accurate understanding of the end-user perception can be measured by measuring new criterion namely, Quality of Experience (QoE). The ITU-Telecommunication Standardization Sector (ITU-T) has defined QoE as, "The overall acceptability of an application or a service, as perceived subjectively by the end-user" [9]. Modeling of QoE cannot be limited to only one users condition. The condition varies from user to user as there are several influential factors involved. These influential factors affect and determine the end-user QoE.

Hosfeld et al. [10] has identified some significant influencing factors for the end-user QoE. These factors are categorized into technical, psychological, content and context level. In a similar way, the authors of [11] have grouped the influential factors in user, system, service, application and context level. Moreover, they have categorized all these influencing factors into three groups; these are human, system and context level influential factors. In paper [12], the author described the context level influential factors into spatial and temporal context. Spatial context is the environment of the user (e.g. location, house, office). Temporal context is the temporal aspects of the user experience (e.g. the time of the day, duration and the usage frequency of any service). The variation in the context of any service (spatial or temporal) causes variation in the user interactions, which mainly determines the end-user perception of that particular service. In the context of web browsing, the temporal aspects play a major role [13]. Understanding of these temporal aspects characteristics and their effect on QoE in web browsing is still evolving.

The study of Hosfeld et al. [10], defined the QoE of web services that is based on the HTTP protocol and accessed via a web browser as Web-QoE. There are many variables that had been extended over different areas and can be instrumental in finding out Web-QoE. Ensuring a better Web-QoE for web services is still a subject of intense research. However, in web browsing, the users often have to wait for the content of a requested web page to be transferred from the server, this is the waiting time. Waiting time is a key factor to determine the Web-QoE [2]. Prolonged waiting times cause dissatisfactions to the end-user. The user can have different experiences with the same service in different point of time by the variation of this key factor. In addition, bad experiences tend to be remembered more by the user because of the memory effect. In the study [10], the author states that the memory effect is considered as a key influencing factor for Web-QoE. The users loyalty towards any service heavily depends on the users memory for that particular service.

Moreover, human intentions, needs and feelings vary for different users and also for a same user over time. Perception of human is highly complex, subjective and context dependent. However, there are certain general threshold. The Web-QoE study shows that, the user satisfaction level breaks when waiting times exceeds 9-10 s [14] in a single session. If any user had this particular threshold experience, the user may or may not use the same service again. The importance of limited waiting times was investigated in the case of e-commerce services [15]. This study has pointed out an 8 s limit of page download time to be kept in order to avoid user dissatisfaction.

Generally, users always expect better services with less waiting time. Their expectations on performance grows up if there are further decreases in the waiting times. This happens based on the knowledge and experience of how quick responses could be given by a service. On the other hand, increasing waiting times are perceived by the user as particularly disturbing. QoE researches have so far been dominated by multimedia services. However, growing user base for web services demands the focus to measure the web experiences and its threshold levels. Reference [16] investigates the relationship between limited access speed (i.e. high waiting time) and the quitting of users from the sessions. They found while facing high waiting times, the user tends to leave or close the service which are clear signals that users lost patience. Also in the study [2], users were given the opportunity to break the download of a picture once they ran out of patience, which typically happened after 10 to 20 s. However, small amounts of waiting times affect the users' perception in very different ways and which are occurring very often for web based services.

There are two types of small waiting time identified by the researcher, one is network induced waiting time and another is user think time. The study of Shaikh et al. [17] outlines the difference between the networks induced traffic gaps and user think time. The duration of traffic gap is generated due to user think time and network outage during a transfer. The network outage that results in the freezes in video transfer of the web are often constituted of duration between 1-4 s. Using wavelet analysis they also observed, the traffic gaps 1-4 s specify the sign of poor quality data transfer. Also, the users perceived quality of any web service is related to the waiting time. All these waiting times, shorter and longer contributes to cross the user tolerance level and the tolerance of a user has particular significance when the user is browsing an e-commerce site.

The estimate for user tolerance of QoS for an e-commerce website is well studied on [5]. They they have discussed the how to improve the server performance for a shopping website. They described the acceptability duration threshold for a user interaction with a website. These user requirements are integrated into the system design for better performance. The users perceived delay may not be similar to the actual delay on the page for any web service. The study [18] focuses on the user perceived delay and the tolerance. The results showed that shorter time delays are overestimated and longer delays are underestimated. These states that the delay between 2-6 s can be estimated accurately by any user. Also the study found an interesting effect between the type of delay and the type of the task. The author in [19] states that, users QoE over a service develops over time with a temporal rim around the user QoE with pre and post experiences. The past experiences and the expectations of the user influence the present user QoE.

Consequently, a number of studies have been conducted with the goal to quantify the relationships between Web QoE, application-level metrics (such as response and download times) and the QoS of the underlying network. In the study [20], the author describes the Quantitative relation between the factors influencing QoS and the user QoE. A generic formula which provides a relationship between the users perceived QoE and QoS connecting through an exponential relationship, which is called IQX Hypothesis [21]. On this IQX hypothesis explain the sensitivity of QoE as a function of QoE itself. Where, the sensitivity of QoE relates to the influential factors of QoS.

Moreover, the performance of the underlying network affects the user experience of any service. Failure or success of a service is determined by the user experience. Shaikh et al [22] tries to correlate these network level QoS to user perceived QoE. Also investigated on the correlation between traffic characteristics and performance criterias which are measured in an operational network. They also observed that the duration of the web surfing session seems less dependent on the throughput and on the perceived QoE. A relation between QoE and traffic characteristics has been drawn on the paper. This helps the service provider to continuously monitor user satisfaction level, reaction time and rectify the performance problem.

Previously, a several number of research are done to quantify the impact of delays on end-user QoE [2], [5], [23]. Similarly, extensive amount of work [20], [21], [24] is done in finding out relationship between end-user QoE and QoS based measurable parameters. The standardization organization ITU-T Recommendation G.1030 [25] provides few guidelines to determine the Web-QoE. However, there is no recommendation providing guidelines for the evaluation of Web-QoE based on the present day scenario. The ITU T Study Group-12 is working on a recommendation for new evaluating methods of Web-QoE [26] along with many researchers around the world.

This research mainly focuses on the transitions of user experience due to the variation of waiting time. It would be interesting to evaluate, how the user's QoE changes for a website, where each page has waiting time. An e-commerce website is to be used for the web browsing experiment with three sessions, including three specific amounts of delay pattern attached to these sessions. The goal is to find out how the user perception is determined for different set of delays.

Chapter 2

Technical Background

This chapter provides an overview of the technical background which is essential for the understanding of this thesis. A common way for an internet user to access a web server is through a web browsing session. The requests from the user side are sent to the web server for resources processed by a web browser. The packet transfer between the server and the client takes place soon after a TCP connection is established. Once a connection gets established, different types of contents are transferred between them in a web browsing session. Often these responses are not instantly getting transferred to the end-user due to the unavailability of resources. On the other hand, users are not interested to wait unnecessarily or unproductively for any resources. Subsequently, the user‘s perceived quality gets affected by the waiting time [2]. A little increase or decrease in these waiting times causes variation in user QoE.

2.1 HTTP Object Transfer

In a web browser, a number of HTTP objects are transferred from server to client, for a request from a user. These packet exchange (i.e. sending and receiving of information between client and server) take place continuously for every request and response between client and server. According to the ITU-T recommendation G.1030 [25], the HTTP transfer is identified as handshake time and the data transfer time. Handshake time includes the DNS Look-up and the TCP handshake. DNS Look-up is where the user requests for a particular web address where it searches the IP for the given human readable Uniform Resource Locator (URL) of the user. Then a connection is established for secure transfer of data and required time is called data transfer time. During the data transfer phase, the packet exchange is constituted of the request for the HTML file (HTTP GET) and the responses of the data packets transferred from the server (i.e. Page download time). A brief illustration is given in the Figure 2.1 for the HTTP Objects transfer. During the entire communication, disturbance at any point in data transfer causes an increase the page load time for the objects to be displayed in the web browser. These issues are discussed below in details.

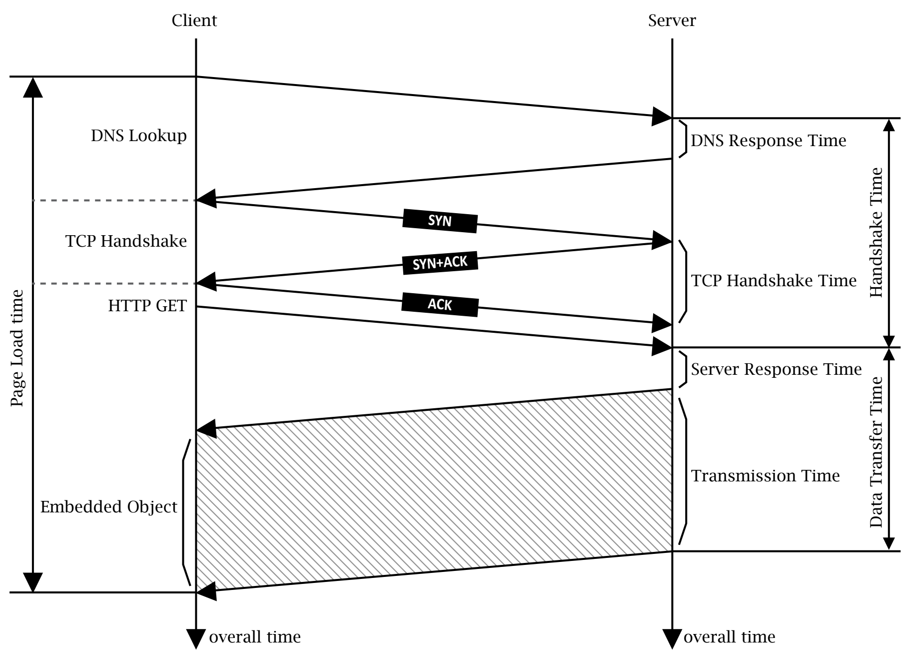

Figure 2.1: HTTP object transfer [25]

2.1.1 Page Load Time

Page load time refers to the entire time taken from a request for a URL or a click by the client till the final data packet (i.e. all content on the web page) transferred by the server is received by the client [27]. Page load time includes many things like DNS lookup, TCP handshake and Data transfer. The time taken to load all these items forms the page load time. The page load times are the waiting times faced by a user before viewing the content of the requested web page.

2.2 User's Web Browsing Experience

Web browsing activities over the internet are producing a bulk of internet traffic. This increase in the internet traffic is making the users to wait for longer time, for having their request served by the server. Due to which the widespread availability of World Wide Web (WWW) is also being named as a World Wide Waiting process. There are several other reasons for causing delays in WWW [28]. The most important user perceived quality indicator of a web based application is the delay. This is observed when a user clicks on a link and having the required content displayed on the screen.

2.2.1 Click, Wait and View

Figure 2.2 shows the click, wait and view process of web surfing. Usually, users browse the website via a web browser. Generally, users refer to the web address of the domain name. When a user writes a URL in the address bar and then a request is sent to the specific DNS server, the DNS server provides the IP address for that particular website.

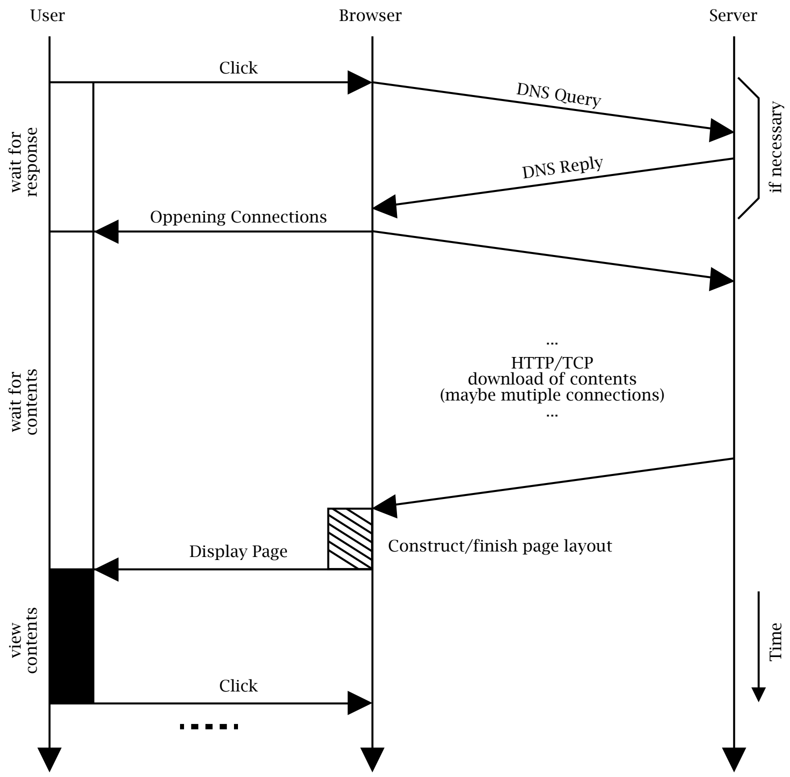

Figure 2.2: Click, wait and view process of web surfing [28]

At this time a lot of transaction is done between the browser and the server. When a new page is loaded, the text, images, embedded objects and icons etc. are also loaded from the server. After users view the contents of the website, they click the hyperlink or fulfil the required task to move on for new pages. In general, it is difficult to measure the viewing time exactly without specific web browsing software and physically observing the user. Most of the browsers give an indication to the users about the current state of the download process. The status of the line like as "*Connect: Looking for host.....*" is displayed during the DNS Look-up. The status is changed to "*Connect : Contacting host.....*" and "*Connect: contacted*" during the TCP connection phase [28]. During the data transfer the download status informs the current state of the users request. The remaining waiting time left before the page is loaded and the content is ready to be viewed.

2.2.2 From Web Pages to Web Sessions

A web page is a Hyper Text Markup Language (HTML) text document, which consists of several objects such as images, scripts and many more. Whereas, HTTP is a messaging protocol for the web, HTML describes the content in a web page, allows the content providers to redirect to other web pages through hyperlinks. A user, while accessing a web page, may refer to several other links or submit forms which redirect them to another web page. In response to this HTTP request, the user gets redirected to a new web page. This results in the user having to view a new page resulting in a new QoE [1], which is based on the time the content gets loaded in that page. As the user moves on surfing, it typically includes a set of pages and even a change in web sites too. For this reason, a web session can be categorized by a series of web pages with a waiting time. The waiting times caused by the loading of the objects may be small or long. The waiting times are sufficient for ensuring a certain degree of user satisfaction [2]. These waiting times in a web session can occur with a low intensity for a longer period of time over the consecutive browsing of web pages. Also, they may occur with higher intensity with a short period of times over the continuous browsing of web pages. The impact caused by this set of pages influence the user‘s QoE for a web sessions.

2.2.3 User’s Flow of Experience

The speed and fluidity of the browsing experience depend on several network parameters in the underlying network. Parameters such as delay or bandwidth may increase the loading time of the objects [5, 29]. This causes unacceptable completion times of page views of the users. The time it takes between a click of an URL (i.e. Request) and the last response from the server to this request is referred to as page load time. This is a key performance metric. Several studies [30, 31], show that another relevant metric is the visual sign of progress. This relevant metric is considered as the duration of time from the user request submission until the rendering of the new page starts. This metric is used directly to correlate to web QoE [25, 32]. The progress of pages being viewed is related to the waiting times, which has a significant impact on user QoE.

The authors [33, 34] state that a web browsing is an interactive process. These interactive processes include a sequential page or URLs to which the users redirect themselves from the particular page. The user perceives the web browsing as an immersive continuous flow of experience instead of restricting to a particular page [34]. Thus the users perceived web experience can be determined based on the sequential page-view events, which is done over a certain period of time. Variations in waiting times in this flow cause an impact over user perceived experience in web browsing. This helps in performing a quality judgement for overall QoE for any web based service.

2.3 Web Session and peak Delay

A web session is formed by making a sequence of request by a client to a server [1]. It starts by requesting a URL or by the click of a link. After a request is sent to the server, it starts responding with a base file and then followed by the embedded objects. The requests for these embedded objects are made by the client side web browser soon after the base file is received. A web session consists of several pages. We used three web session in this web browsing experiment.

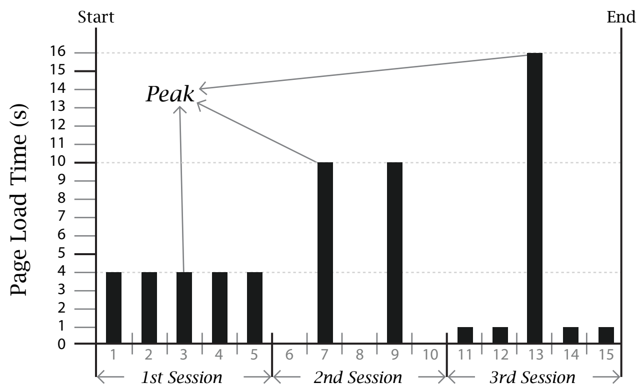

Figure 2.3: Web session with peak delay at different pages

Figure 2.3, the x-axis represent web session and the y-axis represents the page load time for each individual web page, which indicates the amount of delay perceived by the user for each web page. This represents one complete web browsing session of a user, where first, second and third session represents three individual sessions with different sets of *peak* delays respectively at any web page of the session. In this thesis, the term *peak* delay represents the highest page load time faced by a user during a web session. The impact caused by these different *peak* delays might be variant, the total waiting time of every session may be constant. In this research, we want to find out if the users report their experience differently for different distribution of *peak* delays in a session.

During web browsing, the user may go through different web browsing sessions. The user QoE may vary from session to session, as the *peaks* that occurred in the web browsing sessions may not be the same. These may even be influenced by the experience created by the past browsing session of the web user. The main focus in this thesis is to find out the impact of these *peak* disturbances in a user web browsing session

2.4 Time Perception of Web Applications

There are three important time limits which are determined by human perceptual abilities in order to measure the web based application performance. The quality perception related to response time can be grouped according to the following three perceptual time limit [35].

Instantaneous action: When a user experiences smaller time limits such as 0.1 s, then the user considers the system is reacting instantaneously.

Uninterrupted experience: Users assume that the loading is uninterrupted if the time limit experienced by the user is about 1.0 s. Normally, a delay less than 1 s is unnoticed by the user. If the delay exceeds more than 1 s the user thinks that the service provided is interrupted and is not responding.

Loss of attention: Users feel like quitting the service if the waiting time limit experienced by the user is about 10 s. In such case, the users usually opt for another tab and let the service do the loading. This is the time limit when the user looks forward to quit working on the current window. Day by day, the waiting time that causes this loss of attention is decreasing. In the present scenario, the users quit the service if the page download time reaches the 8 s limit [2].

Its observed that these waiting times effect user perceived quality for a web browsing session. During a web browsing session these waiting times are caused by different sets of peak delays. Finding out the impact of these peak delays over a web browsing session seems to be very crucial. The main essence of this thesis is to find out how these peak delays impact user perceived quality over a period of time.

Chapter 3

Experimental Design

In this chapter, the main focus is to investigate the impact of *peak* delay on the end-user perception for a web browsing session. An experimental setup is configured and designed for performing the task and it consists of three main modules. Web site hosting service is provided by the server where an e-commerce website is hosted. The network emulator is configured to shape the network traffic for the web service and the client side is designed for the users to go through the web browsing experiment in different network conditions.

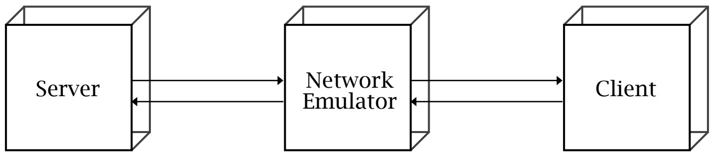

Figure 3.1: Experiment design

The three main components and connections are illustrated in Figure 3.1. The detail experimental process is described in details in this chapter. Furthermore, the chapter describes about the tools and techniques which are used to perform these subjective user tests.

3.1 Experiment Setup Configuration

In order to find out the effect of network disturbances on QoE of web browsing session, an experimental setup is established having a server, a client and a network emulator. The network emulator is placed between the server and the client to generate desired network environment (Figure 3.2). The server is running with Linux environment (Ubuntu 10.10 operating system(OS)) and the client is configured with Windows 7. All the devices are connected though a link of bandwidth 10 Mbps. An Apache [36] web server is configured where server side scripting language Hypertext Pre-processor (PHP) [37] is enabled. Here the software Bind9 [38] is used in the DNS server because of its wide availability and open source nature. The same server machine is also providing the DNS service. This DNS server helps to redirect the user requested URL to its respective IP address. The client machine is configured using Microsoft Windows 7 and it is preferred as its one of the most popular and widely used operating systems [39]. The Google Chrome browser is configured on the client side for users to browse the website.

Network delays are controlled using a network emulator, KauNet [40]. KauNet is specially used to generate packet driven deterministic network shaping as it does not reorder the packets, instead holds them in queue. All the request and reply packets are transferred through the network emulator where the shaping is applied based on the pattern being assigned for that particular run. KauNet acts as a gateway between client and server.

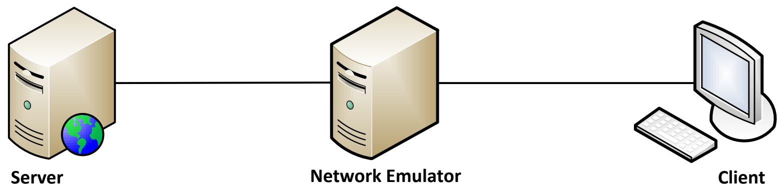

Figure 3.2: Experiment setup

An application level object information capturing tool named as Fiddler is installed on the client side and used to capture the HTTP traffic transferring from server to client. Then all these captured data is saved as a HAR (HTTP Archive) file for further analysis. A HAR is a common format which records information about HTTP traffic. It records information about each object being loaded by the browser and their timestamps.

3.2 Client-Side Design

3.2.1 Client

When a user requests for a URL on the web browser, the browser first checks its own cache and further checks the local OS cache for the DNS resolution. In this experiment, the web browser‘s cache and the local OS cache was disabled to make sure that all DNS requests are always sent to the server. The browser (i.e. Google Chrome) was set to Incognito mode [41] on the client side. This special mode prevents the browser from caching any resource. It deletes all the information downloaded during the session as soon as the browser window is closed.

3.2.2 MySQL

A proficient way to cache, fetch and remodel data is done by a computer program known as database software. MySQL is a free and open-source relational database management system (RDBMS) tool [42]. Using MySQL, web applications that are able to provide service to several thousands of users per second by availing terabytes of data developed by web developers. User friendly interface with other programming languages coupled with a tiny size, high processing power and easy installation of MySQL makes it attractive and demanded. MySQL was used to store the entire user given data and feedback.

3.2.3 CodeIgniter

To build a fully featured dynamic website with PHP, CodeIgniter [43] is used which is an open-source, lightweight, powerful web application framework. The main idea of CodeIgniter is to provide developers with a rich set of libraries and helps to develop a website from the start. CodeIgniter is developed using a well-known Model-View-Controller (MVC) framework. Figure 3.3, shows the basic Workflow of CodeIgniter.

Using this design pattern developers separates the code into three parts:

- In this model database interaction is maintained
- To show data Viewers are used in the user interface
- Interaction with users that affects model and views is handled by the controller

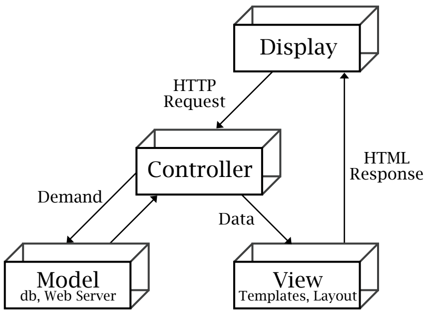

Figure 3.3: Workflow of codeigniter

Updates are viewed by the user on the manipulation of the model of the controller in accordance with the interaction/communication with the user. CodeIgniter was used for developing the web site, where users shop for their required product.

3.2.4 Fiddler

The abduction of all HTTP(s) traffic that is between the client computer and the Internet over the Windows platform is done by an open-source web debugging proxy tool known as a fiddler [44]. To debug web transfers it is a handy tool used by web developers. Analyzing the incoming and outgoing data, inspecting the traffic and showing the timeline for web transfers are among some of the features that it provides. Using WinHTTP, WinINET and Web Socket the traffic is captured by Fiddler. Traffic can be debugged from any application like Internet Explorer, Google Chrome, Mozilla Firefox, Apple Safari and many more which supports proxy using fiddler. Google Chrome browser is used WinINET. The work flow of fiddler is specified in Figure 3.4.

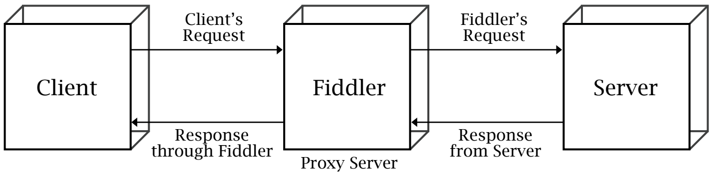

Figure 3.4: Workflow of fiddler

As a proxy server it is even used to remove traffic from Windows Phone, iPhone/iPad/iPod. There are two formats for the session transfer, the 'saz' (zip compatible) and the HTTP ARchive (HAR) file format. Fiddler was used to capture all the HTTP traffic, which is being sent and received on the client meshing along with their respective time stamps for further analysis.

3.3 Server-Side Design

3.3.1 Server

In this experiment, an e-commerce website is hosted on the server where the web service is enabled. This website consists of five pages for the user to purchase products. DNS service is also provided by the same server. The DNS server is prepared to resolve three domain names for the same website for three different web browsing sessions. The users browses the same website in three different sessions of their purchases.

3.3.2 Bind9

This application is installed and configured on the server; it is used to provide Domain Name System (DNS) service. DNS helps in translating the domain names to numerical IP address, which helps to identify the device or the service worldwide. This acts as a telephone directory where the host names are redirected to the IP address. Berkley Internet Naming Daemon (Bind) [38] is a Linux based free open source application. This was developed by the graduate students from the university of California, Berkley (UCB). It is a very handy application to configure and install.

3.3.3 Apache Web Server

In this Client-Server model the web server plays a vital role in communication. One popular web server is Apache [45]. This is an open source software and is available for major OS which are widely used. The main role of the web server is to respond with HTTP(s) response for all the HTTP(s) requests being sent by client (Figure 3.5). In general a web server serves every request with a response in the form of an HTML document [36].

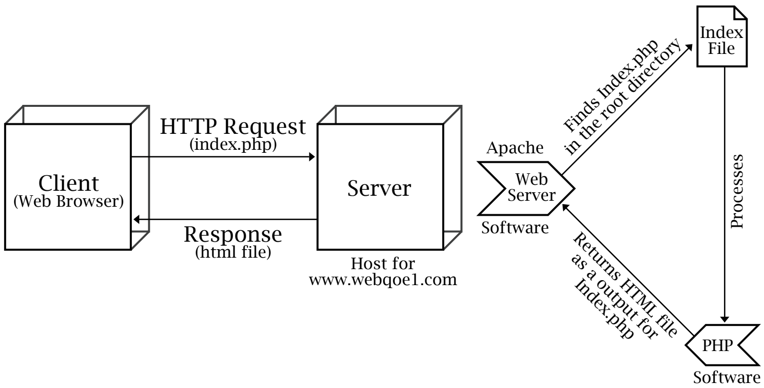

Figure 3.5: Workflow of apache web server

3.4 Network Emulator

The network emulator acts as a network gateway between the client and the server. All the network traffic that is generated from the server to the client and vice versa, passes through the network emulator. The KauNet is installed and configured in Linux environment as the network emulator. KauNet [40] is an open source free to use network emulator tool. It was developed by Johan Garica, Per Hurting and Anna Brunstrm of Karlstad University. KauNet extends the dummy net approach by adding some restrictions on the network parameters like packet loss, bandwidth, bit errors, delay changes and packet reordering. KauNet generates the pattern generation file, which contains the information on how the network packets are supposed to be shaped. This controls the behavior of the network which is based on a per-packet or per-millisecond basis.

3.5 Network Emulation Parameters

A numerous studies have been done to quantify the impact of waiting time on QoE [10, 46, 47]. These three studies used 2 s, 4 s, 8 s, 12 s, and 16 s delay as impairment. In another study [20], the authors used seven levels of the time delay (0 s, 2 s, 6 s, 10 s, 18 s, 26 s and 34 s). These delays were randomly assigned to each iteration of the experiment with the restriction that each delay will occur twice in each session.

Similarly, this study uses three sets of delay pattern produced for three web sessions browsed by each user. Despite the difference in the delay pattern, the cumulative sum of the user perceives waiting time was approximately 20 s in each session. Different delay patterns are used to vary the *peak* delay on every session. The embedded objects in every page of the five page web session were delayed as shown in the Table 3.1. Packet numbers were calculated by using Wireshark [48] and and delay is applied on a specific packet from server to client direction. The total sum of 20 s page load time was not the applied delay, it was the overall delay perceived by every end-user in each of the sessions. The first delay session has five *peaks* of 4 s, the second delay session has two *peaks* of 10 s and the third one has a single *peak* of 16 s. The values of the applied delay for all the three patterns can be viewed in Appendix G.

One session contains a uniformly distributed delay, where users perceive a delay of 4 s in each of the 5 pages. On the second session, 20 s is split into two 10 s delay on the second and the fourth page while the rest of the page does not have any delay. On the third session, users perceived 16 s in the third page while all the other pages have nearly 1 s of delay, as shown in Table 3.1. User opinions for different delay sessions are collected. These are discussed in more details in the chapter 5.

Table 3.1: User's perceived delay in different sessions

Session Name	**Perceived Network Delay**
Session 1	4 s delay in Page 1
	4 s delay in Page 2
	4 s delay in Page 3
	4 s delay in Page 4
	4 s delay in Page 5
Session 2	No delay in Page 1
	10 s delay in Page 2
	No delay in Page 3
	10 s delay in Page 4
	No delay in Page 5
Session 3	1 s delay in Page 1
	1 s delay in Page 2
	16 s delay in Page 3
	1 s delay in Page 4
	1 s delay in Page 5

3.6 Subjective Quality Assessment Scale

User saisfaction of a service is collected by the ACR rating scale recommended by ITU-T P.800 [8] . The user MOS is given based on the five point practically user scale over a five point scale (5) as Excellent, (4) as Good, (3) as Fair, (2) as Poor and (1) as Bad. Table 3.2, provides a detailed description.

Table 3.2: ITU-T scale for quality impairment

Grading Value	Quality	Impairment
5	Excellent	Imperceptible
4	Good	Good perceptible, but not annoying
3	Fair	Slightly annoying
2	Poor	Poor annoying
1	Bad	Very annoying

3.7 Development of Automatic Test Controller

We developed an automated tool 'Test controller' to manage the entire experiment. A PHP framework named CodeIgniter is used to manage the entire system. The design structure of the Test controller is specified in Figure 3.6.

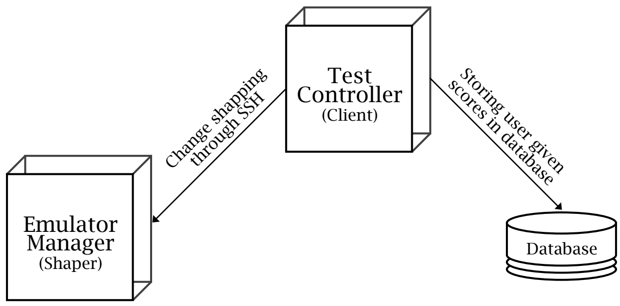

Figure 3.6: Automatic test system

A set of operations is to be performed in conducting the experiment for every user session. They are changing the shaping rules on the network emulator, collect user given information and store in database. A typical web browsing session is a continuous flow of experience where the users browse through a set of pages and completes their task. In this browsing session, the essential thing is that, the user must not be disturbed to have a continuous flow in user experience. To have these operations performed continuously, an experimental controller is used.

The emulator manager is developed for assisting the Test controller. This Emulator Manager is developed using bash script. This script takes a delay pattern as an input from the experimental controller through SSH. The Test controller loads the delay pattern from a text file which is done based on the delay pattern ID which is taken based on the user ID. The delay patterns are as specified in the Table 3.1. Two bash scripts are developed in the Emulator, one to flush all the rules in the emulator and the other is for applying the new rules.

Test controller executes a shell script based on Output-Buffering technique which is done by calling a PHP script. The Output-buffering technique is used to write any data into the output buffer. Using this technique a command can be given to the remote machine while waiting for the response of that particular command. This technique helps the Test controller to collect all packets without placing its call in waiting for the user session.

Test controller is developed to handle the emulator Manager. It also collects the user data of the entire experiment and save every user MOS into MySQL database.

3.8 User's Task-driven Experiment Process

To find out the user perception in web browsing, a task driven experimental procedure is built. Users are given an e-commerce website to browse through it, for buying a product. The entire website is the same for all users, while the delays being applied on the sessions are separate and these delays occur randomly for every session of each user. A training session is conducted to every user for about five minutes to describe the entire experimental procedure. In this training session all the essential steps are clearly explained to every user.

Prior to starting the entire experiment a MATLAB program was run, which generates random sets of delay sessions. Each set contains all the three sessions, but the order of occurrence of these sessions is randomized. According to this generated sequence, delays are applied to every user. A SignUp page is used in the beginning to have the user information saved into database. Every user needs to give a rating and acceptance about the service at the end of every individual session. The ratings and acceptance given by the user are saved into a MySQL database on the client machine. The schematic work flow of the experiment for one user during the entire session is described in the flowchart (Figure 3.7).

The e-commerce website is designed in such a way that the user would have a realistic feeling which resembles the commercial trend of e-commerce website. This web session consists of five pages, where the first page is a category selection page. In this page, the user selects the category for his preferred products. On the second page the user gets all products available in that category, where the user selects a product. On the third page, detailed description of the product is given. Here the user gets a buy now option, where he selects the option to buy. On the fourth page the user gets the payment details of the product. This page also contains the checkout option. Before selecting the checkout option the user needs to insert a security code in a text box from a CAPTCHA image. This image is used to create a realistic payment procedure for the user. On successful insertion of the CAPTCHE image into the text field the user is redirected to a payment confirmation page.

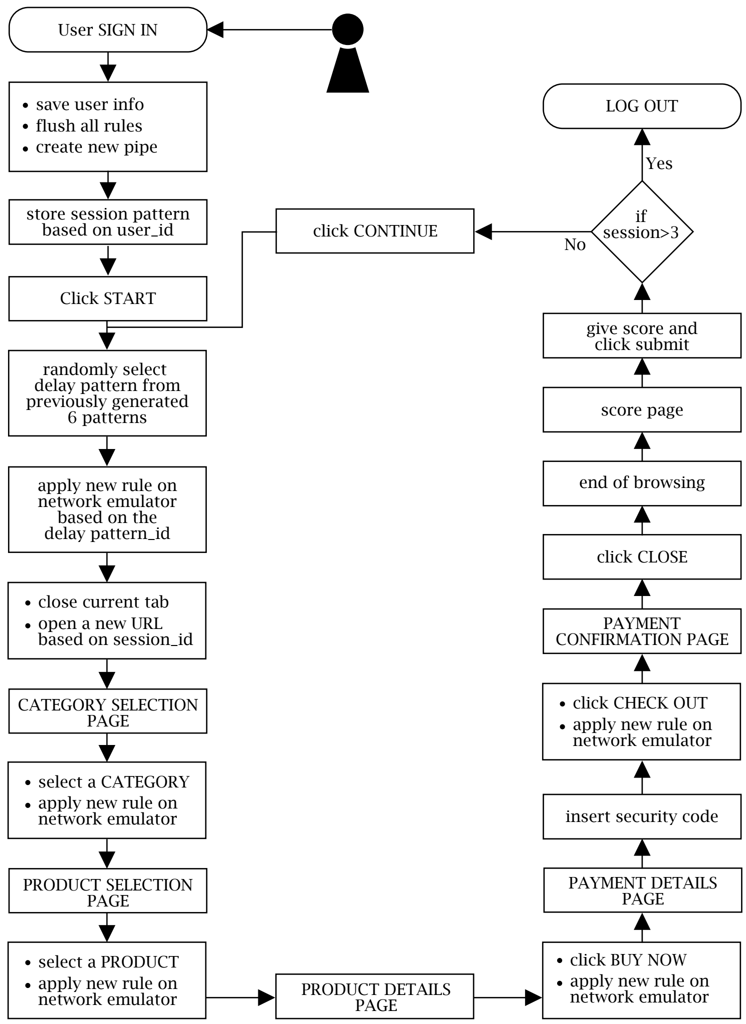

Figure 3.7: Flowchart of the full experiment process of one user

Chapter 4

Post Processing

This chapter describes the post processing of the data obtained from the experiment that explained in the previous chapter. In the experiment, users are asked to browse an e-commerce website consisting of five web pages in order to buy a product. In every session, the user browses the website with different delay pattern (as mentioned in Table 3.1) and at the end of the sessions, they provide their subjective feedback for two different questions (Available in Appendix D).

4.1 Data Collection

Data is captured at three different levels in this experiment. Firstly, The network level packets are captured on the client side using t-shark [49]. Secondly, the application-level data for HTTP objects is captured using Fiddler [44]. Finally, the user level subjective opinion scores are gathered using a MySQL database in the client side. Each entry into the database consists of a User ID, Session ID, User MOS (integer format) and feedback (Yes-No). All collected data are stored based on their respective file formats for further analysis. The details about the analysis process are explained in the following sections.

4.2 Page Load Time Calculation

The time taken for all the contents of a web page to be transferred from server to client is denoted as page load time. In this context, page load time is the time elapsed between a URL-request and the complete display of the response in the browser. When the process gets initiated by the client request, it involves the Handshake time (i.e. DNS lookup, TCP handshake) and the data transfer time (i.e. HTTP GET, Embedded objects). A detailed description of how these page load times are calculated is given further below.

4.2.1 Network Level Page Load Time Calculation

The T-shark is running on the client machine. It captures all the packets from the client-server communication on the network-level. Every transfer of information done between client and server are captured and stored using their time stamps by t-shark. The files are stored locally with a 'pcap format. This 'pcap' file is converted to a 'txt' file. This 'txt' file is used as a input for script to extract the required content. A Perl script was developed to extract the timestamp of the request from the client and the last response being sent by the server to the client (Available in Appendix F). The time difference between these two timestamps gives the page load time for that particular web page on the network-level.

4.2.2 Application Level Page Load Time Calculation

Application level data are also collected for every user in the experiment. A web debugging proxy tool called Fiddler [44] is used for collecting application level data on the client machine. Each of the client request and the corresponding response from the server are collected into a JavaScript Object Notation (JSON) format file. In general every web transfer consists of two parts, these are request and response headers. The Fiddler stores different object like timings, cache etc. The timing section stores the information about the time taken by different phases like waiting time, TCP connection time etc. Load time is taken based on the timestamps on all the objects in a web page while getting loaded. All this data is stored in a 'har' file format. Along with these 'har' files 'saz' (Session Archive Zip) files are captured. These 'saz' files are used to store all the HTTP traffic which helps in cross verifying the data obtained from 'har'. A PHP script is used to extract all the required information presented in the 'har' file. This script collects timestamps based on the first request from the client and the last response from the server. Based on these timestamps the page load time for all five web pages are stored in three different 'csv' files individually based on their session number.

4.3 Data Analysis

A total of 49 participants subjectively took part in the experiment. The maximum age of participants is 33 and the minimum is 19. The mean age of these users is 26. Among the participants, 38 are male and 11 female. All of them are daily users of web browsing services and used e-commerce website for online shopping.

Delays are applied on all the five pages in a session (Table 3.1). This is done based on the location of the session. Therefore, the page load time is calculated for all the pages in the session and finally summed up so that the overall page load time of the session tends to be approximately 20 s. This is done by considering the timestamps from the initial query to the final object being transferred which indicates the completion of fetching of the page. Page load time is calculated for every page based on the network level and application level data.

In this experiment, every user browses same e-commerce website with three different sessions. After extracting the required data, three different data files were created having page load time from application-level, network-level and user given MOS. Mean, Standard Deviation and Confidence Interval of the page load time and MOS are calculated using all the data obtained, in each of the data files.

4.3.1 Selection of the Data

Out of all 49 sets of data, 7 users who gave a constant value for all the three sessions or gave unusual high value in any session is eliminated (Available in Appendix C). All the data related to these users, i.e. page load time and MOS are removed from the calculation. Finally, after the elimination, 42 sets of data are taken into consideration for further analysis.

4.3.2 Mean and Standard Deviation Calculation

The Mean and Standard deviation is calculated for all the 42 users MOS and page load time for each of the sessions.

Mean is given as the arithmetic average of the values, which is calculated by adding all the values and dividing by the total number of scores or values. Mathematical representation of mean can be viewed in the following formula.

X be the overall page load time of any individual session

The Mean page load time is defined as,

$$\bar{X} = \frac{1}{N}\sum_{i=1}^{N} X_i$$

Standard deviation is a single number that tells us the spread or variability of a distribution (group of values). The Standard Deviation of the overall page Load Time for any given session is defined as,

$$\sigma_p = \sqrt{\frac{\sum_{i=1}^{N}(X_i - \bar{X})^2}{N-1}}$$

Where
N = The number of values (42)
X_i = Page load time for the i^{th} user of any session.

Y be user MOS at the end of any individual session

The Mean MOS is defined as,

$$\bar{Y} = \frac{1}{N}\sum_{i=1}^{N} Y_i$$

The Standard Deviation of the user MOS for any session is defined as,

$$\sigma_m = \sqrt{\frac{\sum_{i=1}^{N}(Y_i - \bar{Y})^2}{N-1}}$$

Where
N = The number of values (42)
Y_i = MOS given by the i^{th} user of any session.

4.3.3 Confidence Interval Calculation

Confidence Interval (CI) is also calculated for all the Mean page load time and MOS values. This is done as the Mean is always associated with CI 95%. The calculated Mean of the page load time and MOS values are obtained based on 95% CI.

The 95% CI of any session page load time is defined as,

$$[\bar{X} - \gamma_p, \bar{X} + \gamma_p]$$

The marginal error γ_p is defined as,

$$\gamma_p = 1.96 * \frac{\delta_p}{\sqrt{N}}$$

Where
δ_p = Standard deviation of an overall page load time for any session
N = Number of users

Similarly, The 95% CI of any session page load time is defined as,

$$[\bar{Y} - \gamma_m, \bar{Y} + \gamma_m]$$

The marginal error γ_m is defined as,

$$\gamma_m = 1.96 * \frac{\delta_m}{\sqrt{N}}$$

Where
δ_m = Standard Deviation of MOS for any session
N = Number of users

4.4 Overall Waiting Time in Each Session

The web browsing sessions are named according to the acting delay patterns, '4 s', '10 s' and '16 s'. For example, web browsing session '10 s' indicates that this session has the delay pattern which has two *peaks* of 10 s delay on the second and the fourth page (Table 4.1). All the data had been divided based on the three sessions in the experiment. The acronym mentioned in Table 4.1 is used to represent a particular session with its perceived delay.

Table 4.1: Session acronym for analysis

Session Name	**Session Acronym**	**Perceived Network Delay**
Session 1	4 s	4 s delay in Page 1 4 s delay in Page 2 4 s delay in Page 3 4 s delay in Page 4 4 s delay in Page 5
Session 2	10 s	No delay in Page 1 10 s delay in Page 2 No delay in Page 3 10 s delay in Page 4 No delay in Page 5
Session 3	16 s	1 s delay in Page 1 1 s delay in Page 2 16 s delay in Page 3 1 s delay in Page 4 1 s delay in Page 5

Detailed descriptions of some values obtained for each different session are given in the tables below (Table 4.2-4.4).

Table 4.2: Application, Network level page load time with 95% CI in 4 s session

	App.PLT	**95% CI**	**Net. PLT**	**95% CI**
Page 1	4.15 s	0.41%	4.08 s	0.33%
Page 2	3.97 s	0.98%	3.96 s	1.00%
Page 3	3.84 s	0.17%	3.82 s	0.24%
Page 4	3.98 s	0.22%	3.95 s	0.22%
Page 5	3.94 s	0.25%	3.93 s	0.26%
Total	19.86 s	0.25%	19.81 s	0.25%

Table 4.3: Application, Network level page load time with 95% CI in 10 s session

	App.PLT	**95% CI**	**Net. PLT**	**95% CI**
Page 1	0.29 s	10.05%	0.26 s	10.06%
Page 2	9.66 s	0.10%	9.63 s	0.06%
Page 3	0.13 s	18.74%	0.13 s	18.12%
Page 4	9.64 s	0.08%	9.58 s	0.07%
Page 5	0.13 s	11.40%	0.13 s	11.55%
Total	19.85 s	0.24 %	19.70 s	0.22%

Table 4.4: Application, Network level page load time with 95% CI in 16 s session

	App.PLT	**95% CI**	**Net. PLT**	**95% CI**
Page 1	0.93 s	2.46%	0.90 s	3.95%
Page 2	0.92 s	2.51%	0.93 s	2.61%
Page 3	15.95 s	0.12%	15.92 s	0.13%
Page 4	0.95 s	1.82%	0.95 s	1.80%
Page 5	0.83 s	1.31%	0.83 s	1.28%
Total	19.60 s	0.26%	19.51 s	0.30%

Chapter 5

Results and Analysis

This chapter gives a detailed description about the analysis of the obtained results. These results are based on the data obtained from a subjective web browsing experiment (explained in chapter 3). The data obtained from the user have been extracted and analyzed using some specific graphs. These graphs are generated for each and every setting of network condition in the experiment. The amount of applied *peak* delay on the embedded objects varies within three specific setting on three web sessions. The data obtained from these three specific sessions has been compared with one another. This comparison helps to find out the impact of randomly applied delay settings on end-user QoE. This chapter discusses about how these specific amounts of *peak* delay influence end-user QoE while the overall accumulated waiting times of every session is same. Also, how the occurrence of these *peak* delay sessions at different positions in the experiment shapes end-user perception.

5.1 Impact of Different Peak Delays on End-User QoE

The general relationship between user satisfaction and waiting time in any service is inverse [2]. This relationship persists in the domain of web browsing, with the consideration of some additional factors. However, waiting times do not always linearly translate into QoE. On the web browsing, waiting times caused by the delay occurrences during the transfer are not linearly translated into user perceived quality. There are also other factors, which may influence the user perception of web browsing QoE. In order to understand the dynamics of delay occurrences in web browsing, consideration of the tolerance threshold is very important [25]. If the perceived duration is shorter than the tolerance threshold, the users interpret that as fast and decent. Conversely, if the duration is perceived as longer than the tolerance threshold, the users interpret the duration as slow and insufficient. This experiment considers three patterns of *peak* delay (Section 3.5), one is way below the tolerance threshold, one exactly on the tolerance threshold and the other is way over the tolerance threshold.

This section gives a brief description about the overall impact of *peak* delay on the end-users in the experiment. As explained earlier in chapter 3, all the users of this web browsing experiment went through three browsing sessions encountering three different patterns of *peak* delays. In this section, similar delay patterns are grouped together to exhibit the overall impact of that particular *peak* delay. Figure 5.1, x-axis represents all three delay sessions and the corresponding user-given average MOS is placed on y-axis. Figure 5.1 shows the overall impact of the different *peak* delays of the end-user QoE.

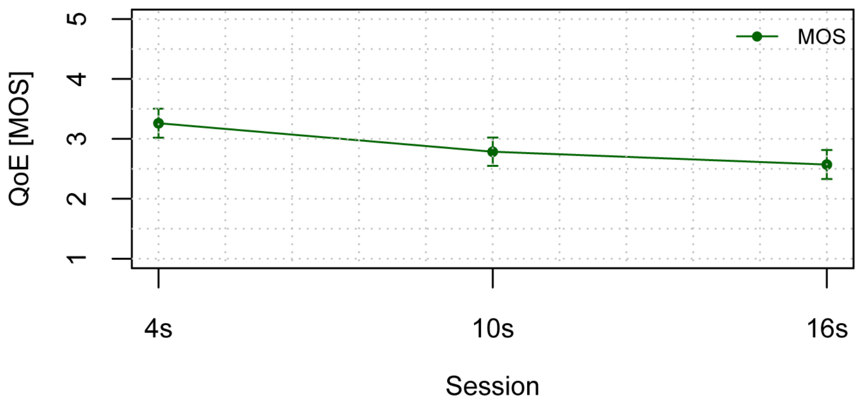

Figure 5.1: Average of user satisfaction for three different delay sessions

The web browsing is an immersive flow of experience beyond a single request-response transaction. In a web browsing session, the user encounters a series of waiting times page by page. Direct mapping of the measured loading time for a single page to the end-user rating cannot give a meaningful insight to the overall user QoE. In general, people do not want to wait for long time unnecessarily. Making a user to wait for a service can be a major source of dissatisfaction to the service as time is a non-renewable entity. The users of a website may leave the website if the content is not served within the expected amount of time. Even a tiny increment of a fraction of a second in the waiting time, significantly affects the overall browsing experience of the user surfing the web page [7]. If the response is provided within the time limit of 1-5 s, then users feel that there is some delay and the computer is still working on their command [50].

Figure 5.1 exhibits the reactions of users to varying durations of network delays. The accumulated waiting time in every session is approximately 20 s. The 4 s sessions have five pages with 4 s delay, which tends to get the highest overall user ratings. However, the small amount of peak delay has to be avoided in any case from a user-centric point of view. If the response time ranges between 5-10 s, then the flow of thoughts gets interrupted and 10 s is about the limit for keeping the users attention on the ongoing session. Delays of several seconds are acceptable [25], but not more than about 10 s. Users likely to be dissatisfied in case of 10 s sessions. The 10 s sessions get overall second highest ratings among the three. In case of 16 s sessions, users tend to be highly dissatisfied for the long waiting time at once. These findings suggest that users don't like long network disturbances at once. Also, users are more tolerant for low intensity network disturbance over a long period of time. Though the accumulated waiting time in every session is approximately same, but the amount of *peak* delay makes the difference in the user-given MOS. Figure 5.1, indicates that the duration of the *peak* delay has an impact on the determination of Web-QoE.

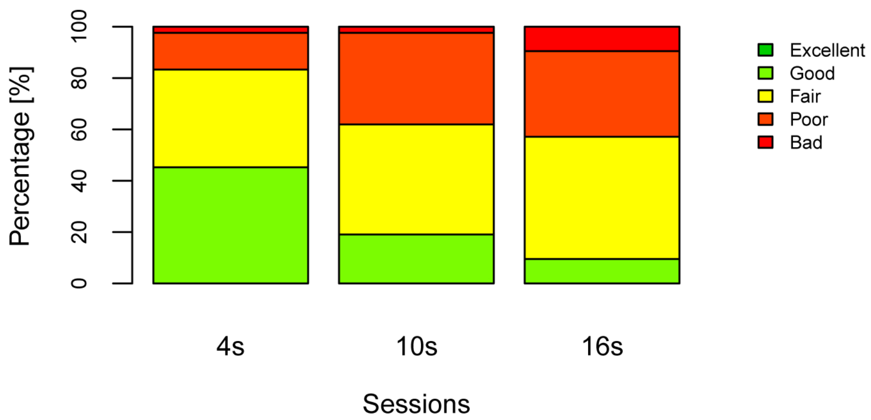

Figure 5.2: User satisfaction for three different delay sessions

Additionally, the user-given ratings for each of the sessions were shown in the bar chart 5.2. The session with 4 s of *peak* delay got more positive ratings compared to other sessions. The 4 s delay session was supposed to be comparatively lower delay than the other sessions, but it was still enough delay to get noticed by the users. Another reason for getting the highest percentage of positive ratings by 4 s delay can be that the delay was uniformly distributed amongst the five pages. As a result, the users have not experienced any variation in the delay. The "Good" and the "Fair" ratings constitute of more than 80% of the total ratings for 4 s of *peak* delay. This distribution of the "Good" and the "Fair" ratings reduces significantly for the 10 s and 16 s *peak* delay sessions. The reduced ratings given by the users were mostly "Poor" in the percentage.

The basic plotting of the obtained data in the above two graphs indicate that the user prefers low intensity network disturbances continuing over a long period of time over long network disturbances occurring all at once. This user preference is more clearly understandable when the total waiting time of every session is taking into account. As it is mentioned earlier that the total waiting time of every session is approximately same. Generally, the user does not want face any waiting time during web browsing. Moreover, the users want to avoid those situations where they have to wait for a long period of time for any content to be visible on the screen. In order to avoid such kind of situations the network resource management systems has to install a mechanism for splitting up the long delays into smaller pieces if the delay cannot be avoided [51]. The later sections will discuss how this user preference gets developed by the different pattern of *peak* network delays.

5.1.1 Impact of Peak Delay on End-User Acceptability

In this task-driven web browsing experiment, we give the users an option to assess whether they would continue to use the service (yes) or not (no). We can see from the results that, the users prefer a network disturbance with short delay with a longer period of time rather than a long network delay with a short period of time. Users tolerance is high for this type of temporary service disturbances. The percentage of acceptability for 4 s delay sessions is 78%. Users perceive the quality of a service with respect to the waiting time. The users attention goes uninterrupted if the waiting time less than 10 s [25].

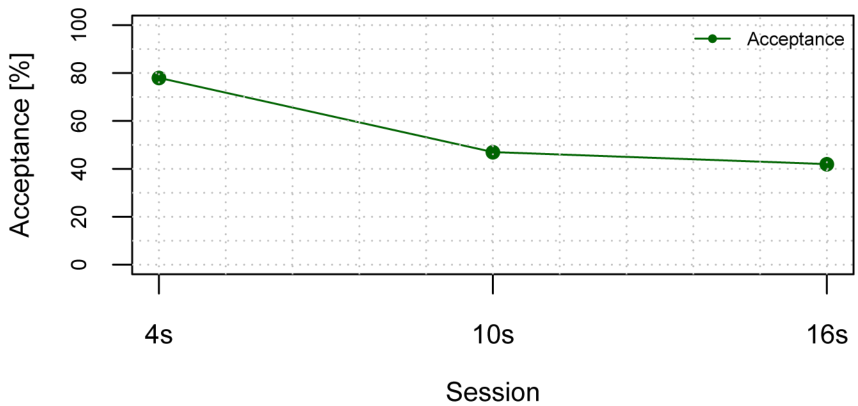

Figure 5.3: User acceptability ratings for different *peak* delay sessions

Users start abandoning their tasks, as the waiting times precede towards the 10 s threshold limit. For instance, the users started issuing the “Bad” rating for the 10 s delay session and the service was unacceptable to the user. The acceptance rate was 47% in 10 s delay session. The 10 s delay session is clearly the noticeable disturbance threshold and users were very sensitive to the long delays at once. The 16 s had the lowest degree of acceptance, 42%. The results show that, longer delay at once achieve a lower acceptance rate. The small durations of delay had a slight impact on the acceptability, whereas long durations of delay at once had a strong impact on acceptability rating.

5.2 Temporal Changes in QoE

This section discusses how user patience levels are decreasing as they are experiencing same delay sessions in different chronological order over time. Despite of approximately the overall waiting times of all the three shopping session, the user ratings vary, significantly. This is an interesting observation, which prompts us to dig further down into our results for detailed analysis about the possible underlying reasons.

Figure 5.4, the x-axis represents the session's positions of appearance of a user during the experiment and y-axis represents the user-given MOS. The MOS value for all the delay sessions in three sequential web browsing sessions that every user has gone through the experiment. The "start", "mid" and "end" indicate the temporal location each session.

Figure 5.4, every line represents the user-given MOS score for the specified delay. It exhibits the impact on user perception in every session individually for a same delay session while the waiting times are approximately same. For example, the green line represents the user feedback, when the 4 s session was encountered at the first session, second session, third session among the three sessions.

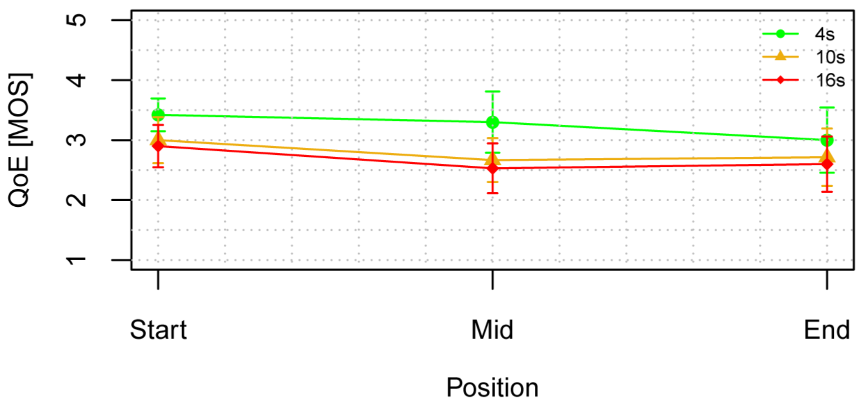

Figure 5.4: MOS at different positions for all three sessions

At the "start" position for every *peak* delay, the overall MOS rating is "Good". In this initial stage, though the users are experiencing higher amount of *peak* delay, the given score is comparatively high. Usually, when a user comes to a new service, the tolerance level tends to be higher as it can be seen from the graph that it doesnt show that much impact on the user QoE. Having the different delay sessions at the "start" position has very small deviation on the impact of the users opinion. On the other hand, the delays occurring in the "mid" position is clearly noticed and distinguished by the users.

At the "mid" position, the rating of the MOS is "Poor" for 10 s and 16 s session while the MOS is still remains “Good” for 4 s session. The MOS for the 10 s and 16 s sessions is almost same; it seems that the user could not distinguish much between these two sessions with higher amount of *peak* delay. The overall patience level of the users was decreasing as they were experiencing delays in the first sessions.

In general, end-user QoE decreases when waiting time increases. In contrast, QoE against decreasing waiting time doesnt increase sharply. The reason behind this is the phenomenon defined as memory effect by HoBfeld et al [10]. The memory effect becomes quite visible in the MOS for a same delay session at “start”, “mid” and “end” position for each of the user. For instance, the MOS value that appears at the “start” position for 4 s delay session is not same at the “end” position, it decreases significantly. As it can be seen in the Figure 5.4, the MOS values are decreasing from 3.42 to 3 for 4 s delay session. It reflects a significant loss of user QoE over time. This significant drop in grading scale states the impact of *peak* delay over the users working memory, which manifested in the recent past [52].

From these observations, it points out that, though the overall waiting times are approximately same for each session, the user MOS is different depending on the position where delay has been perceived. Later, user-given scores for a same amount of delay do not seem to rise above the respective initial MOS score. For example, user rating varies when session position keeps changing from “start” to “end”. The MOS score is shifted to 3.42 to 3 for 4 s delay session.

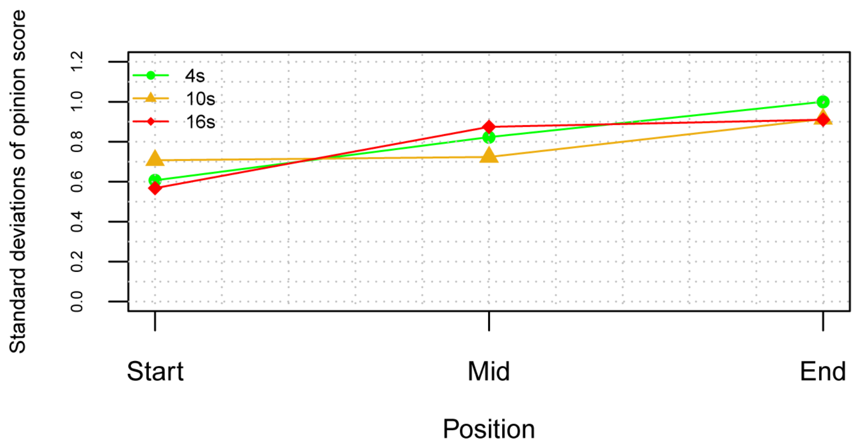

Figure 5.5: Standard deviation of MOS at different positions

Figure 5.5, x-axis represents the session positions and y-axis represents the standard deviation of the user-given opinion score.

In addition, we have to take the standard deviation of the opinion score into account in order to get a meaningful insight into the variation of MOS among users. The author [53] suggested that, the standard deviation of opinion score of users is needed to be considered in order to determine the end-user QoE. In order to further strengthen our understanding of the observed decay in MOS and underlying memory effect among users, we have computed standard deviation of opinion scores for all sessions in different position, where same network disturbances were applied. The total waiting times for these sessions were kept approximately around 20 s. The objective was to determine whether the standard deviation varies significantly over time throughout the course of the experiment.

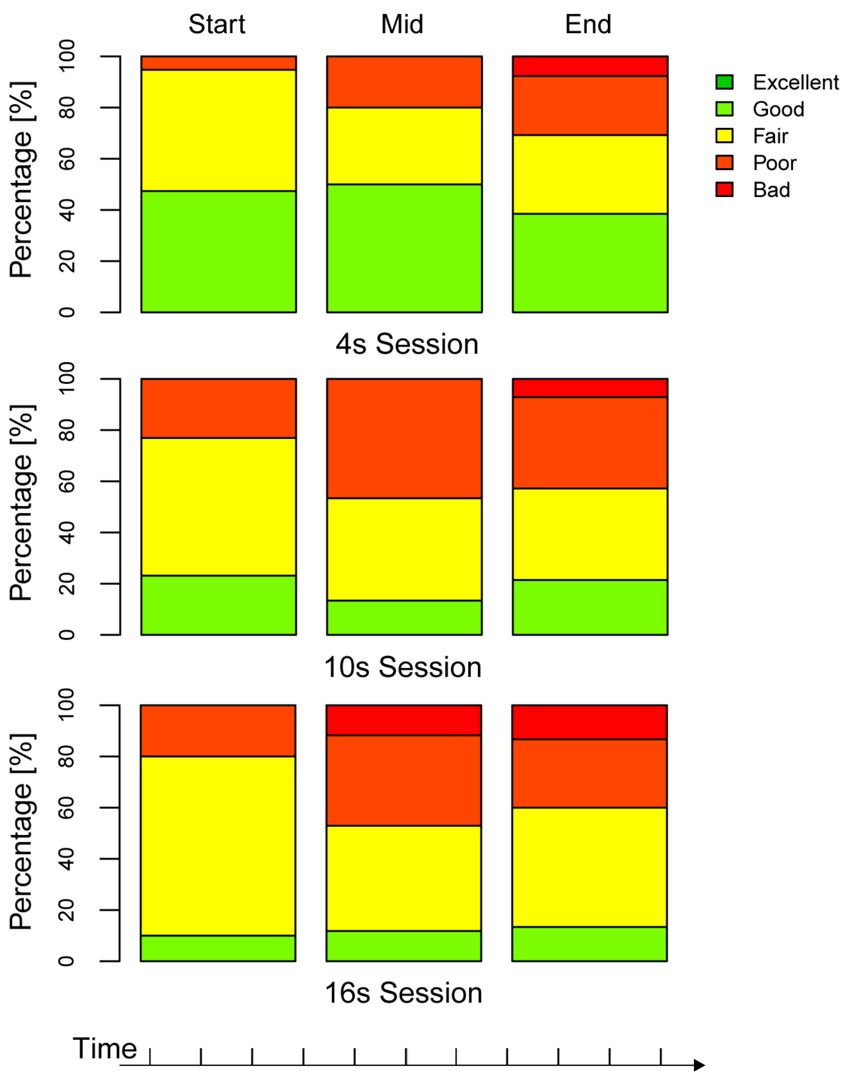

Figure 5.6: User satisfaction at different positions for all three sessions

Figure 5.5 shows the standard deviation of opinion scores for all web sessions with every delay environment. The increase in standard deviation indicates that the difference of opinion scores among subjects gradually increase with experiment time. Opinion difference is higher when a session appears later in the experiment. This shows that the memory effect is working differently among the individual subjects.

Moreover, we have also observe the percentage of the user rating regarding their satisfaction on the same delayed web sessions over time. Figure 5.6 illustrates the percentages of users' opinions have changed over time.

Usually the user tolerance level is high, when they browse any new service. At the first glance it is clear that, at the "start" positions there is no "Bad" opinions. Here, even users give a very small number of "Poor" rating. For example, even the 16 s delay session gets only 20% "Poor" ratings. The users acceptance rate is higher in the "first" position. From the percentage, it can be seen that, a significant proportion of the bar chart is "Fair" rating for all the delay session. Users clearly understand the service disturbance at the "mid" position.

At the "mid" position the user's opinion score is decreasing suddenly. It can be seen that, "Poor" rating is increasing for all the delay sessions. For instance, "Poor" rating percentages have moved from 5% to 20% for the 4 s delay session. And also a minor portion of "Bad" score is added to the 16 s delay sessions. The overall patience level of the users was decreasing as they were experiencing delays from the "first" session.

In general, end-user QoE decrease when waiting time increases. On the contrary, QoE against decreasing waiting time does not increase sharply. Usually, the subjects express their perception in the form of "Bad" ratings when they have experienced network disturbances in the recent past. The MOS ratings had changed significantly each session at the "end" position. From the bar chart it can be seen that, at the "end" position the "Bad" score is increasing. Alternatively, the "Good" rating is decreasing for 4 s, 10 s delay sessions. However, a small amount of "Good" score is increased but the large majority of "Fair" rating is decreased for the 16 s delay session. 16 s delay sessions has 13% of "Bad" rating which is the highest among all three positions. The network disturbance at the "end" position is clearly visible to the users, hence have a strong impact on QoE. An average 7-13% percentage of "Bad" rating is added to the "end" positions for all sessions.

5.3 Variation of QoE for Different Sequences of Session Appearance

This section discusses the shaping of end-user perception of random appearance of different delay sessions at different positions in the experiment. The experiment spans over 3 different sessions with 3 specific delay patterns. The permutation of 3 different patterns results into 6 chronological order of delay patterns (Table 5.1) and MOS values are available in Appendix E.

Table 5.1: Random delay pattern in three sessions

Pattern No	**Patterns Appearance**
Pattern 1	4 s, 10 s, 16 s
Pattern 2	4 s, 16 s, 10 s
Pattern 3	10 s, 4 s, 16 s
Pattern 4	10 s, 16 s, 4 s
Pattern 5	16 s, 4 s, 10 s
Pattern 6	16 s, 10 s, 4 s

Each pattern is given a detailed description in the following sections.

5.3.1 Pattern 4s 10s 16s

Figure 5.7, x-axis represents the position of the sessions and y-axis represents the user-given MOS. This figure shows the average MOS and the corresponding 95% confidence intervals in this particular delay pattern.

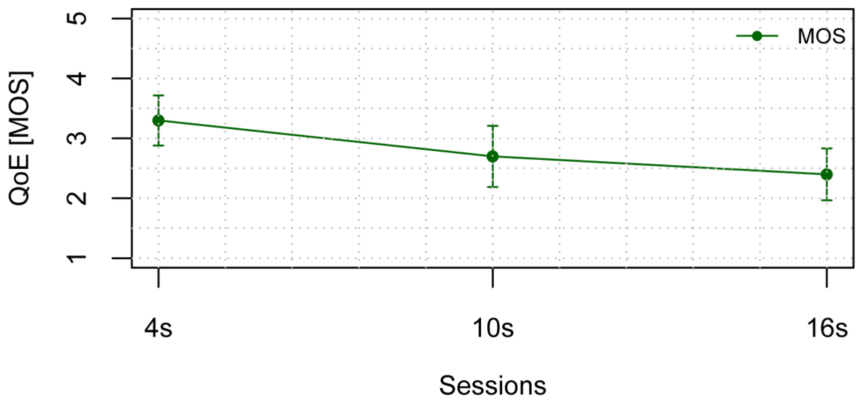

Figure 5.7: MOS for pattern 4s 10s 16s with 95% CI

Figure 5.7 exhibits that, in the initial state, the user may experience somewhere close to the real life continuity. The user perceives short-term network disturbance frequently at the "start" session. The user reflection of the "start" session have "Good" rating. When users browse to next shopping session "mid", there is a transition in the level of assessment from "Good" to "Poor". Users' perception was degrading to the e-commerce website as they were experiencing very long waiting time of 10 s in two of five pages. The waiting time at the "end" session was the longest in third page and applied at once. Although the other four pages had very small delay of 1 s, this session predictably was less tolerable to the user as reflected by worst rating. The patience level of the users is decreasing, as they are experiencing delays in several sessions. This pattern indicates that a slower service with distributed disturbance gets a better user perception than a fluctuating disturbance with higher *peak*. Even the overall allocated resource was the same to provide these two different services.

5.3.2 Pattern 4s 16s 10s

Figure 5.8, x-axis represents the position of the sessions and y-axis represents the user-given MOS. This figure shows the average MOS with 95% confidence intervals in this particular delay pattern.

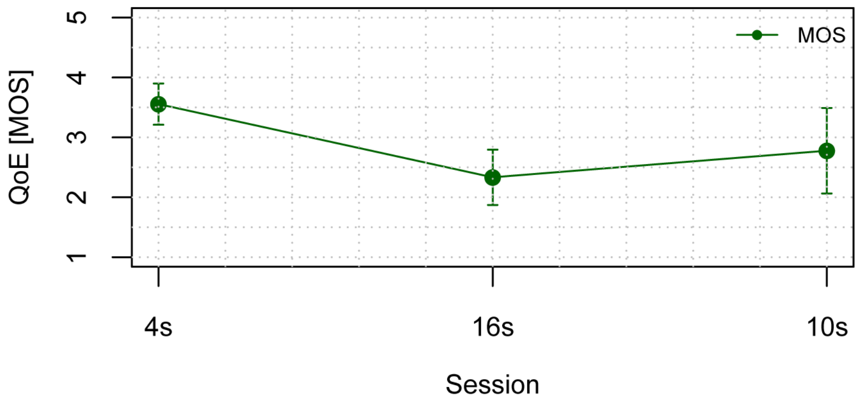

Figure 5.8: MOS for pattern 4s 16s 10s with 95% CI

The variation of a network impairment may not be affecting the QoE parameter at the initial state. For instance, small delays and delay variations may be eliminated by a jitter buffer, without the user noticing the additional delay [20]. Figure 5.8 illustrates that, in the "start" session the users convey "Good" rating. When the network disturbance exceeds a certain threshold, the former quasi-optimal QoE level cannot be maintained anymore. As the network disturbance grows, the QoE and thus the user satisfaction sink [20]. When the users shifted from "start" sessions to "mid" session, users opinion about the service tends to "Poor". In this case, users opinion drops quickly where MOS values are shifted from 3.55 to 2.33 because of the higher *peak* delay variation than the previous session. Worth noticing that the "mid" session with 16 s *peak* delay receives lower MOS than it received in the previous pattern where it appeared at the "end". A users tolerances decrease when previous experience is better [32]. This trend support this statement as in this pattern the user's experience was better (4 s delay) just before experiencing 16 s delay. If the QoE is already low, a further disturbance is not perceived significantly [20]. User's score is "Poor" at the "end" position for 10 s delay session.

5.3.3 Pattern 10s 4s 16s

Figure 5.9, x-axis represents position of the sessions and y-axis indicates the user-given MOS. This figure shows the average MOS and the corresponding 95% confidence intervals in this particular delay pattern.

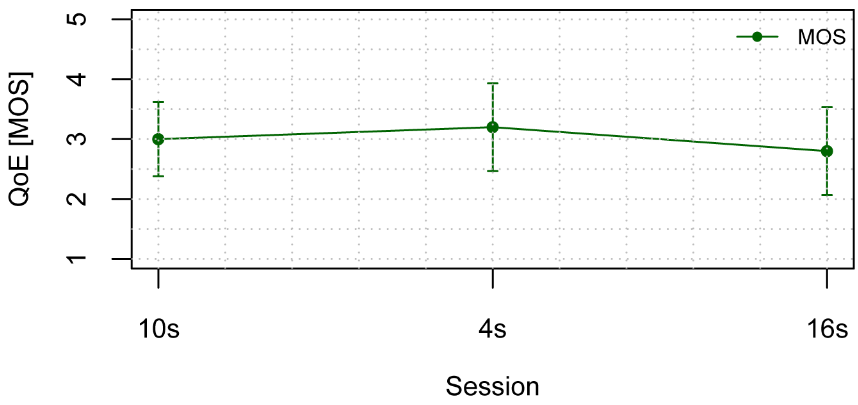

Figure 5.9: MOS for pattern 10s 4s 16s with 95% CI

This session pattern shows, in the initial state users QoE is acceptable for its long waiting times. Figure 5.9 illustrates that, in the "start" session user's level of assessment is "Good". This reflects the higher tolerance level of the user. Also the user satisfaction has less effect with longer waiting time at the "start" position. Generally, when page load time is decreasing, then user QoE is increasing. When the user proceeds from "start" to "mid" session, user ratings are lead to "Good" and MOS increases which is expected. Sudden degradation or rise of network disturbances affects user QoE significantly. The page load time is increased suddenly, when session shifted from "mid" to "end". The user assessment transforms from "Good" to "Poor" in this case. The MOS values are shifted from 3.2 to 2.8. Here, there were two factors namely that reduce the user satisfaction. One is the position of the highest *peak* delay at the end session and another is that the user had better page load time in the previous session.

5.3.4 Pattern 10s 16s 4s

Figure 5.10, x-axis represents position of the sessions and y-axis represents MOS value with 95% confidence intervals.

User-perceived quality suffers from initial delays when applications are launched as well as data delivery. Users QoE is less affected for this type of initial delay. Figure 5.10 illustrates the user satisfaction as "Good" in the "start" session and it reflects the higher tolerance level of the user. Any small increment in the page load time strongly affects the user interests in the webpage [7].

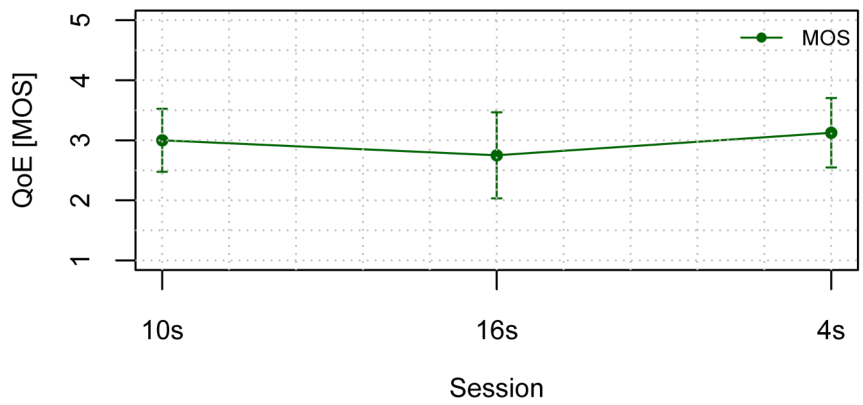

Figure 5.10: MOS for pattern 10s 16s 4s with 95% CI

When users shifted from "start" to "mid" sessions then they perceived a long *peak* delay at once. Users score also changes from "Good" to "Poor". A sudden degradation of network outage impact users QoE. Users satisfaction level progress to "Poor" to "Good" for uniform *peak* delay at "end" position.

5.3.5 Pattern 16s 4s 10s

Figure 5.11, x-axis represents position of the session and y-axis represents the user MOS. This figure shows the average MOS and the corresponding 95% confidence intervals in this particular delay pattern.

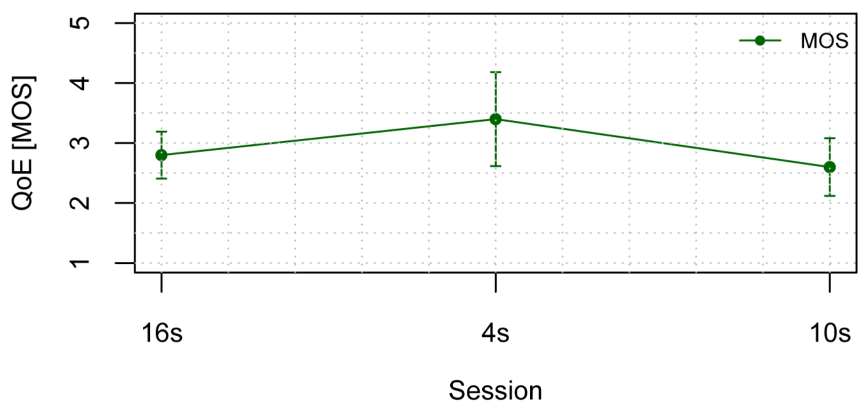

Figure 5.11: MOS for pattern 16s 4s 10s with 95% CI

In the initial state the delay is more acceptable to the user [2]. It shows that the user is ready to tolerate any startup initialization. Figure 5.11 illustrates, the users' assessment is "Good" for evaluating the "start" session. As the network disturbances is low, the assessment of the QoE and the user satisfaction would be high. This graph shows that the users opinion score is increasing instantly when page load times drop immediately as a result of session shifting. If the QoE is very high, a small disturbance will strongly decrease the QoE [20]. For instance, at the "end" sessions the users are less satisfied for an additional delay and the MOS value were shifted from 3.4 to 2.6 with this variance.

5.3.6 Pattern 16s 10s 4s

Figure 5.12, x-axis represents positions of the session and y-axis indicates the user MOS rating. This figure shows the average MOS with 95% confidence intervals in this particular delay pattern.

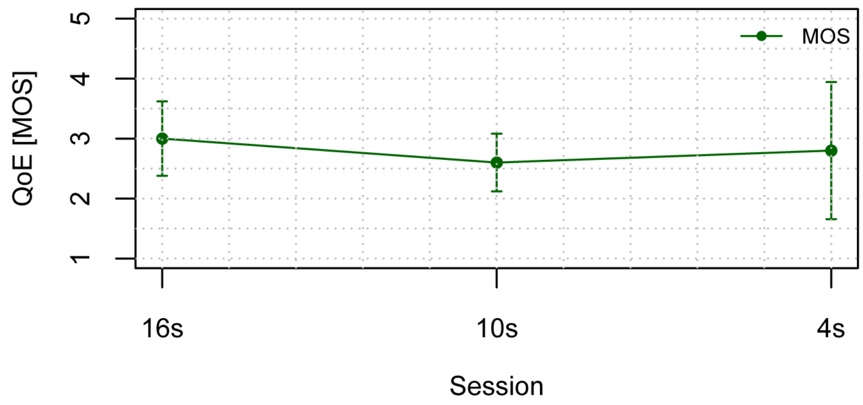

Figure 5.12: MOS for pattern 16s 10s 4s with 95% CI

Unacceptable waiting times may directly translate into user annoyance and churn. In the initial state the delay is more acceptable to the user [2]. The users are ready to tolerate long *peak* delay at the beginning. Figure 5.12 exhibits the user satisfaction level is "Good" at "start" position. User satisfaction is decreasing, when they encounter next shopping session at "mid" position. To compare this two web sessions the "mid" session has a less amount *peak* delay. User satisfaction is "Poor" for "mid" position. Here, the past experience affects the end-user QoE. At the "end" session, a small number of user rating is increasing for 4 s delay session. As the network outages are low, the assessment of the QoE would be high. A 95% confidence intervals shows that, user diversity also increased for 4 s delay sessions, because of the experiencing more disturbance period previously.

The patience level of the user decreases while experiencing consecutive delays in several sessions. We analyzed the impact of service disturbances in the context of random waiting times. From the observation of all web sessions, we can say that users don't like long waiting times at once. But the QoE is not only driven by waiting time, but also depends on the waiting time of the user' previous experience in nearby sessions. Which seems like users get 10 s delay after 16 s delay session, due to which the users QoE get affected. User perception for web browsing activities depend on this time dynamics. It is very important to identify the point of time for the occurrence of any disturbance in the network during a session in the process of execution.

5.4 Network-Level Analysis

In this section, a brief look at the communication occurring at the network-level has been provided. In order to get the network-level packets and to calculate the number of GET requests, TCP flag bits (SYN, SYN ACK, FIN ACK and RST) in the client-server communication a Perl script was developed (Available in Appendix F). Particular packets transmitted from the server to the client, carrying response of the object requested by the client was targeted to introduce the delays. The job of the traffic shaper was to keep the transmitted packet in queue according to the instruction written in the Perl script.

When the delay was applied on the packet carrying the response from the server, the client keeps initiating new TCP connections with SYN packets frequently, until it receives packets from the server. The higher the amount of delay, the greater the number of TCP connection requests from the client. For example, when approximately 10 s of delay on a packet from the server side was applied, 9 additional SYN requests from the client side (Table 5.2) was initiated [51].

Table 5.2: SYN in second page for three sessions

Delay	**Mean**	**STD**	**95% CI**
1 s	3.41 s	1.17 s	10.40%
4 s	5.43 s	1.85 s	10.31%
10 s	12.33 s	1.43 s	3.50%

Similarly, a large number of TCP terminations were initiated by the server when higher than 10 s (Table 5.3) delays were applied. When the server does not get ACKs of the sent packets, it starts terminating the existing TCP connections and keep on retransmitting the FIN packets [51].

Table 5.3: FIN-ACK in third page for three sessions

Delay	**Mean**	**STD**	**95% CI**
No Delay	0.03 s	0.15 s	196.00%
4 s	0.14 s	0.47 s	100%
16 s	12.02 s	4.23 s	10.65%

The client also retransmits the object requests by sending GET requests for the same object recursively. The amount of GET requests from the client side increase significantly with the increasing delay applied on packets. It is visible from the above analysis that, such delay events do not only waste resources, but also produce a multiplicative impact on the end-to-end performance of data transfers, hence, the QoE of the end-user.

Chapter 6

Conclusions

In this thesis, we presented a set of observations with regards to the impact of the same amount of overall waiting time with different variations of *peak* delay on the web browsing QoE over a period of time. The total download time of any particular web page is tightly coupled with web browsing QoE [33]. Web browsing is not a single request-response transaction, but it is considered as a continuous flow of experience where the user come across a series of waiting times [2]. The duration, position of waiting times have different implications in prediction of end-user QoE. It can be seen from the findings of this thesis that despite of the approximately similar overall waiting time for each session, the user overall rating varied significantly. These variations in the user's overall rating lead to several conclusions after careful analysis of the obtained data.

Firstly, the amount of *peak* delay is an important factor for determining the end-user perception for web browsing. Though each session in the experiment had approximately the same amount of waiting time in it, user overall ratings were different. Their MOS score was 3.2 for 4 s of *peak* delay, which reduced to 2.79 and 2.57 for 10 s, 16 s of *peak* delay respectively. The user is given ratings varied along with the amount of *peak* delay. The larger the *peak* delay in a session, the lower the user given score is. The *peak* delay has an inverse relation with the user's given scores despite the fact that there are other delay present in a session which if added together results in approximately similar amount of waiting time for every session in the experiment.

Secondly, the occurring position of the *peak* delay also has an impact on web browsing QoE. Same type of session was rated differently for changed chronological position in the experiment. The main reason can be that the patience level of the end-users were decreasing while experiencing consecutive waiting times in several sessions, hence the memory effect controls the user's given scores. Later, user-given scores for a same amount of delay session do not seem to rise above the respective initial user given scores for the same session. For example, user rating varied when session position changes from start to end . The MOS score is shifted to 3.42 to 3 for 4 s delay session.

In human time perception, it is acknowledged that (subjective) perception of a duration should never be assumed to be accurate and true to the actual duration [2]. The actual duration reflects objective time, which in this experiment was approximately similar for each session. The perceived duration reflects subjective time, which is the psychological time variation depending on the user. The users were more likely to rely on their mental estimations as the objective times were unknown to them. The main event that a user remembers from a web browsing session is the amount of *peak* delay. There were other delays in the sessions but the *peak* delay is playing the key role behind shaping the mental estimations of the user. Thus, the user given score is revolving around the amount of *peak* delay. Also, any increase or decrease in the *peak* delay is clearly visible from the user given scores. In the context of interactive task driven web applications a decrease in the response time can improve the users loyalty as the user has a particular goal to fulfill.

6.1 Answer to the Research Questions

We provided answers to these research questions based on our research results in the following section.

RQ 1. What is the effect of low intensity disturbances occurring continuously for a long period of time in a web browsing session?

Ans 1. The end-user avoids the situations where they have to wait for too long without any response. They want at least some piece of response, without waiting for too long at a time. End-user prefers a network with low intensity disturbances occurring continuously for a long period of time rather than a high intensity disturbances going for a small period of time. We found that the user experience is good in the situation of 4 s of additional delay on each of the five pages compared to the rarely occurring long delays (16 s) at one page. Their MOS score is 3.42 for 4 s of peak delay, which reduced to 2.57 for 16 s of peak delay.

RQ 2. What is the effect of high intensity disturbances going for a small period of time in a web browsing session?

Ans 2. High intensity disturbance in the network breaks users' attention on the task. This can interrupt the users flow of contention during a task. Users, start abandoning their tasks, as the waiting time proceeds towards the 10 seconds limit. User start issuing a bad rating for high intensity disturbances over a small period of time. Users are likely to be dissatisfied in the case of 10 s delay session. The 10 s delay sessions get overall second highest ratings (MOS value of 2.79) among considered three web sessions. Users tend to be highly dissatisfied for a longer waiting time at once. For example, 16 s delay session has a less MOS (MOS value of 2.57).

RQ 3. How the overall user QoE changes in multiple web browsing session span over time?

Ans 3. Users MOS depends on the position of the web session where delay has been perceived. For different positions of a delay session, user's recent past experience and working memory influence to determine the present perception of user QoE. The user MOS value is different in different shopping session position, though user perceived same delay for each position For example, user ratings vary when session position keeps changing from "start" to "end". This MOS value is shifted from 3.4 to 3.

RQ 4. How the users previous experience of a network disturbance affects the present perception in the web browsing QoE?

Ans 4. The patience level of the user decreases while experiencing consecutive delays in several sessions. Users recent past experience and working memory influence the present determination in perceiving the web QoE. Long delays at once in the network has a serious impact on the user's QoE. The QoE is not only driven by the delays, but also depends on the delays of the user's previous experience of the nearby sessions. For example, users opinion drops quickly where MOS values are shifted from 3.55 to 2.33 because of the higher *peak* delay variation than the previous session (Figure 5.8).

6.2 Future Work

In the future work, this research can be conducted by large numbers of users in the real life scenario using crowdsourcing. The experiment can be addressed in the wireless network environment where delay is applied to the traffic flow based on non deterministic mathematical model. Which is crucial to identify what happens when QoS parameters are changed under time varying network conditions and how it affect users QoE. It would be also interesting to create a time varying network conditions in a controlled laboratory environment and investigate the correlation between users QoE and time varying network conditions.

Appendix

APPENDIX A

The /etc/network/interfaces file in server and network emulator machine in the experiment setup has been edited by adding below parameters to set the IP address as permanent.

Server
auto eth0 iface eth0 inet static
address 10.0.1.1
netmask 255.255.255.0
broadcast 10.0.1.255
gateway 10.0.1.2

Emulator
auto eth2 iface eth2 inet static
address 10.0.1.2
netmask 255.255.255.0
broadcast 10.0.1.255
gateway 10.0.1.0
up ip route add 10.0.1.0/24 via 10.0.1.2

auto eth1 iface eth1 inet static
address 192.168.0.100
netmask 255.255.255.0
broadcast 192.168.0.255
gateway 192.168.0.0
up ip route add 192.168.0.0/24 via 192.168.0.100

As client machine was set up with windows based operating system, the permanent IP addresses were setup as following:

Client
IP: 192.168.0.101 Subnet mask: 255.255.255.0
Default gateway: 192.168.0.100
Preferred DNS: 10.0.1.1

APPENDIX B

Shaper Machine configurations

B1. Code for flushing shaping rule

```
#!/bin/bash
ipfw -f flush
ipfw -f pipe flush
```

B2. Code for applying new shaping rule

```
#!/bin/bash
cd /home/shaper/KauNet/KauNet2.0-Linux/patt_gen
rm -rf $1
./patt_gen -del -pos $1 data 1000 $2
ipfw -f flush
ipfw -f pipe flush
ipfw add allow all from any to any
ipfw add 1 pipe 100 $3 from 10.0.1.1 to 192.168.0.101 in
ipfw pipe 100 config bw 10Mbit/s pattern $1
```

B3. PHP script for preparing shaper

```
<?php
$rule_content = $_REQUEST["shape_rule"];
$session_id = $_REQUEST["session_id"];
$fh = fopen("shapping_model.txt","w+");
fwrite($fh,$rule_content);
fclose($fh);
echo json_encode("status:Sucess");
?>
```

B4. PHP script for changing shaping rule

```
<?php
$page_id = $_REQUEST["page_id"];
$shape_array = array();
$file = file("shapping_model.txt");
foreach($file as $line){
        array_push($shape_array,$line);
}
if($shape_array[$page_id] == 0){
        $exec = exec('sudo /home/shaper/KauNet/KauNet2.0-
Linux/patt_gen/Pipeflush.sh');
} else {
        $exec = exec('sudo /home/shaper/KauNet/KauNet2.0-
Linux/patt_gen/IP_pipeflush.sh /home/shaper/KauNet/KauNet2.0-
Linux/patt_gen/temp.dcp ' . trim($shape_array[$page_id]));
}
echo json_encode("status:Sucess");
?>
```

APPENDIX C

These users are filtered out for giving a static and unusually higher MOS.

Table C.1: Static and unusually high MOS

User	**4 s Session**	**10 s Session**	**16 s Session**
User 3	4	4	4
User 15	5	5	4
User 18	3	3	3
User 28	5	4	5
User 32	4	4	4
User 44	3	3	3

APPENDIX D

User were asked to answer the following question end of each session.

Q1. How do you feel about its loading time?
Options:

- Excellent (5)
- Good (4)
- Fair (3)
- Poor (2)
- Bad (1)

Q2. Would you be willing to use this Internet Service again?
Options:

- Yes
- No

APPENDIX E

Table E.1: Overalls application level PLT and MOS with 95% CI in three sessions

Session	**Total App. PLT**	**95% CI**	**MOS**	**95% CI**
4 s	19.86 s	0.25%	3.26 s	7.40%
10 s	19.85 s	0.24%	2.76 s	8.49%
16 s	19.56 s	0.26%	2.57 s	9.42%

Table E.2: Pattern 4s 10s 16s Application level PLT and MOS with 95% CI

Session	**Total App. PLT**	**95% CI**	**MOS**	**95% CI**
4 s	19.87 s	0.24%	3.55	9.68%
10 s	19.56 s	0.41%	2.33	19.80%
16 s	19.91 s	0.47%	2.77	25.70%

Table E.3: Pattern 4s 16s 10s Application level PLT and MOS with 95% CI

Session	**Total App. PLT**	**95% CI**	**MOS**	**95% CI**
4 s	19.98 s	0.73%	3.3	12.68%
10 s	19.83 s	0.68%	2.7	18.90%
16 s	19.58 s	0.53%	2.4	18.06%

Table E.4: Pattern 10s 4s 16s Application level PLT and MOS with 95% CI

Session	**Total App. PLT**	**95% CI**	**MOS**	**95% CI**
4 s	19.91 s	0.42%	3	20.66%
10 s	19.73 s	0.41%	3.2	22.92%
16 s	19.52 s	0.34%	2.8	26.19%

Table E.5: Pattern 10s 16s 4s Application level PLT and MOS with 95% CI

Session	**Total App. PLT**	**95% CI**	**MOS**	**95% CI**
4 s	19.87 s	0.58%	3	17.46%
10 s	19.57 s	1.10%	2.75	26.08%
16 s	19.85 s	0.37%	3.125	18.51%

Table E.6: Pattern 16s 4s 10s Application level PLT and MOS with 95% CI

Session	**Total App. PLT**	**95% CI**	**MOS**	**95% CI**
4 s	19.56 s	0.58%	2.8	14.00%
10 s	19.78 s	0.24%	3.4	23.06%
16 s	19.80 s	0.63%	2.6	18.47%

Table E.7: Pattern 16s 10s 4s Application level PLT and MOS with 95% CI

Session	**Total App. PLT**	**95% CI**	**MOS**	**95% CI**
4 s	19.54 s	0.27%	3	20.66%
10 s	19.73 s	0.17%	2.6	18.47%
16 s	19.81 s	0.95%	2.8	40.82%

APPENDIX F

Available in online repository : `https://github.com/nais11/Web_Browsing_QoE.git`

APPENDIX G

Table G.1: Applied delay on different sessions

Pages	**Session 1**	**Session 2**	**Session 3**
Page 1	8, 3700, 9, 1 tcp	0	8, 500, 9, 1 tcp
Page 2	5, 3700, 6, 1 tcp	6, 9400, 7, 1 tcp	5, 700, 6, 1 tcp
Page 3	4, 3700, 5, 1 tcp	0	5, 15700, 6, 1 tcp
Page 4	4,3800, 5, 1 tcp	5, 9400, 6, 1 tcp	4, 800, 5, 1 tcp
Page 5	5,3800, 6, 1 tcp	0	5, 700, 6, 1 tcp

Bibliography

[1] R. Schatz, T. Hoßfeld, L. Janowski, and S. Egger, "From Packets to People: Quality of Experience as a New Measurement Challenge," in *Data Traffic Monitoring and Analysis: From measurement, classication and anomaly detection to Quality of experience*, M. M. Ernst Biersack, Christian Callegari, Ed. Springers Computer Communications and Networks series, 2012.

[2] S. Egger, T. Hossfeld, R. Schatz, and M. Fiedler, "Waiting Times in Quality of Experience for Web Based Services," in *Fourth International Workshop on Quality of Multimedia Experience (QoMEX)*, pp. 86–96, Yarra Valley, Australia, July 2012.

[3] S. Grondin, "Timing and Time Perception: A Review of Recent Behavioral and Neuroscience Findings and Theoretical Directions," *Attention, Perception, & Psychophysics*, vol. 72, no. 3, pp. 561–582, 2010.

[4] S. C. Seow, *Designing and Engineering Time: The Psychology of Time Perception in Software.* Addison-Wesley Professional, 2008.

[5] N. Bhatti, A. Bouch, and A. Kuchinsky, "Integrating User-Perceived Quality Into Web Server Design," *Computer Networks*, vol. 33, no. 1, pp. 1–16, 2000.

[6] A. Van Moorsel, "Metrics for The Internet Age: Quality of Experience and Quality of Business," in *Fifth International Workshop on Performability Modeling of Computer and Communication Systems (PMCCS 5)*, vol. 34, pp. 26–31, Erlangen, Nürnberg, Germany, September 2001.

[7] S. Lohr, "For Impatient Web Users, An Eye Blink is Just too Long to Wait," *New York Times*, 2012.

[8] "Methods for Subjective Determination of Transmission Quality." `http://www.itu.int/rec/dologin_pub.asp?lang=e&id=T-REC-P.800-199608-I!!PDF-E&type=items`, 1996. [Online; Accessed: 18-Jan-2013].

[9] I. Rec, "P. 10/G. 100 Amendment 1,," *New Appendix IDefinition of Quality of Experience (QoE), International Telecommunication Union*, 2007.

[10] T. Hoßfeld, S. Biedermann, R. Schatz, A. Platzer, S. Egger, and M. Fiedler, "The Memory Effect and Its Implications on Web QoE Modeling," in *Proceedings of the 23rd International Teletraffic Congress*, pp. 103–110, San Francisco, USA, September 2011.

[11] P. Le Callet, S. Möller, and A. Perkis, "Qualinet White Paper on Definitions of Quality of Experience," *European Network on Quality of Experience in Multimedia Systems and Services (COST Action IC 1003), Tech. Rep*, vol. 1, Lausanne, Switzerland, June 2012.

[12] N. Crespi, B. Molina, C. Palau, *et al.*, "QoE Aware Service Delivery in Distributed Environment," in *Workshops of International Conference on Advanced Information Networking and Applications (WAINA)*, pp. 837–842, Biopolis, Singapore, March 2011.

[13] S. Kujala, M. Minge, A. Pohlmeyer, and M. Vogel, "Temporal Aspects of User Experience: Models and Methods Beyond a Single Use Situation," *TU Delft: Industrial Design Engineering: Industrial Design*, December 2012.

[14] I. Ceaparu, J. Lazar, K. Bessiere, J. Robinson, and B. Shneiderman, "Determining Causes and Severity of End-User Frustration," *International Journal of Human-Computer Interaction*, vol. 17, no. 3, pp. 333–356, 2004.

[15] A. Sieminski, "Changeability of Web Objects-Browser Perspective," in *Proceedings of the 5th International Conference on Intelligent Systems Design and Applications (ISDA'05)*, pp. 476–481, Wroclaw, Poland, September 2005.

[16] S. Khirman and P. Henriksen, "Relationship Between Quality-of-Service and Quality-of-Experience for Public Internet Service," in *Proceedings of the 3rd Workshop on Passive and Active Measurement*, March 2002.

[17] J. Shaikh, M. Fiedler, P. Arlos, and D. Collange, "Modeling and Analysis of Web Usage and Experience Based on Link-level Measurements," in *Proceedings of the 24th International Teletraffic Congress (ITC 24)*, pp. 1–8, Krakow, Poland, September 2012.

[18] E. Wang, "User's Delay Perception and Tolerance in Human-Computer Interaction," in *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 46, pp. 651–655, SAGE Publications, Baltimore, Maryland, September 2002.

[19] I. Koskinen, K. Battarbee, and T. Mattelmäki, *Empathic Design: User Experience in Product Design.* IT Press Helsinki, 2003.

[20] M. Fiedler, T. Hossfeld, and P. Tran-Gia, "A Generic Quantitative Relationship Between Quality of Experience and Quality of Service," *IEEE Network*, vol. 24, no. 2, pp. 36–41, 2010.

[21] T. Hoßfeld, P. Tran-Gia, and M. Fiedler, "Quantification of Quality of Experience for Edge-Based Applications," in *Managing Traffic Performance in Converged Networks*, pp. 361–373, 2007.

[22] J. Shaikh, M. Fiedler, and D. Collange, "Quality of Experience from User and Network Perspectives," *Annals of Telecommunications*, vol. 65, no. 1-2, pp. 47–57, 2010.

[23] C. Lorentzen, M. Fiedler, H. Johnson, J. Shaikh, and I. Jorstad, "On User Perception of Web Login—A Study on QoE in the Context of Security," in *Australasian Telecommunication Networks and Applications Conference (ATNAC)*, pp. 84–89, Auckland, New Zaeland, November 2010.

[24] T. N. Minhas and M. Fiedler, "Quality of Experience Hourglass Model," in *Proceedings of the International Conference on Computing, Management and Telecommunications (ComManTel)*, pp. 87–92, Ho Chi Minh City, Vietnam, January 2013.

[25] I. Rec, "G. 1030,," *Estimating end-to-end performance in IP networks for data applications,*, November 2005.

[26] "ITU-T SG12: Performance, QoS and QoE." `http://www.itu.int/en/ITU-T/studygroups/2013-2016/12/Pages/default.aspx`. [Online; Accessed: 02-Aug-2013].

[27] "Essential Web Performance Metrics-A Primer, Part 1 (Componets of page load time)." `http://blog.smartbear.com/web-performance/essential-web-performance-metrics-a-primer-part-1/`. [Online; Accessed: 26-Aug-2013].

[28] J. Charzinski, "Web Performance in Practice–Why We are Waiting," *AEU-International Journal of Electronics and Communications*, vol. 55, no. 1, pp. 37–45, 2001.

[29] M. Andrews, J. Cao, and J. McGowan, "Measuring Human Satisfaction in Data Networks.," in *INFOCOM*, 2006.

[30] H. Cui and E. Biersack, "Trouble Shooting Interactive Web Sessions in a Home Environment," in *Proceedings of the 2nd ACM SIGCOMM workshop on Home networks*, pp. 25–30, Toronto, ON, Canada, August 2011.

[31] D. Olshefski and J. Nieh, "Understanding the Management of Client Perceived Response Time," in *ACM SIGMETRICS Performance Evaluation Review*, vol. 34, pp. 240–251, June 2006.

[32] E. Ibarrola, F. Liberal, I. Taboada, and R. Ortega, "Web QoE Evaluation in Multi-Agent Networks: Validation of ITU-T G. 1030," in *Proceedings of the 5th International Conference on Autonomic and Autonomous Systems (ICAS'09)*, pp. 289–294, Valencia, Spain, April 2009.

[33] H. Weinreich, H. Obendorf, E. Herder, and M. Mayer, "Not Quite the Average: An Empirical Study of Web Use," *ACM Transactions on the Web (TWEB)*, vol. 2, no. 1, p. 5, February 2008.

[34] Y. X. Skadberg and J. R. Kimmel, "Visitors Flow Experience While Browsing a Web Site: Its Measurement, Contributing Factors and Consequences," *Computers in Human Behavior*, vol. 20, no. 3, pp. 403–422, 2004.

[35] "Response Time Limits: Article by Jakob Nielsen." `http://www.nngroup.com/articles/response-times-3-important-limits/`, 2013. [Online; Accessed: 02-Sep-2013].

[36] "Welcome! - The Apache HTTP Server Project." `http://httpd.apache.org/`, 2013. [Online; Accessed: 21-Jul-2013].

[37] "PHP: Hypertext Preprocessor." `http://php.net/`, 2013. [Online; Accessed: 14-Jul-2013].

[38] "Bind9- Debian WikiBind9." `https://wiki.debian.org/Bind9`, 2013. [Online; Accessed: 18-Jul-2013].

[39] "OS Statistics." `http://www.w3schools.com/browsers/browsers_os.asp`, 2013. [Online; Accessed: 08-Jul-2013].

[40] "KauNet." `http://www.kau.se/en/kaunet`, 2013. [Online; Accessed: 04-Jul-2013].

[41] "Incognito mode (Browse in Private) - Chrome Help." `https://support.google.com/chrome/answer/95464?hl=en`, 2013. [Online; Accessed: 10-Jul-2013].

[42] "MySQL?:: The Worlds Most Popular Open Source Database." `http://www.mysql.com/`, 2013. [Online; Accessed: 14-Aug-2013].

[43] "CodeIgniter / EllisLab." `http://ellislab.com/codeigniter`, 2013. [Online; Accessed: 21-Aug-2013].

[44] "Fiddler: Web Debugging Proxy." `http://www.fiddler2.com/fiddler2/`. [Online; Accessed: 26-Aug-2013].

[45] "November 2013 Web Server Survey — Netcraft." `http://news.netcraft.com/archives/2013/11/01/november-2013-web-server-survey.html`, 2013. [Online; Accessed: 02-Nov-2013].

[46] S. Egger, P. Reichl, T. Hosfeld, and R. Schatz, "Time is Bandwidth? Narrowing the Gap Between Subjective Time Perception and Quality of Experience," in *Proceedings of the International Conference on Communications (ICC'12)*, pp. 1325–1330, Ottawa, Canada, June 2012.

[47] D. Strohmeier, S. Jumisko-Pyykko, and A. Raake, "Toward Task-Dependent Evaluation of Web-QoE: Free Exploration vs.Who Ate What?," in *Globecom Workshops (GC Wkshps)*, pp. 1309–1313, California, USA, December 2012.

[48] "Wireshark: The Network Protocol Analyzer 1.10.6.." `http://www.wireshark.org/`. [Online; Accessed: 20-Nov-2014].

[49] "Tshark - The Wireshark Network Analyzer 1.10.0.." `http://www.wireshark.org/docs/man-pages/tshark.html`. [Online; Accessed: 16-Nov-2014].

[50] R. B. Miller, "Response Time in Man-Computer Conversational Transactions," in *Proceedings for AFIPS Spring Joint Computer Conference, part I*, pp. 267–277, December 1968.

[51] N. Islam, V. J. D. Elepe, J. Shaikh, and M. Fiedler, "In Small Chunks or All at Once? User Preferences of Network Delays in Web Browsing Sessions," *to appear in IEEE ICC Workshop on QoE-centric Network and Application Management*, Sydney, Australia, June 2014.

[52] J. Shaikh, M. Fiedler, P. Pangkai, S. Egger, and F. Gyuard, "Back to Normal? Impact of Temporally Increasing Network Disturbances on QoE," in *IEEE Workshop on Quality of Experience for Multimedia Communications*, Atlanta, USA, December 2013.

[53] T. Hobfeld, R. Schatz, and S. Egger, "SOS: The MOS is not Enough!," in *Proceedings of the 3rd International Workshop on Quality of Multimedia Experience (QoMEX)*, pp. 131–136, Mechelen, Belgium, September 2011.