

# Examensarbete: Fördjupning inom streckkoder i Android

Författare: Urban Kertes

Uppdragsgivare: MMT AB

Utbildning: Javautvecklare, Teknikhögskolan

Kurs: Examensarbete, 25 YH-poäng

Handledare: Hans Sandström

Version 2. Göteborg 2014-03-31

# 1 Innehållsförteckning

1	Innehållsförteckning .....	2
2	Lista på förkortningar.....	3
3	Bakgrund .....	4
3.1	problemformulering.....	4
4	Avgränsning .....	6
4.1	Bibliotek .....	6
4.2	Androidenheter.....	7
4.3	Androidversion.....	7
4.4	Funktionalitet.....	7
5	Metod .....	8
5.1	Marknadsundersökning .....	8
5.2	Utveckling.....	8
6	Resultat .....	13
6.1	ZBar .....	13
6.2	ZXing.....	15
6.3	Resultat av test och övervägande .....	16
6.4	Testade android-enheter .....	16
6.5	Resultat i förhållande till förstudie.....	17
7	Diskussion .....	18
7.1	Allmänt.....	18
7.2	Mina lärdomar .....	19
8	Bilagor .....	20
8.1	Användarmanual.....	20
8.1.1	Installation .....	20
8.1.2	Användande.....	20
8.2	Källförteckning .....	21
8.3	Koddokumentation .....	22
8.4	Timrapport .....	22

## 2 Lista på förkortningar

LIA: Lärande i arbete

1D: En-dimensionell streckkod

2D: Tvådimensionell streckkod

RFID: Radio frequency identification

UPC: Universal Product Code

JNI: Java Native Interface

GNU LGPL: GNU Lesser General Public License

QR-code: Quick Response code

### 3 Bakgrund

Jag har valt att fördjupa mig inom streckkoder främst för att förbereda mig inför min LIA. Under min LIA-period kommer jag att arbeta med streckkoder för en inventeringsapplikation på företaget MMT (Marin MätTeknik) AB. Jag kommer också att jobba med Android och förhoppningsvis hinna med att utveckla ett flertal applikationer.

På grund av detta, samt att jag skulle kunna tänka mig att arbeta med utveckling för Androidplattformen efter min utbildning så föll sig examensarbetets fokus naturligt.

Applikationen jag skall utveckla skall kunna läsa in streckkoder och med hjälp av denna kod kunna hitta inventarier i en databas. Ett antal av inventariernas fält skall kunna ändras i appen, och huvuddatabasen skall kunna uppdateras.

För att detta skall kunna fungera smidigt behövs en snabb och kraftfull kod-skanner. Jag har för detta syfte undersökt vad som finns på marknaden, både betalversioner och öppen källkod, och valt ut att testa två stycken lösningar i en Android-app som medföljer mitt examensarbete.

#### 3.1 problemformulering

MMT behöver ett system för att bättre kunna hantera inventarier när flyttas mellan olika platser och olika fartyg. De har i nuläget en databas kallad Amos för att kunna hålla reda på inventarier, och har själva utvecklat en web-version; Wamos. Men de har behov av en smidigare lösning när inventarier byter plats, och en app skulle kunna lösa detta.

Appen behöver kunna skanna 1D streckkoder av typen "CODE 39" av behöver använda sig av ett skanningsbibliotek för detta ändamål. Det finns flera sådana bibliotek att tillgå, och vilken som skall användas måste bestämmas.

Vidare behöver det testas att "vanliga" mobiltelefoners och läsplattors kameror klarar av att skanna streckkoder på ett snabbt och smidigt sätt, och att ingen specialutrustning förutom en vanlig telefon eller läsplatta alltså skulle behövas.

Det är även önskvärt om biblioteket skulle kunna skanna de mer komplicerade 2D-koderna som QR koder, för eventuella framtida behov.



FIGUR 1: 1D STRECKKOD AV TYPEN CODE39



FIGUR 2: 2D STRECKKOD AV TYPEN QR

## 4 Avgränsning

Jag har valt att göra följande avgränsningar i mitt examensarbete.

### 4.1 Bibliotek

Jag har valt att avgränsa de testade skanningsbiblioteken till de två populäraste på Androidplattformen. Vilka dessa är har jag undersökt genom utvecklingsforumet Stack Overflow (<http://stackoverflow.com/>), Googles webbutik Google Play samt olika Android -bloggar och hemsidor.

#### **ZBar**

Zbar är ett skanningsbibliotek som erbjuds som öppen källkod under licensen GNU LGPL ([http://en.wikipedia.org/wiki/GNU\\_Lesser\\_General\\_Public\\_License](http://en.wikipedia.org/wiki/GNU_Lesser_General_Public_License)).

ZBar är programmerat i språket C och det finns versioner till många olika plattformar, som Windows, Linux, Iphone och Android. För att kunna använda biblioteket på Androidplattformen används JNI, som möjliggör exekvering av C-kod i Java.

#### **ZXing**

ZXing ("zebra-crossing") är ett bibliotek för skanning av de flesta typer av streckkoder, som kan användas som öppen källkod under licensen Apache 2.0 (<http://www.apache.org/licenses/LICENSE-2.0.html>).

Det går att implementera ZXing direkt i sin app, men upphovsmännen förespråkar att man hellre skall ladda ner appen Barcode Scanner (<https://play.google.com/store/apps/details?id=com.google.zxing.client.android>) och använda sig av dess skanningsmotor via Intents.

## 4.2 Androidenheter

Min andra avgränsning är antalet typer av mobiltelefoner och läsplattor som jag har kunnat testa skanningsbiblioteken på. Detta är en självklar praktisk avgränsning då jag inte har fysisk tillgång till alla de Android-enheter som finns på marknaden i dagsläget. Jag har dock kunnat få tillgång till de modeller som skall användas på MMT, LG Nexus 5 samt Acer Nexus 7.

## 4.3 Androidversion

Den tredje och sista avgränsningen är vilken Android-version som appen skall utvecklas för.

Jag har i samråd med MMT utvecklat appen från version 4.03 (Ice Cream Sandwich) och högre versioner. Detta beslut togs då MMT har nyare mobiltelefoner och läsplattor bland sina användare, och ingen annan publik är tänkt för appen i nuläget.

## 4.4 Funktionalitet

Jag har begränsat appens funktionalitet till att skanna streckkoder - av alla de vanligast förekommande typerna - samt att kunna behandla, spara och skicka dessa vidare. Angränsande funktionalitet som RFID-koder har inte utforskats på grund av brist på tid samt att MMT i nuläget inte har något intresse av detta.

## 5 Metod

Min metod har bestått av två delar.

### 5.1 Marknadsundersökning

Jag har undersökt marknaden för skanningsbibliotek och genom sökningar på forum, bloggar och webbutiker och hittat två stycken lämpliga kandidater.

Jag har satt mig in i dessa biblioteks fördelar och nackdelar och läst om andra utvecklarens erfarenheter med dem.

### 5.2 Utveckling

Jag har utvecklat en Android-app som testar de skanningsbibliotek jag valt ut under verkliga förhållanden. Jag har med appens hjälp kunnat välja ut det bibliotek som jag anser passa MMTs behov bäst.

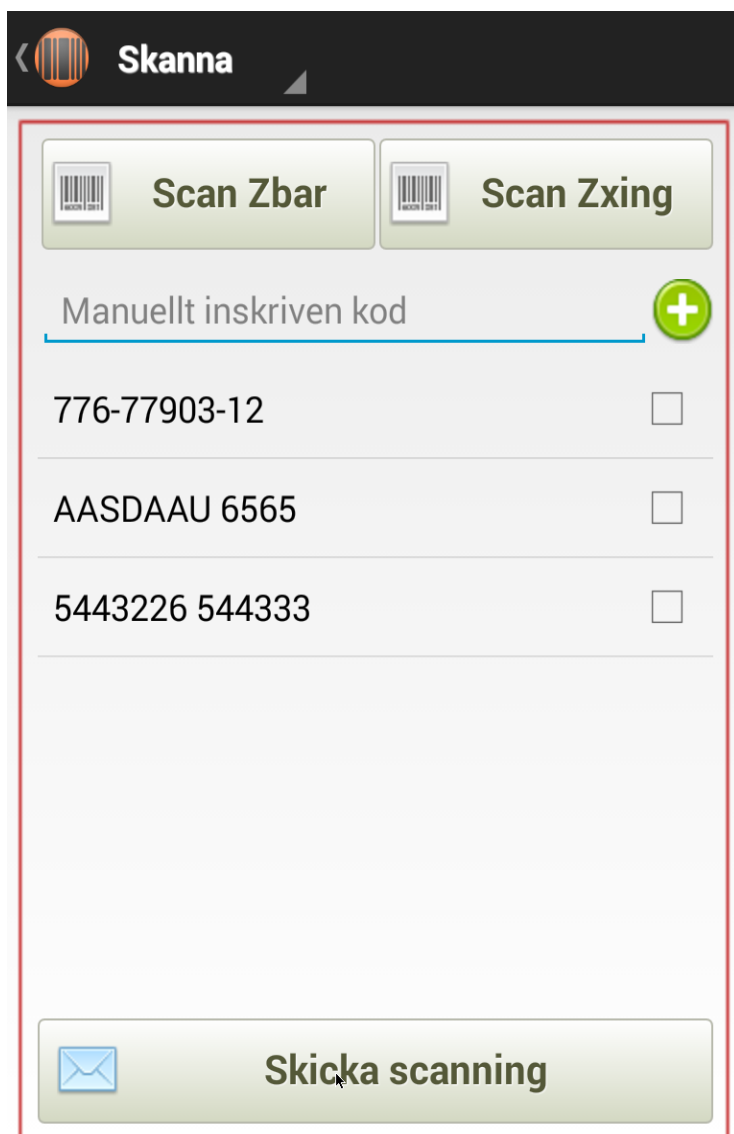


Appen består av en Home-sida, från man kan välja att skanna koder, se tidigare koder, läsa information om appen samt avsluta.



FIGUR 3: SKANNINGSAPPENS STARTSIDA "HEM"

Från Skanna Kod-sidan kan man testa de två skanningsbiblioteken. Alla koder man skannar sparas automatiskt i en databas, som går att nå från Tidigare Skanningar.



FIGUR 4: SKANNINGSAPPENS "SKANNA" -SIDA

En extra-inlagd funktion utöver det som bestämts i Förstudien är en databas. Alla koder som skannas sparas i denna. Det går att se gamla koder samt radera dem från sidan ”Tidigare skanningar”.



The screenshot shows a mobile application interface with a dark header bar containing a back arrow, a barcode icon, and the text 'Tidigare'. Below the header is a large, empty rectangular area with a light gray background and a red border. At the bottom of this area is a table with six rows of data. Below the table is a note in italics: '(lång-klicka på en rad för att radera den)'. At the very bottom is a green button with the text 'Radera allt i databasen'.

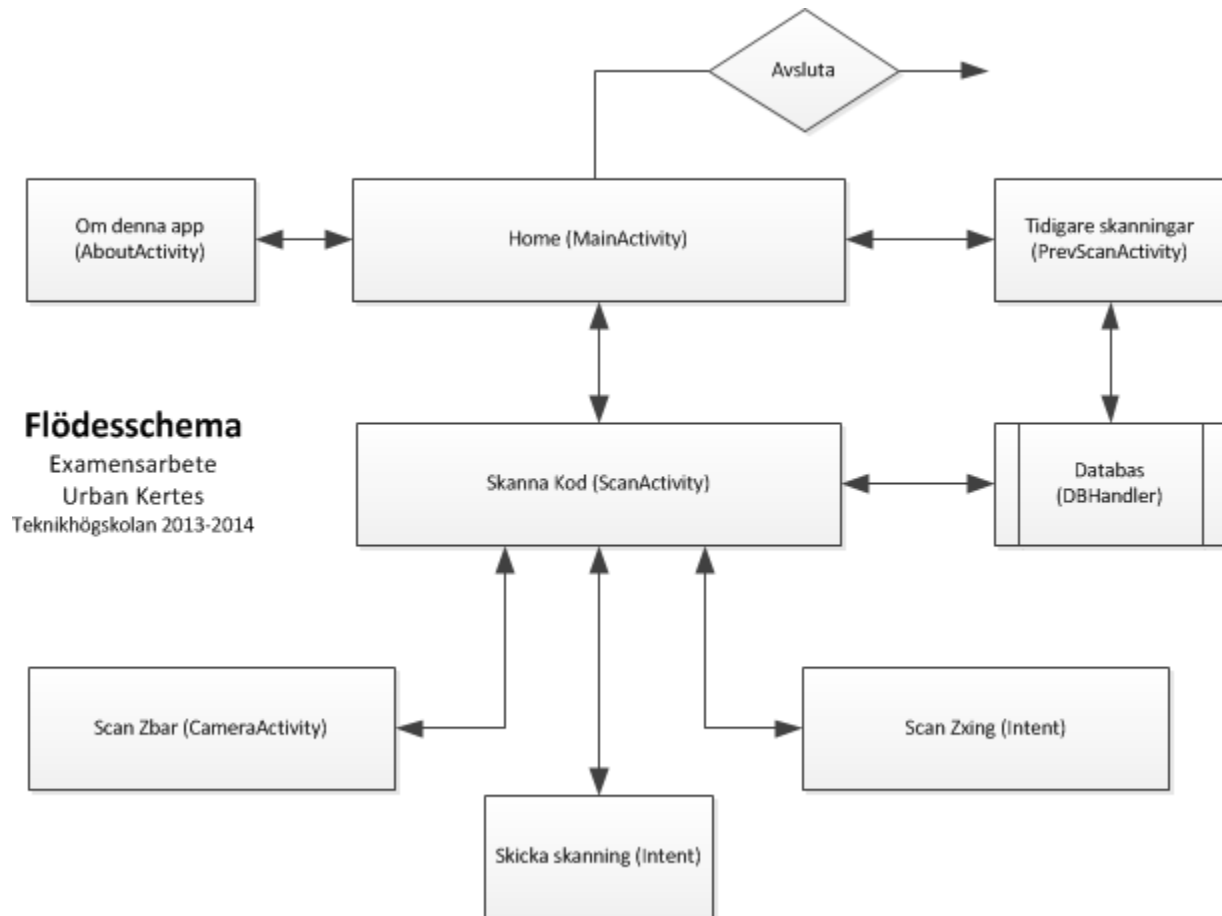
1	5443226 544333	2014-03-31 16:06:23
2	AASDAAU 6565	2014-03-31 16:06:39
3	776-77903-12	2014-03-31 16:06:52
4	711234113	2014-03-31 16:12:37
5	NTRE 21653	2014-03-31 16:12:49
6	88904 TRWW	2014-03-31 16:12:59

*(lång-klicka på en rad för att radera den)*

**Radera allt i databasen**

FIGUR 5: TIDIGARE SKANNINGAR

Man kan även skicka vidare sin skanning från ”Skanna kod”-sidan till de flesta appar man har installerade på sin Android-enhet via ett Intent. Man markerar de koder man skall skicka och trycker på knappen ”Skicka skanning”, där man väljer destination.



FIGUR 6: FLÖDESSCHEMA ÖVER SKANNINGSAPPEN

## 6 Resultat

Efter att appen programmerats och testats har beslutet tagits att använda skanningsbiblioteket Zbar framför Zxing Detta beslut grundas på följande slutsatser:

### 6.1 ZBar

ZBar är ett enkelt och robust bibliotek som inte kraschat en gång under min testperiod. Det är relativt enkelt och smidigt att använda och integrera i en app, och det skannar snabbt.

Under min marknadsundersökning fann jag att det är populärt och det finns mycket hjälp att tillgå på forum och websidor. Det är också öppen källkod under licensen GNU LGPL (GNU Lesser General Public License)

Det används i porträtt-läge, vilket passar MMTs krav väl.

Nackdelar jag fann är det är skapat och underhålls av en person, och därför kanske är mer sårbart än ett bibliotek utvecklat av en grupp.

Det är programmerat i språket C och används på Android via JNI (Java Native Interface). Detta kan möjligen göra det mindre robust än ett bibliotek skrivit direkt i Java och gör det svårare att debugga på Android.



FIGUR 7: ZBar SKANNAR EN KOD. NOTERA ATT SKANNINGEN ÄR INKORPORERAD I APPEN.

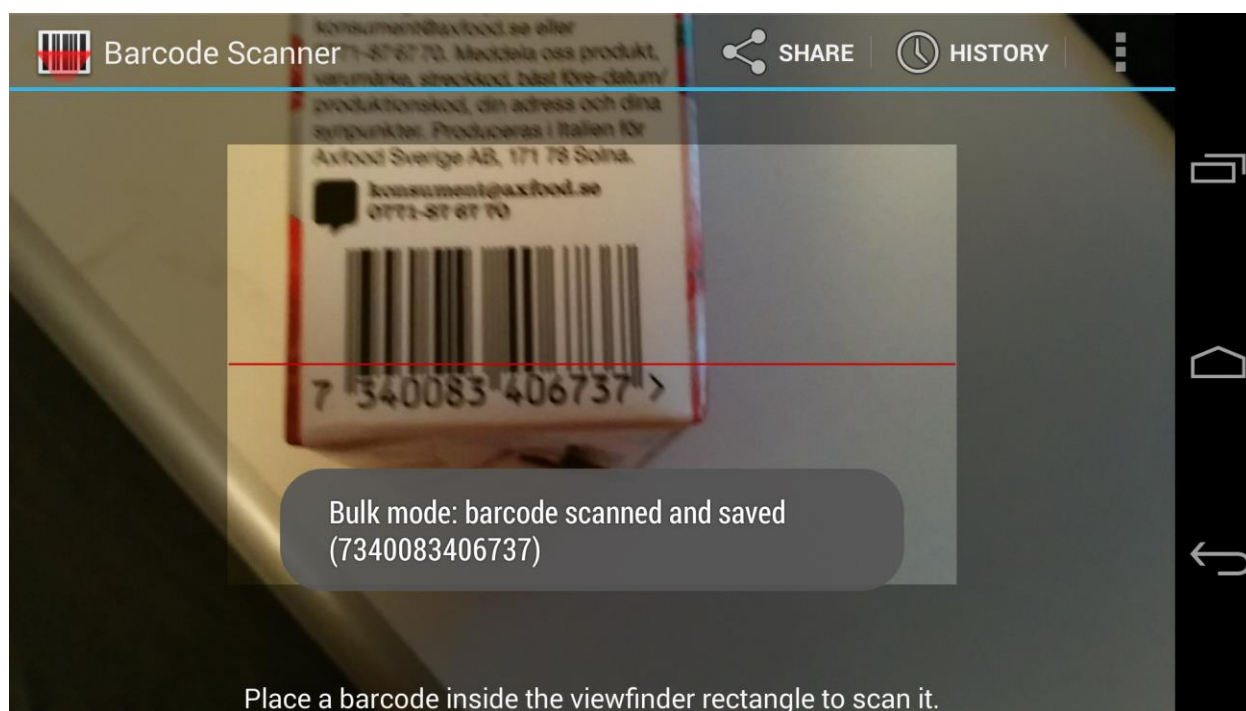
## 6.2 ZXing

ZXing är även det mycket populärt (dess tillhörande app Barcode Scanner har 80 miljoner nedladdningar på Google Play), vilket talar väl för uppdateringar och hjälp. Det har många funktioner och är ett moget bibliotek.

Det utvecklas av ett team istället för av bara en person vilket ger goda chanser till fortsatta uppdateringar. Det är öppen källkod under Apache 2.0-licensen.

En specifik nackdel i mitt fall är att det har många fler funktioner än vad MMT efterfrågar och är onödigt avancerat. Det är också komplicerat att integrera direkt i en app: skaparna förespråkar att man installerar en separat app – Barcode Scanner – och skannar genom den appen istället för att inkorporera ZXings kod direkt i sitt eget program.

En annan nackdel är att det används i landskaps-läge, och man måste alltså vända på telefonen varje gång man skall skanna.



FIGUR 8: ZXING BARCODE SCANNER. EN SEPARAT APP ÖPPNAS NÄR MAN SKANNAR MED DETTA BIBLIOTEK.

### 6.3 Resultat av test och övervägande

Baserat på analyserade och testade fördelar och nackdelar väljer jag att använda ZBar. Det är det bibliotek som enligt mig passar kundens behov bäst av de två som jag har prövat.

### 6.4 Testade android-enheter

Vidare har jag provat min app på följande plattformar:

<b>Typ</b>	<b>Tillverkare</b>	<b>Modell</b>	<b>Resultat Bibliotek</b>
Mobiltelefon	LG	Nexus 5	Fungerar med båda
Mobiltelefon	HTC	One	Fungerar med båda
Mobiltelefon	Samsung	Galaxy S4	Fungerar med båda
Läsplatta	Samsung	Galaxy tab 3	Fungerar inte med något testat bibliotek
Läsplatta	Acer	Nexus 7	Fungerar med båda

Jag fann att inget av de skanningsbibliotek jag testat fungerade på läsplattan Samsung Galaxy Tab 3. Anledningen tror jag är att kameran har en dåligt fungerande autofokus, vilket är en funktion som används vid skanning av streckkoder.

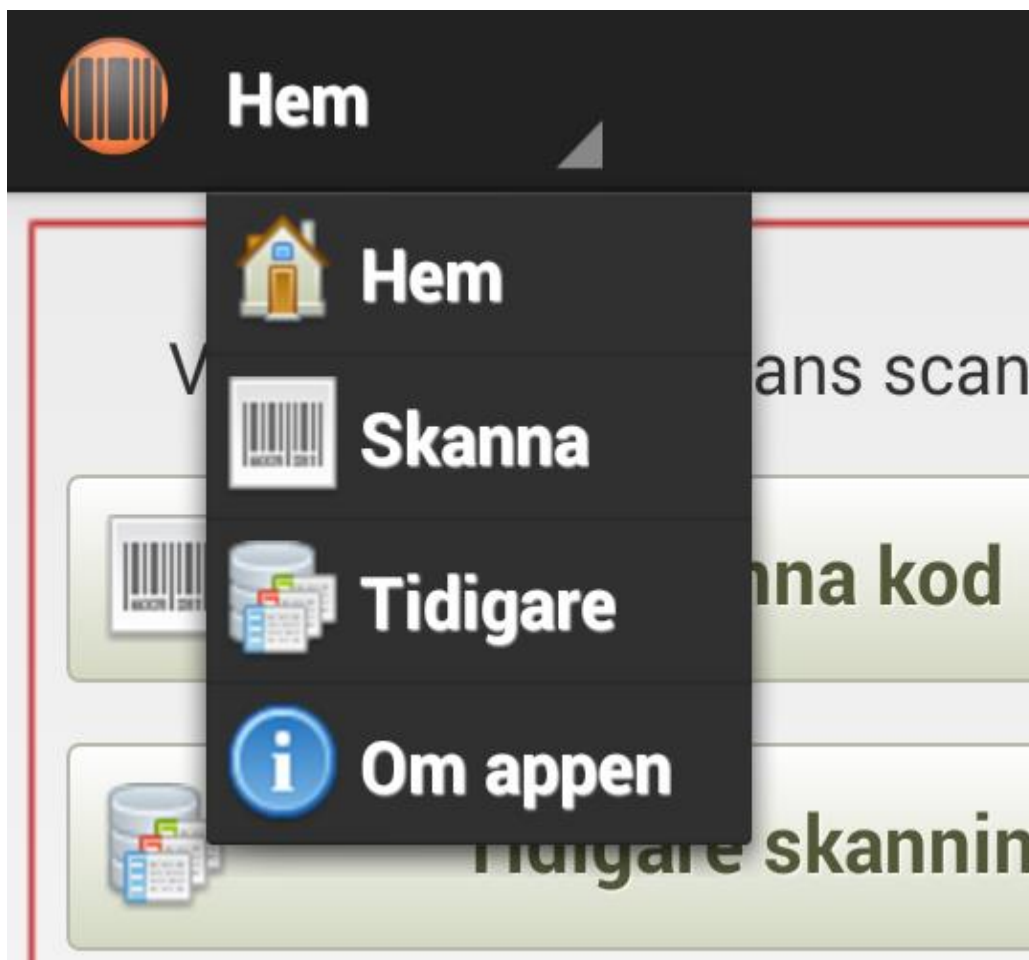
Båda skanningsbiblioteken fungerade lika bra på alla andra testade enheter.



## 6.5 Resultat i förhållande till förstudie

Jag tycker att jag lyckats införliva allt som min handledare och jag bestämde under arbetet med förstudien. Det blev dock några förseningar i arbetet och jag fick justera planeringen två gånger. Detta på grund av arbete med andra kurser.

Jag har kunnat utveckla allt vad som bestämdes och även längre tekniskt än vad som först bestämdes – jag har lagt till en databas och databashantering i appen, samt mer komplicerad navigering via en Spinner i appens Action Bar.



FIGUR 9: NAVIGERING VIA EN SPINNER I ACTION BAR I SKANNINGSAPPEN

## 7 Diskussion

### 7.1 Allmänt

Möjliga förbättringar hade varit att undersöka fler skanningsbibliotek på marknaden. Anledningen till att så inte skedde är examenarbetets tidsomfattning, och att jag ansåg att de två mest populära biblioteken skulle räcka för att få en uppfattning.

Hade ingen av de två utvalda biblioteken fungerat efter MMTs krav hade jag fått undersöka marknaden mer ingående, men det var nu inte fallet då båda gick att använda tillfredställande.

Vidare hade det funnits en möjlighet att köpa en skanningslösning, såsom Barcode Scanner Plus (<https://play.google.com/store/apps/details?id=com.srowen.bs.android>). Detta behövdes nu ej då de två utvalda biblioteken fungerade.

Jag hade också kunnat testa appen på fler olika Android-enheter, men där hindrades jag av ekonomiska och praktiska begränsningar. De enheter som skall användas på MMT blev testade och fungerade bra.

En sista tanke är att jag hade kunnat implementera ZXing direkt i min app (istället för att kalla på Barcode Scanner via Intent) såsom jag gjorde med ZBar. Anledningen till att jag inte gjorde så var att utvecklarna inte rådde till detta, och tidsbrist: ZXing är ett mer avancerat bibliotek än ZBar och utvecklat för att användas via Intent (även om fullständig integration enligt utvecklarna är möjlig).

## 7.2 Mina lärdomar

Jag har först fått en uppfräschning av mina gamla Android-kunskaper från våren 2013. Utvecklingen av appen har rört vid ett flertal områden som krävt återvändande till gamla kunskaper, men jag har också behövt titta på och fördjupa mig inom (för mig) nya områden inom Android-utveckling - bl.a skanningsbibliotek, JNI, databashantering samt navigering via Spinners.

Jag har lärt mig mycket om streckkoder, både allmänt som historia och användningsområden, men också detaljer som hur streckkoder fungerar tekniskt samt skillnader mellan olika standarder.

Det har också varit lärorikt att arbeta med min handledare, fastställa och begränsa projektets omfattning samt få lyssna och anpassa arbetet efter löpande feedback. Detta känns som en realistisk övning i vad ett kund/konsultutbyte skulle kunna innehålla.

Jag har under examensarbetet fått använda och utveckla min analytiska förmåga. Detta när jag fått undersöka marknaden för skanningsbibliotek och införskaffa information om vad som finns tillgängligt, samt när jag vägt fördelar och nackdelar mellan de två produkter som jag valde ut.

Jag har sist men inte minst fått öva och slipa på min presentationsteknik under själva opponeringen. Här har jag också lärt och använt mig av ett för mig nytt presentationsverktyg; [www.prezi.com](http://www.prezi.com).



## 8 Bilagor

### 8.1 Användarmanual

#### 8.1.1 Installation

Ladda ner appen Examensarbete.apk från

<https://bitbucket.org/ukertes/examensarbete2/downloads#download-321277>

Det lättaste sättet att installera appen på en telefon eller läsplatta är att maila filen Examensarbete.apk till sig själv via Gmail. Då kan man klicka på filen direkt på sin Android-enhet och välja att installera den.

Det går också att använda sig av appinstallationsverktyg som APK Installer (<https://play.google.com/store/apps/details?id=com.potente.apk.installer>) eller likande från Google Play. Följ sedan appinstallationsverktygets instruktioner.

#### 8.1.2 Användande

Skanningsappen ”Examensarbete” är i sig lätt att använda. För att skanna en kod välj ”Skanna Kod”.

Välj sedan du vill skanna med hjälp av skanningsbiblioteket Zbar eller Zxing genom att trycka på respektive knapp.

För att använda Zxing behöver man installera Barcode Scanner (<https://play.google.com/store/apps/details?id=com.google.zxing.client.android>)

Rikta kameran mot en streckkod och den kommer att skannas. Om du använder Zbar kan du välja att skanna fler koder utan att gå tillbaka.

Koderna skrivs in i en lista. Dra en kod åt höger i listan om du vill radera den. Tryck på knappen ”Skicka skanning” om du vill skicka iväg koden/koderna t.ex via email.

Du kan konsultera tidigare skannade koder via sidan ”Tidigare Skanningar”. Härifrån kan du även radera alla skanningar.

## 8.2 Källförteckning

Android Programming – The Big Nerd Ranch Guide av Bill Phillips och Brian Hardy (ISBN 0-321-80433-3)

Android in Practice av Charlie Collins m.fl. (ISBN 1-935182-97-7)

Programming Android av Zigurd Mednieks m.fl. (ISBN 978-1-449-31664-8)

<http://stackoverflow.com/>

<https://github.com/zxing/zxing>

<https://github.com/zxing/zxing/wiki>

<https://play.google.com/store/apps/details?id=com.google.zxing.client.androidtest>

[http://en.wikipedia.org/wiki/Barcode\\_Scanner\\_\(application\)](http://en.wikipedia.org/wiki/Barcode_Scanner_(application))

<https://play.google.com/store/apps/details?id=com.srowen.bs.android>

<http://sourceforge.net/projects/zbar/>

<http://sourceforge.net/p/zbar/discussion/>

<https://groups.google.com/>

<http://en.wikipedia.org/wiki/ZBar>

<http://developer.android.com/index.html>

<http://developer.android.com/training/camera/cameradirect.html>

<http://android-developers.blogspot.se/>

<http://www.vogella.com/tutorials/android.html>

[http://www.keepautomation.com/code\\_39/](http://www.keepautomation.com/code_39/)

<https://code.google.com/p/barcode-android/source/browse/#svn%2Ftrunk%2Fsrc%2Fcom%2Fsugree%2Fbarcodegen>

<http://www.howstuffworks.com/gadgets/high-tech-gadgets/upc.htm>

<http://www.barcodesinc.com/generator/index.php>

<https://github.com/romannurik/Android-SwipeToDismiss>

### 8.3 Koddokumentation

Var vänlig se Index.html i bifogad fil JavaDocUrban.zip eller på <https://bitbucket.org/ukertes/examensarbete2>

För att se all koden gå till <https://bitbucket.org/ukertes/examensarbete2>

Presentationen finns som PDF även på denna plats, eller på [http://prezi.com/4vd-keiydzwm/edit/#26\\_24309637](http://prezi.com/4vd-keiydzwm/edit/#26_24309637)

### 8.4 Timrapport

Tidsrapport Examensarbete Urban Kertes		
		130
Datum	Kommentarer	Timmar
19/11, 6/12 2013	Möte med handledare, skrivande av förstudie	6
21/11/2013	Undersökning av skanningsbibliotek	6
22/11/2013	Undersökning av skanningsbibliotek	4
24/11/2013	Undersökning av skanningsbibliotek	8
25/11/2013	Skapande av basic-app	8
26/11/2013	App: planering av uppbyggnad	6
27/11/2013	App: planering av gränssnitt	8
28/11/2013	Implementering av gränssnitt och MainActivity	8
29/11/2013	Implementering av gränssnitt och ScanActivity	4
1/12/2013	Implementering av gränssnitt och ScanActivity och listview	6
2/12/2013	Implementering av gränssnitt och ScanActivity och listview	8
7/12/2013	Implementering av Zbar i CameraActivity	8
4/12/2013	Implementering av Zbar i CameraActivity, CameraPreview	6
9/12/2013	Implementering av Zbar i CameraActivity, CameraPreview	6
10/12/2013	Gränssnitt, swipe to dismiss	4
11/1/2014	Implementering av Zxing	4
19/1/2014	Implementering av Zxing	3
10/2/2014	Test på flera Android-enheter	6
1/3/2014	Dokumentation, rapport, utvärdering	7
2/3/2014	Dokumentation, rapport	6
3/3/2014	Bugfix	8