

RFID och streckkodsläsning med Androidtelefoner

AML ABBAS

AISHA UGLJANIN

BACHELOR'S THESIS

DEPARTMENT OF ELECTRICAL AND INFORMATION TECHNOLOGY |

FACULTY OF ENGINEERING | LTH | LUND UNIVERSITY



axiell

RFID och streckodsläsning med Androidtelefoner

**Kan Androidtelefoner användas till att läsa
RFID-taggar och streckkoder i en biblioteksmiljö?**



**LUNDS
UNIVERSITET**
Lunds Tekniska Högskola

LTH Ingenjörshögskolan vid Campus Helsingborg
Institutionen för Elektro- och Informationsteknik

av

Aml Abbas
Aisha Ugljanin

© Copyright Aml Abbas, Aisha Ugljanin

LTH Ingenjörshögskolan vid Campus Helsingborg
Lunds universitet
Box 882
251 08 Helsingborg

LTH School of Engineering
Lund University
Box 882
SE-251 08 Helsingborg
Sweden

Tryckt i Sverige
Lunds universitet
Lund 2021

Sammanfattning

Syftet med examensarbetet var att undersöka huruvida RFID-tekniken och streckodsläsning i Androidtelefoner gick att tillämpa i en biblioteksmiljö. Fem bibliotekarier intervjuades för att utreda vilka funktioner som hade varit av värde i det vardagliga arbetet.

Utvecklingen skedde i Android Studio med Java där verktyg som Gradle och J-Unit användes. CI/CD tillämpades i utvecklingsprocessen. En webbsida utvecklades för att testa prototypens integration mot Axiells webbaserade bibliotekssystem Axiell Quria®. Användartestning utfördes med tre bibliotekarier i en biblioteksmiljö för att utvärdera och verifiera de framtagna funktionerna.

Examensarbetet resulterade i en prototyp till en mobilapplikation för Android med funktioner som hanterar biblioteksmaterial på följande vis: återlämning, utlåning inkl streckodsläsning, programmering av RFID-taggar samt scanning med presentation. Integrationen mellan webbsidan och prototypen indikerar att prototypen till mobilapplikationen kan integreras med Axiell Quria®.

Slutsatsen är att prototypen till mobilapplikationen kan användas i en biblioteksmiljö för att utföra ovan nämnda uppgifter genom scanning av RFID-taggar på biblioteksmaterial. Prototypen till mobilapplikationen ska ses som ett underlag för vidareutveckling.

Nyckelord: RFID, NFC, streckkod, Android, java, Library Service Management.

Abstract

The purpose of this study was to examine the possibility of implementing RFID and barcode scanning in mobile Android phones for usage in a library setting. In order to investigate the functions that would be useful to librarians in their daily work, five interviews were conducted with said target group.

Development was made with Java in Android Studio with tools such as Gradle and J-Unit. CI/CD was applied in the development process. For evaluation and verification of the prototype, user tests were performed with three librarians in a library setting. In order to test the possibility of integrating the prototype to the web based Library Service Management system called Axiell Quria®, a website was developed containing some of the functionalities found in the system.

As a result, a prototype to a mobile Android application was developed with functions that handle library items in the following ways: check in, check out including barcode scanning, programming of RFID tags and presenting the item when scanning. The integration between the prototype and the website suggests that an integration of the prototype to Axiell Quria® is possible.

The conclusion is that the prototype to the application can be used in a library setting. However, further investigation is needed in order to explore its full potential.

Key words: RFID, NFC, barcode, Android, java, Library Service Management.

Förord

“Vi arbetade hårt och ibland flera timmar i sträck, men det var hela tiden ett sant nöje (...) det är mycket lättare att sträva mot fulländning när man aldrig har tråkigt” skriver nobelpristagaren Daniel Kahneman om sin kollega tillika vän Amos Tversky och om forskningen som de bedrev tillsammans [41]. Det är något som vi inspireras av. Vi vill därför först och främst tacka varandra för ett gott samarbete, för alla lärdomar vi dragit av varandras olika kunskaper och för det trevliga sällskapet under arbetet.

Vi vill rikta ett stort tack till Axiell Group för möjligheten att få utveckla ett nytt koncept. Ett varmt tack till vår handledare Daniel Bohman som bistått med rådgivning, kunskap och tålamod under hela processen. Tack till Claes Källqvist för att ha hjälpt oss i allt från intervjumetodik till UX-design. Vi vill också tacka Pascal Collberg för förtroendet.

Ett särskilt tack till våra handledare Christin Lindholm och Christian Nyberg för all vägledning inom det akademiska. Tack till alla bibliotekarier som har ställt upp i vår studie, ingen nämnd, ingen glömd.

Sist men inte minst vill vi tacka våra familjer och nära vänner för all stöd, kärlek och uppmuntran. Det har betytt mycket för oss och för det här arbetet.

Aml Abbas & Aisha Ugljanin

Innehållsförteckning

1. Inledning	9
1.1 Bakgrund	9
1.2 Syfte	10
1.3 Målformulering	11
1.4 Problemformulering	11
1.5 Motivering av examensarbetet	11
1.6 Avgränsningar	12
2. Teknisk Bakgrund	14
2.1 Continuous integration & continuous deployment	14
2.2 Gradle	15
2.3 J-Unit och enhetstester	17
2.4 RFID-tekniken	17
2.5 NFC	18
2.6 Webbsida	18
2.7 WebSocket	19
2.8 Intent	19
2.9 OCR-läsning	20
2.10 Foreground service	20
3. Metod	22
3.1 Arbetets faser	22
3.1.1 Arbetets upplägg och fördelning	22
3.1.2 Förstudier	24
3.1.3 Intervjuer	25
3.1.4 Analys av intervjuer	27
3.1.5 Utveckling	29
3.1.6 Användartestning	33
3.1.7 Utvärdering	36
3.1.8 Examensrapportskrivandet	37
3.1.9 Presentation	37
3.2 Källkritik	38
4. Resultat	39
4.1 Prototypen	40
4.1.1 Före användartest	41
4.1.2 Efter användartest	42
4.2 Quria (webbsidan)	43

4.2.1 Före användartest	44
4.2.1.1 Anslutning	45
4.2.1.2 Program	46
4.2.1.3 Check in	46
4.2.1.4 Check out	46
4.2.1.5 Show book	46
4.2.2 Efter användartest	47
4.3 Presentation av bok	49
4.3.1 Före användartest	50
4.3.2 Efter användartest	52
4.4 Programmering av NFC-tag	54
4.5 Återlämning (incheckning)	56
4.6 Utlåning (utcheckning)	57
4.6.1 Före användartest	58
4.6.2 Efter användartest	61
5. Diskussion och slutsats	65
5.1 Slutsats utifrån frågeställningar	65
5.2 Reflektion över etiska aspekter	68
5.3 Framtida utvecklingsmöjligheter	69
6. Terminologi	72
7. Källförteckning	73
8. Bilagor	79
Bilaga 1 - Testprotokoll	79
Bilaga 2 - Intervjuguide	81

1. Inledning

Examensarbetet utfördes i samarbete med Axiell Group AB. Axiell inriktar sig på IT-tjänster för bibliotek och institutioner. Syftet är att sprida kultur och kunskap världen över. De vill bidra till ett ökat engagemang genom att bland annat uppmuntra till läsning, bevara kulturarvet och förbättra förutsättningar för lärande. Kunderna består av bibliotek, museum, arkiv, skolor och förlag i 58 länder. Axiell tillhandahåller digitala verktyg som ämnar att förändra och förbättra den dagliga interaktionen med besökare. Huvudkontoret finns i Lund och har i övrigt 28 kontor i världen [1].

1.1. Bakgrund

Bibliotekens uppdrag är att göra kunskap och kultur lättillgänglig för allmänheten och det är av stor vikt att tekniken möjliggör detta. Det finns en önskan hos bibliotekspersonal att frigöra tid från tidskrävande hantering av material för att kunna arbeta med detta uppdrag genom att exempelvis anordna högläsning för barn. I dagsläget behöver bibliotekspersonal hämta böcker till receptionen för att scanna av dem för hämtning och uppdatering av information. Bibliotekspersonal måste bära eller köra en vagn med ett antal böcker fram och tillbaka mellan hyllorna och receptionen. Det är både tidskrävande och icke-ergonomiskt. Axiell har dock en digital tjänst Axiell Quria® som stödjer RFID med NFC. Fördelen med Axiell Quria® är att det ger möjlighet att utveckla en prototyp av en mobilapplikation som tillåter interaktion med olika hårdvarukomponenter på en mobiltelefon. Tanken är att det ska underlätta för biblioteksanställda i deras dagliga arbete genom att utveckla en prototyp av en mobilapplikation som kommunicerar med Axiell Quria®. I detta examensarbete kommer en sådan prototyp att utformas, utvecklas och utvärderas.

Axiell Quria® är en molnbaserad, digital tjänst av typen Library Services Platform (se kapitel 2) som är framtagen för folkbibliotek [1]. Syftet med tjänsten är att underlätta för bibliotekspersonal i sitt vardagliga arbete genom att minska på antalet system som används. Med hjälp av Axiell Quria® kan exempelvis bibliotekspersonal boka event och aktiviteter, såsom högläsning för barn, vilket sedan syns på bibliotekets hemsida. Dessutom hanterar Quria påminnelser och skickar räkningar för försenat eller ej återlämnat material. På så sätt slipper bibliotekspersonalen lära sig många system för olika ändamål eftersom Axiell Quria® tillåter en kombination av ovan nämnda funktioner. Det är önskvärt att utvidga Axiell Quria® så att en mobiltelefon kan läsa av en tagg på exempelvis en bok för att hämta information om den.

Axiell Quria® möjliggör läsning och skrivning till NFC-taggar. NFC baserar sig på RFID-tekniken. Eftersom Axiell Quria® är webbaserad och agerar framför allt som gränssnitt för bibliotekspersonal behövs en integration mellan Axiell Quria® och prototypen som hanterar läsningen av NFC-taggar. Detta kan uppnås med android intents. Ovan nämnda tekniker finns beskrivna i kapitel 2.

1.2. Syfte

Syftet med examensarbetet är att utveckla en prototyp till mobilapplikation där biblioteksanställda ska kunna skanna material med hjälp av en mobiltelefon. Vid skanning ska dem presenteras med relevant information om det skannade materialet.

1.3. Målformulering

Examensarbetet ska undersöka om en prototyp till mobilapplikation på en mobiltelefon kan scanna bibliotekets material på ca 2-3 cm avstånd och på så sätt hämta information. Examensarbetet kommer fokusera på skrivning och läsning till NFC-taggar av typen NfcV av standarden ISO 15693.

1.4. Problemformulering

Examensarbetet kommer att besvara följande frågeställningar:

1. Vilka arbetsuppgifter och på vilket sätt kan RFID med NFC på en mobiltelefon att underlätta för bibliotekarier i sitt vardagliga arbete?
2. Hur ska arbetsflödet för en mobilapplikation utformas för att underlätta bibliotekariernas arbete?
3. Vilka kriterier kan användas för utvärdering av prototypen?
4. Kan prototypen vidareutvecklas till en mobilapplikation som kan integreras med Axiell Quria?®

1.5. Motivering av examensarbetet

Det finns en stor samhällsnytta med att göra kunskap och kultur lättillgängligt, öppet och åtkomligt för allmänheten. Författarna för examensarbetet vill hjälpa personalen som arbetar aktivt med det. Förhoppningen är att detta examensarbete kommer att spara tid för bibliotekspersonal när det gäller hantering och inventering av material. På så sätt öppnar det för möjligheten för bibliotekspersonal att interagera med låntagare både i och utanför biblioteken. Från författarnas intervjuer med biblioteksanställda har det framkommit att de önskar exempelvis att kunna anordna fler evenemang där de åker ut med

bokbuss eller cyklar till offentliga ställen i samhället för att nå ut till fler personer och engagera till bokläsning. Med en teknik som exempelvis möjliggör enkel avlarmning av materialet så skapas både mer tid och fler förutsättningar för detta. Det kan även underlätta i det ergonomiska arbetet för bibliotekarier om de slipper bära på tungt material.

En annan förhoppning är att examensarbetet kan bidra med ökad kunskap kring bibliotekssystemen och hur de kan utvecklas.

Dessutom ger samarbetet med bibliotekarier möjlighet till vidareutveckling av författarnas bemötande, eliciteringsteknik och utformning av en produkt utifrån en kunds behov [7].

1.6. Avgränsningar

Det ursprungliga syftet var att ta fram en mobilapplikation med stöd för NFC-tekniken för att öppna för användning för samtliga operativsystem. Då Chrome enbart stödjer NDEF som är en enklare variant av RFID-tekniken så kommer systemet avgränsas till en mobilapplikation som stöds av android. Prototypen ska således utvecklas för Android.

RFID-chip är av standarden ISO 15693 och datamodellen som används är ISO 28560. Protokollet som kommer användas är NfcV. NfcV är en kommunikationsstandard som är en implementation för att kunna läsa/skriva ISO 15693-baserade chip. De är framtagna av samarbetsgruppen NFC Forum

vars uppdrag är att komma överens om hur mobiltelefoner bör kommunicera med RFID-chip. [8]

2. Teknisk Bakgrund

I det här kapitlet presenteras valda tekniker under varje delavsnitt.

2.1. Continuous integration & continuous deployment

Continuous integration (CI) och continuous deployment (CD) är två principer som sammanlänkas med ett agilt arbetssätt, ofta skrivet som CI/CD. Syftet med CI/CD är att effektivisera projektarbeten där fler än en person bidrar för att upptäcka eventuella fel i koden tidigt innan projektet expanderar. Det handlar också om att kod automatiskt kan distribueras (deploy) till produktionsmiljön om så önskas. Det innebär att nya ändringar i koden testas gentemot andra ändringar i en versionshanterad utvecklingsmiljö såsom t.ex. GitHub [10].

Med hjälp av CI/CD kan det komplexa och tidskrävande underhållet av utvecklingen av prototypen hanteras på ett enklare sätt. För examensarbetet har GitHub valts som utvecklingsmiljö eftersom det innehåller API:er som stödjer implementationen av CI/CD. GitHubs verktyg för CI/CD kallas GitHub Actions [13]. Med pull requests kan koden visas för resterande bidragande medlemmar i projektet vilka ändringar som har gjorts. En del organisationer har processer uppsatta där en annan utvecklare utvärderar koden och först när denne har gett sitt medgivande, slås förändringarna ihop med resten av koden. Ett alternativ är att den andre utvecklaren har synpunkter på förändringarna som författaren till de ursprungliga förändringarna behöver korrigera först, innan ihopslagningen sker [14]. Fördelen med CI/CD är att en pull requests kan avbrytas om ett fel i koden upptäcks [10].

En annan grund för CI/CD är automatiska tester. De automatiska testerna kan sättas upp en enda gång så att testerna inte behöver skrivas varje gång en ny kod publiceras. Vid skapandet av en ny pull request körs de automatiska testerna mot förändringarna i koden. Körningarna för automatiska tester specificeras vid exempelvis varje begäran om överlämnande av egen kod till huvudkoden (*push*) [15].

Med CI/CD säkerställs att utvecklingen kontrolleras automatiskt gentemot en standard under hela utvecklingsprocessen. Standarden specificeras av utvecklarna och testerna som ingår som medarbetare i det aktuella projektet (*contributors*). Ett exempel kan vara att tester alltid körs med J-Unit (se figur 2.1).

För examensarbetet har GitHubs lösning i form av GitHub Actions använts [13].

2.2. Gradle

Gradle är ett byggautomationsverktyg som kan användas för att kompilera kod, koppla ihop olika bibliotek och köra automatiska tester. Det installeras i instruktionerna till GitHub actions vilket ska underlätta kontroll och underhåll av ny källkod som läggs till i projektet [9].

I examensarbetet används Gradle för att implementera principen om CI/CD (Continuous integration & continuous deployment) genom automatiska tester. De automatiska testerna upprättas manuellt av författarna och anpassas så att koden som läggs till alltid håller samma standard. Förutom att testa källkod för syntaxfel och inkorrekt indentation så upprättas exempelvis logiska tester. De

logiska testerna kontrollerar att metoder och parametrar returnerar korrekta typer genom olika testfall som antingen går igenom eller misslyckas [10]. Testerna körs i JUnits ramverk för tester [11], men startas av Gradles byggprocess där ett av stegen är att kompilera Java-koden. Därefter körs de automatiska testerna mot den kompilerade koden. Ett av de automatiserade, logiska testerna som utförs är att skicka ett exempel på en CRC-kod (Cyclic Redundancy Check, en metod som kontrollerar att data inte har manipulerats under en sändning [12]) som lyckas avläsas eller inte, se kod nedan.

```
@Test
public void setCRCTest() {
    byte[] data = new byte[32];
    String barcode = "AA987654321876LL";
    byte[] afterSetBarcode = NfcTagUtil.setBarcode(barcode,
data);
    byte[] afterSetCRC =
NfcTagUtil.setCRC(Utilities.calculateCRC16(afterSetBarcode),
data);
    byte[] correct = new byte[32];
    for (int i = 0; i < 16; i++) {
        correct[i + 3] = (byte) barcode.charAt(i);
    }
    correct[19] = -5;
    correct[20] = -40;
    Assert.assertArrayEquals(afterSetCRC, correct);
}
```

Kodenhet 2.2: Kod som visar ett logiskt test.

```

dependencies {
    implementation 'androidx.appcompat:appcompat:1.2.0'
    implementation 'com.google.android.material:material:1.2.1'
    implementation 'androidx.constraintlayout:constraintlayout:2.0.4'
    testImplementation 'junit:junit:4.+'
    androidTestImplementation 'androidx.test.ext:junit:1.1.2'
    androidTestImplementation 'androidx.test.espresso:espresso-core:3.3.0'
    implementation 'org.java-websocket:Java-WebSocket:1.5.1'
}

```

Figur 2.2: Skärmdump över dependencies som visar hur Gradle är konfigurerat för examensarbetet, vilka verktyg som används, versioner etc.

2.3. J-Unit och enhetstester

J-Unit är ett API utformat för enhetstestning av Java. Med hjälp av J-Unit kan mindre delar av ett större program testas kontinuerligt under utveckling. Testerna utförs för att kontrollera att en klass eller metod gör det som den är avsedd för. På så sätt kan testerna även fungera som en dokumentation för programmet [11].

2.4. RFID-tekniken

RFID-tekniken (Radio Frequency Identification) läser av ett objekt med hjälp av radiovågor vilket kräver att tagg och läsare arbetar på samma frekvens, i det här fallet 13,56 MHz. I biblioteken används passiva RFID-taggar vilket innebär att taggarna saknar egen strömförsörjning och kommunicerar genom induktion. Således syftar de passiva RFID-taggar till att enbart lagra och presentera information. Handskakningen initieras när taggen väcks av en signal från en läsare. En implementation med RFID-tekniken kan göras med NFC-taggar som

beskrivet nedan [2].

2.5. NFC

NFC, som står för Near Field Communication, låter tagg och läsare kommunicera med varandra på ett visst avstånd. Idag på biblioteken är RFID fysisk hårdvara med antenn där kommunikation kan ske med 50 cm avstånd [2] och i vissa fall upp till 1 meter [6] beroende på hur stor taggen är. I mobila enheter där protokollet NfcV med standard ISO 15693 används är räckvidden vanligtvis 10 cm [3]. För material på biblioteket behövs ett avstånd på 2-3 cm för att mobiltelefonen ska kunna generera så mycket effekt som möjligt. Redan idag används tekniken i exempelvis matbutiker där personal kan scanna av matvaror med mobiltelefoner som stöder NFC och få information om varan. [4]

I examensarbetet används NFC-läsning av olika anledningar. Dels för att biblioteksmaterial innehåller NFC-taggar med information om materialet som kan vara relevant för bibliotekarier att ta del av som t.ex. bokens unika nummer (item ID), dels för att androidtelefoner är utrustade med NFC-läsare. Tanken är att det ska underlätta för bibliotekarier att använda känd och befintlig teknik.

2.6. Webbsida

För att kunna testa prototypens funktioner som t.ex. in- och utcheckning har en webbsida utvecklats med hjälp av JavaScript [16], HTML [17] och CSS [18]. Webbsidan agerar klient i kommunikationen via WebSocket.

I en vidareutveckling av prototypen och en eventuell integration med Axiell Quria® eller annat webbaserat bibliotekssystem ersätts webbsidan med Qurias®

gränssnitt.

2.7. WebSocket

En WebSocket upprättar en tvåvägskommunikation via en kanal mellan en server och klient, i det här fallet en webbläsare, över en TCP-koppling. Genom den öppna kanalen kan anrop skickas åt båda hållen samtidigt. Exempelvis kan servern uppdatera klienten vid behov så att klienten slipper använda polling, se kapitel 3. Men även klienten kan skicka anrop till servern [19].

I examensarbetet kommer WebSocket stå för kommunikationen mellan prototypen och webbsidan. WebSocket används för att skicka text i JSON-format [20]. Texterna som skickas med JSON innehåller värden beroende på mellan vilka enheter de skickas. Om de skickas mellan webbsidan till taggen så skickas värdena `todo` och `value`. Värdena beskriver vad användaren vill göra med taggen. Från prototypen till webbsidan skickas två andra värden, `done` och `value`. Med `done` skickas information om vad som gjordes (ex incheckning eller programmering av taggen) och `value` informerar om det lyckades eller inte.

2.8. Intent

Ett intent är en datastruktur som används för kommunikation mellan olika applikationer i ett program. Intents anropar framför allt aktiviteter som ska utföras. Här används intent för att anropa `startActivity()` i prototypen vilket sedan håller kommunikationen öppen mellan prototypen och NFC-läsaren. Ett intent kan vara implicit eller explicit och används för att starta en aktivitet i en annan komponent. Med implicita intent specificeras endast

vilken aktivitet som ska utföras. Explicita intent specificerar både aktivitet och dess tillvägagångssätt inom samma mobilapplikation. Om en användare vill aktivera platstjänster i sin enhet så skickas en begäran med hjälp av ett implicit intent till en lämplig mobilapplikation som visar platsen man befinner sig på. Med explicita intent kan en aktivitet startas som en reaktion på en annan aktivitet, exempelvis att en sida i samma mobilapplikation visas genom ett knapptryck. [5]

I examensarbetet används explicita intents för att kontrollera att kommunikationen sker mellan prototypen och NFC-läsaren i hårdvaran. Exempelvis sker läsning och skrivning till NFC-taggen med hjälp av intents och av- och påarmning av taggen. När en NFC-tag kommer i kontakt med mobiltelefonens NFC-läsare skickas ett intent från NFC-läsaren som väcker applikationen. Därefter skickas ett meddelande med hjälp av WebSocket till SocketServer och webbsidan (klienten).

2.9. OCR-läsning

För avläsning av streckkod som finns på låntagares bibliotekskort kommer mobiltelefonens kamera att användas. Det behövs för att låna ut böcker som tillhör en låntagare så att boken kan registreras i låntagarens namn. Med hjälp av biblioteket QuaggaJS kan metoder för avläsning implementeras. QuaggaJS är webbaserad och använder kameran för att analysera statiska bilder, som läser många typer av streckkoder, och avkoda dem från en live stream [21].

2.10. Foreground service

En foreground service är ett program som körs i bakgrunden men agerar som ett användargränssnitt som är synlig för användaren. Dess uppgift är att samla

information eller utföra operationer medan användaren interagerar med programmet. Ett exempel på en foreground service är en musikspelare som spelar musik i bakgrunden men låter användaren justera volym eller byta låt [22].

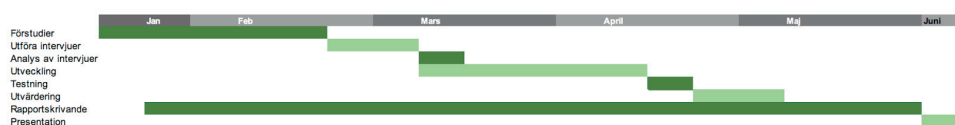
För examensarbetet är fördelen med att använda en foreground service att användaren hela tiden kan arbeta i Axiell Quria®.

3. Metod

I det här avsnittet presenteras tillvägagångssättet för arbetet med examensarbetet. Delavsnitten beskriver arbetets faser, hur resultatet i faserna har använts i andra faser, kommunikationen mellan examensarbetarna sinsemellan samt mellan examensarbetarna och företaget.

3.1. Arbetets faser

Arbetets faser delades upp enligt figur 3.1 nedan och innehöll: förstudier, utförda intervjuer, analys av resultat, utveckling, testning, utvärdering, rapportskrivande och presentation.



Figur 3.1: Gantt-schema som visar arbetets faser med start i mitten av januari och slut i början av juni.

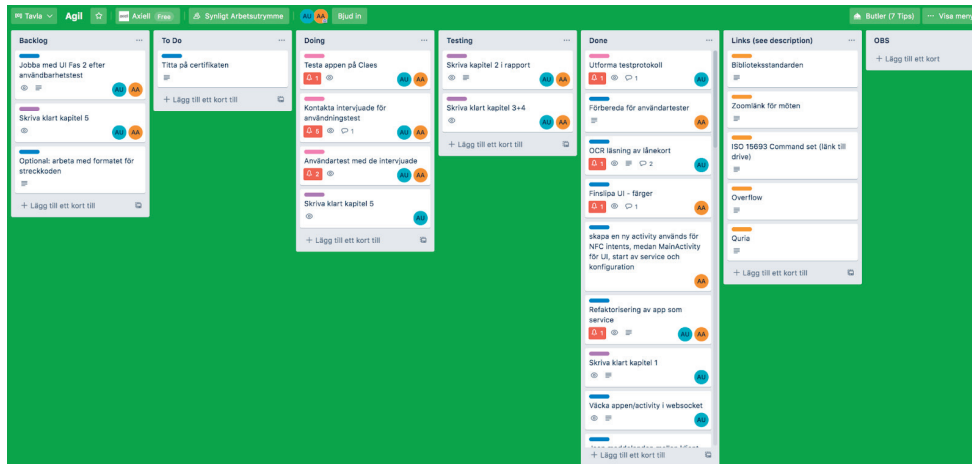
3.1.1. Arbetets upplägg och fördelning

Examensarbetarna valde att arbeta agilt med examensarbetet av flera anledningar. Först och främst, det agila arbetssättet främjar en regelbunden kommunikation och värnar om relationen mellan individerna som är involverade i projektet, dvs författarna och handledare, istället för att fokusera på processer och omfattande dokumentation [23].

En fördel med den agila arbetsmetoden är att arbetet kan delas upp i flera mindre mål. Det innebär att arbetet fortlöpte iterativt, dvs utvecklingen följde en process där en ny funktion producerades, testades och justerades tills den ansågs vara färdig.

Regelbundna möten fördes med handledaren två gånger i veckan. De fungerade också som en typ av deadline. Inför varje möte skulle författarna presentera vad som hade utförts hittills och föra diskussioner om vidare arbete i examensarbetet. Exempelvis kunde examensarbetarna bestämma arbetsuppgifter tillsammans med handledaren som ingick i en viss fas, t.ex. att utveckla OCR-läsning. Den här uppgiften skulle vara färdig till nästa möte. På så sätt agerade nästa möte deadline för uppgiften. Under mötet presenterades funktionen med OCR-läsning för handledaren. Om någonting inte fungerade hjälpte handledaren till med allt från att bidra med information till kodskrivande. Därefter, om examensarbetarna och handledaren var överens om att uppgiften var klar, bestämdes en ny uppgift som skulle utföras till nästa möte.

För arbetets organisering och struktur användes en Trello *board* med olika tavlor innehållandes uppgifter som var planerade (“Backlog” och “To do”), under utförande (“Doing”), under utvärdering (“Testing”) eller klara (“Done”), se figur 3.1.1.



Figur 3.1.1. En ögonblicksbild över Trello som ett agile board innehållandes tavlorna “Backlog”, “To do”, “Doing”, “Testing”, “Done”, “Links” och OBS.

En daglig kommunikation fördes med samtliga involverade i examensarbetet via kommunikations- och samarbetsverktyget Microsoft Teams [24]. I faserna för intervjuer och användartester involverades även en UX-designer. UX-designern hjälpte examensarbetarna att förbereda för intervjuerna, användartesterna och gav även återkoppling på prototypens utseende.

3.1.2. Förstudier

I förstudier ingick läsning av litteraturstudien “RFID på Malmö Högskola? En konsekvensanalys” utförd av Tornbjer, Ericson Lagerås och Nilén (2010) [2] samt utformning av en liten applikation som läser och skriver till NFC-taggar.

Litteraturstudien syftade till att öka förståelsen om RFID-teknikens implementering i biblioteken. Den handlar om hur RFID-tekniken tidigare har använts i bibliotekmiljö och hur arbetet för bibliotekarier har underlättats. Bland annat menar Tornbjer et al. [2] att RFID har bidragit till tidsbesparing, större kontroll över bibliotekets samlingar och ökad säkerhet. Tidsbesparingen kunde ses i form av en utvidgad självservice där låntagare själva kan låna och återlämna böcker vilket alltså sparar tid för bibliotekarier. Den nya kontrollen över samlingarna innebär att det blev lättare för bibliotekarier att upptäcka om material hade felplacerats på biblioteket. Även säkerheten ökade genom att det blev lättare att upptäcka stulna böcker. Studien bidrog med en nyförvärvad kunskap hos examensarbetarna om hur RFID-tekniken i biblioteken fungerar, vilken nytta tekniken har för bibliotekarier samt insikter i hur prototypen skulle kunna utvecklas. Med detta i åtanke kunde utvecklingen av prototypen till applikationen formas kring en idé om vad som skulle kunna vara till nytta för bibliotekarier.

Utvecklingen av en liten applikation som kunde läsa av och skriva till NFC-taggar ämnade dels att skapa en grund för en vidareutveckling till det som blev prototypen, dels att få en uppfattning om hur NFC-läsningen med en mobil enhet kan gå till. Genom att testa NFC-läsningen i ett tidigt skede kunde författarna skapa sig en bild av hur flödet skulle kunna se ut i framtiden.

3.1.3. Intervjuer

Utgångspunkten för examensarbetet är kvalitativ där intervjuer användes som undersökningsmetod. Fördelen med intervjuer som kvalitativ undersökningsmetod är att de ger kunskap om hur någonting uppfattas av den

riktade målgruppen [29], i det här fallet, hur RFID-tekniken uppfattas av bibliotekarier. Målet med intervjuerna var att få information om hur de arbetar med RFID idag, vilka hjälpmedel de använder som implementerar nämnd teknik och dess för- och nackdelar som upplevs, samt hur en applikation i en mobiltelefon hade hjälpt att underlätta deras arbete.

Informantintervjuer innebär att de intervjuade är verksamma i sitt område och har relevant kunskap [25]. I detta fall bestod informanterna av bibliotekarier. Inklusionskriterierna för deltagande var bibliotekarie med minst fem års erfarenhet från arbete i folkbibliotek. Sammanlagt fem intervjuer med bibliotekarier genomfördes. Samtliga intervjuer utfördes digitalt via kommunikationsverktyget Zoom [26]. Tre av de intervjuade var verksamma vid folkbibliotek, en var verksam vid ett universitetsbibliotek och en arbetade som produktägare i Axiell. Rekryteringen utfördes som ett strategiskt urval. Det innebär att individer med särskild erfarenhet och kunskap tillfrågas om deltagande i studie [42]. Mejl skickades ut till bibliotekarier verksamma vid olika folkbibliotek. Axiell bistod också med rekryteringen.

Semistrukturerade intervjuer utförs när en kontext behöver avgränsas [27]. Syftet med intervjuerna var att samla information om vilken funktionalitet som hade underlättat i bibliotekariernas vardag. Som en del av den semistrukturella metoden användes en intervjuguide för att säkerställa att informanterna besvarade samma frågor (bilaga 2). Intervjuguiden bestod av frågor med öppna svarsalternativ som lät de intervjuade uttrycka sina tankar fritt [25]. Följdfrågor ställdes för att få en djupare förståelse för vad den intervjuade hade för tankar om användning av RFID-tekniken [27].

Innan intervjun påbörjades fick samtliga deltagare information om att de när som helst kunde avbryta intervjun samt att deltagandet sker helt frivilligt. Dessutom spelades intervjuerna in med en ljudupptagare (mikrofon) för att kunna uppfatta tonfall och ordval [28]. De inspelade intervjuerna transkriberades efteråt.

Båda examensarbetarna närvarade vid intervjuerna. Ena examensarbetaren verkade som aktiv intervjuare medan den andra examensarbetaren hade en passiv roll. I den aktiva rollen ingick det att ställa frågor och föra diskussion med informanten för att bibehålla ett naturligt flyt i samtalet [27]. Den passiva intervjuaren agerade observatör och ställde följdfrågor. Rollerna skiftades åt vid olika intervjuer.

3.1.4. Analys av intervjuer

Analysen fokuserade på att besvara frågan: “Vad finns det att vinna på att använda RFID i en smartphone? Vilka funktioner hade underlättat mest i bibliotekariers arbete?”

Det transkriberade materialet bearbetades genom kvalitativ analys [27]. För att få en uppfattning om textens innehåll och helhet lästes materialet på nytt flera gånger. Meningsbärande enheter lyftes ur texten där de ansågs vara relevanta för att besvara ovanstående frågeställning. Exempel på meningsbärande enheter ses nedan:

Gällande gallring av böcker

“Behöver ofta bära böcker till RFID-platta från hyllan ... vid gallring går jag igenom alla böcker som ska rensas ut ... vi har en viss hylla som vi är ansvariga för ... tar hyllan steg för steg. Om man kan sköta allt vid hyllan så hade det gått

smidigt“

Informant 4

Vilka funktioner som önskas i en mobilapplikation

“Utlåning, återlämning, läsa av rfid chippet och komma till katalogen (vilken hylla boken finns i vilken status har boken).”

Informant 3

Scenarier där en mobilapplikation hade underlättat

“Om en låntagare frågar om en bok, hämta den så kan man låna ut den på gång, behöver inte trängas eller vänta i kö till automat. Tidseffektivt också ... de som inte kan språket eller är rädda för teknik behöver inte skickas till automat eller till kö”

Informant 1

“Kånkande av böcker mellan kontor och bibblan, det hade kunnat förenklas”

Informant 2

Ovan ses olika exempel och scenarier där bibliotekarier hade sett en vinst med att använda RFID-teknik i mobiltelefoner. Samtliga informanter uttryckte önskemål och beskrev scenarier på liknande sätt.

Resultatet av analysen visade att intervjuade upplevde en tidsvinst i att använda RFID i vardagen. RFID underlättar självservice där utlåning och återlämning hanteras av låntagarna själva. Detta leder till en tidsvinst för bibliotekarier. Tornbjer et al. [2] drar samma slutsats om tidsvinsten med att använda RFID i vardagen för bibliotekarier. Således var bibliotekarierna positivt inställda till RFID-tekniken i mobiltelefoner.

De intervjuade var eniga om vilka funktioner de hade velat se i en mobiltelefon, en sammanställning av önskade funktioner kan ses i figur 3.1.4 nedan. Intervjuerna visade att det fanns en nytta med att skapa en applikation. Dock fick de önskade funktionerna begränsas i antal under implementation pga tidsbrist.



Figur 3.1.4. En visuell sammanställning av de funktioner som önskats i en mobiltelefon av de intervjuade bibliotekarierna.

3.1.5. Utveckling

I utvecklingsfasen låg fokus på att vidareutveckla den lilla prototypen av NFC-läsaren som togs fram under förstudierna för att säkerställa att NFC-läsningen är möjlig att genomföra. Utvecklingsfasen inkluderade den mest omfattande delen av projektet.

Utvecklingen utgick från ett iterativt arbetssätt. I början av varje vecka bestämdes i samråd med handledaren vilken funktionalitet som examensarbetarna skulle utveckla. Flera funktioner kunde bestämmas under en

vecka och lades till i Trellons backlog (se figur 3.1.1). Uppgifter flyttades efterhand till “To Do” när examensarbetarna i samråd med handledare bestämde vilka uppgifter som skulle ligga i fokus för veckan. När en uppgift påbörjades flyttades den från “To do” till “Doing”. Under veckans gång producerades kod som sedan testades och reviderades. Vid testning flyttades uppgiften från “To do” till “Testing”. “Testing” innebar att funktionaliteten behövde utvärderas vilket kunde ske på olika sätt. Exempelvis testades att anslutningen mellan prototypen och Quria höll under en hel användning av både examensarbetarna, handledare, familjemedlemmar och kursare. I slutet av veckan diskuterades den nyutvecklade funktionen med handledaren, dvs positiva aspekter av funktionen, aspekter som behövde ses över eller utvecklas på nytt eller om funktionen ansågs vara helt färdig. Vid behov av hjälp bistod handledaren med sin kunskap och expertis. En helt färdig uppgift flyttades i Trello från “Testing” till “Done”.

Tester som utfördes med hjälp av GitHub Actions hjälpte att upptäcka logiska fel i koden. Alla fel som upptäcktes i en kod innebar en omskrivning innan den kunde slås samman med resterande kod. Exempel på ett logiskt test kan ses i avsnitt 2.1, kodenhet 2.1. Nya tillägg till kodbasen fick inte ändra strukturen på CRC eftersom CRC syftar till att kontrollera att data inte har manipulerats under en sändning.

De funktioner som har utvecklats är av- och påarmning (check out och check in), läsande av och skrivande till NFC-taggen, presentation av boken med bild på hemsidan samt läsande av låntagare med hjälp av streckkod. Även språköversättning (svenska och engelska) har utvecklats för ökad användbarhet.

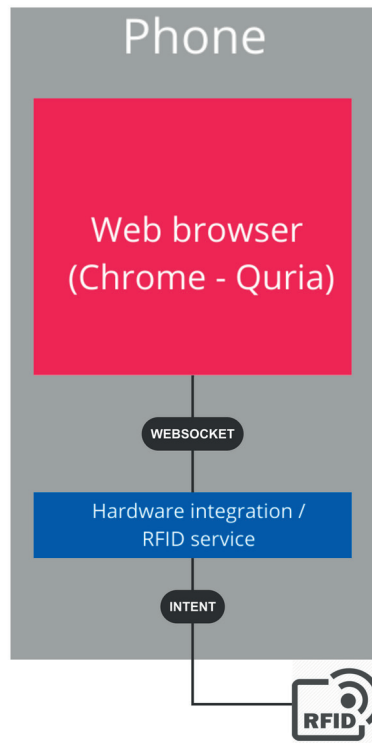
I början av utvecklingen skapades en hemsida med HTML och JavaScript för att skicka data till webbläsaren och testa prototypens funktioner. Hemsidan utvecklades som en prototyp till bibliotekssystemet Axiell Quria®. Anledningen till att mobilapplikationens prototyp under utvecklingen inte kommunicerade med Axiell Quria® direkt var för att inte riskera att påverka gränssnittet som används i verkligheten. På så sätt kunde examensarbetarna få fritt spelrum att utveckla och testa prototypen till applikationen.

Användningen av GitHub motiverades med att det kunde användas för versionshantering. Fördelen med versionshantering i projekt är att det underlättar att ta reda på vilka ändringar som har gjorts mellan olika versioner, låta flera utvecklare arbeta med projektet samtidigt och från olika datorer samt kunna återgå till äldre versioner om det skulle behövas [30]. Med funktioner som GitHub Actions och Pull requests kunde ett workflow arrangeras och utvärderas vilket bidrog till en ökad transparens för samtliga involverade i utvecklingen, dvs för examensarbetarna och handledaren. Användningen av Gradle med J-Unit blev ett självklart val eftersom dessa API:er stöds av Java och dessutom ingår i en grund *template* med Android Studio [31].

Eftersom prototypen till applikationen används till att kommunicera med NFC-taggar så behöver den här funktionen alltid finnas tillgänglig i interaktionen med webbsidan. Av den anledningen har foreground service-tekniken använts så att prototypen kan finnas latent i bakgrunden och vänta på att få kommunicera med NFC-taggar under tiden webbsidan används, se figur 3.1.5 för en översikt av kommunikationen mellan systemets olika komponenter.

Explicita intents är det enda sättet att upprätthålla kommunikationen mellan NFC-läsaren och prototypen per standard [5]. Med explicita intents riktas anropen till en specifik del av programmet som ska startas. I det här fallet startas klassen `NfcActivity` från `MainActivity`, där `NfcActivity` ansvarar för hanteringen kring taggen medan `MainActivity` hanterar anslutning till server samt lagrar taggens data. Då kan metoderna i `MainActivity` skicka explicita data som krävs för hanteringen av taggen med hjälp av den inbyggda metoden `putExtra()` [32]. Exempel på data kan vara programmering av item ID eller in- och utcheckning. Därför finns det ingen anledning att låta användare själv välja hur en aktivitet ska utföras genom implicita intents.

För anslutningen mellan prototypen och hemsidan används `WebSocket`. Fördelen med `WebSocket` över t.ex. polling är att den eliminerar problemen med latens. Med polling skickar klienten, i det här fallet webbsidan, konstant en begäran till en server för att kontrollera om någon data ska skickas. Det är resurskrävande och skapar en onödig väntetid när en riktig anslutning ska upprättas. `WebSocket` däremot kringgår problemen med latens genom att låta servern anropa klienten när ny data ska skickas och även låta klienten skicka data tillbaka till servern [33]



Figur 3.1.5: Översiktsbild på systemets uppbyggnad och kommunikationen mellan de olika komponenterna

3.1.6. Användartestning

Utvärdering av prototypen är nödvändig för att användarupplevelsen ska kunna vara optimal. Det utförs för att säkerställa ett smidigt arbetsflöde i användningen av applikationen. En applikation som anses användarvänlig är lätt att lära sig, enkel att navigera och skapar en trevlig användarupplevelse. Eftersom dessa kriterier baseras på subjektiva uppfattningar kan det vara svårt att bedöma användarvänligheten hos en applikation [34].

Applikationen vänder sig i första hand till bibliotekspersonal i alla åldrar, men kan upplevas olika beroende på bland annat användarnas erfarenhet av teknik, deras förväntningar på vad applikationen ska leverera och personliga preferenser i användbarhet. Även humöret hos användarna under själva testningen kan påverka resultatet. Exempelvis om användarna känner sig stressade, distraherade eller påverkade av andra testares åsikter. Med detta i åtanke anses det fortfarande viktigt att utföra användartester [34].

En applikation som anses vara lätt att lära sig innebär exempelvis att funktionerna syns tydligt genom knappar och kontroller, att terminologin är rätt så att användaren kan upptäcka mönster under navigationen och skapa sig en bild av hur systemet är uppbyggt samt att få tydlig feedback när någonting utförs [34].

Ovanstående faktorer togs hänsyn till vid utformningen av ett testprotokoll, se bilaga 1. Testprotokollet syftade att säkerställa att användartestningen följde samma mönster med samtliga testare.

Användartestningen ska helst utföras i flera omgångar för en grundlig kvalitetskontroll. Vanligtvis utförs 2-3 omgångar med användartester under ett projekt [35], här utfördes 2 omgångar med användartester, en med familjemedlemmar och vänner och en med målgruppen dvs bibliotekarier. Även en demonstration hölls med en av Axiells bibliotek som använder Axiell Quria®. Användartesterna ämnade att ge återkoppling på flödet i användningen i applikationen, huruvida applikationen uppfyller sitt syfte och är enkel att

förstå.

Innan användartesterna med bibliotekarier utfördes testades prototypen på familjemedlemmar och kursare för att verifiera flödet i hanteringen av NFC-taggen. Sammanlagt utfördes sju användartester i två omgångar. Testarna bestod av personer med varierande teknisk bakgrund och kunskaper. En av testarna var en kursare som var minst lika tekniskt kunnig som examensarbetarna. På så sätt kunde examensarbetarna ta del av återkoppling relaterade till användarflöde, UX-design, latens etc som kan påverka flödet i användningen.

Testprotokoll i bilaga 1 användes för samtliga användartester. Scenarier ur testerna lästes upp för användarna och anteckningar fördes under tiden uppgifterna utfördes. Användarna fick bland annat tala högt under testerna för att examensarbetarna kunde ta del av hur uppgiften, funktionen och flödet uppfattades. Efter användartesterna utfördes en sammanställning i testprotokollet och åtgärder vidtogs som beskrivet nedan.

Exempelvis behövde författarna bland annat kontrollera att en användare förstod hur programmering av en NFC-tag med ett nytt unikt ID (*item ID*) gick till med hjälp av namnet på knappen, instruktioner under förfarandet och bekräftelse på att programmeringen lyckades. Med hjälp av de tidiga användartesterna kunde författarna hitta en del oklarheter såsom att det inte var lätt att förstå hur nära mobiltelefonen skulle läggas intill NFC-taggen. Detta rättades genom att införa en animation och text i prototypen som visar hur NFC-läsningen skall gå till. Liknande oklarheter rättades innan användartester utfördes med målgruppen, i det här fallet bibliotekarier.

Användartestningen utfördes därefter på tre bibliotekarier i ett bibliotek för att göra det så likt en realistisk arbetsmiljö som möjligt. NFC-taggar klistrades fast på böcker och bibliotekarierna fick låna Androidtelefoner där prototypen fanns installerad. Under testningen uppmanades bibliotekarierna att genomföra de uppgifter som finns beskrivna i testprotokollet i bilaga X. De skulle också dela med av sina tankar och funderingar under utförande av en viss uppgift. Allting observerades och antecknades av författarna. Efter testningen fördes en diskussion med bibliotekarierna om ytterligare tankar och idéer för hur användningen kunde förbättras. Materialet användes sedan som ett underlag för åtgärder [36], såsom ändring av knappar, text och terminologi, se kap 4 för fler detaljer.

Till slut utfördes en demonstration med en av Axiells kunder som använder Axiell Quria®. Dels för att ytterligare verifiera syfte, funktioner och användning, dels för att kontrollera att flödet i användbarheten är synkroniserad med användningen i Axiell Quria®. Demonstrationen utfördes på distans via Zoom och kamera där en mobiltelefon var uppkopplad via skärmdelning. På så sätt kunde kunden ta del av prototypen i realtid. En av examensarbetarna ansvarade för handhavande av prototypen under demonstrationen. Kunden uppmanades att testa de olika funktionerna (scan item, utlån, återlämning och programmering av tagg) genom att peka var i prototypen examensarbetaren skulle trycka. Avslutad demonstration ledde till att prototypen ansågs uppfylla sitt tänkta syfte (se avsnitt 1.2), att ovan nämnda funktioner samt flödet i användningen fungerade väl.

3.1.7. Utvärdering

Utvärderingen innebar ändringar efter testningen. Om minst en uppgift i testprotokollet under minst ett användartest inte lyckades genomföras av bibliotekarierna skulle funktionen ses över och refaktoriseras. Om exempelvis en bibliotekarie inte lyckades läsa av en NFC-tagga pga bristande instruktioner ändrades dessa i prototypen till att innefatta en tydligare text om hur läsningen skulle gå till. När hela utvärderingen var färdig visades prototypen under en demonstration med en bibliotekarie som använder Axiell Quria®, se avsnitt 3.1.6.

Under varje omgång av användbarhetstester åtgärdades allt från större buggar till mindre estetiska fel, se kap 4 för prototypens funktion och utseende före och efter användartester. Åtgärderna syftade att förbättra flödet i användningen av prototypen.

3.1.8. Examensrapportskrivandet

Rapportskrivandet löpte parallellt med utvecklingen som en dokumentation av processen.

3.1.9. Presentation

Som en del av examensarbetet ingår en presentation och opponering. På grund av restriktionerna med covid-19 utfördes presentationen digitalt i realtid via en uppkopplad skärm. På samma sätt deltog även handledare, examinator, opponenter och övrig publik.

3.2. Källkritik

Examensarbetet ska vila på vetenskaplig grund där samtliga påståenden och yttranden i examensarbetet ska stöttas av legitima källor. Således är det av största vikt att samtliga källor utvärderas för sin tillförlitlighet.

Källförteckningen i examensarbetet består av tryckt litteratur, vetenskapliga artiklar samt dokumentationer.

Tryckt litteratur och vetenskapliga artiklar anses tillförlitliga, dels eftersom författarna till litteraturen anses vara experter inom sina respektive områden, dels för att innehållet i texterna granskas före publicering. Det gäller för källorna [10], [25], [27], [28], [29], [34], [35] samt [40]. Även källorna [11], [30] samt [42] anses uppfylla samma kriterier som ovan då innehållet författats av professorer på Uppsala Universitet respektive Karlstads Universitet. Det gäller även för [2] som är ett examensarbete.

Källor som består av dokumentationer och teknisk beskrivning anses vara pålitliga eftersom företag inte får bedriva falsk marknadsföring. Detta gäller för källorna [1], [3], [4], [5], [6], [8], [9], [13], [14], [15], [16], [17], [18], [20], [21], [22], [24], [26], [31], [32], [36], [37], [38], [39].

Källorna [7], [12], [19], [23] samt [33] är tekniska bloggar som författats av yrkesprofessionella och anses således trovärdiga. De bidrar med kunskap och praktiska tillämpningar av tekniken som använts i examensarbetet.

4. Resultat

Resultatet är en prototyp av en applikation som kan användas för att läsa NFC-taggar. Resultatet redovisas med skärmdumpar och text i de olika avsnitten av det här kapitlet. I avsnittet visas och förklaras även ändringar av prototypen efter användartester, samt även en förklaring till varför ändringarna gjordes.

Resultatet från intervjuerna med bibliotekarier ledde till utveckling av en prototyp av en applikation med följande funktioner:

1. Presentation av bok
2. Programmering av tagg
3. Återlämning (incheckning)
4. Utlåning (utcheckning)

En enkel webbsida utvecklades för att testa ovanstående funktioner som beskrivet i kapitel 2, avsnitt 2.6. Webbsidan har i examensarbetet döpts till Quria och ska ej förväxlas med bibliotekssystemet Axiell Quria®.

Användningen mellan Quria och prototypen till applikationen går till enligt följande:

1. Prototypen startas och ser till att anslutningen är igång (se figur 4.1.2)
2. Användaren lämnar prototypen och återgår till *Home*, dvs startsidan för Androidtelefonen
3. Från *Home* öppnas Quria (se figur 4.2.2).

Steg 1-3 ovan finns illustrerade i samtliga figurer för användarflöden (exempelvis figur 4.3.1).

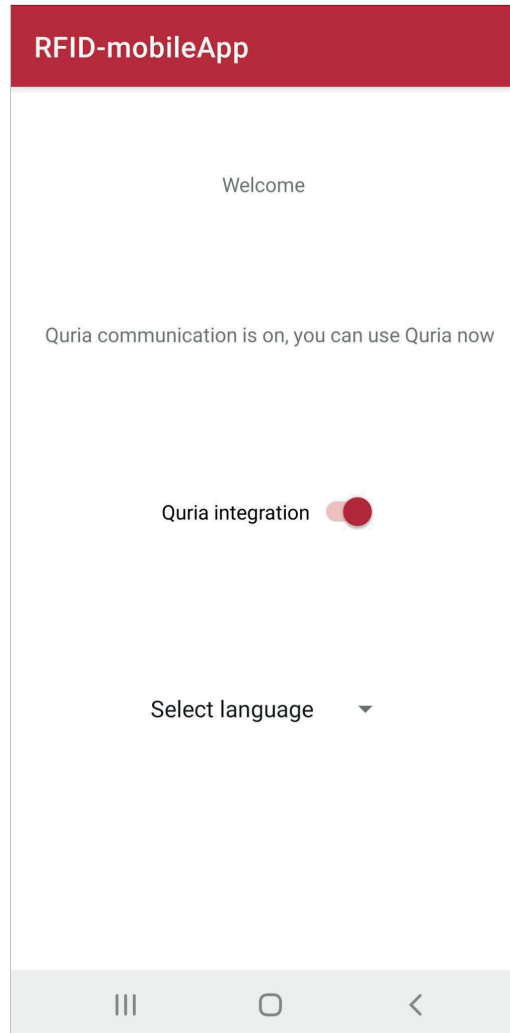
Varje delavsnitt av det här kapitlet beskriver funktionerna som har utvecklats. Funktionerna för presentation av bok samt utlåning (avsnitt 4.3 respektive 4.6) har genomgått omfattande förändringar efter användartesterna och kommer således att presenteras med illustrationer både före och efter användartesterna. Resterande funktioner i prototyp som inte ändrades efter testerna kommer enbart presenteras i en bild utan jämförelse.

Under figurerna 4.3, 4.4, 4.5, 4.6 finns beskrivning av stegen som är numrerade enligt deras ordning under flödet (markerat med lila färg i figuren). För illustrationerna som visar flöden efter användartest finns även en beskrivning över de ändringar som har gjorts (markerade med rosa färg i figuren) i alfabetisk ordning.

4.1. Prototypen

Här kommer skärmdumpar av applikationens prototyp att visas. Utseendet och UX-designen för prototypen kommer att jämföras med före och efter användartester.

4.1.1. Före användartest

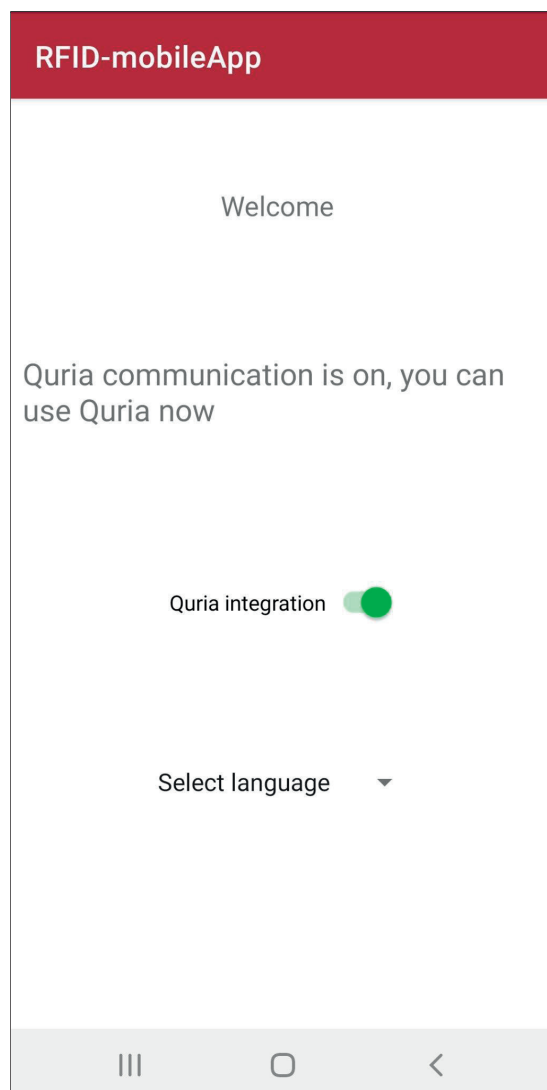


Figur 4.1.1: Prototypens utseende före användartestet.

Som illustrerat i figur 4.1.1 syns prototypens startsida med namnet “RFID-mobileApp”, en välkomsttext och status för WebSocket-anslutningen mellan prototypen och Quria. Vid start av prototypen upprättas en anslutning

automatiskt. Anslutningen kan stängas av och sättas på med switchen “Quria integration”. Det går även att byta språk i prototypen (svenska och engelska).

4.1.2. Efter användartest



Figur 4.1.2: Prototypens utseende efter användartestet.

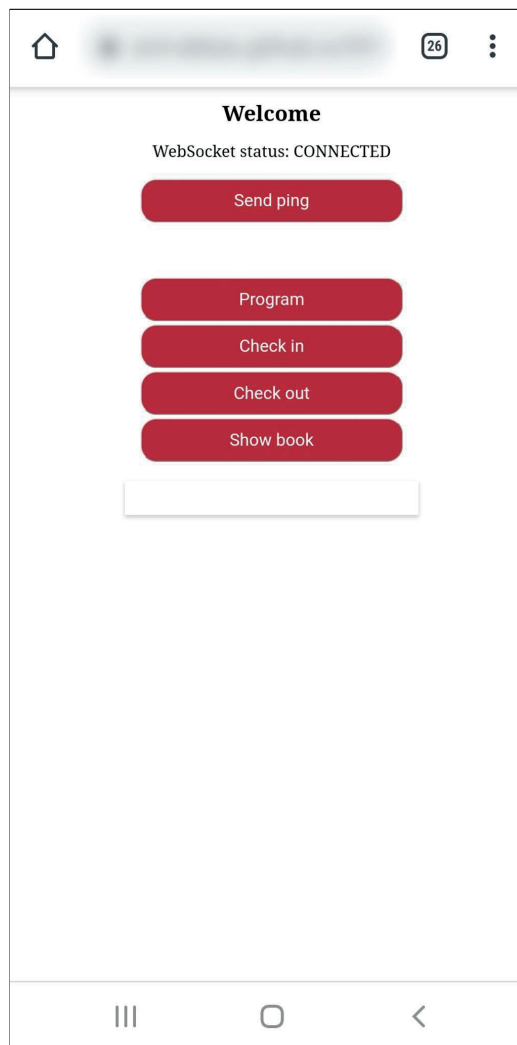
Efter användartestet utfördes en ändring i prototypens utseende. Switchen “Quria integration” ändrades från röd till grön färg (figur 4.1.2) för att visa en aktiv WebSocket-anslutning mellan prototypen och Quria. Vid avstängning ändras switchens färg från grön till grå.

Denna ändring medförde att användarna lättare kunde se att anslutningen var igång eftersom användarna är vana vid grön färg på en switch som indikation på en aktiv anslutning. Innan ändringen utfördes upplevde användarna att det var svårt att veta att anslutningen var på eftersom switchen var röd.

4.2. Quria (webbsidan)

Här kommer skärmdumpar av Qurias prototyp att visas. Dessutom en jämförelse av utseende och UX-design före och efter användartest. Quria är en prototyp av den nuvarande webbsidan som företaget använder nu, den innehåller utvalda funktioner som testades i examensarbetet.

4.2.1. Före användartest



Figur 4.2.1: Qurias utseende före användartest.

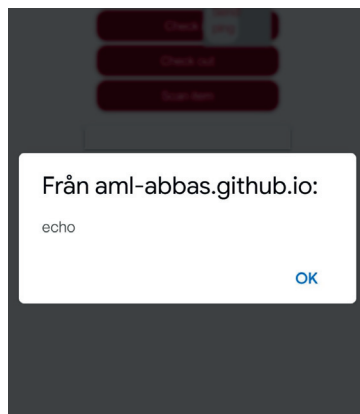
Figur 4.2.1 visar webbsidan Quria före användartestet. Här finns samtliga funktioner markerade med röd knapp.

4.2.1.1. Anslutning

Överst står texten “Welcome - WebSocket status: CONNECTED” med röd knapp med texten “Send ping”.

“WebSocket status: CONNECTED” innebär att det finns en upprättad anslutning mellan prototypen och Quria. Om anslutningen är bruten visas texten “WebSocket status: DISCONNECTED”.

“Send ping” syftar att skicka ett ping till servern för att kontrollera att servern svarar på handskakningar. Vid en lyckad sändning av ett ping fås svaret “echo” tillbaka [37], här visas “echo” i en pop-up-ruta, se figur 4.2.1a.



Figur 4.2.1a. Pop-up-fönster som visar en lyckad ping-sändning till server med “echo” som svar.

4.2.1.2. Program

Med funktionen “Program” kan användaren, exempelvis en bibliotekarie, skriva ett *item ID* till en tagg. Det utförs om taggen är tom från början och ett biblioteksmaterial ska chippas för första gången. Det kan också göras om ett biblioteksmaterial ska gallras ur bibliotekets samling, då makuleras *item ID* med hjälp av programmeringsfunktionen.

4.2.1.3. Check in

Funktionen “Check in” avser att ändra status hos en tagg när ett biblioteksmaterial återlämnas till bibliotekets samling efter utlån. Då ändras status från “checked out” till “checked in”.

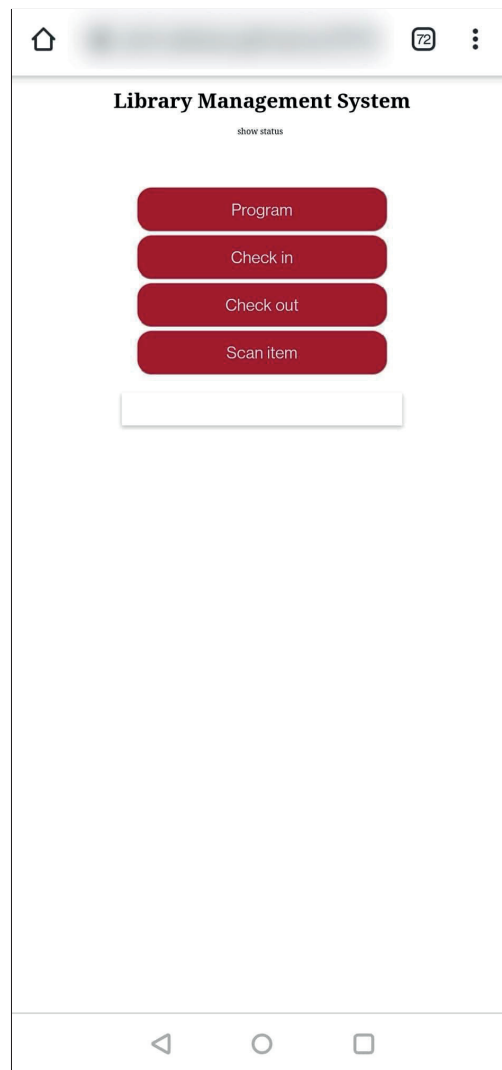
4.2.1.4. Check out

Funktionen “Check out” avser att ändra status hos en tagg när ett biblioteksmaterial ska lånas ut till en låntagare och ska således lämna bibliotekets samling under en viss tid. Då ändras status från “checked in” till “checked out”.

4.2.1.5. Show book

Funktionen avser att presentera användaren med biblioteksmaterialets omslag och *item ID*. Observera att funktionen enbart är en prototyp för en liknande funktion i Axiell Quria® och innehåller begränsat med information. Exempelvis innehåller Axiell Quria® information om biblioteksmaterialets placering i biblioteket, vilket Quria inte gör.

4.2.2. Efter användartest



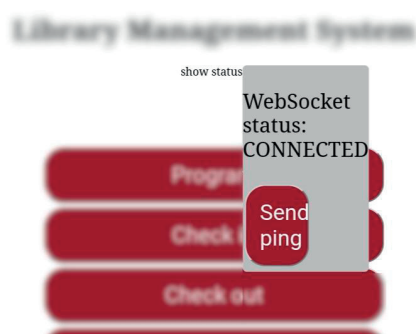
Figur 4.2.2: Qurias utseende efter användartest.

Figur 4.2.2 visar Qurias utseende efter användartestet. Samtliga ändringar gjordes för att undvika att skapa förvirring och för att varje knapp ska ha en

tydlig koppling till domänet för bibliotekarier. Namnet på knappen “Show book” ändrades till “Scan item” för att det stämde bättre överens med bibliotekariers terminologi för funktionen, se avsnitt 4.2.1.5 samt avsnitt 4.3.

Texten på översta raden ändrades till “Library Management System - show status” från ursprungliga texten i figur 4.2.1. Den nya texten “Library Management System” visar tydligt att webbsidan är ett bibliotekssystem istället för “Welcome!” (se figur 4.2.1) som var otydligt.

Den lilla texten “show status” (se figur 4.3.2 samt figur 4.3.2a) visar med en tooltip [38] om anslutningen är på (se figur 4.2.2a) med texten “CONNECTED” eller “DISCONNECTED”. Den röda knappen med texten “Send ping” (figur 4.2.1) togs bort för att inte skapa förvirring då den liknar övriga knappar med funktioner som har tydliga kopplingar till bibliotekariers arbete. Knappen ersattes istället med en liten och diskret text.



Figur 4.2.2a: Tooltip som visar WebSocket-anslutningens status, här: connected, dvs ansluten.

4.3. Presentation av bok

Funktionen avser att presentera information om det inscannade biblioteksmaterialet som kan vara relevant för bibliotekarier att ta del av. I prototypen till Quria begränsades mängden information till att enbart presentera biblioteksmaterialets omslag och *item ID*. I Axiell Quria® presenteras exempelvis även information om materialets placering i biblioteket, när det senast lånades ut etc. Eftersom Quria är tänkt att ersättas av Axiell Quria® utvidgades inte den här funktionen ytterligare.

Nedan presenteras illustrationer över flöden för funktionaliteten före och efter användartest.

4.3.1. Före användartest



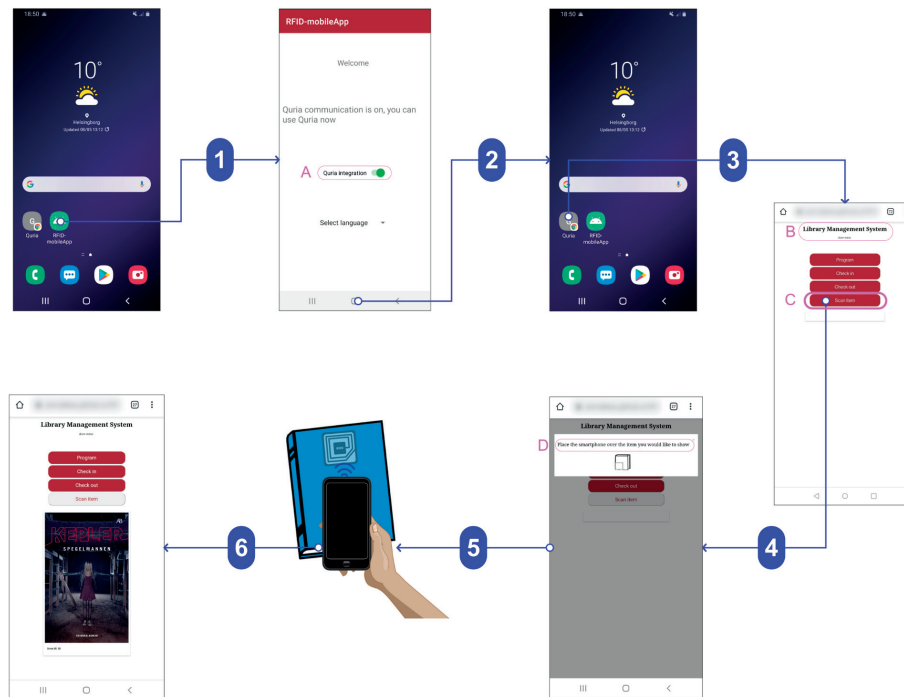
Figur 4.3.1: Illustration över flödet för presentation av ett biblioteksmaterial (här: bok), före användartestet med bibliotekarier.

Flödesbeskrivning för figur 4.3.1:

1. Från hemskärmen nås prototypen (här kallad “RFID-mobileApp”).
2. Från prototypen nås hemskärmen.

3. Från hemskärmen nås Quria (webbsidan).
4. Vid knapptryck av "Show book" visas ett pop-up-fönster med texten:
"Place the smartphone over the item you would like to scan".
5. Användaren placerar Androidtelefonen så att ytan nuddar taggen.
6. Vid lyckad avläsning av taggen presenteras materialet (här: bok) med omslag och "item ID" (här: 36) under omslaget. Knappen "Show book" är aktiv vilket ses i form av en ljusgrå bakgrund till skillnad från resterande knappar som är icke-aktiva och har således röd bakgrund.

4.3.2. Efter användartest



Figur 4.3.2: Illustration över flödet för presentation av ett biblioteksmaterial (här: bok) efter användartestet med bibliotekarier.

Flödesbeskrivning för figur 4.3.2:

1. Från hemskärmen nås prototypen (här kallad “RFID-mobileApp”).
2. Från prototypen nås hemskärmen.
3. Från hemskärmen nås Quria (webbsidan).

4. Vid knapptryck av "Scan item" nås ett pop-up-fönster med texten: "Place the smartphone over the item you would like to show".
5. Användaren placerar Androidtelefonen så att ytan nuddar taggen.
6. Vid lyckad läsning av taggen presenteras materialet (här: bok) med omslag och "item ID" (här: 36) under omslaget. Knappen "Scan item" är aktiv vilket ses i form av en ljusgrå bakgrund till skillnad från resterande knappar som är icke-aktiva och har således röd bakgrund.

Beskrivning av ändringar i figur 4.3.2, rosa-markerade med ringar och i alfabetisk ordning, se avsnitt 4.2.2.

- A. Ändring av färg på switch, se avsnitt 4.1.2 för jämförelse.
- B. Ändring av text till "Library Management System - show status", se avsnitt 4.2.2.
- C. Texten på knappen ändrades till "Scan item" från "Show book" (figur 4.3.1) för att terminologin ska stämma för bibliotekarier vilket inte var fallet från början.
- D. Texten ändrades till "Place the smartphone over the item you would like to show" för att tydliggöra att omslaget för materialet kommer att visas i nästa steg.

4.4. Programmering av NFC-tag

Här illustreras tillvägagångssättet för programmering av en NFC-tag, se avsnitt 4.2.1.2 för en utförlig beskrivning av funktionen.

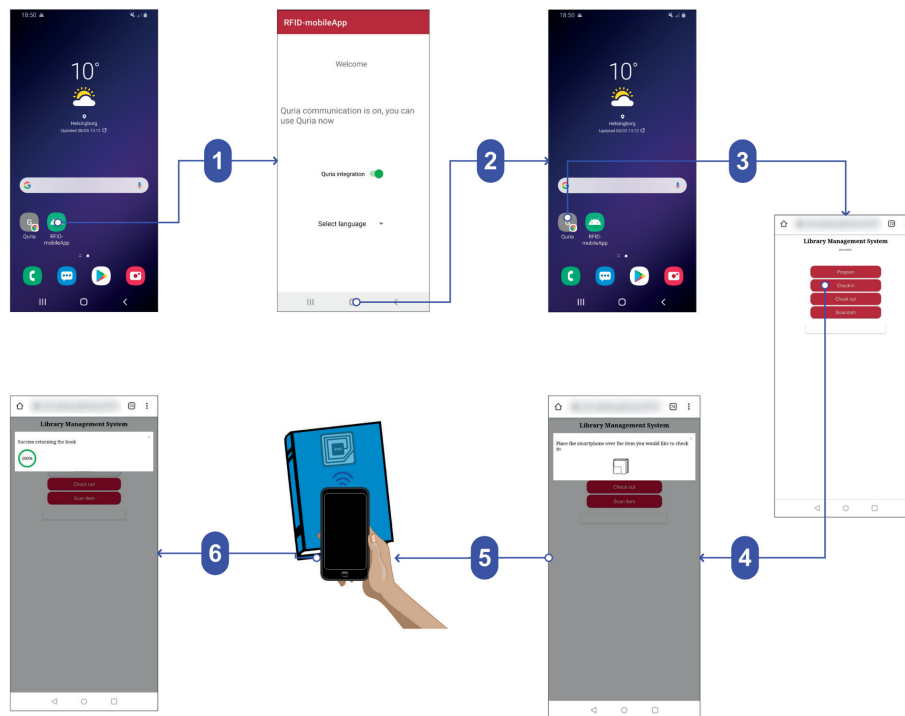


Figur 4.4: Illustration över flödet för programmering av en NFC-tag.

Flödesbeskrivning för figur 4.4:

1. Från hemskärmen nås prototypen (här kallad "RFID-mobileApp").
2. Från prototypen nås hemskärmen.
3. Från hemskärmen nås Quria (webbsidan).
4. Vid knapptryck av "Program" nås ett pop-up-fönster med en textruta med instruktionen: "Write a new item id". Användaren fyller i ett item ID (exempelvis nr 18).
5. Vid knapptryck av "Program the chip" visas ett pop-up-fönster med: "Place the smartphone over the item you would like to program".
6. Användaren placerar Androidtelefonen så att ytan nuddar taggen.
7. Vid lyckad läsning visas ett pop-up-fönster med texten: "Success to write to the tag" tillsammans med en grön progress indicator.

4.5. Återlämning (incheckning)



Figur 4.5: Illustration över flödet för återlämning (incheckning) av ett biblioteksmaterial.

Flödesbeskrivning för figur 4.5:

1. Från hemskärmen nås prototypen (här kallad “RFID-mobileApp”).
2. Från prototypen nås hemskärmen.
3. Från hemskärmen nås Quria (webbsidan).

4. Vid knapptryck av “Check in” visas ett pop-up-fönster med texten: “Place the smartphone over the item you would like to check in”.
5. Användaren placerar Androidtelefonen så att ytan nuddar taggen.
6. Vid lyckad läsning visas ett pop-up-fönster med texten: “Success returning the book” tillsammans med en grön progress indicator.

4.6. Utlåning (utcheckning)

Avsnittet visar scenariot för utlåning före och efter användartestningen. En av de huvudsakliga funktionerna som har utvecklats, utöver NFC-läsning, är streckodsläsning. Streckodsläsningen utförs med kamera. Funktionen syftar att låta ett utlån ske med hjälp av en mobiltelefon där en specifik låntagare kan få exempelvis en bok registrerad på sitt namn.

Ändringarna innebar att vi implementerade en stepper som hjälper användaren att förstå stegen i uppgiften, dvs att i första steget läses en *patron* (låntagare) och i andra steget läses taggen, se avsnitt 4.6.2. Dessutom hjälper det att inte tillåta användaren att checka ut en tagg utan att läsa in *patron* innan dess, dvs första steget måste utföras innan användaren kan gå vidare till andra steget.

4.6.1. Före användartest



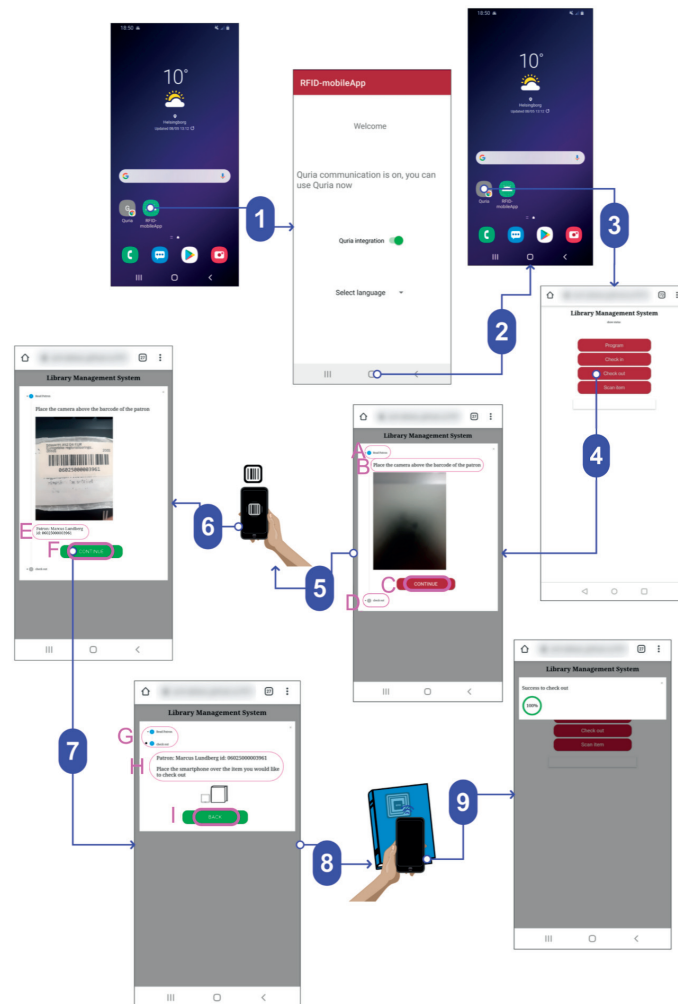
Figur 4.6.1: Illustration över flödet för utlåning av biblioteksmaterial (här: bok) inklusive scanning av streckkod och tagg, före användartest.

Flödesbeskrivning för figur 4.6.1:

1. Från hemskärmen nås prototypen (här kallad "RFID-mobileApp").
2. Från prototypen nås hemskärmen.
3. Från hemskärmen nås Quria (webbsidan).
4. Vid knapptryck av "Check out" nås ett pop-up-fönster med en röd knapp med texten "Read patron". Undertill visas texten: "Place the smartphone over the item you would like to scan".
5. Vid knapptryck av "Read patron" visas ett pop-up-fönster med ett kamerafönster. Under kamerafönstret visas en knapp med texten "Use patron".
6. Användaren förbereder för scanning av streckkod. Streckkoden representerar streckkoden som finns på ett lånekort hörande till en låntagare.
7. Läsning av streckkoden sker med kamera. Under kamerafönstret visas en syntolkning av streckkoden. Streckkodstolkningen under kamerafönstret ska överensstämma med den riktiga streckkoden.
8. Vid knapptryck av "Use patron" visas ett pop-up-fönster med text som överst visar streckkodstolkningen, en knapp med texten "Read patron" och texten: "Place the smartphone over the item you would like to scan".

9. Användaren placerar Androidtelefonen så att ytan nuddar taggen.
10. Vid lyckad läsning visas ett pop-up-fönster med texten: “Success to check out” tillsammans med en grön progress indicator.

4.6.2. Efter användartest



Figur 4.6.2: Illustration över flödet för utlåning av biblioteksmaterial (här: bok) inklusive scanning av streckkod och tagg, efter användartest.

Flödesbeskrivning för figur 4.6.2:

1. Från hemskärmen nås prototypen (här kallad “RFID-mobileApp”).
2. Från prototypen nås hemskärmen.
3. Från hemskärmen nås Quria (webbsidan).
4. Vid knapptryck av “Check out” visas ett pop-up-fönster som visar att streckkodsläsning sker i två steg: 1. Read patron, 2. Check out.

Steg 1. Read patron, en grå cirkel med en “1” inuti:

Ett kamerafönster visas i mitten. Ovanför kamerafönstret visas texten “Place the camera above the barcode of the patron”. Under kamerafönstret visas en knapp med texten “Continue”. När steget är färdigt ändrar cirkeln färgen från grå till blå.

5. Användaren förbereder för scanning av streckkod med kameran.
6. Steg 2. Check out, en grå cirkel med en “2” inuti:

Vid scanning av streckkod visas texten “Patron: “ + låntagares namn och “id: “ + streckkodstolkning (figur 4.6.2 visar “Patron: Marcus Lundberg” och en streckkodstolkning). Under kamerafönstret visas en grön knapp med texten “Continue”. När steget är färdigt ändrar cirkeln färgen från grå till blå.

7. Här visas både steg 1 och 2 som ifyllda blå rutor som indikerar att båda stegen är klara.

Vid knapptryck av "Continue" visas texten "Patron: " + låntagares namn "id: " + streckkodstolkningen. Undertill visas texten "Place the smartphone over the item you would like to check out".

8. Användaren placerar Androidtelefonen så att ytan nuddar taggen.
9. Vid lyckad läsning visas ett pop-up-fönster med texten: "Success to check out" tillsammans med en grön progress indicator.

Beskrivning av ändringar i figur 4.6.2, rosa-markerade med ringar och i alfabetisk ordning, som tillkom efter användartestning, se inledning till 4.6.

- A. Ändring till stegvis utlåning. Här: steg 1. "Read patron" som visar att användaren först måste läsa streckkoden med kameran.
- B. Tillägg av texten "Place the camera above the barcode of the patron" som förtydligar tillvägagångssättet för streckkodsläsning. Undertill visas även ett bredare kamerafönster jämfört med kamerafönstret i figur 4.6.1 som möjliggör en enklare streckkodsläsning.
- C. Tillägg av knapp med texten "Continue" som visar att knapptryckningen leder till steget efter streckkodsläsning.

- D. Ändring till stegvis utlåning, här: steg 2. “Check out”. (Obs: I detta steg är utcheckningen inte färdig).
- E. Tillägg av text som visar låntagaren vars namn är kopplad till streckkoden/lånekortet.
- F. Knappen “Continue” ändrar färg från röd till grön när inläsningen är klar.
- G. Tillägg av illustration med två blå cirklar som visar att läsning av streckkoden är klar och att biblioteksmaterialet är redo att lånas ut (“Check out”).
- H. Ändring så att kamerafönstret är avstängt. Tillägg av text som bekräftar låntagaren och streckkoden, “Patron: “ + låntagares fulla namn “id: “ + streckkodstolkningen. Undertill visas texten “Place the smartphone over the item you would like to check out” som låter användaren veta att telefonen är redo att läsa av taggen.
- I. Tillägg av knappen “Back” som hänvisar tillbaka till kamerafönstret för streckkodsläsning, om användaren vill scanna på nytt.

5. Diskussion och slutsats

Detta kapitel sammanfattar det viktigaste i resultatet genom en diskussion utifrån problemformuleringen som beskrivs i avsnitt 1.4, etiska aspekter och framtida utvecklingsmöjligheter.

5.1. Slutsats utifrån frågeställningar

Detta avsnitt ämnar att besvara de inledande frågorna som beskrivs i avsnitt 1.4. Slutsatser kommer att dras utifrån resultaten och utifrån ett nyttoperspektiv för både bibliotekarier och bibliotek.

1. Vilka arbetsuppgifter och på vilket sätt kan RFID med NFC på en mobiltelefon att underlätta för bibliotekarier i sitt vardagliga arbete?

De arbetsuppgifter som kan underlättas med RFID med NFC-läsning på en mobiltelefon är följande enligt resultat från intervjuer: återlämning, utlåning och scanning av material. Dessa funktioner har också utvecklats, testats av bibliotekarier och ändrats efter målgruppens önskemål.

Utifrån resultatet från användartesterna med bibliotekarier kan det konstateras att funktionerna hade avsevärt underlättat i det dagliga arbetet genom tidsbesparing och ergonomisk avlastning. Bibliotekarierna fann prototypen enkel att använda, dvs enkel och tydlig UI med klar information och relevant terminologi som ledde till ett intuitivt användande såsom beskrivet i avsnitt 3.1.6 om användartestning. För att ytterligare förbättra användarflödet i prototypen utfördes ändringar efter användartesterna, se kapitel 4 för fler detaljer. Därefter genomfördes en demonstration av prototypen för en

bibliotekarie som också är kund hos Axiell och använder Axiell Quria® dagligen. Prototypen fick positiv respons och idén om att den skulle användas i det dagliga arbetet mottogs väl.

2. Hur ska arbetsflödet för en mobilapplikation utformas för att underlätta bibliotekariernas arbete?

Arbetsflödet är en av de viktigaste aspekterna att ta hänsyn till vid utformningen av prototypen. För att utforma en prototyp med ett smidigt arbetsflöde utfördes målgruppsanalys [idéfasen]. Som beskrivet i avsnitt 3.1.6 innebär en användarvänlig applikation att arbetsflödet fungerar bra och skapar en trevlig användarupplevelse. För att utreda vad en användarvänlig och trevlig användarupplevelse är för bibliotekarier har det således varit viktigt att utföra intervjuer och användartester vilka utgör en del av målgruppsanalysen [7].

Då målgruppsanalysen syftar till att samla verkliga behov hos målgruppen har intervjuerna och användartesterna utgjort den viktigaste delen i utformningen av applikationen. Det var av stor vikt att utföra intervjuerna med målgruppen innan utvecklingen kunde påbörjas. Fokus låg på att utveckla funktioner som skulle uppfylla målgruppens önskemål, vilka funktioner som hade använts mer frekvent än andra, och samtidigt att undvika bygga funktioner som i slutändan inte hade haft ett värde för bibliotekarier.

Genom användartester kunde målgruppen bekräfta eller dementera att prototypen uppfyllde de behov som uttrycktes i intervjuerna. Under användartestningen fick författarna specifik återkoppling på det som

målgruppen anser fungerar bra, mindre bra och varför. Detta kan ses i testprotokollet i bilaga 1 där det anges att testarna ska tala om vad de tänker när de utför ett specifikt scenario.

Exempelvis uttryckte en bibliotekarie att terminologin för funktionerna inte stämde överens med den vedertagna terminologin för bibliotekarier och att detta kunde orsaka förvirring. En sådan förvirring kan leda till en försämrad användarvänlighet och arbetsflöde [34]. Utifrån aspekten om användarvänlighet och arbetsflöde kunde denna typ av återkoppling leda till en förbättring av prototypen.

3. Vilka kriterier kan användas för utvärdering av prototypen?

Kriterierna som kan användas för utvärdering av prototypen är funktionalitet och användarvänlighet.

När det gäller funktionalitet var det viktigt att se till att de framtagna funktionerna fungerar väl och uppfyller sina syften. Detta kunde verifieras både genom enhetstester och användartester. Enhetstesterna syftade till att kontrollera att all ny kod som lades till uppfyllde de logiska kraven. Användartesterna användes för verifiering av de funktionella kraven. Utifrån testerna kunde prototypen utvärderas och refaktoriseras om exempelvis ett enhetstest inte gick igenom eller kraven från målgruppen ändrades.

Användarvänlighet var ett annat kriterium för att utvärdera prototypen. Med en användarvänlig prototyp ökar sannolikheten att prototypen kommer att

användas [34]. Utifrån användartesterna kunde examensarbetarna få återkoppling på det som hade gjort att bibliotekarier inte hade använt prototypen. En avbruten anslutning till servern mitt under en användarsession var en sådan anledning. Att behöva kontrollera anslutningen flera gånger kunde skapa en frustration och leda till försämrad användarvänlighet. Med återkoppling som denna kunde författarna åta sig att ändra och refaktorisera problemen som uppstod med anslutningen till servern. Denna typ av återkoppling anses värdefull eftersom den ger en indikation på de viktigaste åtgärderna som behöver utföras för att öka sannolikheten att applikationen kommer att användas i verkligheten.

4. Kan prototypen vidareutvecklas till en mobilapplikation som kan integreras med Axiell Quria®?

Författarna till examensarbetet anser att prototypen kan vidareutvecklas till en fullständig mobilapplikation som kan integreras med Axiell Quria®. I dagsläget är prototypen integrerad med webbsidan Quria vilket är en stark indikation på att integration mot Axiell Quria® är möjlig då Axiell Quria® också är webbaserad.

5.2. Reflektion över etiska aspekter

I ingenjörernas hederskodex konstateras följande: “Ingenjören bör i sin yrkesutövning känna ett personligt ansvar för att tekniken används på ett sätt som gagnar människa [...] och samhälle” [39].

Examensarbetet strävar efter att utforska nya teknikområden för att bidra till ökad samhällsnytta. Med prototypen kan bibliotekarier snabbare betjäna låntagare både i och utanför biblioteket. Bibliotekarier får möjlighet att röra sig fritt från diskarna med RFID-utrustning och istället öka interaktionen med låntagare. På så sätt skapas möjligheter till att sprida kunskap och kultur vilket gagnar människa och samhälle.

Prototypen lagrar inte personuppgifter i dagsläget då prototypen inte är kopplad till någon databas. Dock finns det stöd för hantering av personuppgifter i prototypen, nämligen vid utlåning/utcheckning där streckkoder läses och låntagarens namn visas på skärmen. All form av hantering av personuppgifter regleras av GDPR.

Etiska aspekter har tagits hänsyn till under delar av examensarbetet där det ingick interaktion med bibliotekarier. Under intervjuer fick bibliotekarier information om att det inspelade materialet inte kommer att användas till någonting annat än som underlag för utveckling av funktioner till prototypen. De blev också informerade om att de när som helst kunde avbryta intervjun utan att ange en orsak. Samma principer tillämpades under användartesterna.

5.3. Framtida utvecklingsmöjligheter

Applikationen som är i ett prototypstadium ska ses som ett underlag till vidareutveckling. I framtiden skulle prototypen kunna användas för fler funktioner än de som har utvecklats i detta examensarbete. Bibliotekarier har önskat funktioner som underlättar gallring av material, plocklistor och sökning

av material på hyllorna i biblioteken. Om prototypen skulle integreras med Axiell Quria® skulle hantering med fysiskt media i bibliotekssystemet kunna ske med hjälp av en mobiltelefon.

En funktion som i synnerhet har önskats är att kontrollera om böcker står på sin rätta plats genom att dra mobiltelefonen över flera böcker och utföra en scanning i ett svep. Att hitta försvunna böcker kan hjälpa att hålla nere utgifterna med upp till 170 000 kr per år för ett medelstort bibliotek [2]. I dagsläget finns det inga möjligheter att utveckla en sådan funktion i en likvärdig prototyp. Det beror på mobiltelefonens fysiska begränsningar där antennen enbart klarar en fokuserad avläsning. Däremot skulle prototypen kunna vara en del av en lösning tillsammans med tekniker såsom smarta bokhyllor med inbyggda RFID-läsare [2].

En funktion som skulle kunna utvidgas till låntagare är scanning av material. I en studie utförd av Setiabudi, Wiguno & Palit [40] utvecklades en applikation till besökare på museum där de kunde ta del av information om utställningar och utställningsobjekt via sin mobiltelefon med hjälp av RFID-tekniken. Låntagare skulle kunna, på samma sätt som i studien, ta del av information om exempelvis en bok genom att scanna dess NFC-tag.

Även utlåningsfunktionen hade kunnat utvidgas så att låntagare själva kan låna böcker med sin mobiltelefon. Med en ökad självbetjäning ökar även tillgängligheten av biblioteksmaterial för allmänheten. Hänsyn bör tas vid att prototypen inte lyckades läsa av streckkoder på lånekort till skillnad från streckkoderna på taggarna. Problemet tros ligga i att lånekortets streckkod som

är av typen Plessey inte kan läsas av det använda biblioteket QuaggaJS. Utifrån detta kan en lösning bestå i av en eventuell utvidgning av biblioteket med eget bidrag till den öppna källkoden. I övrigt gick det att utföra streckkodsläsning av andra typer utöver Plessey, exempelvis EAN.

På Axiell i Finland görs ett försök till att utvidga utlåning med RFID-tekniken i mobiltelefoner genom att låta en användare checka ut (låna) en bok direkt efter att en annan användare checkat in (återlämnat) den, dvs om ingen annan står i kö till boken. En sådan funktion kan öka tillgängligheten för biblioteksmaterial. Examensarbetet visar att det skulle vara helt möjligt att inkludera funktionen i en likvärdig prototyp.

6. Terminologi

Arbetsflöde	beteende hos en användare som följer ett arrangerat och repetitivt mönster pga en systematisk ordning av komponenter i en applikation
Biblioteksmaterial	typiska material som finns på bibliotek tillgängliga för nyttjande av allmänheten, material som kan chippas med RFID-taggar. Exempel på material kan vara böcker, tidskrifter, ljudböcker, CD/DVD-skivor, VHS etc.
Item ID	en unik kombination av siffror och bokstäver som används som en unik identifierare för ett biblioteksmaterial
Patron	engelskt ord för låntagare, en användare med ett konto i bibliotekssystemet som kan nyttjas för utlåning av biblioteksmaterial

7. Källförteckning

- [1] Axiell Group AB (n.d), tillgänglig: <https://www.axiell.com/se/> [Hämtad: 2021-01-28]
- [2] Tornbjer E., Ericson Lagerås, K. & Nilén, P. (2010). RFID på Malmö Högskola? En Konsekvensanalys. Malmö Högskola, Malmö, Sverige..[Online]. Tillgänglig: https://muep.mau.se/bitstream/handle/2043/10863/RFID-rapport%20september%202010_PUBLIKA.pdf?sequence=1&isAllowed=y [Hämtad: 2021-01-28]
- [3] Beaufort, F. (2021). Interacting with NFC devices on Chrome for Android. [Online]. Tillgänglig: <https://web.dev/nfc/> [Hämtad: 2021-01-28]
- [4] Löfqvist, M. Lär dig mer om RFID-teknik. (2012). [Online]. Tillgänglig: <https://www.minabibliotek.se/sv/article/1%C3%A4r-dig-mer-om-rfid-teknik>. [Hämtad: 2021-01-28]
- [5] Intent. (n.d). [Online]. Tillgänglig: <https://developer.android.com/reference/android/content/Intent> [Hämtad: 2021-01-28]
- [6] iCODE ISO 15693 Taggar. (n.d). Tillgänglig: <https://rfidprodukter.com/cat/rfid-taggar/hf-rfid-taggar-13-56-mhz/icode-iso-15693-taggar/> [Hämtad: 2021-01-28].
- [7] Idéfasen. (n.d). Tillgänglig: <http://anvandbarhet.se/bok:idefasen> [Online] [Hämtad: 2021-01-28]

- [8] NFC-V Just got better - Here's how. (2015). [Online]. Tillgänglig: <https://nfc-forum.org/nfc-v-just-got-better-heres-how/> [Hämtad 2021-02-08]
- [9] What is Gradle?. (n.d). [Online]. Tillgänglig: https://docs.gradle.org/current/userguide/what_is_gradle.html [Hämtad 2021-05-19].
- [10] Rangnau, T, v. Buijtenen, R., Fransen, F. & Turkmen, F. (2020). Continuous Security Testing: A Case Study on Integrating Dynamic Security Testing Tools in CI/CD Pipelines. 2020, IEEE 24th International Enterprise Distributed Object Computing Conference (EDOC), doi: 10.1109/EDOC49727.2020.00026
- [11] Nyström, S. (n.d). JUNIT. [Online]. Tillgänglig: <https://www.it.uu.se/edu/course/homepage/oopjava/st09/notes/f06-junit.html>. [Hämtad 2021-05-19]
- [12] McDaniel, B. (2003). An Algorithm for Error Correcting Cyclic Redundance Checks. [Online]. Tillgänglig: <https://web.archive.org/web/20170720165847/http://www.drdoobs.com/an-algorithm-for-error-correcting-cyclic/184401662>. [Hämtad 2021-05-19].
- [13] Automate your flow with GitHub Actions. (n.d). [Online]. Tillgänglig: <https://github.com/features/actions>. [Hämtad 2021-05-19].
- [14] About pull requests. (n.d). [Online]. Tillgänglig: <https://docs.github.com/en/github/collaborating-with-issues-and-pull-requests/About-pull-requests>. [Hämtad 2021-05-19].

- [15] Git Push. (n.d). [Online]. Tillgänglig:
<https://github.com/git-guides/git-push>. [Hämtad 2021-05-19].
- [16] JavaScript. (n.d). [Online]. Tillgänglig:
<https://developer.mozilla.org/sv-SE/docs/Web/JavaScript>. [Hämtad 2021-05-19].
- [17] HTML: HyperText Markup Language. [Online]. Tillgänglig:
<https://developer.mozilla.org/sv-SE/docs/Web/HTML>. [Hämtad 2021-05-19].
- [18] CSS: Cascading Style Sheets. (n.d). [Online]. Tillgänglig:
<https://developer.mozilla.org/en-US/docs/Web/CSS>. [Hämtad 2021-05-19.]
- [19] Creating a WebSocket Server with the Java API for WebSockets. (2018). [Online]. Tillgänglig: <https://learn.vonage.com/blog/2018/10/22/create-websocket-server-java-api-dr/?fbclid=IwAR2INxWItKsUvgOsZAN1QB0ZytYnh5pg526umeojfWdpBG4d1csTwlcFTm0#>. [Hämtad 2021-05-19].
- [20] JSON. (n.d). [Online]. Tillgänglig: <https://www.json.org/json-en.html>. [Hämtad 2021-05-19].
- [21] QuaggaJS. (2021). [Online]. <https://github.com/ericblade/quagga2>. [Hämtad 2021-05-19].
- [22] Foreground Services. (n.d). [Online]. Tillgänglig:
<https://developer.android.com/guide/components/foreground-services>
[Hämtad 2021-04-23]

- [23] Comstedt, M (2018). Arbeta Agilt - vad innebär det?. [Online]
Tillgänglig: <https://onbird.se/arbete-agilt-vad-innebar-det/>. [Hämtad 2021-05-19].
- [24] Teams. (n.d). [Online]. Tillgänglig: <https://support.microsoft.com/sv-se/office/v%C3%A4lkommen-till-microsoft-teams-b98d533f-118e-4bae-bf44-3df2470c2b12>. [Hämtad 2021-05-19].
- [25] Kvale, S. & Brinkmann, S. (2014). Den kvalitativa forskningsintervjun. Lund: Studentlitteratur.
- [26] Zoom. (n.d). [Online]. Tillgänglig: <https://zoom.us/>. [Hämtad 2021-05-19].
- [27] Lantz, A. (2013). Intervjumetodik. (3rd ed). Lund: Studentlitteratur.
- [28] Trost, J. (2010). Kvalitativa intervjuer. Lund: Studentlitteratur.
- [29] Lauesen, S. (2002). Software Requirements - Styles and Techniques. Addison-Wesley.
- [30] Nyström, S. (n.d). Versionshantering. [Online]. Tillgänglig: <https://www.it.uu.se/edu/course/homepage/oopjava/st10/notes/f08-version.html>. [Hämtad 2021-05-19].
- [31] Create an Android project. (n.d). [Online]. Tillgänglig: <https://developer.android.com/training/basics/firstapp/creating-project>. [Hämtad 2021-05-19].
- [32] Sending the user to another app. (n.d). [Online]. Tillgänglig: <https://developer.android.com/training/basics/intents/sending>, [Hämtad 2021-05-19].

- [33] Singh-Kilbride, K. (2021). [Online]. WebSockets vs Long polling. Tillgänglig: <https://ably.com/blog/websockets-vs-long-polling>. [Hämtad 2021-05-19].
- [34] Matz, K. (2013). Designing Usable Apps: An agile approach to User Experience design. Winchelsea Press.
- [35] Krug, S. (2014). Don't Make Me Think, Revisited: A Common Sense Approach to Web Usability. (3rd ed). New Riders Publishing.
- [36] Testa! - Att tänka på inför, under och efter test. En praktisk handbok om testmetoder och Prototypande. (n.d). [Online]. <https://tillvaxtverket.se/download/18.9f781f21626464cda18cfad/1522769480524/Testa%20&%20prototypa%20-%20en%20handbok.pdf>. [Hämtad 2021-05-19].
- [37] Ping. (2018). [Online]. Microsoft. Tillgänglig: <https://docs.microsoft.com/en-us/windows-server/administration/windows-commands/ping>. [Hämtad 2021-05-19].
- [38] CSS tooltip. (n.d). [Online]. Tillgänglig: https://www.w3schools.com/css/css_tooltip.asp. [Hämtad 2021-05-19].
- [39] Hederskodex. (n.d). [Online]. Sveriges Ingenjörer. Tillgänglig: https://www.sverigesingenjorer.se/om-forbundet/sveriges-ingenjorer/hederskodex/?gclid=CjwKCAjwv_iEBhASEiwARoemvB8frpd0LtlInHwctTGzy0vUXEIVZA6dJ_0wONmB8a66Y9IP PfsOjBoCi7sQAvD_BwE. [Hämtad 2021-05-19].
- [40] Setiabudi, D. Wiguno, R.C., Palit, H.N. (2017). Near Field Communication Technology in Delivering Information in Museums. 2017 International Conference on Soft Computing, Intelligent System and

Information Technology, pp 173-177, doi: 10.1109/ICSIT.2017.52

- [41] Kahneman, D. (2017). Tänka, snabbt och långsamt. Volante.
- [42] Axelsson, J. (2011). [Online] Föreläsning kvalitativ metod. Karlstads Universitet. Tillgänglig: <https://www3.kau.se/kurstorg/files/f/C10B99641b0dc28104JKw182E0B1/Forelasning%20kvalitativ%20metod2012.pdf> [Hämtad 2021-05-25].

8. Bilagor

I detta kapitel finns bilagor med material som har använts i examensarbetet.

Bilaga 1 - Testprotokoll

Användartest för prototypen RFID mobilapp

Testvärd:

Testare:

Tekniskt kunnig: 1 - 2 - 3 - 4 - 5 (1 = inte alls, 3 = använder mejl och surfar på nätet, 5 = väldigt kunnig)

1. *Läs upp följande introduktion:*
”Välkommen! Vi har bitt dig att användartesta vår app RFID mobilapp. Jag kommer att visa dig en androidtelefon och en tagg och sedan ge dig instruktioner för olika uppgifter som du ska utföra. Under testets gång ber jag dig att tala högt om dina tankar och funderingar kring appen.
2. *Öppna appen och ge användaren en minut att kolla på förstasidan. Ställ följande frågor och anteckna svaren:*
 - a. “Vad är det första du lägger märke till när du börjar använda appen?”
3. *Läs upp scenariorna. Börja med 1 och instruera användaren att lösa problemet som givits. Ställ sedan följande frågor:*
 - a. “Vad tänker du nu?”
 - b. “Vet du att det fungerade?
Om ja: “Hur då?”
Om nej: “Hur hade du velat veta att det funkar?”
4. *Om det tar för lång tid:*
 - a. “Vad tänker du?”
5. *Avslutningsvis fråga användaren om deras funderingar kring appen.*

Scenario:

1. Du vill läsa av taggen som finns på boken för att visa booken.
2. Du vill nu ändra taggens item ID. alltså programmera item id till ett nytt id.

3. Du ändrar till item id = 14 eller 18. Vad får du upp? *Hint: 14 = Mark Levengoods bok, 18 = Harry Potter.*
4. Du vill checka in taggen.
5. Du vill checka ut
6. Stäng ner appen i mobilen, öppna Quria och försök checka in.
7. Öppna appen igen, starta kommunikationen och försök checka in.
8. Du vill byta språk i RFID-appen.
 - a. Byt till svenska.
 - b. Byt till engelska

Bilaga 2 - Intervjuguide

Intervjuguide inför möte med bibliotekarier

Hej och tack för att du har valt att medverka till den här studien. Vi är två högskolestudenter som ska utreda om det går att använda en mobil enhet för att scanna av material på biblioteket mha RFID-tekniken. Den här intervjun är värdefull för oss eftersom den ger oss vägledning i hur den här utvecklingen ska ske, ffa vilka funktioner som vi ska fokusera på. Vi vill gärna ta fram något som ska hjälpa er i ert vardagliga arbete som bibliotekarier.

Vi har avsatt en timme för intervjun mest för att du ska få prata i lugn och ro om ditt arbete, men det kan hända att intervjun avslutas tidigare om det inte finns något mer att tillägga. Du kan också avbryta intervjun när som helst utan att ange en orsak till det.

- Vad använder du disken till?
 - Vilka funktioner använder ni mest i ert bibliotekssystem?
 - Vilka tycker du att du behöver ha för att kunna vara ute bland folk?
 - Vilka funktioner använder ni minst?
- Hur ofta använder ni systemet?
- När är du i behov av en RFID-platta?
- Hur långt behöver du gå från hyllorna till en RFID-platta?
- Hur ofta behöver du bära böcker bort till en RFID-platta?
 - Varför gör ni det?
- Vad är smidigast med systemet?

- Om du hade kunnat gå bort till en bokhylla och scanna av den med en platta eller mobil, vilken information hade du velat se på skärmen?
 - Vad hade du velat göra med den informationen?
- Vilka funktioner hade du gärna sett i en mobilapp?
- Föredrar du mobil eller surfplatta?
- Använder du tjänstemobil i ditt arbete idag?

Tack för din medverkan. Om det skulle vara så att du kommer på något mer som du gärna hade velat förmedla så är du välkommen att höra av dig. Du kan skriva till antingen mig eller Aml på nedanstående adresser:

xx@gmail.com

yy@hotmail.com



LUND
UNIVERSITY

Series of Bachelor's theses
Department of Electrical and Information Technology
LU/LTH-EIT 2021-823
<http://www.eit.lth.se>