ByggletMobil

Android application for the construction industry

Annika Larsson
**Sammanfattning**

Den digitala tekniken utvecklas till olika användningsområden och antal mobila applikationer blir fler och fler både för privatlivet och yrkeslivet. Nu vill projektets uppdragsgivare Bygglet AB expandera mot den mobila marknaden med sin affärside. Företaget Bygglet AB arbetar i huvudsak mot byggbranschen och tillhandahåller det webbaserade projekthanteringsverktyget Bygglet.

Denna uppsats behandlar examensarbetet med prototypen mobilapplikationen ByggletMobil. Rapporten innefattar både teoretisk och praktisk bakgrund, arbetet med implementationen och en motivation till att denna typ av digital teknik behövs inom byggnadsområdet.
Abstract

Digital technology has been developed for different application areas and the number of mobile applications for both private and professional life is increasing.

Now, the project's sponsor Bygglet AB is expanding to include the mobile market with their business idea. The company Bygglet AB works principally in the construction industry and provides the Web-based project management tool Bygglet.

This dissertation presents a final year project on a prototype mobile application ByggletMobil. The report includes both a theoretical and practical background, work towards an implementation and a motivation for the need for this kind of digital technology in the construction area.
# Innehåll

1 Inledning .................................................................................................................. 1
   1.1 Bakgrund ........................................................................................................... 1
   1.2 Syfte .................................................................................................................. 2
   1.3 Utredning av begrepp ....................................................................................... 2
   1.4 Avgränsningar .................................................................................................. 3
   1.5 Disposition ........................................................................................................ 3

2 Bakgrund teori ......................................................................................................... 4
   2.1 Grafiska Användargränssnitt .......................................................................... 4
      2.1.1 Dator GUI vs Surfplatta GUI vs mobil GUI ............................................. 5
      2.1.2 Vad är en mobilapplikation ....................................................................... 5
      2.1.3 Design av GUI ............................................................................................ 7
   2.2 Android ............................................................................................................... 10
      2.2.1 Androids Operativsystem ......................................................................... 10
      2.2.2 Android applikation .................................................................................. 11
      2.2.3 En aktivitets livscykel ............................................................................. 11
      2.2.4 Ett fragments livscykel .......................................................................... 13
      2.2.5 Ett Android-projekts uppbyggnad .......................................................... 14
      2.2.6 Databas ...................................................................................................... 16
   2.3 Säkerhetsaspekter ............................................................................................... 18
      2.3.1 Kryptering ................................................................................................. 18
      2.3.2 Autentisering ............................................................................................ 20
      2.3.3 Säker lagring av data .............................................................................. 20
      2.3.4 Validera input ........................................................................................... 20
      2.3.5 Testa säkerheten ...................................................................................... 20
   2.4 Databassäkerhet .................................................................................................. 21
   2.5 Androidsäkerhet ................................................................................................ 21
   2.6 Sammanfattning ................................................................................................ 22
3 Projekt bakgrund.................................................................23
  3.1 Projektet.............................................................................23
  3.2 Projektavgränsningar.........................................................24
  3.3 Bygglet AB.........................................................................25
  3.4 Bygglet................................................................................26
  3.5 Tillgången på IT-teknik inom byggnadssektorn.....................28
  3.6 Befintliga applikationer.......................................................29
  3.7 Bygglet vs ByggletMobil......................................................31
  3.8 Uppdraget applikationen ByggletMobil.................................31
     3.8.1 Flödesdiagram...............................................................31
  3.9 Sammanfattning.................................................................32

4 Projektets design och Implementation..................................33
  4.1 Planeringsfas.....................................................................33
  4.2 Initieringsfas......................................................................34
  4.3 Designfas............................................................................35
     4.3.1 Designbegränsningar....................................................35
     4.3.2 Designmönstret MVC...................................................36
     4.3.3 Flödesdiagram...............................................................36
  4.4 Programmeringsfas............................................................37
     4.4.1 Klasser – Java.................................................................37
     4.4.2 Designmönstret MVC...................................................37
     4.4.3 Grafiskt gränssnitt Autentisering....................................38
     4.4.4 Grafiskt gränssnitt ByggletMobils funktionsdel................38
     4.4.5 Databas...........................................................................45
     4.4.6 Klasser – API..................................................................46
     4.4.7 Projektets säkerhetsaspekter..........................................46
  4.5 Sammanfattning..................................................................47

5 Resultat.................................................................................48
  5.1 Testkörning.........................................................................49

6 Slutsats..................................................................................50
  6.1 Summering av arbetet.........................................................50
  6.2 Problem och dess lösningar under projektet........................51
  6.3 Framtida arbete...................................................................52
References........................................................................................................53

Bilagor........................................................................................................58
  Bilaga A - Beskrivning av examensarbete......................................................58
  Bilaga B - Godkännande av projektet applikationen......................................59
  Bilaga C - Konfidentiell programkod..............................................................60
  Bilaga D - Förfrågan om tillgänglighet till Bygglet AB Databas....................61
Figurlista

Figur 1: Surfplatta i horisontellt och vertikalt läge.................................................8
Figur 2: Androids arkitektur......................................................................................10
Figur 3: En aktivitets livscykel................................................................................12
Figur 4: Ett fragments livscykel................................................................................13
Figur 5: Designmönstret MVC (Model-View-Controller) ........................................14
Figur 6: Kommunikation mellan databas och mobil via PHP som API................................17
Figur 7: Översikt Bygglet..........................................................................................26
Figur 8: Personal Bygglet..........................................................................................27
Figur 9: Flödesdiagram ByggletMobilapplikation....................................................32
Figur 10: Flödesdiagram av ByggletMobils design................................................36
Figur 11: Emulatorn med ByggletMobils ikon.............................................................39
Figur 12: Inloggningssida till ByggletMobil..............................................................40
Figur 13: Inloggningssida till ByggletMobil vid felaktig inmatning...............................41
Figur 14: ByggletMobils första sida...........................................................................42
Figur 15: ByggletMobil när användaren tryckt på knappen kunder.............................43
Figur 16: ByggletMobil när användaren tryckt på knappen personal..........................44
Figur 17: Databas MySQL.........................................................................................45
1 Inledning

Idag lever vi i en värld där den digitala tekniken utvecklas i användningsområden. Datorerna har blivit mindre i storlek och trenden är att stationära datorer försvinner till förmån för så kallade surfplattor och mobiltelefoner som är enklare att bära med sig. Konsekvensen är att människor kan leva ett mer flexibelt liv där de alltid kan vara uppkopplade mot de intressenter som de vill kommunicera med på alla tänkbara sätt. Det innebär också att användningen av ”appar”, dvs. mobila applikationer ökar i takt med att försäljningen av den mer flexibla utrustningen ökar i popularitet.

1.1 Bakgrund

1.2 Syfte

Syftet med detta projekt är att skapa en mobilapplikation, för det mobila operativsystemet Android. Applikationen baseras på en redan befintlig webbapplikation, ett projektledningsverktyg för byggsektorn.

1.3 Utredning av begrepp

För att göra denna rapport tydligare vill jag förklara följande återkommande begrepp:

Bygglet AB är namnet på uppdragsgivarens företag.

Bygglet är namnet på webbapplikationen.

ByggletMobil är namnet på mobilapplikationen som detta examensarbete handlar om.

Android är ett mobilt operativsystem.

Java är ett objektorienterat programspråk.

Eclipse är en utvecklingsplattform.

MySQL är namnet på databashanteraren som använts.

SQLite är databashanteraren som ingår i Androids Operativsystem

PHP (Hypertext Preprocessor) är ett skriptspråk.

API (Application Programming Interface) är regler som beskriver hur mjukvara kan kommunicera med annan mjukvara och fungerar även som filtret mellan databasen och applikationen.

JSON (JavaScript Object Notation) är tolken som används för att mjukvarornas kommunikation ska fungera.
1.4 Avgränsningar

Det beställande företaget har sekretessbelagt allt gällande kodning samt resultat av den kod som produceras i detta projekt. Däremot kommer vi att kunna presentera skärmbilder på delar av resultatet och utföra en provkörning av prototypen vid presentationen av examensarbetet. Orsaken till detta är att företaget inte vill ge eventuella konkurrenter en marknadsfördel genom att lämna information eftersom uppsatser lätt kan bli spridda och ej kan hållas hemliga. Företaget har också begärt att få en prototyp samt att den enbart skall fungera för Androids mobila plattform.

1.5 Disposition

Uppsatsen är uppdelad i 6 kapitel. Kapitlen är Inledning, Bakgrund teori, Bakgrund projekt, Projektets design och implementation, Resultat och Slutsats.

2 Bakgrund teori

Detta kapitel handlar om den teoretiska bakgrunden. Inledningsvis finns en beskrivning av grafiska användargränssnitt och en jämförelse mellan dator, surfplatta och mobil. Sedan följer en kort introduktion till vad en mobilapplikation är och några problemområden vad gäller designen av GUI. Därefter kommer en fördjupning inom området Android och som avslutning till detta kapitel behandlas säkerhet i allmänhet och speciellt Androidsäkerhet.

2.1 Grafiska Användargränssnitt

Teknik gällande datorer och kommunikation idag är komplex och av mycket hög standard. Det finns en risk att människan och hennes begränsningar kommer i andra hand vid utveckling av program och applikationer då hjärnan inte kan ta in hur mycket intryck som helst. Det är för människor som applikationer ska utvecklas till och då krävs det att designen är utvecklad för människans skull. En viss fara ligger i att teknikutvecklingen går förbi människors förmåga. Fokus måste ligga på användarvänlighet, vad människor vill ha i stället för vad tekniken klarar av.

Fler och fler människor kommer att använda sina mobiler och surfplattor till de flesta användningsområden. Man kan idag utföra i stort sett samma operationer på en mobil eller surfplatta som man kan på en pc. Detta ställer då krav på programvaran till dessa plattformar och detta innebär då också ett ökat behov av ett förstklassigt GUI med god design och användarvänlighet [4]. Programvaran till en mobil eller surfplatta kallas för en applikation eller ”en app”.

2.1.1 Dator GUI vs Surfplatta GUI vs mobil GUI

Klassificeringen av en surfplattas funktionalitet hamnar någonstans mitt emellan en dator och en mobil. Användningsområden för en surfplatta är i huvudsak att surfa på websidor [51] och även att läsa mail, se film, läsa nyheter och böcker, spela vissa spel och interagera med facebook eller andra sociala medier.

2.1.2 Vad är en mobilapplikation
I en mobiltelefon kan det även finnas widgets [50] som enligt svenska datatermgruppen är en grafisk användargränssnittskomponent [6]. Det är också en typ av applikation som ofta är ett litet dynamiskt program som visas på hemskärmen hela tiden. En widget kan vara en klocka, kalender eller visa väderleken [7].

En del appar är förinstallerade av mobiltelefontillverkaren och det går mycket enkelt att ladda ner och installera fler appar som man själv behöver från specifika onlinebutiker eller app-förråd. En del appar är gratis och andra får man betala för [5].


1. Google Maps 54 %
2. Facebook 44 %
3. Youtube 35 %
4. Google+ 30 %
5. Weixin/WeChat 27 %
6. Twitter 22 %
7. Skype 22 %
8. Facebook Messenger 22 %
9. Whatsapp 17 %
10. Instagram 11 %
2.1.3 Design av GUI

I detta delkapitel beskrivs en del problematik runt designen av GUI som programutvecklare har krav på att lösa vid utveckling av program, webbsidor och applikationer. Här belyses en del svårigheter som applikationsutvecklare ställs inför och det nämns en modell som det går att använda sig av i utvecklingsarbetet.


Metoden går ut på att man använder HTML, Hypertext Markup Language [55] och CSS, Cascading Style Sheets [54], och skapar en dynamisk layout. Innebörden av detta är att samma sida med samma innehåll och samma URL kan se olika ut beroende på vilken enhet som används [53]. Istället för att ange precisa storlekar och placeringar vid design av GUI så anger man procentuella storlekar och placeringar istället [52].

Ett klassiskt problem i arbetet med programutveckling är att mobiler och surfplattor har två olika lägen, ett horisontellt läge och ett vertikalt läge (se figur 1). Applikationen måste vara användarvänlig och ge ett trevligt intryck i båda lägena [4].
En ytterligare faktor att ta hänsyn till är att ytan på bildskärmen är mindre på en mobil och en surfplatta än på en datorskärm. Det får helt enkelt inte plats lika mycket information samtidigt om det ska gå att läsa så utvecklaren måste välja vilken information som ska vara med. Annars kommer användaren att vara tvungen att zooma in och ut hela tiden och det är inte bra för användarvänligheten [11].


Miljö - I vilken miljö kommer användaren att använda applikationen? Kommer det att vara i solljus, mörker eller annat?
Kontext - Vad gör användaren vid användning av applikationen? Är den tillräckligt lätthanterlig och intuitiv så att användaren snabbt kan få ett grepp om de olika stegen i uppbyggnaden av applikationen? Även om telefonen ringer skall applikationen kunna hantera avbrottet och det fortsatta arbetet med applikationen skall kunna genomföras utan att störas av telefonsamtalet när användaren återvänder till applikationen.

Inmatningsmetoder - Vid uppbyggnaden av applikationen måste man ta hänsyn till hur användaren kommer att hålla i sin telefonenhet. Exempel på det ärstående eller liggande format, knappar och liknande skall fungera med applikationen och fungera med användarens hantering av enheten.

Förmågor - Den smarta mobilen (enhetens) förmågor är t.ex. mikrofon, kamera, virtuellt tangentbord, GPS, skakningar samt vridningar av telefonens läge (lodrät eller vågrät). Allt detta måste man ta hänsyn till och använda i applikationen för att eventuellt förbättra användarupplevelsen.

Metaforer - Vid design av applikationer bör man använda så vardagliga och kända egenskaper som möjligt (t.ex. en ikon som visar ett brev för att visa användaren att det är en meddelandefunktion). Detta bör genomsyrar hela applikationen för att underlättta för användaren när den skall interagera med applikationen [1].
2.2 Android


2.2.1 Androids Operativsystem

Androids operativsystem är ett öppet källkodsprojekt till mobila plattformar.

Dess mjukvarustack brukar delas in i fem sektioner. (se figur 2) [13].

![Figur 2: Androids arkitektur](image-url)
I det understa lagret i stacken finns en Linuxkärna som förser mobilen med basfunktionalitet som processhantering och minneshantering.

I nästa lager finns biblioteken som inkluderar till exempel webbläsaren, databasen SQLite, bibliotek för att spela upp och spela in audio och video och Internet säkerhet [13].

I det tredje lagret hittar vi Android runtime. Där finns en viktig komponent, DVM, Dalvik Virtual Machine. Android-applikationer körs inom en DVM som tillåter att varje applikation blir tilldelad sin egen process. Detta medför viss säkerhet i och med att applikationen blir isolerad från andra applikationer och får sitt eget lagringsutrymme samt minimal tillåtelse att nå andra delar av systemet. För att kunna nå eller utnyttja andra delar av systemet måste applikationen i fråga be om sådan tillåtelse [14].

I det fjärde lagret finns Application Framework och här får man till förfogande många tjänster som utvecklare kan dra nytta av som till exempel Java-klasser.

I det femte och sista lagret som befinner sig överst i mjukvarutacken hittar vi applikationslagret och där befinner sig alla applikationer [13].

2.2.2 Android applikation

2.2.3 En aktivitets livscykel
En vanlig Android-applikation består av en eller flera aktiviteter som representerar vyerna som användaren interagerar mot (se figur 3).
Aktiviteterna som inte befinner sig på toppen av stacken är i pausat eller stoppat tillstånd och löper risk att bli förstörda av operativsystemet för att exempelvis frigöra minne. Denna funktionalitet medför att om det ringer på telefonen så pushas den nya samtalsaktiviteten på stacken och den föregående aktiviteten övergår till pausat tillstånd. Förutom pausat tillstånd kan aktiviteten övergå till stoppat eller aktivt tillstånd varvid metoder som onPause, onStop eller onResume åkallas [14].

Figur 3: En aktivitets livscykel
2.2.4 Ett fragments livscykel
Ett fragment kan, till skillnad från en aktivitet, vara en del av ett användargränssnitt och tillsammans med andra fragment utgöra en hel vy i en applikation varav mer flexibla och varierande gränssnitt kan byggas (se figur 4).

Figur 4: Ett fragments livscykel

Ett eller flera fragment hanteras av en huvudaktivitet som sköter dess livscykel vilket medför att fragmenten indirekt påverkas av aktivitetens livscykel men har även ett par ytterligare lägen som associeras med att ta bort, förstöra, skapa eller lägga till fragment [14].
2.2.5 Ett Android-projekts uppbyggnad

Designmönstret Model-View-Controller (MVC) är en arkitektur som separerar användarens interaktion med applikationens data och dess presentation (se figur 5) [16].

![Diagram MVC](image)

Figur 5: Designmönstret MVC (Model-View-Controller)

Model är det första uttrycket i designmodellen och står för applikationens datahantering. På engelska kallas uttrycket content provider som kapslar in och tillhandahåller data till applikationer genom ett interface som kallas ContentResolver. Denna datahantering behövs om man behöver dela data mellan flera applikationer som exempel kan nämnas då man vill uppdatera en extern databas [59].

View är det andra uttrycket i designmodellen och är en komponent som hör till UI där varje fönster i en applikation ärvs från javaklassen android.app.Activity som i sin tur är en container för android.view.View. En aktivitet är en komponent där användarna kan utföra en operation i applikationen som till exempel ta ett foto eller skicka ett mail. En applikation består av flera aktiviteter som hör ihop med varandra men en aktivitet kallas ”main” som är den aktiviteten som drar igång applikationen. Sedan kan varje aktivitet utföra olika uppgifter och varje gång en ny aktivitet startar så stoppar den föregående men systemet ”sparar”aktiviteten i en stack som beskrivs i kapitel 2.2.3 som handlar om en aktivitets livscykel [59].
Controller är det tredje uttrycket i designmodellen och är en komponent som kan utföra långvariga operationer i bakgrunden på applikationen och behöver inte något interface för utförandet. En annan applikation kan starta en service och den kommer att fortsätta att köras i bakgrunden även om användaren byter till en annan applikation. Som exempel på en service vara nätverksöverföring [59].

Med den här modellen får inte datahanteringen (model) konsekvenser på grund av att det sker förändringar i presentationslagret (view) samt att data kan omorganiseras utan behöva ändra i presentationslagret. Detta löses genom den mellanliggande komponenten Controllern. Detta innebär att användaren påverkar en kontroll som i sin tur påverkar en modell. Modellen består av applikationsdata och affärslogik. Kontrollen tolkar användarens input och översätter det till kommandon för modellen [59].

Vyerna representeras med hjälp av statiska xml-filer som beskriver dess initiala utseende och en aktivitet. Förutom att uppdatera och visa vyn har aktiviteten även ansvaret att fånga och behandla användarens interaktioner såsom knapp-tryckningar och andra händelser som kan tänkas inträffa [14].

Ett projekt består av och byggs upp med hjälp av en specifik xml-fil som heter manifest. Manifestet innehåller information om applikationens aktiviteter, vilken lägsta och högsta API-nivå man avser stödja samt krav på systemresurser [14].
2.2.6 Databas
MySQL - MySQL är en relationsdatabas och används ofta i webbapplikationer [60].

SQLite - Som man kan se i figur 2 som visar Androids arkitektur är SQLite en databas som ingår i Androids Operativsystem. Det specifika med denna databas är att den är komprimerad till en enda fil där all data hanteras och lagras direkt på filen istället för på en extern server. Avsikten med SQLite är att den ska vara liten och kompakt, resursbesparande och enkel att administrera och underhålla. Den är en relationsdatabas likt SQL men har begränsat minne vid exekvering. SQLite kan bäddas in i andra applikationer [17] [18].


PHP - Det vanliga vid kommunikation med en databas och en applikation är att använda sig av skriptspråket PHP (Hypertext Preprocessor) [21] som API (Se figur 6) [20]. Dess uppgift är att vara filtret mellan databasen och applikationen som ska tolkas och översättas av en interpreter för att sedan omvandlas till en textström som skrivs ut. Detta skriptspråk används mest på webbservrar för att driva internetsajter med dynamiskt innehåll. Interpreteren är för det mesta JSON som beskrivs i nästa stycke.

JSON - JSON (JavaScript Object Notation) [22] används då man vill utväxla data oftast mellan en server och en webbapplikation. JSON är ett språkoberoende format som består av två grundstrukturer som nästlas in i varandra för att bygga upp och beskriva avancerade objekt eller liststrukturer.

Den andra strukturen är en lista av värden där värden kan vara grundvärden som strängar och nummer men även komplexa strukturer som hundobjektet ovan eller andra innästlade listor. Listor startar och slutar med hakparenteser och värdena är separerade med kommatecken. En enkel lista med namn som även den är tagen från referens [14] skulle kunna se ut såhär: ["olle" , "kalle" , "pelle" , "nisse"].

Kommunikation - Ett anrop mellan en mobil och en databas kan ske enligt följande (se figur 6). En applikation anropar ett PHP-script för att utföra någon av basoperationerna. PHP-scriptet kodar informationen med hjälp av JSON och ansluter sedan till MySQL-databasen för att verkställa operationen så att data kan flöda mellan applikationen till PHP-scriptet för att till slut lagras i MySQL-databasen. SQLite-databasens uppgift är att lagra användarinformation lokalt på android-telefonen [20].

![Diagram](image-url)

Figur 6: Kommunikation mellan databas och mobil via PHP som API
2.3 Säkerhetsaspekter


Nämns bör att i mitt examensarbete ingår det att bygga en mobilapplikation till operativsystemet Android. Android ägs av Google som numera via Google play kontrollerar och skannar igenom alla applikationer, om de kan vara skadliga och utgöra hot på olika sätt, innan utgivning [23]. I Kapitel 2.5 berättas mer om Androids säkerhet.

2.3.1 Kryptering

Om man vill skicka information som är hemlig och som inte andra ska ta del av bör man kryptera denna information [23]. Detta gäller mobila enheter likväl som stationära enheter. Vilken krypteringsmetod man ska använda beror på vad man ska kryptera men målet är att det inte ska gå att knäcka metoden för en utomstående [24].

De vanligaste varianterna av kryptering är av typerna symmetrisk och asymmetrisk kryptering. Vid båda dessa metoder använder man sig av en algoritm och en eller flera nycklar. Vid symmetrisk kryptering måste både sändare och mottagare ha tillgång till samma nyckel och då uppstår problemet med hur nyckeln ska överföras på ett säkert sätt så att den inte kommer i orätta händer. Det finns dock en fördel med symmetrisk kryptering och det är att den är snabb. 3DES (Triple-DES) och AES
(Advanced Encryption Standard) är två exempel på symmetrisk kryptering [25].


Symmetrisk kryptering används i huvudsak då det är mycket information som ska krypteras. För att kringgå problemet vid nyckelhanteringen i detta fall kan man skicka nyckeln med asymmetrisk kryptering [25].

En krypteringstandard som tillämpas i första hand vid kommunikationen mellan webbläsaren och webbserver är SSL/TLS (secure socket layer/transport layer security). Man kan se att en webbsida har säker förbindelse och använder denna säkerhetsmekanism om dess adress börjar på “https://” istället för “http://”. TLS är en vidareutveckling av SSL men det brukliga är att man benämner standarden SSL/TLS. SSL/TLS grundar sig på asymmetrisk kryptering där den publika nyckeln används vid kryptering och skickas till webbläsaren för anslutning till servern och den privata nyckeln används vid dekryptering av informationen som krypterats med den publika nyckeln. För att komma förbi problemet med att serverns nyckel tillhör rätt server används ett certifikat som är ett bevis på ägandet. Certifikaten har skapats av CA (Certificate Authority) som agerar som tredje part och verifierar att förbindelsen är korrekt [27].
2.3.2 Autentisering

2.3.3 Säker lagring av data
Mobilapplikationerna sköter mer och mer lagringen lokalt i applikationen än lagring i servern. Det innebär att man måste ta hänsyn till säkerheten även i detta fall. Ett exempel med denna problematik kan vara hantering av bankkonton. En aspekt är att inte lagra lokalt som inte behöver lagras lokalt. Bara för att man har en bankapplikation behöver inte kontonumren lagras i applikationen. Om informationen måste lagras lokalt på applikationen bör man använda någon av krypteringsmekanismerna [23].

2.3.4 Validera input
Inmatningen till applikationer bör kontrolleras. En ytterligare säkerhetsaspekt är om inte användarna är pålitliga. Det som skrivs eller skickas till applikationen kan vara ett systematiskt sätt att skada. Utvecklare bör ha uppsikt över det som matas in till programmet och anta att det alltid kan komma från någon som medvetet vill skada [23].

2.3.5 Testa säkerheten
Det är viktigt att inte bara testa funktionaliteten, utan även säkerheten i mobilapplikationer innan de släpps på marknaden. Annars kan någon annan som har onda avsikter upptäcka säkerhetshålen [23].
2.4 Databassäkerhet

Det är essentiellt att skydda en databas av flera orsaker. Det kan vara fel i indatan, fel i mjukvaran eller hårddvaran som leder till korrumpertad data, regelbrott mot generella policyn eller användare som vill göra intrång. Man kan införa databasskydd genom att införa säkerhetsåtgärder och integritetsvillkor för innehållet.


2.5 Androidsäkerhet


En ytterligare säkerhetsaspekt i operativsystemet, som jag beskrev i Kapitel 2.2.5, handlade om designmönstret Model-View-Controller. Där får en applikation inte använda funktioner som kan inverka på andra applikationer, operativsystemet eller användaren. Men det går att kringgå dessa skyldigheter av utvecklaren genom att göra en specifikation i Manifest-filen [31].
2.6 Sammanfattning

Kapitlet beskriver projektets teoretiska bakgrund och innehåller teori som ingår direkt och indirekt i examensarbetet. Det är teoretiska områden som belyses för att öka förståelsen över vad denna rapport handlar om och en del av dessa områden används sedan vid det praktiska arbetet vid framställningen av ByggletMobil.

Nästa kapitel handlar om projektets bakgrund och en motivation till varför applikationen som denna rapport handlar om behövs.
3 Projekt bakgrund


3.1 Projektet

Det webbaserade programmet Bygglet är ett projektanteringsverktyg inom området byggnad och riktar sig i huvudsak till företag som arbetar i projektform. Programmet Bygglet har utvecklats av Bygglet AB som har som försäljningsidé att sälja Bygglet till både stora och små företag inom byggbранchen. Bygglet AB tillhandahåller även utbildning av programmet, support och löpande uppdateringar till sina kunder.

I detta program kan man idag ha kontroll över tidsregistrering, projekt, arbetsorders, artiklar, offerter, kunder, personal, lön och bokföring. Eftersom Bygglet AB har beslutat att expandera sina tjänster för att nå ut till mobiltelefonanvändare och samtidigt surfplatteanvändare beslöts det att det skulle utvecklas en Android mobilapplikation som tillägg till sina tjänster och som resulterade i mitt examensarbete. En essentiell förutsättning för denna tjänst är att inriktningen endast ska gälla mot enheter som har pekskärm.

Idag finns en mobilapplikation där projektledaren endast kan registrera byggnadspersonalens arbetade tider på olika arbetsplatser. Programmet behöver nu utvecklas eftersom en projektledare inom byggnadsområdet reser mellan olika arbetsplatser och har ofta hand om flera projekt och
flera olika personalgrupper. Istället för att ha med sig en eller flera portfolios i form av dokumentation ska projektledaren kunna klara sig med sin mobil.

Examensarbetet består av att skapa en prototyp av en mobilapplikation i programmeringsspråket Java Android. Prototypen ska fungera som en första version som sedermera ska utvecklas till en större applikation och byggas vidare på av uppdragsgivaren själv. Prototypen ska endast innehålla de mest elementära delarna som behövs till ett användargränssnitt till Bygglet. Prototypen har blivit namngiven ByggletMobil.

ByggletMobil ska vara ett verktyg till en projektledare på ett stort eller litet företag då han kan använda sin mobil som hjälpmedel i sitt dagliga arbete. Det mobila verktyget är tänkt att fungera som ett tittskåp där användaren kan se valda delar från Bygglet.

Informationen i systemet kräver, av säkerhetsskäl, en inloggningssida med användarnamn och lösenord givna av Bygglet AB.

### 3.2 Projektavgränsningar

Följande omfattats ej av projektet:

- Ingen kod av mobilapplikationen får visas för tredje part enligt uppdragsgivaren Jonas Wånggren VD, Bygglet AB.
- All hantering av databasen, inkluderat dess säkerhet etc. sköts av Bygglet AB och ingår ej i uppdraget.
- Mobilapplikationen skall visas i vertikalt läge.
- Vad gäller personalinformation skall applikationen endast visa personalens namn, mobilnummer och e-postadress.
I kundinformationen skall man endast se kundens namn, mobilnummer och e-postadress.

All eventuell övrig funktionalitet, tex en sökfunktion, ingår ej i uppdraget.

Prototypen innehåller ingen specifik design med avseende på Bygglet AB.

Säkerhet med avseende på HTTPS, dvs krypterad kommunikation mellan Bygglet Mobil och Bygglet, ingår ej i projektet.

3.3 Bygglet AB


I februari år 2011 lanserades Bygglet och har idag 5 st anställda med huvudkontor i centrala Göteborg. Verksamheten är rikstäckande och riktar in sig idag på utveckling av programvaran till byggsektorn som entreprenadföretag, snickare, målare, golvläggare, markanläggare, rörmokare osv. Bygglet har på sin hemsida en slogan ”Vi hjälper dig och ditt företag att bli mer effektivt. Släng alla pärmar och sätt upp lite blommor istället” [32].
3.4 Bygglet


![Figur 7: Översikt Bygglet](image_url)
Figur 8: Personal Bygglet

I fliken personal (se figur 8) kan man samla information om all personal. Där finns personuppgifter, namn, adress, telefonnummer, utbildningar och dokument. Det går att skapa och skriva ut anställningskontrakt.

I fliken kunder kan man samla information om alla kunder.


I fliken offerter kan man skapa offerter som man sedan kan maila till kunden direkt ifrån systemet. Sedan kan kunderna via en länk i mailen svara på offerten och svaret registreras då direkt i Bygglet.
I fliken projekt kan man samla information om alla projekt. Man kan lägga upp nya projekt som får ett löpnummer. Alla arbetade timmar, använt material och leverantörsfakturor styrs mot projektnumret. För varje projekt kan man skapa arbetsorders till de anställda.

I fliken tidregistrering kan man fylla i tidrapporter direkt via en mobiltelefon eller en dator. Här ges möjlighet att ange arbetad tid, använt material samt skriva kommentarer och dagboksanteckningar för varje projekt. Sedan kan man ta ut ett färdigt löneunderlag direkt ur systemet.

I fliken kundfakturor har man tillgång till ett faktureringssystem. Offerter, tidrapporter och leverantörsfakturor ligger som färdiga faktureringsunderlag och där kan man skapa fakturor och skicka dem till kund.

I fliken leverantörsfakturor kan man få alla leverantörsfakturor inskannade och tolkade direkt i systemet. Leverantörsfakturorna kopplas till rätt projekt för vidarefakturering och kostnadskontroll.

3.5 Tillgången på IT-teknik inom byggnadssektorn

Jag ställde mig frågan om IT-tekniken och dess tillämpningar är hög inom byggnadssektorn. Använder entreprenörer mobilapplikationer för att underlätta sitt arbete? Efter att ha läst ett urval artiklar [33], [34], [35] på undersökningar som gjorts tidigare dras slutsatsen att entreprenörer inom byggnadssektorn är den yrkeskategori som använder IT-teknik minst.

Förekomsten av IT är dock hög inom branschen men då inom administration, kommunikation och projektering. Olle Samuelsson skriver i sin licentiatavhandling ”IT-användning i byggande och förvaltning” [33] att den största orsaken att använda IT ska vara om man kan ha stor nytta av det direkt i arbetet och en av de största motgångarna för bruk av IT är av finansiell art. Det är inte många företag som vill göra en ekonomisk satsning till något som inte genererar direkt avkastning. I
doktorsavhandlingen ”IT-Innovationer i svenska bygg- och fastighetssektorn” [34] som är författad av samme Olle Samuelsson kan man läsa att företag investerar i IT för att göra redan existerande arbetsmetoder och förlopp bättre men fokus för ökad IT-teknik ligger mot området administration.

En annan teori utläses i magisteruppsatsen ”IT-stöd för anläggningsarbetare i byggbranschen” [35]. Där påstår Mattias Grävare att byggbranschen är konservativ och människorna där i första hand tänker på att prestera sitt hantverk och en produkt och i andra hand dess administration. Det har även saknats bra applikationer som är tillämpningsbara inom sektorn. Det är flera aktörer inblandade i varje bygghandelsprojekt både direkt och indirekt med huvudleverantörer och underleverantörer så det har tills idag varit svårt att få varje aktör som befinner sig i sin del av systemet att ”prata” med en aktör som befinner sig i en annan del i systemet. Det finns dock pågående projekt som arbetar med lösningar av nämnda problem.

3.6 Befintliga applikationer

I projektet gjordes en egen undersökning genom att göra enkla sökningar på Internet med avseende på att undersöka om det finns några webbaserade applikationer liknande Byggle. En jämförelse gjordes mellan applikationernas funktioner på ett visuellt sätt.

Applikationerna var specter [36], projectplace [37], Qbis [38] och Visma [39].

Specter är ett affärssystem som inriktar sig mest mot ekonomin. I Specter finns det funktionalitet till offerter, fakturering, kassahantering, lagerhantering, inköp, leverantörer och bokföring.

Projectplace är ett projekthanteringsverktyg som innehåller funktioner kring projektledning, dokumentation och kommunikation.
Qbis är ett affärssystem innehållande funktioner till projekthantering, tidrapportering, styrning, organisering och administration för kunder och ärendehantering.

Visma är även det ett affärssystem som riktar sig mot ekonomin, bl.a. med fakturering, lönehantering och bokföring.

I denna undersökning hittades ett antal affärssystem som är inriktade mot de ekonomiska bitarna i ett företag som löner och bokföring och ett antal projekthanteringssystem som är inriktade mot företag som arbetar i projektform. Inget annat befintligt system hittades som har samma funktionalitet som finns i Bygglets applikation. Bygglets prestanda täcker in tidsregistrering, projekt, arbetsorders, artiklar, offerter, kunder, personal, lön och bokföring. De applikationerna som undersöks innehar några av Bygglets bitar som de ekonomiska delarna eller delarna med projektverktygen men inte alla.

I Kapitel 3.3 nämns att byggsektorn behöver fler bra tillämpningsbara applikationer som de som arbetar inom byggnadssektorn kan använda i sitt arbete. Målet är också att ByggletMobil i framtiden ska utvecklas till en bra tillämpningsbar applikation som förenklar arbetet i byggnadsbranschen.
3.7 Bygglet vs ByggletMobil

Som det beskrivs i kap 3.1 är Bygglet ett projekthanteringsverktyg vars huvudinriktning är mot företag inom byggnadsbranschen. Verktyget är tänkt att vara till hjälp och underlätta vid dokumentationen runt personal, ekonomi, löner och bokföring. ByggletMobil är tänkt att vara en hjälp till projektledarna som arbetar med Bygglets ordinarie verktyg.

ByggletMobil är ett interface mot Bygglets server och är i detta projektarbetes stadium en prototyp som ska fungera som en första version som sedermera ska utvecklas till en större applikation och byggas vidare på av Bygglet själv. Prototypen ska innehålla funktioner för att kunna lista och skriva ut på skärmen personal och kunder som är sparade i en databas.

3.8 Uppdraget applikationen ByggletMobil

I uppdraget rörande Bygglet mobilapplikation ingår att ta fram en mobilapplikation, en prototyp i operativsystemet Android, bestående av ett tittskåp som primärt skall visa kund- och personalinformation från det webbaserade programmet Bygglets databas. Prototypen ska kunna vidareutvecklas av updragsgivaren.

Följande funktionalitet vill uppdragsgivaren att applikationen ska innehå: 

- Inloggningstvåning med användarnamn och lösenord
- Lista kundinformation och personalinformation med namn, adress och telefonnummer
- Användarinformation, kundinformation och personalinformation ska sparas och hämtas i en extern mySQL-databas
- Utloggningstvåning
- Applikationen ska vara robust och användarvänligt
3.8.1 Flödesdiagram
Nedanstående flödesdiagram visar ur ett användarperspektiv den funktionalitet som utvecklas i detta projekt (se figur 9). Flödesdiagrammet ska läsas uppifrån och ner.

![Flödesdiagram ByggletMobilapplikation](image)

Figur 9: Flödesdiagram ByggletMobilapplikation

3.9 Sammanfattning
Kapitlet handlar om projektets bakgrund och delkapitel 3.3 visar att det behövs mer IT-teknik inom byggbranschen, speciellt inom byggnadsområdet för projekthanteringssystem. Det finns mycket teknik inom administration, kommunikation och projektering men inte som direkt hjälpmedel för projektledaren ute på fält. Detta belyses i Kapitel 3.4. Det finns ett behov av en applikation som är robust, användarvänlig och lätt hanterlig och det är de egenskaper som mobilapplikationen ByggletMobil ska ha.
4 Projektets design och Implementation

I detta delkapitel beskrivs hur det praktiska arbetet med projektet ByggletMobil genomfördes.

Kapitlet är indelat i faser och inleds med att redogöra för planeringsfasen där idén fastställdes och de inledande diskussionerna med uppdragsgivaren tog plats. Sedan följer initieringsfasen som gick ut på bland annat val av utvecklingsplattform och inläsning av litteratur. Den tredje fasen kallas designfasen som innehöll arbetet med programmets design och arkitektur. Den fjärde och sista fasen heter programmeringsfasen och där beskrivs hur implementeringen gick till och visar resultatet i form av skärmbilder.

4.1 Planeringsfas

Uppdraget bestod i att skapa en mobilapplikation, en prototyp, bestående av ett användargränssnitt till Bygglet i programmeringsspråket Java. ByggletMobil fungerar som ett tittskåp där användaren kan se kundregistret och personalregistret från Bygglets databas. På grund av sekretess av Bygglets egen databas har jag skapat en egen databas som jag fyllt med fiktiv data.


Då uppdraget blev godkänt för start både från uppdragsgivaren och från skolans examinator började jag leta efter vilken utvecklingsplattform som skulle användas. Det visade sig efter lite eftersökning att ett bra val var Eclipse [56] där Android SDK, Software Development Kit, var ett inbyggt

Hårdvaran som har använts till examensarbetet är en Ultrabook Aser i7 4 GB RAM [57]. Operativsystemet som använts är Ubuntu Linux 14.04 [58]. Trots att denna dator är av god prestanda gick exekveringen långsamt då Eclipse och emulatörn tar mycket av datorns minne.

4.2 Initieringsfas

I början av denna fas i projektet gick den mesta tiden, cirka två månader, åt till att installera ADT och lära sig att använda mjukvara och rent praktiskt hur det går till att programmera Java Android. Mycket tid, mer än tre månader, gick åt till att studera litteratur om Androidprogrammering och hur utvecklingsplattformen med emulatörn fungerar.

Installation av utvecklingsplattformen fanns på Developers hemsida [40]. Där fanns ADT Bundle tillgänglig via zipfilen ”adt-bundle-linux-x86_64-20140321.zip”. I den zipfilen fanns Eclipse IDE med en inbyggd ADT. Zipfilen packades upp i hemkatalogen och sedan var det bara att starta Eclipse. ADT finns redan inbäddad i utvecklingsmiljön. Emulatörn som var nödvändig att ha fanns också inkluderad. Där tillåts utvecklingsarbetet och testningsarbetet av applikationerna utan att använda en fysisk telefon vid sidan av datorn så det var bara att sätta igång att testa.

Ett flertal enkla testapplikationer programmerades för att få lärdom rent praktiskt hur alltihop arbetade tillsammans. Fyra webskolor användes i huvudsak. De var Developers [41], Vogella [42], androidhive [44] och coderzheaven [45].
4.3 Designfas

Denna fas i projektet krävde en hel del planering. Planeringsarbetet utgick från punkterna i kapitel 2.1.3 som handlade om design av GUI och från punkterna i kapitel 3.2 som handlade om projektets avgränsningar. Texten i detta kapitel beskriver tankegången runt designen av applikationen.

4.3.1 Designbegränsningar

Eftersom applikationen ska användas i vertikalt läge har designen av gränssnittet och implementationen av xml-filerna utgått utifrån det alternativet.

Hänsyn har tagits till att det är en storleksmässigt mindre bildskärm på en mobiltelefon så valet har blivit att det ska vara så få inmatningsrader och knappar som möjligt i varje fönster. Det medför också hänsynstagande till problemet med ”fat fingers”.

Applikationen kommer att användas i de flesta byggnadsmiljöer. Den fungerar i både lju och mörker.


Det ska användas enkel grafisk design till interfacen som hör ihop med autentisering och då man är inloggad i funktionsdelen. Orsaken till detta var att göra ByggletMobil så användarvänlig som möjligt. Till exempel har ett typiskt autentiseringsfönster (se figur 12) som redan är känd av de flesta användare implementerats och till gränssnittet i funktionsdelen har fokus legat på enkelhet (se figur 14).
4.3.2 Designmönstret MVC
Den övergripande arkitekturen är baserad på designmönstret Model-View-Controller, MVC som beskrivs i Kapitel 2.2.5. Vyerna representeras av statiska XML-filer som tillsammans med respektive aktivitet utgör en dynamisk vy. Aktiviteten har även ansvaret som controller då den tar emot användarens interaktioner och förmedlar dem vidare i rätt ordning till modellen.

4.3.3 Flödesdiagram
Bilden nedan visar ett flödesdiagram av ByggletMobils design (se figur 10).

![Flödesdiagram av ByggletMobils design](image)

Man startar applikationen genom att klicka på ByggletMobils ikon i telefonen. Man kommer då till ett autentiseringsfönster där man ska logga in med e-postadress och lösenord. Om inloggningsuppgifterna inte stämmer efter kontroll med de uppgifter som redan finns på databasen skrivs ”Fel användarnamn/lösenord” ut på skärmen. Om inloggningsuppgifterna stämmer kommer användaren vidare till ByggletMobils UI. Där kan användaren välja att lista personal eller lista kunder eller logga ut. Om valet är att logga ut kommer användaren tillbaka till inloggningsfönstret.
4.4 Programmeringsfas

I denna fas ägde ByggletMobils implementering rum som följe designmönstret MVC. Uppbyggnaden består av 4 delar, Autentisering med användarnamn och lösenord, ByggletMobils funktionsdel med knappar, databaserna med SQLite och MySQL och API.

4.4.1 Klasser - Java

LoginActivity- hanterar inloggning och utloggning med användarnamn och lösenord.

DashboardActivity - kollar om användaren är inloggad och listar kunder och personal på skärmen.

DatabaseHandler - använder databasen SQLite för att lagra användarinformation tillfälligt då användaren är inloggad.

UserFunctions - hanterar och kontrollerar loginstatus med användarnamn och lösenord.

JSONParser - hanterar parsningen av strängen till ett JSON objekt.

GetDataFromDB - hämtar kunder och personal från databasen MySQL.

Users – används för att hantera objektet kunder och personal från databasen MySQL.

4.4.2 Designmönstret MVC

Implementeringen följde designmönstret Model-View-Controller som förklarades i kapitel 2.2.5.

Den första termen Model härledde till datahanteringen mellan telefonen och MySQL-databasen. Klasserna android.content.ContentValues och android content Context importerades till klassen DatabaseHandler och android content Context importerades till klassen UserFunction för
hantering av applikationsdata och status av användarinformation och databasinformation i form av data från tabellerna kunder och personal.


Den tredje termen controller härledde till servicen med nätverksöverföringen i klassen GetDataFromDB där anslutningen upprättas med den externa MySQL-databasen.

4.4.3 Grafiskt gränssnitt Autentisering
Detta delkapitel demonstrerar i form av skärmdumpar hur de olika skärmsvisningarna åskådliggörs då ByggletMobil exekveras.

Bilden nedan visar ”startläget” efter kompilering av den javakod som skrivits in i utvecklingsmiljön. Vad som visas är emulatorm som simulerar en Androidtelefon. Här väljs att klicka på ByggletMobils ikon, för att sedan komma vidare i processen (se figur 11).
Figur 11: Emulatorn med ByggetMobils ikon

När man tryckt på ByggetMobils ikon på androidtelefonen kommer man till autentiseringsfönstret. Detta fönster består av två rader och en knapp. På raderna ska man skriva in sitt användarnamn som i detta fall är e-postadressen och sitt lösenord (se figur 12).
Figur 12: Inloggningssida till ByggletMobil

Om inte användaruppgifterna stämmer skrivs det ut på skärmen ”Felaktig e-mail/lösenord” (se figur 13).

Figur 13: Inloggningssida till ByggletMobil vid felaktig inmatning

Man får mata in på nytt tills uppgifterna stämmer och först då har man tillträde till ByggletMobil.
4.4.4 Grafiskt gränssnitt ByggletMobils funktionsdel
När användaren har blivit autentiserad kommer man till nästa del i applikationen där funktionaliteten finns. I detta fönster skrivs ”Välkommen” ut på skärmen så att användaren vet att han har blivit inloggad. I fönstret presenteras också tre knappalternativ. Knapparna är ”Kunder”, ”Personal” och ”Logga ut mig” (se figur 14).

Figur 14: ByggletMobils första sida
Knapparna ”Kunder” och ”Personal” länkar vidare till respektive tabell i databasen och knappen ”Logga ut mig” länkar tillbaka till inloggningsfönstret.

Då användaren har tryck på knappen ”Kunder” listas kunderna med deras namn, adress och telefonnummer, från MySQL-databasen upp på skärmen (se figur 15).

**Figur 15:** ByggletMobil när användaren tryckt på knappen kunder
Samma sak händer då användaren trycker på knappen “Personal”. Man kan då se en lista med personalens namn, adress och telefonnummer (se figur 16).

Figur 16: ByggletMobil när användaren tryckt på knappen personal
4.4.5 Databas

I applikationen används Androids egen databas SQLite och den externa databasen MySQL. SQLite används för att lagra användarinformation så att programmet senare kan kontrollera om en användare är inloggad eller ej och i MySQL finns tabellerna som hör till kunder och personal.

Eftersom Bygglet AB på grund av sekretess inte vill att jag ska ha tillgång till deras egen MySQL-databas har jag själv skapat de nödvändiga tabellerna som applikationen behöver i en annan extern server. MySQL-databasen som jag använt befinner sig på en server i Stockholm på webbhotellet Binero [43]. I figur 11 finns en bild av tabellen kunder från databasen MySQL.

![Figur 17: Databas MySQL](image)
4.4.6 Klasser - API
API-kontraktet skrivs i programmeringsspråket PHP och innehar följande klasser:

- **config** - denna klass innehåller de konfigurationskonstanter som behövs för att ansluta till databasen.

- **DB_Connect** - denna klass används för att ansluta eller koppla från databasen.

- **DB_Functions** - denna klass innehåller funktioner för att nå användare från databasen. De funktionerna heter `getUserByEmailAndPassword` som hämtar användaren från databasen om inloggningsuppgifterna stämmer och `isUserExisted` som kontrollerar om användaren existerar.

- **index** - denna klass innehåller funktioner för att hantera förfrågningar. Vid varje förfrågning kommer den att tala med databasen och ge den rätta responsen i JSON format.

4.4.7 Projektets säkerhetsaspekter
Bygglet Mobils säkerhetsaspekter utgår från samma säkerhetstäck som beskrivs i kapitel 2.3 som handlar om säkerhetsaspekter allmänt och i kapitlet 2.5 som handlar om Androids säkerhet.

- **Kryptering** - Någon kryptering har inte utvecklats i implementationen eftersom det inte ingick i uppdragsspecifikationen.

- **Autentisering** - Implementationen innehåller ett inloggningsfönster där det efterfrågas användarnamn/e-mail och lösenord.

- **Säker lagring av data** - Data lagras i en extern server.

- **Validera input** - Det finns ingen specifik kontroll implementerad för att validera vad som matas in till applikationen eftersom det inte ingick i uppdragsspecifikationen.
• Testa säkerheten - Applikationen har blivit testad och godkänd av projektets uppdragsgivare.

• Databassäkerhet – Ingen hänsyn har tagits till skydd av databasen eftersom det inte ingick i uppdragsspecifikationen.

4.5 Sammanfattning

Detta kapitel handlar om prototypen ByggletMobils hela utvecklingsarbete från idé till färdig produkt. Kapitlet är indelad i faserna planeringsfas, initieringsfas, designfas och programmeringsfas där problem och lösningar som uppstått under utvecklingsarbetet beskrivs.

Redovisning av resultatet behandlas i kapitel 5.
5 Resultat

Enligt uppdragsspecifikationen skulle en prototyp av en mobilapplikation utvecklas i programmeringsspråket Java. Prototypen är ett tittskåp till det webbaserade projekthanteringsprogrammet Byggle. Alla punkter i uppdraget har uppnåtts och godkänts av uppdragsgivaren, se bilaga 2.

Det färdiga resultatet har uppnått den funktionalitet som uppdragsgivaren önskade som kan läses i kap 3.8.1.

- Inloggningsfönster med användarnamn och lösenord har implementerats. Användaren har autentiserats på en extern MySQL-databas.
- Lista kundinformation och personalinformation med namn, adress och telefonnummer. Detta har levererats och fungerar enligt specifikation.
- Användarinformation, kundinformation och personalformation ska sparas och hämtas i en extern MySQL-databas. Detta har levererats och fungerar enligt specifikation.
- Utloggningsfunktionalitet. Detta har levererats och fungerar enligt specifikation.
- Applikationen ska vara robust och användarvänlig. Detta har levererats och fungerar enligt specifikation.
5.1 Testkörning

Applikationen har blivit testad och provkörd enligt designprinciperna i kapitel 2.1.3. Inga fel har upptäckts och programmet fungerar utan anmärkning.

Bygglet AB har också testat och godkänt applikationen med positivt resultat. Se bilaga från Bygglet AB. Vad gäller Bygglet AB:s testmetoder kan dessvärre inte redovisas på grund av att Bygglet AB inte vill offentliggöra sina testmetoder av säkerhetsskäl.
6 Slutsats

Detta delkapitel är en utvärdering av projektet som helhet. Kapitlet är en summering av arbetet, en sammanställning av en del av erfarenheterna som införskaffats under projektets gång, några problem som uppstått och några idéer för framtida utveckling kommer att behandlas.

6.1 Summering av arbetet

Examensarbetet vars uppdragsgivare var Bygglet AB, bestod i att skapa en prototyp till en mobilapplikation. Arbetet har utförts i Karlstad medan att företagets huvudkontor befinner sig i Göteborg.

Applikationen skulle sedermera fungera som hjälpmedel till en projektledare som arbetar i projekthanteringssystemet Bygglet. Projektledaren ska kunna hämta kunduppgifter och personaluppgifter i en befintlig databas.

Kontakten med uppdragsgivaren har flutit på ett kontinuerligt och bra sätt via e-post och Skype. Bygglet AB har även vissa gånger fungerat som det bollblank som en projektpartner skulle varit.

Vid starten av examensarbetet hade jag begränsade kunskaper om både programmeringsspråket Java, Android och utvecklingsplattformen Eclipse. Dessa faktorer gjorde det svårt att bedöma tidsåtgången av arbetet. Den första tiden gick åt till inläsning av litteratur och lära Java, Android och hur Eclipse fungerar.

Det slutliga resultatet har nått upp till kravspecifikationens mål och Bygglet AB är nöjd med resultatet.

Genom att utveckla denna applikation har jag själv fått stora erfarenheter i programmering och hur en applikation kan vara uppbyggd med API och databas.

Tidsåtgången har visat sig vara högre än vad jag till en början antog pga ovanstående anledningar.
6.2 Problem och dess lösningar under projektet

I och med att projektet har avslutats med en demonstrationsprototyp, har projektet lyckats.


Ett problem som jag haft vid implementationen var arbetet med SQLite-databasen. Det uppstod svårbegripliga fel och en databas som inte uppdaterad sig efter att jag hade ändrat i koden på specifikt ställe. Detta problem var svårt att felsöka men till slut hittade jag svaret på programmeringsforumet stackoverflow. [46].

Lösningen var så simpel att jag bara skulle ändra databas versionen.

private static final int DATABASE_VERSION = 2;

Ett annat problem som tog lång tid att förstå på grund av en missuppfattning från min sida var varför man skulle använda två olika databaser, både SQLite och MySQL.

SQLite-databasen är telefonens interna databas och hanterar om användaren är inloggad eller ej medan i MySQL-databasen finns tabellerna som lagrar kund och personalinformation.
6.3 Framtida arbete

Fortsättning av utvecklande av ByggletMobil kan vara att utveckla den design som jag valt att använda mig av i projektet. Här finns det mycket att göra för att inlemma applikationen i Bygglets egen företagsprofil.

Att utveckla säkerheten vad gäller nätverksöverföringen med HTTPS, dvs krypterad överföring av data till applikationen.

Vidare kan man fundera kring om man vill begränsa antalet inloggningsförsök innan konton låses. En inaktivitetsspärr, tidsmässig, kan också vara av värde då många användare glömmer att logga ut ur applikationen. Applikationen loggar då automatiskt ut användaren från applikationen och avbryter kommunikationen med databasen.

Möjligheten att skapa en möjlighet för användaren att registrera sig via applikationen finns, men då måste Bygglet skapa en funktionalitet för att godkänna registreringen.

En sökfunktion, för att enklare hitta sökt data, om databasen är omfattande kan vara av värde att implementera.

Sist, men inte minst, kan en filtreringsfunktion skapas. Här tänker jag att man tex kan visa alla kunder i en specifik ort.
References


[8] Johnson, Niklas och Smogner, Peter, **En modern mobilapplikation, från idé till produkt**, KTH Information and Communication Technology 2011 (Artikel)


[10] Johnsson, Marta, **Adapt or Die Användarnas preferenser av mobila webbplatser** 2012 (Artikel)


[27] IT-säkerhet.nu http://www.it-säkerhet.nu/archives286 20131028


[29] Uppsala Universitet Department of Information Technology http://user.it.uu.se/~torer/kurser/dbt/integritet.pdf 20131028


[33] Samuelsson, O, IT-användning i byggande och förvaltning, KTH Industriell ekonomi och orgnisation 2003 (Artikel)

[34] Samuelsson, O, IT-Innovationer i svenska bygg- och fastighetssektorn En studie av förekomst och utveckling av IT under ett decennium 20130923


[36] Specter http://www.specter.se 20140505

[37] Projectplace https://www.projectplace.se 20140506

[38] QBIS http://www.qbis.se 20140507

[39] VISMA http://www.visma.se 20140508
[40] Developer http://developer.android.com 20130905

[41] Developer
   http://developer.android.com/training/basics/firstapp/creating-project.html 20130903

   20131022

[43] Binero http://www.binero.se 20140303


[48] Android www.android.com 20140924

   20140923

[50] Developer
   http://developer.android.com/design/patterns/widgets.html 20140925

[51] .se\Internetstatistik, Findahl, Olle. Svenskarna och Internet 2011,
   DanagårdsLiTHO, Ödeshög, 2011

[52] Marcotte, E. Responsive web design, A Book Apart, New York, USA
   2011

[53] Daytona http://www.daytona.se/responsiv-webbdesign 20140926

   20140927


[57] Acer http://us.acer.com/ac/en/US/content/aspire-s3-ultrabook
       20140919

[58] Ubuntu Linux http://www.ubuntu.com/download 20140918

[59] Bogotobogo
       www.bogotobogo.com/DesignPatterns/mvc_model_view_controller_pattern.php 20141031

Bilagor

Bilaga A - Beskrivning av examensarbete

Utvecklande av ett användargränssnitt tillhörande till det webbaserade programmet Bygget

Programmet Bygget är ett verktyg inom området byggnad och riktar sig i huvudsak som verktyg till företag som arbetar i projektsform. I detta webbaserade program kan man idag ha kontroll över tidsregistrering, projekt, arbetsordrar, artiklar, offertur, kunder, personal, lön och hulköring.

Idag finns en mobilapplikation där projektledaren endast kan registrera byggnadspersonalens arbete, räder på olika arbetsplatser. Programmet behöver nu utvecklas eftersom en projektledare inom byggnadsmiljön räder mellan olika arbetsplatser och har ofta hand om flera projekt och flera olika personalgrupper. Trots att man nu har möjlighet att ha med sig en eller flera portföljer i form av dokumentation ska projektledaren kunna klara sig med sin mobil.

Mitt examensarbete består av att skapa en användargränssnitt till Bygget i programmeringsuppsättning Java där projektledaren ska kunna använda sin mobil som verktyg och hjälpmedel i sitt dagliga arbete. Det mobila verktyget är sålunda att fungera som ett tikskaft där användaren kan se vilka delar av Bygget.

Målet med denna uppgift är att få ett godkännande från min uppdragsgivare senast i april 2014 och att få till ständig min examen som Datatjänst.

För att målet ska kunna nås krävs dessa huvudsakliga arbetsuppgifter:

- Identifiering av utrening av beställaren's behov.
- Skapande av användargränssnittet.
- Skriva rapport.
- Leveransgodkännande av beställaren.
- Presentation - opponering.
- Examen

Jag anser att arbetet i förhållande till tiden och arbetetslina stämmer överens på ett bra sätt eftersom jag valt att arbeta ensam och att mitt arbete stäcker sig över 2 terminer.

Min uppdragsgivare/ kontaktperson kommer att vara tillgänglig löpande under examensarbete:
Jonas Wångberg
Mobil: 0705355022
Bygget AB
Kyrkegatan 44
411 15 Göteborg www.bygget.com jonas.wangberg@bygget.com

Armska Larsson
Beskrivning av examensarbete DVG:G:25
Trenderna 1A
66330 Skellefteå
601826-0249
Mobil: 070/127775
armaska40@hotmail.com
2013-05-28
Gällande examensarbete av Annika Larsson

Vi har haft det stora nöjet att fästa Annika i hennes examensarbete, och vi lycker hon har
arbetat systematiskt och på ett metodiskt sätt. Vi är på det hela taget nöjda med resultatet,
och vi ger henne godkänning i det arbete som avser Bygglet AB.
Såligen vill vi önska Annika lycka till i framtiden.

Med vänliga hälsningar

Jonas Wånggren
VD
Gällande programkod

All programkod som Annika Larsson skrivit är konfidentiell och får inte lämnas ut till tredje par.

Med vänliga hälsningar

Jonde Wåggren
VD
Bilaga D - Förfrågan om tillgänglighet till Bygglet AB Databas

Göteborg 2014-04-09

Angivande förfrågan om tillgänglighet till vår databas

Vi har fått denna förfrågan från Annika Larson, och vi vill på detta sätt meddele att vi inte kan erbjuda denna database uppgifter utan att vara testande using grundligt text och godkänt materialet.

Med vänliga hälsningar

Jonas Wängren
VD