

# Moving the museum outside its walls

An Augmented Reality Mobile Experience

TOMMY MARSHALL



**KTH Information and  
Communication Technology**

Degree project in  
Communication Systems  
Second level, 30.0 HEC  
Stockholm, Sweden

# Moving the museum outside its walls:

An Augmented Reality Mobile Experience

T. Marshall

marshall@kth.se

November 29, 2011

Supervisor:

Lars Lundberg Attentec AB

Examinator and academic supervisor:

Gerald Q. Maguire Jr.

KTH Royal Institute of Technology  
School of Information and Communication Technology  
Stockholm, Sweden



## Abstract

This thesis project examines how to design an Android application that uses augmented reality to help users navigate in the City of Stockholm. The main users of the application are assumed to be first time visitors to Stockholm.

The thesis project concerns design, technologies, and usability issues - while focusing on how to design a suitable application. The application will maybe be used by the Nobel Museum to offer information, and assistance to tourists who are looking for and visiting different locations connected with the Nobel Prize.

The focus will be to achieve an easy to use system for these users, as the user will not be able to get support after leaving the museum. The application is meant to be distributed as freeware that will contribute to a positive experience by museum visitors. One of the aims is that the users should find the application appealing, motivating them to learn more about the Nobel Prize following their visit to the Nobel Museum.

For tourists this means that they get the possibility to experiencing Stockholm in a new way, where information is stored at different locations. They get a guided tour of Stockholm without a guide, they can take the time they want. The museum gets the possibility to spread their message about the Nobel Prize around in Stockholm in a new and exciting way. With the use of technology it is possible to spread the museum experience outside its walls, which will be shown in this report.

**KEYWORDS:** Augmented reality, museums, handheld, usability, Nobel Museum, Android, city tour, navigation, low-fi prototype, high-fi prototype, usability measurements

## Sammanfattning

Det här examensarbetet undersöker hur man kan designa en Android-applikation som använder förstärkt verklighet för att hjälpa användare navigera i Stockholms stad. Den huvudsakliga användaren antas vara förstagångsbesökare i Stockholm.

Examensarbetet avser design, teknik och användarvänlighet - med fokus på hur man utformmar en lämplig applikation. Applikationen kommer eventuellt att användas av Nobelmuseet för att erbjuda information och hjälp turister som är intresserade av olika platser som är relaterade till Nobelpriset.

Fokus kommer vara att uppnå ett lättanvänt system för dessa användare. Användaren kommer inte kunna få något stöd efter att de lämnat museet. Applikationen är tänkt att delas ut som freeware, som bidrar till en positiv upplevelse för museets besökare. Ett av syftena är att användarna ska anse att applikationen är intressant och motivera dem att lära sig mer om Nobelpriset efter deras besök på Nobelmuseet.

För turister innebär detta att de får möjlighet att uppleva Stockholm på ett helt nytt sätt där information lagras på olika platser. De får en guidad tur i Stockholm som sker utan en guide, de kan ta den tid de vill. Museet får möjlighet att sprida sitt budskap om Nobelpriset runt i Stockholm på ett nytt och spännande sätt. Med hjälp av teknik är det möjligt att sprida museiupplevelse utanför dess område, som kommer att visas i denna rapport.



---

## Acronyms and Abbreviations

<b>Acronyms</b>	<b>Description</b>
Android	A operating system created for smartphones from Google, open source.
AR	Augmented reality (learn more in Section 2)
API	Application Programming Interface
FPS	Frames per second
Geo	Geographical
GPS	Global Positioning System
High-fi	High fidelity prototype
HMD	Head mounted device, a display mounted on the head of a user
iOS	A operating system created for smartphones by Apple, created for their own devices.
KML	Keyhole Markup Language
Low-fi	Low fidelity prototype
OS	Operating System, a required software that makes it possible to use computers and applications.
OSM	Open Street Map
PDA	Personal Digital Assistant, is a small hand held device used for storing personal information.
POI	Point-of-interest, a location of importance
SD	Secure Digital (memory)
SMS	Short Message Service, usually used for text messages on mobile phones
QR	Quick Response markers
XML	eXtensible Markup Language



## Acknowledgements

This project could not have been completed without the support and help from Aron Ambrosiani at the Nobel Museum whom has been the project owner. He supported the project with information, and more specifically provided the images that are used in the application.

A lot of aid has been given from Attentec and especially through Lars Lundberg's contributing expertise and enthusiasm.

Patrik Ragnarsson and Anton Hedström has given a lot of support, answering questions and making it all possible through their contributions to Lara.

Gerald Q. Maguire Jr. through his contributions to the academic aspects of this thesis project.

Kim Fouchenette and Chengcheng Huang for their feedback as opponents on this master thesis project report.

Olof Somell for finalising the part as owner for the Nobel Museum when Aron received new assignments.



# Contents

<b>1. Introduction</b>	<b>1</b>
1.1. Background . . . . .	1
1.1.1. The Nobel Museum . . . . .	1
1.1.2. Tourism in Stockholm . . . . .	2
1.1.3. Augmented Reality . . . . .	2
1.1.4. What is usability? . . . . .	2
1.2. Problem Definition . . . . .	3
1.3. Goals . . . . .	3
1.4. Limitations . . . . .	4
1.4.1. Users . . . . .	4
1.5. Structure . . . . .	4
<b>2. Augmented Reality</b>	<b>7</b>
2.1. How Augmented Reality Works . . . . .	7
2.1.1. History of Augmented Reality . . . . .	7
2.1.2. Technology . . . . .	8
2.1.3. Tracking . . . . .	8
2.1.4. Graphics . . . . .	10
2.2. Augmented Reality Framework for smartphones . . . . .	10
2.2.1. Qualcomm's QCAR . . . . .	10
2.2.2. String . . . . .	11
2.2.3. Unifeye Mobile SDK . . . . .	11
2.2.4. Wikitude SDK . . . . .	11
2.3. Similar Projects . . . . .	12
2.3.1. Lara . . . . .	12
2.3.2. Spike Island . . . . .	12
2.3.3. Louvre-DNP Museum Lab . . . . .	12
2.3.4. Uppåkra exhibit . . . . .	13
2.3.5. My Digital World . . . . .	13
2.3.6. Websign . . . . .	13
2.3.7. Augmented Reality Browsers . . . . .	13
<b>3. Navigation and Interaction</b>	<b>15</b>
3.1. Interaction with smartphones . . . . .	15
3.1.1. Haptic . . . . .	15
3.1.2. Movement . . . . .	15
3.1.3. Geographical location . . . . .	15
3.1.4. Sound . . . . .	15
3.2. Navigation . . . . .	16
3.2.1. Maps . . . . .	16
3.2.2. Directional Navigation . . . . .	16

3.2.3.	Turn-by-Turn Navigation . . . . .	16
3.3.	Augmented Reality Navigation . . . . .	16
3.3.1.	Google KML . . . . .	16
3.3.2.	Augmented Reality Markup Language . . . . .	17
3.4.	Navigation with maps . . . . .	17
3.4.1.	Google Navigation . . . . .	17
3.4.2.	Open Street Map . . . . .	17
<b>4.</b>	<b>Method</b>	<b>19</b>
4.1.	Design Methods . . . . .	19
4.1.1.	Users . . . . .	19
4.1.2.	Low-fidelity prototype . . . . .	19
4.1.3.	High-fi prototype . . . . .	20
4.2.	Usability testing . . . . .	21
4.2.1.	Conducting tests . . . . .	21
4.2.2.	Measuring usability . . . . .	22
4.3.	Work Method . . . . .	23
4.3.1.	Brainstorming . . . . .	23
4.3.2.	Scrum . . . . .	24
4.3.3.	Scrum Implementation . . . . .	24
4.4.	Tools . . . . .	25
4.4.1.	Eclipse . . . . .	25
4.4.2.	Unity 3D . . . . .	25
4.4.3.	Mono Develop . . . . .	26
4.4.4.	SQLite . . . . .	26
4.4.5.	Cygwin . . . . .	26
<b>5.</b>	<b>Evaluation of the Underlying Technology</b>	<b>27</b>
5.1.	Augmented Reality Frameworks . . . . .	27
5.2.	Navigation . . . . .	27
5.3.	Reference Image . . . . .	28
<b>6.</b>	<b>Implementation</b>	<b>31</b>
6.1.	Users . . . . .	31
6.2.	Paper prototype . . . . .	31
6.3.	High definition prototype . . . . .	34
<b>7.</b>	<b>Usability Testing</b>	<b>43</b>
7.1.	Paper Prototype . . . . .	43
7.1.1.	User Study . . . . .	43
7.1.2.	Evaluation . . . . .	44
7.2.	High definition Prototype . . . . .	44
7.2.1.	User Study . . . . .	44
7.2.2.	Evaluation . . . . .	46

---

7.2.3. Suggested Improvements . . . . .	48
<b>8. Conclusions and Future work</b>	<b>49</b>
8.1. Conclusions and Results . . . . .	49
8.2. Future Work . . . . .	50
<b>A. Reference Images</b>	<b>59</b>
<b>B. Reference Images with markings</b>	<b>61</b>



---

## List of Figures

1.	An example of a QR Code for <a href="http://www.attentec.se">http://www.attentec.se</a> . . . . .	9
2.	Frame marker and Natural-feature tracking . . . . .	10
3.	Comparing reference images . . . . .	29
4.	Paper Prototype flow . . . . .	32
5.	Navigation paper prototype . . . . .	33
6.	Paper prototype . . . . .	34
7.	The improved Paper Prototype flow (with the new states indicated in color) . . . . .	35
8.	Changes made to the paper prototype . . . . .	36
9.	Start screen comparison . . . . .	37
10.	Overview screen comparison . . . . .	37
11.	Navigation . . . . .	38
12.	Info overview for the Nobel Museum . . . . .	39
13.	Puzzle overview - Completed . . . . .	39
14.	Unity . . . . .	40
15.	The high definition prototype flow chart (with the new states indicated in color) . . . . .	41
16.	Picture of the changed scen at the Nobel Museum . . . . .	47



---

## List of Tables

1.	Usability definition . . . . .	3
2.	Scrum Sprints . . . . .	25
3.	Number of stars given to the different reference images . . . . .	29
4.	Summary of the test subjects' answers to the questions . . . . .	48



---

# 1. Introduction

This chapter provides background information about the Nobel Museum and augmented reality. Details of the technology of augmented reality will be presented in Chapter 2. Chapter 1 also defines the problem, states the goals of the thesis project, and explicitly describes the limitations.

## 1.1. Background

This section will describe what the Nobel Museum is, points of interest, tourism, augmented reality in short, and define usability.

### 1.1.1. The Nobel Museum

The Nobel Museum is dedicated to the life of Albert Nobel, the Nobel Prize, and the Nobel prizewinners. The Nobel Museum wishes to spread knowledge and start discussions about science and culture. The Nobel Museum is located in the old town of Stockholm[1].

The Nobel Prize is awarded in five categories; Physics, Chemistry, Psychology or Medicine, Literature, and Peace. In addition, the Swedish National Bank (Sveriges Riksbanken) awards the "Sveriges Riksbank Prize in Economic Sciences in Memory of Alfred Nobel" [2]. The Nobel Prize is associated with different parts of Stockholm. Some of the most important points-of-interest (POI) are:

- The Nobel Museum - a place to learn more about the Prize.
- Grand Hôtel - the Nobel Prize winners stay in this hotel during the time that they are in Stockholm for the prize ceremony; additionally this hotel was the location of the Nobel Banquet until 1920.
- Nybrokajen 11 - Nobel Prize ceremony was at this location until 1921.
- Nobel Foundation - the location of the foundation that manages the assets left by Alfred Nobel in his will, income from these assets are used to fund the prize.
- Concert Hall - the Nobel Prize Ceremony is held in this location.
- City Hall - Nobel Banquet is held in the City Hall after the prize ceremony.

Many of the museum's visitors ask for directions to these locations. Most visitors to the museum are tourists that are in Stockholm for the first time, and many of them happen to walk past the Nobel Museum when walking around in the old town.

There is a value for the museum to find out if tourists are interested of experiencing Stockholm with the aid of technology. If it is a way for them to spread there influence outside the walls of the museum. The tourists will get an entirely new experience of Stockholm in a unique way.

### 1.1.2. Tourism in Stockholm

In 2010, there were more than 10,000,000 over night visitors that came to the Stockholm Region[3]. Most of these visitors came from Germany, the United Kingdom, and the United States. There are several different types of tourists: enterprise tourists, private tourists, and home tourists. Enterprise tourists come to a place for work or a conference. The costs of enterprise tourists are usually paid by the company they work for. Private tourists are tourists that travel to a new place. These tourists generally pay their own travel costs. Home tourists are people who live in or near the area that they are visiting, but they are visiting locations that they usually do not go to[4].

### 1.1.3. Augmented Reality

In this section we will briefly define what augmented reality is. Details about augmented reality technology are given in Chapter 2.

The simplest way of explaining what augmented reality is, is to give an example: Imagine that you have an advanced navigation system in your car. To illustrate the path that you should follow, information is projected on the front window indicating the turns that you should take while you are driving. This information may include a moving map that shows where you are located, and indications of exactly where you should turn. The system combines information from the real world (the view out the front window) with additional information (projected onto this window) giving the driver an extra layer of virtual information, i.e., the additional information augments the real-world information.

A commonly accepted definition for augmented reality is based on[5, 6]:

- Generating a virtual image on top of a real image
- Enabling interaction in real-time
- Seamlessly blending 3D (or 2D) virtual objects with real objects.

As a result an augmented reality system has to be designed so that it interacts in real time with the real world combining both worlds.

Augmented reality as a technology has evolved. However, until the introduction of mass market smartphones few devices could actually be used to provide a mobile user with an augmented reality experience (outside of laboratories and highly controlled environments)[5, 7, 8, 9].

### 1.1.4. What is usability?

A system that is easy to use is a system that has high usability, while a system that is hard to use has low usability. In a low usability system the user does not know if it is the user or the system that is stupid. In this section a description of usability, and factors contributing to high usability will be explained. These six factors of usability[10] are illustrated in Table 1.

**User friendliness** is a combination of these six factors. It is often hard to develop systems that have high scores on all factors, and can thus it may be better to prioritize the different factors in term of their importance.

Name	Description
Fit for use	The system supports the tasks that are necessary for the user.
Ease of learning	How easy the system is to learn.
Task efficiency	The efficiency of tasks for frequent users.
Ease of remembering	How easy is it to remember the system for the sporadic user.
Subjective satisfaction	The satisfaction of the system for the user.
Understandability	How easy is it for the user to understand what the system does.

Table 1: Usability definition

In this thesis project the main focus of usability will be on **Subjective satisfaction**, **Understandability**, and **Ease of learning**. It is important that *fit for use* is fulfilled so that all necessary functions are available. *Task efficiency* is not very relevant for the thesis project as users are using the application for relatively small tasks. *Ease of remembering* is of low priority as the user is not expected to use the application more than a few times.

## 1.2. Problem Definition

There are two problems that must be solved in this thesis project:

1. Designing and evaluating an augmented reality application that offers both high usability, and navigation within the city of Stockholm.
2. Specifically the project should demonstrate an Android application that extends the museum experience of a Nobel Museum visitor to outside the walls of the museum.

These two problems will be addressed throughout this thesis. The first problem primarily concerns design, implementation, and evaluation; while the second problem concerns the information (content) that is presented to the user via the application. It is important that there be adequate material and that it be suitably presented in order to have a valid evaluation of users' experience when using the application.

## 1.3. Goals

The primary goal of this thesis project is to build an Android application which uses augmented reality with a strong focus on usability (definition in Section 1.1.4) to provide tourists with an improved experience *outside* of the Nobel Museum (while building on the content and furthering the interests of the museum). The application should be designed in such a fashion that it is easy for a user (read more about the users in Section 1.4.1) to move about, locate, and learn about the different locations in the city that are associated with the Nobel Prize.

## 1.4. Limitations

Due to the limited amount of time available to generate a completed application, this project will build upon an earlier prototype called "Lara"[11] (see Section 2.3.1).

The application will be created for the Android operating system. The Android operating system used was Froyo 2.2 (API level 8), due to the fact that applications could not be saved to the SD card in earlier versions[12]. The decision to use Android was based on discussion with the Nobel Museum and statistics on that Android was a quick growing operating system.

### 1.4.1. Users

The Nobel Museum presented their idea of the main user for the application, a brief description will be presented below also the typical museum visitor will be described. In Section 6.1 the potential user is described in more detail.

The typical Nobel Museum visitor is a tourist from all over the world, in all ages, and with a range of different spoken languages. Most users usually go into the museum to find out more about the Nobel Prize, but some visitors are more interested of finding important related locations. These visitors are the intended target group and is described in the next paragraph.

The main user of the application is assumed to be a male tourist (private or enterprise) between 20 and 35 years of age. The user is used to using a smartphone. The user would like to be able to show the use of the application and what they learned while touring about in Stockholm at home to their friends and family. The user's second language is English. The user is aware of the Nobel Prize, but the user does not know very much about where the prize ceremony is held, who is involved in selecting the prize winners, what activities the Nobel laureates do when in Stockholm, nor what they were awarded their prize for.

## 1.5. Structure

The thesis consists of the following eight chapters:

- **Chapter 1 - Introduction:** Introduction to the subject, and goals of the thesis project.
- **Chapter 2 - Augmented Reality:** Descriptions of how augmented reality works, similar projects, and augmented reality frameworks for smartphones.
- **Chapter 3 - Navigation and Interaction:** Information about different ways of interacting with smartphones and a description of different navigation technologies.
- **Chapter 4 - Method:** How the thesis project was executed, what methods & tools were used, and an introduction to methods of evaluating usability.
- **Chapter 5 - Evaluation of Technology:** This chapter addresses the following questions: What augmented reality framework will be used? What navigation solution is appropriate? Does the image quality for reference maps effect user experience?
- **Chapter 6 - Implementation:** The creation of a paper prototype and a (beta) high-fidelity prototype that is ready for testing.

- **Chapter 7 - Usability Studies:** User studies of the paper (low-fi) and the beta (high-fi) prototype.
- **Chapter 8 - Conclusions and Future work:** The conclusions and results of this project. This chapter answers the questions stated in the introduction, and what that has been left for future work.



---

## 2. Augmented Reality

There are a lot of different augmented reality applications for museums today. In this chapter we will review some of them, examine some details of how the technology works, and which components (specifically which platforms) are needed.

### 2.1. How Augmented Reality Works

To understand how augmented reality works we will start by reviewing some of the history of augmented reality. Following this we will introduce two of the most important questions: how to track objects and generate images. We also describe the major issues in user interaction, and how graphics technology is used to realize augmented reality.

#### 2.1.1. History of Augmented Reality

As mentioned in Section 1.1.3, augmented reality is not a new field. Scientists have been working with augmented reality for over 40 years[13]. Augmented reality has evolved over this period from head mounted displays connected to computers in backpacks, to PDAs, and smartphones. Augmented reality has traditionally required a lot of computer power (further details in Section 2.1.2). Before the introduction of smartphones this computing power was provided by using computers in backpacks. These computers were frequently connect to a head mounted display (called HMD). This approach was not socially accepted, and setting such up was expensive, and difficult[14]. One project that used a HMD was ARQuake. ARQuake is a modification of the game Quake, that is played with a HMD, and backpack computer in the real world. Looking around in the world through the headset the monsters will act as expected in the game. A virtual gun was available so that monsters could be killed[15]. Over a period of years the volume of the necessary equipment decreased with the development of computers, laptops, PDAs, and mobile telephones. The backpack and HMD setup was superior to hand held solutions, but have not yet had a major breakthrough in the market (HMD has had success within the military and healthcare)[11, 16].

High speed wireless networks enabled a new solution - do the computation remotely, and only capture images, render, and realize the user interface locally. This was convenient since many PDAs had a wireless network interface built into them. The presence of a high speed wireless network interface made it possible to perform the heavy processing on a remote server, while minimizing the processing that needed to be done locally. In this way the computations could be speed up, but this solution requires a high speed wireless connection. By performing the majority of the computations remotely this approach reduced the time needed for processing by 41%[14]. By suitably sharing the work with servers, applications could execute with sufficiently low delay that the user had a good experience.

Today many people have smartphones, thus making it possible to realize augmented reality, and to share this experience with anyone that has a smartphone. The smartphone together with a high speed wide area network, and remote servers facilitates the use of augmented reality by a single user and extends it to a *connected set of users*. Earlier augmented reality was mainly used by scientists and people interested in technology[11, 17]. However, the transition

of augmented reality to smartphones opens many new doors, for applications in infrastructure, advertisement, and other businesses. Augmented reality has been ranked in a recent Gartner report as *one of the ten most important* disruptive technologies in the next five years[18].

### **2.1.2. Technology**

What happens when an object is displayed on the smartphone? In this subsection we will look into the details of how this process works.

There are a number of important components that are involved in displaying an object on the screen of a smartphone, specifically: camera, image processing, display, and application. In order for an application to recognize something a reference model or image is needed (unless specific markers are used, more information about the use of such markers is given in Section 2.1.3 - see also the image of a marker in Figure 1). The image acquired by the camera can be processed to see if it contains something that the application recognizes (i.e., something that matches either an image it already knows or something that matches a model it already has). The details of this process can be found in [9].

If the application finds a match, then it can take some particular action. For example, if the application recognizes a specific marker, then it might overlay some information over the image of the identified marker. Alternatively, the application could play an audio or video clip that is associated with this marker, display a set of time tables associated with transportation that is available nearby, display a restaurant menu, display historical facts associated with this location, etc.

The application being developed in this thesis project, the camera acquires an image of the surrounding area. Generally this image will be displayed on the device's screen. The application processes the captured image and matches it against a set of reference images and markers. If a match is found, then the application overlays a virtual object on top of the displayed image from the camera. More details about graphics and rendering are given in Section 2.1.4.

Rendering graphics is quite processor intensive. Such rendering was a problem for older systems, however recent smartphones have special hardware to improve the performance of the device when rendering graphics. This additional hardware has made it possible to enhance the visual quality of the graphics that a device can render within a given amount of time and, with a limited amount of energy consumption.

### **2.1.3. Tracking**

There are several ways that an object can be identified by the augmented reality application, the two main methods are based upon special markers or natural features of the target. In both cases we need to have a suitable image (generally one with high contrast) and sufficient (but limited) complexity. They should not be in a repeating pattern and preferably with angles or other characteristic that make them unique. It is important that the image is tracked well so that the virtual objects are located at the right place in the real world[19].

There are a number of different markers that have been used for tracking objects. Recently Quick Response (QR) codes have been used on posters, and advertisements to provide links

to web pages. Figure 1 illustrates a QR code that points to a homepage address (in this case <http://www.attentec.se>). An algorithm is used to encode the text of the URL into a QR code. Another algorithm is used to translate the QR code back into text when the camera views a QR code[20, 21]. In augmented reality the application identifies an object imaged by the camera. In the case of augmented reality rather than rendering a web page from a website the augmented reality could render a 3D object on top of the marker.



Figure 1: An example of a QR Code for <http://www.attentec.se>

Fiducials are markers, similar to a QR code, Figure 1, but instead for representing specific information the fiducials are used by the application for various different purposes depending on what the application is programmed to do. Fiducials generally have high contrast so that they are easy to identify[14, 11].

A frame marker is a marker that is different from a fiducial or QR code. The frame of the frame marker is used to both identify the marker and to encode information (see Figure 2(a)). Inside the frame any type of information can be stored. Each frame marker (just like fiducials) are programmed locally within the application to have its own identification. A frame marker has a recommended physical size of 3-10 cm[22]. Consider Figure 2(a), the entire pink area could be used for information, as long as it does not disturb the black, and white fields along the frame. There are only 512 frame makers (According to Qualcomm's Developer guide - specifically the Frame Maker page[23]).

All three types of markers can be identified from different angles, upside down, and from the sides. This makes the markers quite flexible and makes it possible to use the markers in a dynamic way.

Natural-feature tracking uses the characteristics of the image as a marker (rather than using a pre-defined marker). An example of an image with high contrast is shown in Figure 2(b). The figure is a image of an model at the Nobel Museum of Palytoxin, but many other things could be tracked with natural-feature tracking e.g. a book or a building. For this kind of tracking

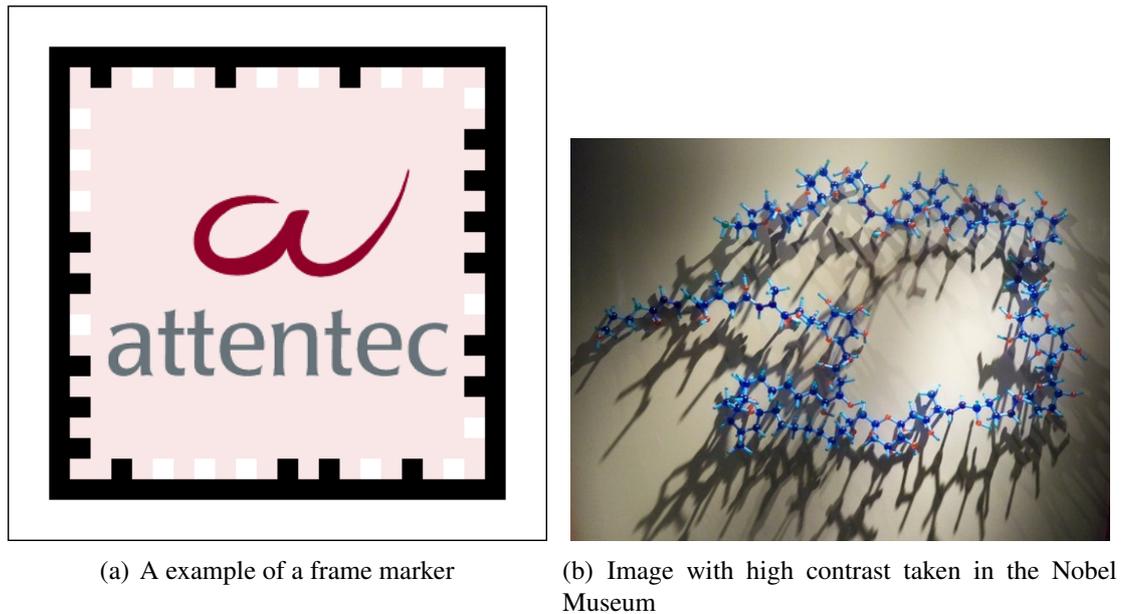


Figure 2: Frame marker and Natural-feature tracking

to work a reference image is required[9]. More details of the required image processing are given in Section 2.1.2.

#### 2.1.4. Graphics

Many application have some sort of graphical interface. This interface can range from simple menus with information to moving objects. The graphics itself is an important part of the application.

Most applications only have simple menus that display boxes containing information. However, more and more applications provide richer graphical user interfaces, especially in the case of games and video applications. In the case of augmented reality the interface can combine what the camera sees with objects rendered as artefacts on top of the image from the camera. Several 3D engines can be used within a smartphone, but the most common specifications for 3D interfaces is OpenGL[24].

## 2.2. Augmented Reality Framework for smartphones

There are several different augmented reality frameworks available for different smartphones and operating systems. This section reviews some of these frameworks.

### 2.2.1. Qualcomm's QCAR

Qualcomm has developed an augmented reality platform that is called QCAR. QCAR is currently available for Android, and Apple's iOS (the iOS release came during the thesis project).

QCAR can be used with natural-feature tracking, frame markers, and fiducials. QCAR makes it possible to track up to five markers at the same time. However, it is recommended that you utilize less than 50 tracked reference images in an application due to performance issues. Another important feature is that QCAR is able to track a box (or something with a similar shape). QCAR has built in functionality so that a user can press on surfaces, and QCAR understands where the user pressed (i.e., to realize virtual buttons). QCAR is free to use both for non-commercial and commercial use. QCAR uses OpenGL ES<sup>1</sup> for rendering 3D objects. It is possible to use Unity 3D (see Section 4.4.2 technology to create and manipulate 3D objects for QCAR[25].

To add a marker the developer uploads the reference image to Qualcomm's homepage. Qualcomm's Target Management System converts the image, and identifies all the angles that it can find. A high contrast image is required in order to find as many angles as possible. The reference image is then converted into an image with marks where angles have been identified. That information is saved in a file that is subsequently used by an application[23].

### 2.2.2. String

String[26] has been developed by the company String to provide augmented reality for mobile applications. String is designed for Apple's iOS, and uses OpenGL for rendering (this can be implemented with any 3D graphics engine) or the application can use Unity Technologies' Unity for rendering.

String uses natural-feature tracking, image markers, frames, and fiducials to track targets. The image used should have high contrast and be fairly unique. String can track up to 10 objects simultaneously. It is possible to use String with virtual buttons. These buttons are easy to integrate within existing projects. String must be purchased for both commercial and non-commercial use.

### 2.2.3. Unifeye Mobile SDK

Unifeye Mobile Software Development Kit (SDK)[27] is an augmented reality platform that can be used with Apple's iOS, Google's Android, Nokia's Symbian, and Microsoft's Windows Mobile operating system. The Unifeye Mobile SDK is designed so that it can easily be used on new platforms. This SDK offers marker-based, and 2D-texture tracking of images. The platform was designed for mobile devices, thus it uses fast and modular libraries, and algorithms. The 3D library used is OpenGL. Currently this SDK only offers global positioning system (GPS) and compass support for location and orientation of an Apple iPhone[28]. The Unifeye Mobile SDK platform was created by metaio and must be purchased for both commercial, and non-commercial use.

### 2.2.4. Wikitude SDK

Wikitude Software Development Kit (SDK)[29] is a platform created for Android by Mobilizy. The platform is free to use, but an application needs to be registered before the application

---

<sup>1</sup>OpenGL ES is a version of OpenGL optimized for embedded systems.

is distributed via Android Market. Wikitude SDK can be used in any Android application, making it possible to create POIs. The POI are viewed through the camera with overlays providing augmented reality. The platform does not support natural-feature tracking, as currently the SDK only supports marker tracking and GPS. There are some restrictions on the use of 3D objects, as textures will not be shown. This SDK also uses OpenGL ES. Wikitude SDK is part of Wikitude Worlds (see Section 2.3.7).

## **2.3. Similar Projects**

There are many projects that are similar in some ways, be it in concept, execution, or usage to the goals of this thesis project. This section describes several similar projects; specifically describing what they are and how they are used. There are few projects that have the same (or similar) focus as this thesis project, but several parts of these other projects are interesting for this thesis project. The projects that have the most similarities are augmented reality browsers (see Section 2.3.7).

### **2.3.1. Lara**

Lara[11] is a treasure hunt designed for new students at Linköpings University. The purpose is to motivate the new students to walk around the campus, in order to find new and exciting places that they will need during their time at the university. The user gets clues about how to find the different locations, and is expected to know something about the area that they are located in. The user has a distance indicator that can be used to learn if they are heading in the right direction. When at the correct location the user looks through their smartphone's camera for the virtual treasure. Lara uses QCAR running on Android to implement the application. What parts of Lara that has been used and implemented are described in Section 6.3.

### **2.3.2. Spike Island**

Spike Island[30] located in Ireland is an island with a lot of history, smuggling, prisons, and Christianity. An application for Android phones was created using two types of augmented reality: geographical location and frame markers. The geographical augmented reality uses the Wikitude SDK to get the user to the correct location. Frame markers were created with AndAR to provide the user with information. The application helps users get to different parts of the island, then shows them information that is not directly available in the real world.

### **2.3.3. Louvre-DNP Museum Lab**

The Louvre-DNP Museum has created an augmented reality guide that helps the user get information when looking through a hand held computer's screen. The user walks around an object and looks at the object through the hand held computer, but the users also sees a character standing there who will guide the user by providing information about the object. To guide the user around in the museum a virtual balloon is used. The user follows this balloon

by looking at the screen of their hand held computer. This project has also been developed by metaio, and was built with the Unifeye SDK[31].

#### **2.3.4. Uppåkra exhibit**

At Lund University's Historical Museum an exhibit about Uppåkra utilizes augmented reality. Uppåkra was a village from the Iron age that was discovered in the 1930s. A temple is the center of the exhibit, and the idea was to create a virtual temple that could be seen during the exhibit. To guide the user through the exhibit a virtual person was added to the application with a voice, and storyline[16].

#### **2.3.5. My Digital World**

At the Arizona Science Center an augmented reality element was introduced by Total Immersion to the exhibit "My Digital World". The exhibit shows how everyday items can be used in order to show the possibilities of technology.

A stationary screen was put up and a user can move fiducials about on a surface and see what happens on the screen. The object was a virtual puzzle of video screens, where each side of a fiducial represents a video. To look at a video users had to place them together in the correct order[32, 33].

#### **2.3.6. Websign**

Websigns are virtual beacons that are deployed in space, and time. The purpose of Websign is to guide intended visitors to a specific location. Websigns are connected to a resource located on the web. Depending on the location of the user different Websigns are displayed. A proxy server is needed to connect the device to a server from which information can be retrieved about interesting Websigns in the area[34, 35]. Physical locations are marked with tags that are stored by a server, the server then send the information associated with a virtual beacon to the user's device.

#### **2.3.7. Augmented Reality Browsers**

An augmented reality browser is an application that displays location based information in a browser. Such a browsers have been implemented in hand held devices, and have become widely available with the introduction of smartphones[36]. These browsers enable user's to add content to enrich the user's experience.

There are three mayor browsers on the market: Layar, Wikitude World, and junaio. They are all very similar to each other. Each of the browsers lets the users create content and add layers that users can share to other users.

Layar[37] uses something called layers and offers support for 3D models, animated POI, auto triggers, and image recognition. It works for several operating systems (Symbian, Android, and iOS). Layar also offers navigation through Google maps to the different venues[38, 9].

Wikitude World[39] uses something called 'world' that users can share. A world is created in XML, hence no advanced knowledge of programming is needed. Google KML or ARML (see Section 3.3) can be used to create worlds[9]. Wikitude shows where objects are when looking around and indicates how far away the user is from the object.

Junaio[40] for Android and iPhone smartphones. Junaio offers natural-feature tracking and marker tracking. Junaio uses channels as layer. Junaio can be used to create an Augmented City to help users with travelling, shopping, finding museums, and much more.

---

## 3. Navigation and Interaction

In this chapter we will look into navigation and how humans and smartphones can interact with each other.

### 3.1. Interaction with smartphones

What types of interaction can a user participate in? In this section we will look at the different types of interactions that are available between humans, and smartphones.

#### 3.1.1. Haptic

Haptic means the sense of touch[41]. All smartphones have the ability to vibrate when commanded to. The most common use is to notify a user of a new SMS (short message service) message arriving or when there is an incoming call to a mobile phone. Most phones have the possibility to create complex vibration patterns that could be used for example when navigating around in a city environment. With complex patterns users need to understand the meaning of different patterns.

#### 3.1.2. Movement

Many smartphone can detect movement[42]. This is usually done through an accelerometer that can identify different types of movement: up, down, left, right, forward, and backwards (or a combination of these). This makes it possible to turn the smartphone, when the phone detects in which orientation the smartphone is held the software can react by rotating the orientation of the screen, turning off the audible ring signal, etc.

#### 3.1.3. Geographical location

It is possible to use the smartphone as a compass if the smartphone has a magnetometer. Although this method could be tricked by strong magnetic fields it should give a general sense of direction. Knowing the phone's orientation in space can be used to help the user find the correct direction to look through the camera[43].

A alternative method is to use the built-in Global Positioning System (GPS) receiver that is increasingly commonly in smartphones. The GPS system utilized a network of 24 satellites with different orbits around the world. There are always atleast four satellites overheard at any given point in time. Four satellites are needed to compute the device's correct GPS coordinates[44]. With four satellites it is possible to determine the device's longitude, latitude, altitude, and precise time.

#### 3.1.4. Sound

Smartphones can play sounds, melodies, or music. This can be used to notify the user[45] or for interacting with the use. Notifying the user of an important event, can be important to an

application. With sound it is possible to get the user's attention, even when they are focused on something else or to indicate that a specific event has occurred.

## **3.2. Navigation**

There are several ways that navigation can be assisted by a smartphone. All smartphones today have a GPS receiver that can provide geo-spatial coordinate information that can be used by applications. There are two main methods of using this information that are relevant to this thesis project: navigation with maps, and with augmented reality navigation. This section consider these methods.

### **3.2.1. Maps**

There are several ways that navigation with maps could be done. For example turn-by-turn directions, and directional navigation help the user get to the places they wish to go to. A deeper look into the relevant concepts are presented below.

### **3.2.2. Directional Navigation**

Showing a user where they are on a map and indicating where different POIs are located on that map, helps the user to know in which direction to move and how far to move. By giving a general direction for the user to move in the user can choose their own path, rather than being forced to walk along a predefined path. For example Layar, (see Section 2.3.7) uses similar technology, but uses augmented reality to indicate the direction that a user should move.

### **3.2.3. Turn-by-Turn Navigation**

Turn-by-turn navigation technology is the most common way of navigation of a car. In this approach the navigator tells the driver by voice when to turn and in which direction to turn. The voice directs the driver along the correct path. The turn information is usually also displayed on a monitor to increase the chance that the user will drive in the correct direction[46]. A similar display of turning information could be used together with maps in a smartphone enabling the user to get the same support as with a car navigator.

## **3.3. Augmented Reality Navigation**

There are an two standards that are used for displaying positioning data. In this section we will briefly describe each of them.

### **3.3.1. Google KML**

Google KML (Keyhole Markup Language)[47] is used together with Google Earth[48] and other tools to add information to geo-spatial displays. KML makes it possible to add your own locations of interest to Google Earth, Google Maps, and other tools. Google KML offers many options and makes it possible to add a variety of information (including Google Sketchup

rendering of buildings or other entities). The KML file is viewed with Google Earth, Google Maps, or other software and the KML file can be hosted on a server.

### **3.3.2. Augmented Reality Markup Language**

Augmented Reality Markup Language (ARML) allows developers to create contents that can be displayed in mobile browsers[49]. It builds on the tags from KML, but is a slimmer version of Google KML. Wikitude uses ARML as its native standard for data. ARML allows turn-by-turn augmented reality navigation (see Section 3.2.3) and can include a POI list. ARML is the latest standard for augmented reality displayed data[50, 36].

## **3.4. Navigation with maps**

This part will describe two available navigation/map systems that can be used and integrated with the application.

### **3.4.1. Google Navigation**

Google Navigation[51] makes it possible to start an application that is part of the standard software that comes with Android. Google Navigation is a complete working navigation tool that can be implemented in an application. Google Navigation is an internet based navigation system.

### **3.4.2. Open Street Map**

Open Street Map (OSM)[52] is an open source editable map. OSM allows applications to use their maps for non-commercial applications. It is possible to export maps with POI and as images to be downloaded on the computer.



---

## 4. Method

This chapter describes the methods that were used in this thesis project, specifically: design methods, work methods, and usability testing. The chapter also motivates why these methods were selected, and how they were implemented. The tools that were used for completing the project will also be described. The chapter also describes how using these methods will enable the project to reach the goals that have been set.

### 4.1. Design Methods

This section describes which design methods were used to create and complete this project. In particular this section will describe how the ideas about the user evolved, the design of the low fidelity & high fidelity prototypes, and the concepts of usability testing, and usability measurements.

#### 4.1.1. Users

One of the earliest design considerations was how to represent one or more users within the system. The first method considered was personas[53]. Personas means that several virtual users would be created for use by the application. Personas are created at the beginning of a project, so they are part of an project from start to end. These personas are created based on interviews with potential users. Each persona was created as if they were an individual. This means that each persona has a name, image and personal information (family status, work, characteristics, etc.). During the development of the project the developers talked about these personas by using their first name, rather than calling them "User 1" or using some other artificial naming scheme. This perspective makes it possible to remember several different types of users with different needs. Personas are usually used for creating systems for specific user groups.

Reasons for not using personas include the effort and time required to capture the state of mind of the users with regard to augmented reality, and navigation. Creating each persona takes about a week. Unfortunately, this was considered too long period of time for this thesis project. This has to be considered in the context of the time to create multiple personas and the time required to develop a complete application, and the time required to evaluate this application.

Instead of using personas it was decided to do shorter interviews with users to grasp the most important aspects that these users could communicate regarding the application. Using the interviews as concept from personas for creating user group from those interviews. Additionally, user tests were conducted during each of the different stages of the project.

#### 4.1.2. Low-fidelity prototype

Low-fidelity (low-fi) prototypes[54] are created in an early stage of a project. The reason for creating such a prototype is that it should be obvious that it is in the early stage of development. When creating low-fi prototypes it is extremely important that the prototype is created in such

a fashion that the user can not mistake the prototype for an almost ready product. The low-fi prototype should be created without the use of digital media. It is better to draw the prototype, for example to create a paper prototype. When creating a paper prototype for mobile devices it is important to create the prototype with the same proportions as the device.

The benefits are that test subjects understand that the prototype is in an early stage, thus they do not mind giving a lot of feedback. It is also inexpensive (and cost effective) to make changes to the design at this stage since the development has not yet been commenced. The drawbacks are that it can be hard for users to understand the concept of the design. Using paper prototypes of applications that are location based and whose behaviour depends upon physical motion can be problematic. It is hard to illustrate how a map would rotate with a low-fi prototype or how augmented reality works when using paper prototypes.

One way of creating low-fi prototypes is to create storyboards[55]. Storyboards are used for getting an understanding of the flow of the application and eliciting user requirements. What happens when a user pushes on a button or does something special? The storyboard can make it clear if a design is missing some important aspects. The story board can help ensure that the purpose of the application or device is clear. Storyboarding originated from the film industry, but has also been moved into software development. 3D printers have become inexpensive and is an alternative to use for creating low-fi prototypes. With a low-fi prototype created with a 3D printer it will not be possible to use the prototype to illustrate how an application works. The purpose would rather be to show how the physical parts of the prototype would function.

### **4.1.3. High-fi prototype**

High-fidelity (high-fi) prototypes are created in a later stage of the project. Such a prototype is used to demonstrate a near complete version (or look alike) of the final product. The prototype consists of a dynamic and usable prototype. If a user can interact with a prototype in any way that substantially exceeds the possibility of a low-fi prototype, then it should be defined as a high-fi prototype.

A high-fi prototype could be created either through a visualization tool (such as Microsoft Office PowerPoint), that allows a user to press on different items causing the prototype to react (for example moving to a new page of a PowerPoint file). An emulation could be used to enable a person to virtually control the device, while seeing examples of what screens would appear or by building a prototype (beta version) using programming or other media. This means that a high-fi prototype can be manipulated. In many case, it is often possible to interact with a high-fi prototype in advanced ways, e.g. actually moving a object or pressing a button.

For hi-fi prototypes it is also possible to create augmented reality (AR) storyboards[56]. With the smartphone's interface images are captured and a storyboard is created using some tool. The user can test the storyboard with their own computer using blocks of information (similar to fiducials or other markers) the user can test the storyboard.

Virzi, Sokolov, and Karis have shown that a high-fi prototype and low-fi prototype are equally effective at finding usability problems[57]. In this thesis project these two types of prototypes will be used to find usability issues regarding the application. However, these different prototypes will each have a different focus. The low-fi prototype is mainly used

to see how the user interacts with the application, specifically what problems might occur during the basic part of the application (in this part of the application augmented reality and navigation are not the focus of the application). For the high-fi prototype the focus is on how well the augmented reality and navigation parts of the project actually work (in this case that part of the application is only of interest if problems arise).

## 4.2. Usability testing

In Section 1.1.4 the definition of usability was explained. This subsection will go into more detail about usability testing.

For each prototype a user test was conducted. The reason for conducting usability tests is to find usability problems[10]. After the test the problems that were identified should be addressed to improve the system. There are different usability tests that could be conducted: **think-aloud test**, **real system**, and **prototypes and mock-ups**. A **think-aloud test** occurs when a test user is given access to a real-system or a low-fi prototype, and asked to say what they think about while using the system. The test user is asked to describe what they are doing and why. For **real system** testing the user tests the complete (or almost completed) system while trying to solve/perform different tasks. **Prototypes and mock-ups** are actual tests of low-fi and high-fi prototypes. No system will reveal no usability problems, but high usability could be observed - even with extensive testing. The more extensive the testing and the greater the diversity of users the greater confidence that the device/system really has high usability.

### 4.2.1. Conducting tests

The ideal test team would consist of two to three people that have different tasks: a *facilitator* who talks to the user and leads the test. A *log keeper* that writes down everything that the user does, including the problems encountered and what the user says. A third person would be able to help with either of the other two roles when needed. It is important that all members of the team understand the system and note where the problems arise. An alternative would be to use a camera where the log keepers task is to document the test through a recording.

**Planing the test**[10] consist of several parts. Whom will be the test subject for the test? The *test subject* should be actual potential users for the system.

In what context will the user use the system? *Test tasks* should be created so they are realistic. The test tasks are ideally based upon giving the test subject a real problem to solve using the system. A test task should be formulated such that the test subject would think about performing this task when put in the same situation. The test subject should get the same information as the actual user would when in the same situation. Thus the task should not contain information about how the system works or any other technical information (if this would not be provided as part of the solution). The test task should be meaningful, thus the test subject should accomplish something through completing the task. No hidden help should be part of the description of the test task.

When the test is conducted it is important to explain the reason why the test is done. This is important so that the user will not be afraid or nervous about doing something stupid. Thus an

introduction should be held during which the test task and system to be tested are explained that the test is being conducted to find flaws in the system, rather than testing the user.

#### **4.2.2. Measuring usability**

There are different ways of measuring usability through tests. One way is to focus on the different factors (explained in Section 1.1.4 and in Table 1), and how well the system performs with respect to these factors. One way would be to simply count the number of usability problems that are discovered. An alternative way would be to consider how long it takes to complete a specific test task. This method could be used to discover if a certain task takes too long to complete or if there are other problems that prevent the test user from completing the task. When measuring usability it is important to set up specific requirements. The requirements are used to understand if the device/system is good enough to be used by the target users as a real product. In terms of test task execution time this might mean that 15 of 20 users have to complete a specific task within a specific time in order for the system to be approved. The exact number of test subjects, proportion of test subjects that must complete the test task within a specific time period needs to be considered thoroughly. This means that these values should not be taken out of thin air. How accurate does the system test results have to be? These questions need to be answered so the numbers of test subjects and testing parameters can be established. The usability factors that can best be covered with a time bound task test are: ease of learning, task efficiency, and ease of remembering.

As mentioned above, one method is simply to count usability problems, this method suggests conducting a think-aloud test. This is a more qualitative method rather than the quantitative task time test. In a think-aloud test each test subject is asked to complete a task, with the test subject telling the test team what the test subject is thinking. All problems that occur are written down, no matter how severe the problem is. Later these problems can be stratified into different categories: critical problems (test subjects can not complete the task, or the user gets annoyed), medium problems (several users have the same small problem), and small scale problems (these problems are overcome with little difficulty). The requirements could be that two out of three test subjects can complete the task (without encountering a critical problem), and that at most three medium problems are found out of all of the problems that were found. These numbers also should not just be taken out of thin air. The number of test subjects should be considered. However, the cost (and time) limits often reduce the practical number of user tests that can be performed. The number of problems that occur should be considered proportionally to the actual system purpose and its intended usage. The usability factors can be evaluated in terms of the problem count: fit for use (testing raises the importance of choice of task), ease of learning, and ease of remembering. Subjective satisfaction is not perfect as an indicator of usability, but it is possible to get an understanding of if the user likes the system or not. Furthermore, understandability evaluation would reveal if the test subject understands what happens, but this can not be measured with this method.

An alternative method is to conduct an opinion poll. The user is asked through a questionnaire or other tool what they think about the system. It is recommended to use the

Likert scale[10]<sup>2</sup>. In a five level Likert scale the alternatives are: Strongly disagree, Disagree, Neither agree nor disagree, Agree, and Strongly agree. Each answer is encoded as a value. The questions can include: How easy is the system to use? This method requires that the system have been used for some time by the users (so that they can form opinions). The requirements determine the questions, and requirements are based upon the expectations of the system. The usability factors that work well with opinion pools are: Fit for use (specific questions about function can be asked, or test subjects can identify what functions are a problem) and subjective satisfaction (just asking what the user thinks about the system, and if they would recommend it to others). Many of the other factors can seem to give good response, but might not be accurate, and can thus not be trusted (e.g. the user thinks the system is easy to learn now that they have learned how to use it).

A way of finding out how well the test subject understands the system is to ask what they think the system does. The measurement is how accurate an answer the test subject gave. This can only be used to measure understandability.

When conducting usability tests on big scale some sort of completion rate[58] should be used. If 8 out of 10 users manage to complete the task this does mean that 80% of the population can use the system without problems. In this thesis project this will not be used since the focus for this study is not to test the system for a broad user group but to identify usability issues.

In this thesis project think-aloud tests were conducted with the use of test tasks (this is explained in further detail in Chapter 7). This test method was used on the two prototypes, and later on the real system. As stated earlier the focus of this thesis project was on **Subjective satisfaction, Understandability, and Ease of learning**. These factors are measured through a combination of: opinion polls (a slightly modified version), system understandability, and problem count. Task time testing is not used due to the fact that for this thesis project how long it took the test subject to perform different tasks was not a focus.

## 4.3. Work Method

This section describes what methods were used during this thesis project.

### 4.3.1. Brainstorming

Brainstorming[59] is a method that is used to collect ideas about a specific problem. During a brainstorming session the group should: not criticise any ideas (no matter how crazy or impossible they are), produce as many ideas as possible, combine ideas, think aloud, and write down everything. After the session a discussion about the ideas should take place. Sometimes the most crazy idea may be the best solution.

During the start up meeting a brainstorming session was carried out. A lot of ideas were discussed. Technical restrictions were investigated in the next meeting where the main functions were decided. Following this second meeting the project could be started.

---

<sup>2</sup>The Likert scale is a grading system where each answer represents a value (often 1 to 5), these values can be used to evaluate the answers.

Inspiration was taken from similar projects, and augmented reality browsers were used for the brainstorming session.

### **4.3.2. Scrum**

Scrum[60] is an agile software development methodology. Scrum is used for developing software with varying focuses. Scrum is a way of executing a project to produce a good final product with the promised features.

Every project that uses Scrum has a team. The team consists of: the Product Owner, the Scrum Master, and a production team. The product owner is the person that ordered the project. This person knows what the project is supposed to accomplish. The scrum master's assignment is to keep the team focused on their tasks, and that Scrum methods are used as intended. The team is recommended to be between five to nine persons with different skills. The team is self lead, and responsible for delivering the product.

A sprint is a period of time that the team works with the project. Usually this period is several weeks. Before a sprint start a sprint meeting is held. During this meeting the goals, and objectives for this sprints are defined collectively by the team, product owner, and scrum master. The objectives and goals are divided into smaller tasks. The team then estimates the amount of time that will be needed for completing each task. Each task is written on an index card with an ID, description, goals (what that needs to be fulfilled for the task to be considered completed), estimated time, and priority. The priority is set by the product owner. It is important that all dependencies are known so that the priorities are correct.

During a sprint, scrum meetings should be held every day within the team. During each meeting the team members each stand up for 15 minutes; talks about what they have done, what they need to do, and if they need any help.

In the place where the team sits a taskboard on the wall shows all the tasks, with their priorities. These tasks are grouped into tasks that are important, less important, and not important. The prioritized tasks are in one column, sorted into rows: not started, started, and completed. It should also be obvious whom in the team that is working on each task. A task is completed when the task's goals are fulfilled.

A product backlog is created listing functionalities, goals, objectives, and other aspects that could be of interest. The person responsible for the product backlog is the owner of the project who (as noted above) prioritizes the different objectives. A sprint backlog represents each sprint, while the product backlog represents the entire project.

### **4.3.3. Scrum Implementation**

The reason for using Scrum in this project was partly that Attentec AB uses it as work method and that the Nobel Museum preferred that work method. But there were problems implementing Scrum. The main reason was that the application was developed and evaluated by a single person. This subsection describes how Scrum was used during this thesis project.

Meetings were held once a week during the project. The meetings were held mainly between the project owner (a representative from the Nobel Museum) and the team (in this case, the author who carried out this thesis project). The scrum master (a representative from

Attentec AB) attended meetings during the start up, and approximately one meeting each month during the rest of the project. No scrum meetings were held with the production team. During the meetings the discussions focused on what would be the prioritized functions, and what would be tasks for the next meeting.

No taskboard was used during the project, but the tasks were written down and updated in the backlog. The thesis project was divided into three sprints. Each of the sprints had different objectives and goals. The sprints were each a few weeks long (see Table 2).

Sprint	Description
Sprint 1	During this sprint the objectives were to evaluate, and work with the future users. Additionally a low-fi prototype was created, evaluated, and improved. A goal for this sprint was to learn more about augmented reality, and available frameworks.
Sprint 2	This sprint was for creating the high-fi prototype, with only the necessary functions for the application to work sufficiently. Additional goals were to learn more about the chosen framework, and learn the Android syntax.
Sprint 3	During this sprint an evaluation of the high-fi prototype needed to be done, and suggestions for improvement for the final product.

Table 2: Scrum Sprints

## 4.4. Tools

This section presents the different tools that were used during the project.

### 4.4.1. Eclipse

Eclipse is an integrated development environment that is used for programming[61]. To develop Android applications with Eclipse an Eclipse add-on[62] is required so that an Android smartphone can be used for development. The add-on also provides the necessary libraries. Eclipse offers support for debugging applications running on the phone. Being able to debug on the smartphone is important when developing Android applications. Eclipse enables the programmer to use the libraries necessary for QCAR ([63] describes how to set this up). It is possible to integrate Eclipse with the Unity 3D software (the integration process is described in [64]).

### 4.4.2. Unity 3D

Unity 3D[65] is a 3D graphics engine that makes it possible to manipulate objects and place these objects where they need to be on a reference image. Unity 3D allows objects to be added, created, and manipulated. Unity 3D projects can be programmed and developed to do different things depending on what is needed by the application. The Eclipse framework offers

a plug-in for Unity 3D projects. Details for this plug-in can be found in [64]. To communicate between the Eclipse project and the Unity project JavaScript or C# scripts had to be used.

#### **4.4.3. Mono Develop**

Mono Develop[66] is a programming environment integrated with Unity 3D. The environment is used to create, and handle all the JavaScript and C# that where needed through out this project.

#### **4.4.4. SQLite**

The application uses a database for all the data that is needed by the application. This database is implemented using SQLite. The database is created before the application. This enables the application to utilize the information stored in the database. Further details about SQLite can be found in [67].

#### **4.4.5. Cygwin**

Cygwin is a console with Linux-like commands. Cygwin is used for debugging, sending data to, and from the Android phone. Cygwin operates in the background when Eclipse interacts with the smartphone. Further details about Cygwin can be found in [68].

## 5. Evaluation of the Underlying Technology

This chapter will discuss the decisions that were made concerning the technical aspects of the project. In particular the chapter will discuss augmented reality frameworks, navigation, and tests on what camera to use to collect reference images.

### 5.1. Augmented Reality Frameworks

As described in Section 2.2 there are several different frameworks that could be used for augmented reality for smartphones. When choosing which framework to use, the fact that Lara used QCAR was an important consideration - as we wanted to build upon the Lara functionality. However, it was not obvious that QCAR should be the framework, hence an evaluation was conducted to select the most appropriate framework.

Because we had selected Android as the operating system String was eliminated (as it only runs on iOS). The Unifeye Mobile SDK offers many features that are of interest for this project, but this SDK did not offer support for GPS and compass with Android. Although GPS and compass functionality was seen as important to have even in the start up phase of the project, it was not clear how the different design decisions would utilize these functionalities. The lack of GPS and compass functionality lead to discarding the Unifeye Mobile SDK. Wikitude SDK is a framework primarily used for augmented reality browsers. The application was not supposed to be created as a browser based application, but as an application with augmented reality that functions when at a specific location. Wikitude was consider for navigation, but not as the augmented reality solution for the application due to the weaknesses of this SDK with respect to natural-feature tracking.

The decision was to use QCAR, partly because Lara used QCAR, but also because QCAR offered good natural-feature tracking, and advanced 3D rendering. The functionalities that QCAR offers met the needs of the project, specifically was a framework that supported Android, had natural-feature tracking, offered the ability to render 3D objects into a view, and supported different ways of tracking objects. QCAR offered all these functionalities. Additionally, QCAR supports iOS - a potential platform for implementation in the future. For all these reasons QCAR was chosen as the framework to be used for the project.

### 5.2. Navigation

This section will describe which solutions were considered, and what solution was selected the navigation aspect of the application. The alternatives that were discussed were: navigation with augmented reality, maps without navigation (only showing users position and destination position), Google Navigation, and a radar screen.

The augmented reality solution was quickly dismissed. The main reason was that it would not be very user friendly for a user who is not acquainted to Stockholm to look in the correct direction without not knowing where the POI is located. Forcing a user to turn around in order to look in all different directions every time they what to find a POI might lead to the user quickly loosing interest in the application. Augmented reality browsers uses this technology, but they usually have several POIs that are of interest for the user - rather than a

target consisting of a single POI. As a result augmented reality was not considered to be an appropriate solution.

The radar screen was compared to the map with, and without navigation. This comparison was conducted during the paper prototype user study (see Section 7.1.1). During this user study it was clear that users trusted a map solution more than a radar screen. Some of the users during the test did not think that navigation was needed if a map was present, if the user's current position was displayed on the map.

As a result of these above evaluations Google Navigation was initially selected as the navigation tool, because it was easy to implement and it was easy to include as a part of the application with out a major effort. The problem was that the map is located on the internet - hence using this solution would require the device to have Internet connectivity. There are solutions for this programmatically but that meant writing over Google Maps saved cached application data (which should be avoided). Google Navigation still has no real offline mode which means that users still have to connect to the internet when calculating how to walk (unconfirmed rumors say that offline mode for Google Navigation should come sometime during 2011).

OSM offers a way to download maps locally to the device and adding them to the application, as images. The decision fell on using OSM as a temporary solution until Google Navigation is released in a true offline mode.

### 5.3. Reference Image

The reference image that is used needs to be of good quality. This raised the question of: Does the quality of the image depend upon the camera that takes the photo? To examine this question two cameras were used to test if the choice of camera determines how well an image was tracked. The cameras that were used were the built in camera in the HTC Desire and a Kodak M1033. The Kodak M1033 is a 10 megapixel HD digital camera with 3x optical zoom[69]. The HTC Desire has a 5 megapixel camera with autofocus[70].

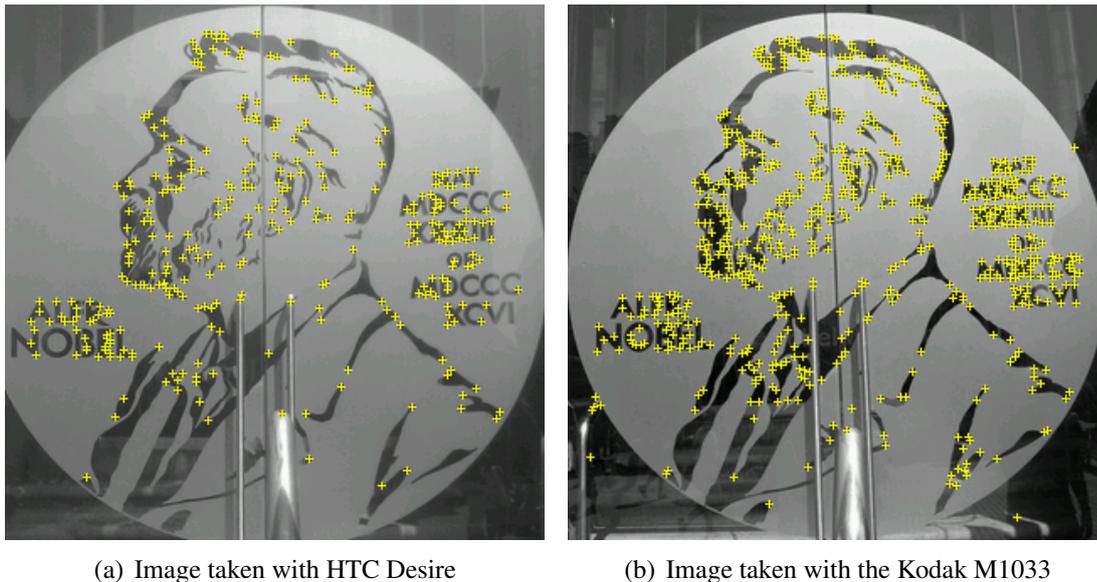
The reference images used for testing were three objects from the Nobel Museum; a model of "Börhuset" (were the Nobel Museum is located), the entrance to the museum, and a illustration of Palytoxin. It was harder to acquire a good image of small objects with the HTC Desire than with the Kodak camera. Additionally, the Kodak camera handled darker places in a better way than the HTC. The original unaltered images are included in Appendix A.

After uploading a reference image to Qualcomm's homepage each reference image is graded. This grading is based on how many angles can be located in the image. The grades are represented with stars where one star represents not working very well, while five stars means that the object can be tracked very well. The three objects got different grades as shown in Table 3.

As can be seen in Table 3 the images from the Kodak camera got consistently better ratings. To illustrate the differences between how many angles were located in each image see Figure 3 (the remaining two pairs of images can be found in Appendix B). Looking closer at the images shows that the image taken with the Kodak camera resulted in more angles.

Image	Camera	Rating
Model of Börshuset	HTC	2
	Kodak	3
Entrance to Nobel Museum	HTC	1
	Kodak	2
Palytoxin	HTC	1
	Kodak	3

Table 3: Number of stars given to the different reference images



(a) Image taken with HTC Desire

(b) Image taken with the Kodak M1033

Figure 3: Comparing reference images

An important aspect of comparing the two cameras is to see if the application can quickly identify an object and how well the object is tracked. This test utilized the reference images of the same objects, but with the reference images from the two different cameras. The test was conducted to learn if the users would experience any difference in behaviour due to the choice of reference images.

The test was conducted at the Nobel Museum. No users participated in this test, the author made a qualitative decision of which set of reference images worked better. It was clear throughout the test that objects were identified more quickly, and tracked better when using the Kodak M1033 reference images rather than the reference images from the HTC Desire. An object took about half a second longer to be identified when using reference images from the HTC Desire. The smartphone was held upright. To test how well the image was tracked the phone was rotated and moved so the angle of which the phone changed. The image from the HTC Desire was sometimes identified at an angle of 45 degrees while the Kodak could even go further before losing the tracked image.



## 6. Implementation

This chapter describes how the idea of who are the users has developed through the project. The paper prototype and high definition prototype will be described, along with the development that was carried out and what has been done to develop the beta prototype.

### 6.1. Users

This section describes how the target user has evolved during the project.

Information from the Nobel Museum indicated that the user was: a man, 20 to 35 years old, acquainted with Android, limited knowledge of Stockholm, and with basic knowledge of the Nobel Prize. Read more about the information given by the Nobel Museum in Section 1.4.1 on page 4.

To get a better model of the user three interviews were conducted. These interviews were used to evolve the view of the target user, and to get a better understanding of the user's needs. The objective was to understand their technological level, their purpose in Stockholm, their livelihood, and what kind of information they are likely to be interested in. The interviews took place at the Nobel Museum. A day was reserved for conducting the interviews, during that day three appropriate candidates who fit the minimal requirements (mentioned in the previous paragraph) and who had time to participate were chosen for the interviews.

The interviewees were students or had just started working and they all had an academic background. The interviewees had common knowledge of technology and good understanding of what a smartphone is and how a smartphone worked. The main purpose of their visit to Stockholm was for vacation and to find out more about Sweden. The information that was of interest was mainly about: the Nobel Prize, Alfred Nobel, the Nobel Prize Ceremony, and the Nobel Prize Banquet.

During the first user test (see Section 7.1.1) it was clear that there were two different states of mind: the competitive and the information seeker. The competitive user's priority was to complete the game, and at a later stage their interest shifted to finding more information about the visited places. The information seeker prioritised learning more information about the location and the objects that were illustrated.

### 6.2. Paper prototype

The paper prototype was created with the help of a flow chart (see Figure 4) of how the application might work. With an understanding of the basic flow of the application each step could be recreated in a paper prototype. The paper prototype was created in the form of a story board.

Creating a basic template of the Android phone was, made it possible to create paper prototypes that were consistent and proportional. Each view was created using the template as a base. Every place that users pressed on (assuming that place was a button) would lead to a new view that was pre-prepared.

Figure 4 illustrates the flowchart of the application at the paper prototype stage. A short description of the flow will be given below. The welcome view is the initial window that

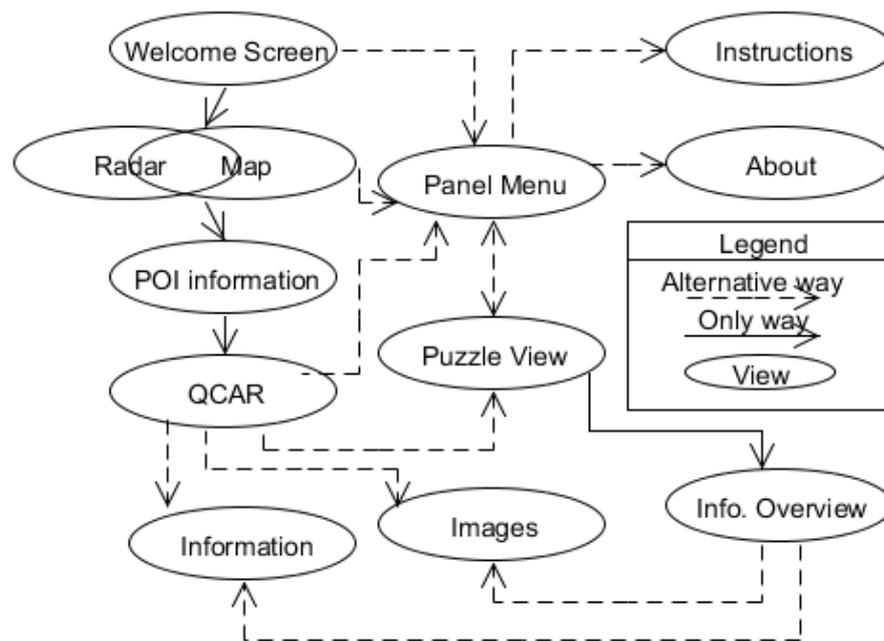


Figure 4: Paper Prototype flow

appears when the application is started. A short description of the application is presented in this window. The welcome view leads to the next view that consists of the map/radar view. When the user presses on a POI, and a pop up was started. From this state the user could go on to navigation. How the navigation was supposed to be done was not decided, hence this was one question for the user study. This was formulated as a choice between of which view the user would prefer (map or radar), see Figure 5. When the user reached a specific location (or virtually reached the location in the case of the paper prototype) the QCAR screen starts up (see Figure 6(a)). Now the user can start looking for special objects and information. In this screen three objects were shown: a puzzle piece, an image frame, and an information frame. It was possible to press on each of the POIs when they were in view. The puzzle piece lead to the puzzle view (Figure 6(b)) that added the found puzzle piece to the puzzle. In the puzzle view a puzzle about the Noble Prize is displayed. When a puzzle piece is located found it leads to an information overview being shown for that location and the screen displays both the image frame and an information frame. The image view displays the images associated with this location as a slide show. The information view displays information relevant to this location.

From every window it is possible to get to the Panel Menu through the menu button on the Android phone (this is the alternative way that is illustrated in Figure 4). The panel menu offers three menu items: Instructions, About, and the puzzle view. The menu item Instructions gives general instructions for how the application works. The About menu describes why the

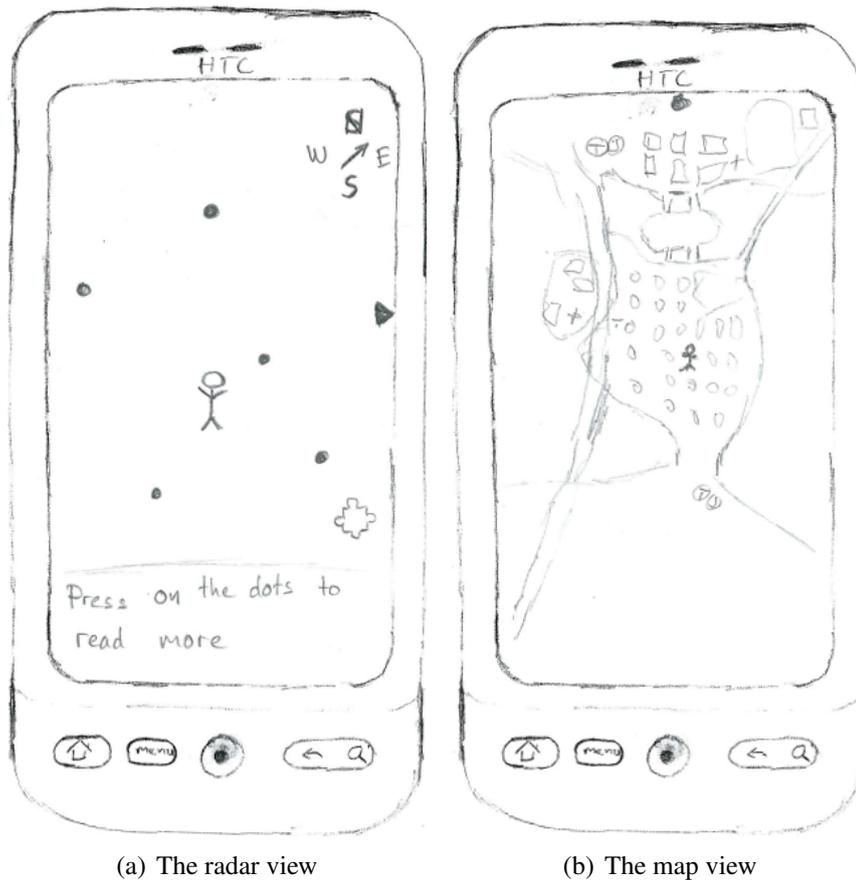


Figure 5: Navigation paper prototype

application was created and other relevant information.

With the flow chart of the application clearly in mind it was possible to draw what would be shown on each screen, what buttons, basic information, and other design issues could be of interest when testing the design. Drawing each view indicated in the flowchart created the first paper prototype. This prototype was then used in a user study and the feedback was evaluated (described in Section 7.1).

After the user studies an evaluation was made to decide upon some improvements that needed to be made. These changes evolved into a new paper prototype. This new prototype was based upon a modified flow chart (see Figure 7).

The views that had a "Back" button were edited, and these buttons were removed. The users could only use the built in back button supported by the phone. This was a suitable solution, as the test subjects mentioned that they would try the back button if no other option to go back was available.

The radar view was removed from the flowchart and instead the map view was chosen for the navigation. The POI pop up was removed, and overview view (Figure 8(b)) was added that enables the user to choose where to go based upon a short description of the point of interest, direction, and distance. This change made it easier for a user to find out where they wish to

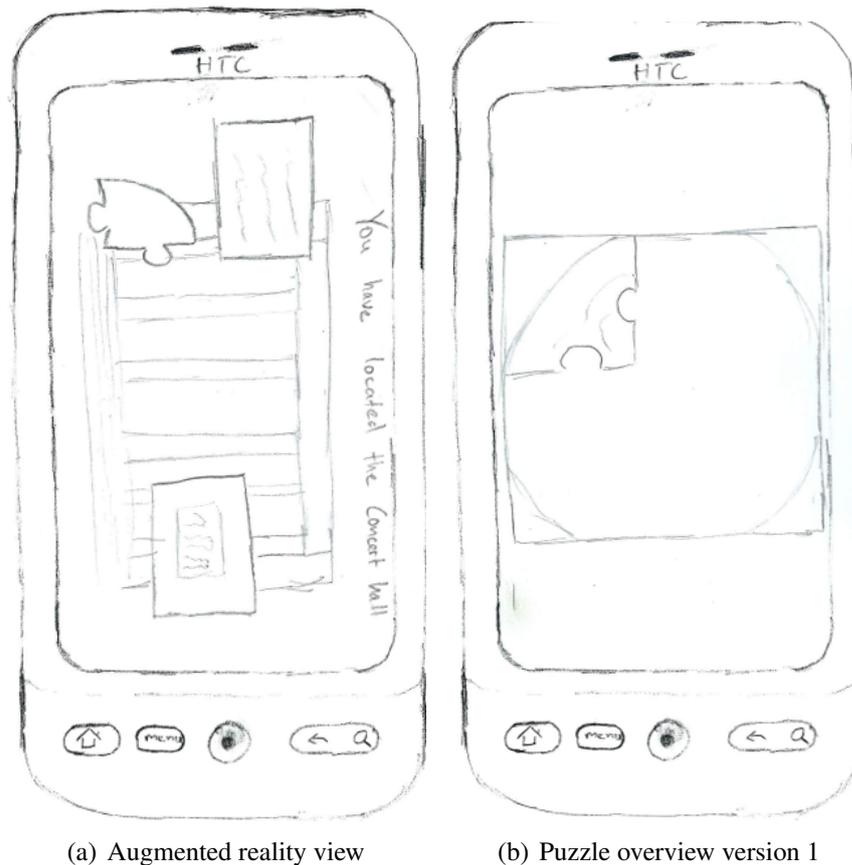


Figure 6: Paper prototype

go, and removed the problems caused when the user had to guess which POI was which. An additional information screen was added as a menu button to display more information about the different prizes. These changes are illustrated in Figure 8.

When a puzzle piece was found two buttons are added: "Find next piece" and "Learn more". These buttons help satisfy the requirements of the two different user groups that participated in the evaluation, these changes are illustrated in Figure 8(a).

### 6.3. High definition prototype

The high definition prototype was constructed to be a prototype of the final product for this thesis project. Improvements from the evaluation will be suggested from the user study of the high definition prototype (Section 7.2) for the final application as future work. The finished application will not be part of this thesis project.

The high definition prototype was constructed according to the evolved flow chart and paper prototypes (described in Section 6.1). The prototype was built on the Lara application (described in Section 2.3.1). Several of the views that were part of Lara could be re-used with some modifications, while other views had to be built from scratch. The views that could be modified from Lara were: welcome screen, overview, and menu. Other background functions

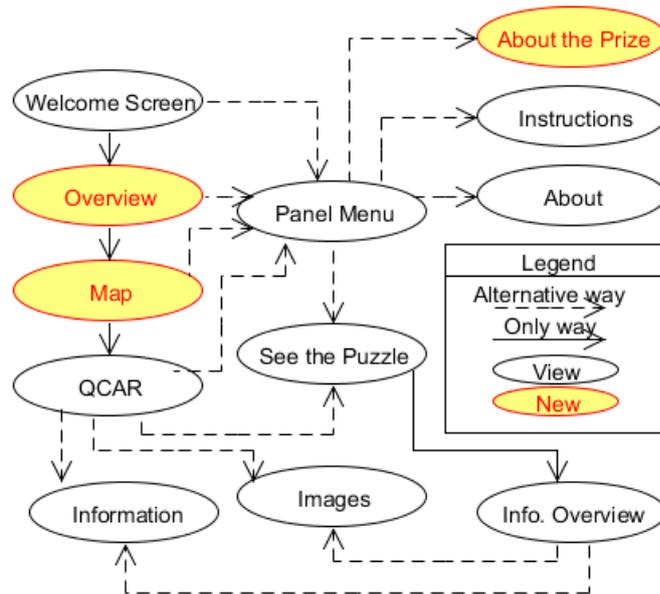


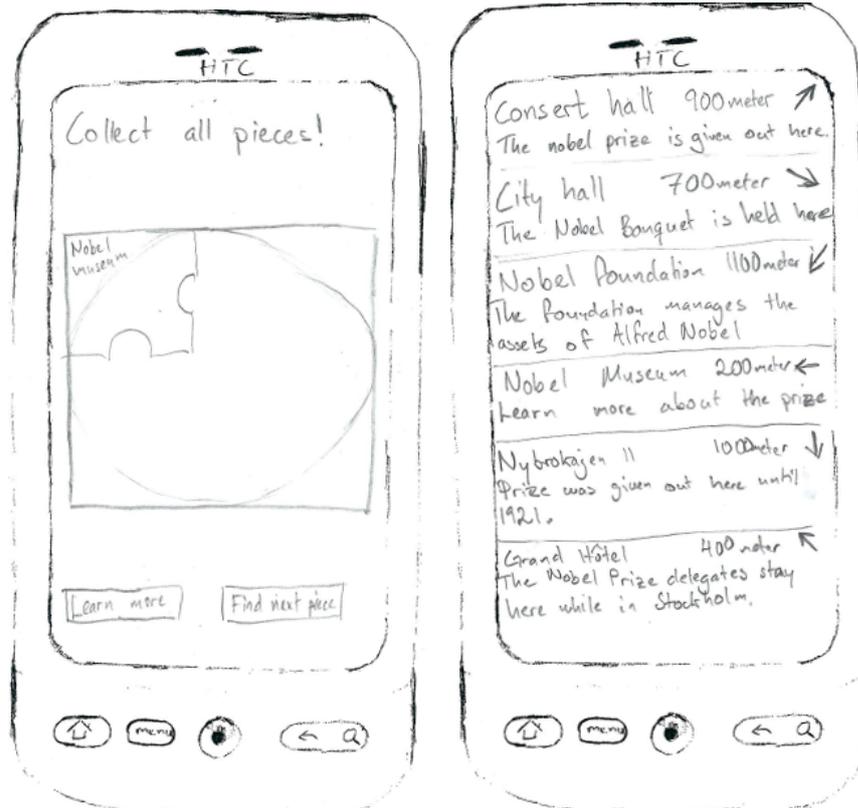
Figure 7: The improved Paper Prototype flow (with the new states indicated in color)

that were re-used with the application include the database and the selection functionality. The new views that were needed were: puzzle overview, information overview, location information view, location image view, and augmented reality view.

There was a function for saving and loading a POI when it was selected. This functionality remains unaltered, and was used in the application for each place where a option was chosen. The database was modified to better suit the information and the relevant data was added. Fields were removed or added depending on what that was needed by the application.

The welcome screen was redesigned to fit the application. The welcome screen was a Nobel medal instead of a start button, a welcoming text, and a brief description of the application. The changes between Lara's welcome screen and the application's welcome screen can be seen in Figure 9. The background color was set to "Nobel Blue" (hex color code #024664), and this color is used throughout the application. The theme for the basic colors were selected after designing the welcome screen, hence the gold color of the medal was used as color for all text.

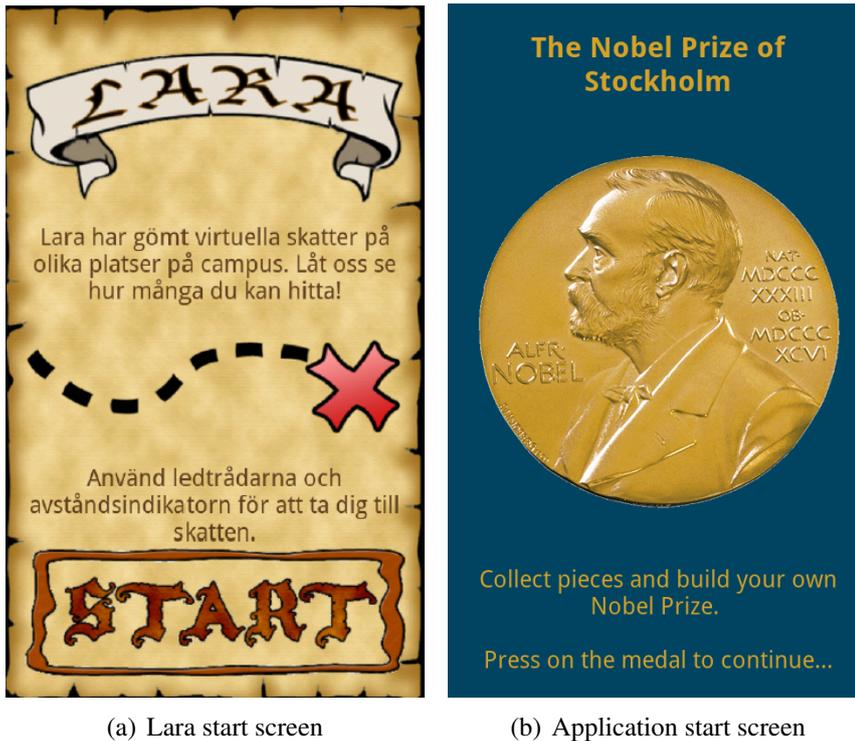
The overview screen was changed in many ways (as illustrated in Figure 10), the original view was a list of locations (which remains). There were several changes made, specifically: the distance and direction to each specific POI was added. The short description for each POI remained. In Lara each row could be found, locked, or be not yet found. This status was illustrated through an icon that changed depending on the status of each item. Each item in the new application list leads to an OSM (Google Navigation when it is released in offline mode) screen with directions to that specific POI. The augmented reality screen reaches activated when a user got to that specific POI.



(a) Puzzle overview version 2

(b) POI Overview

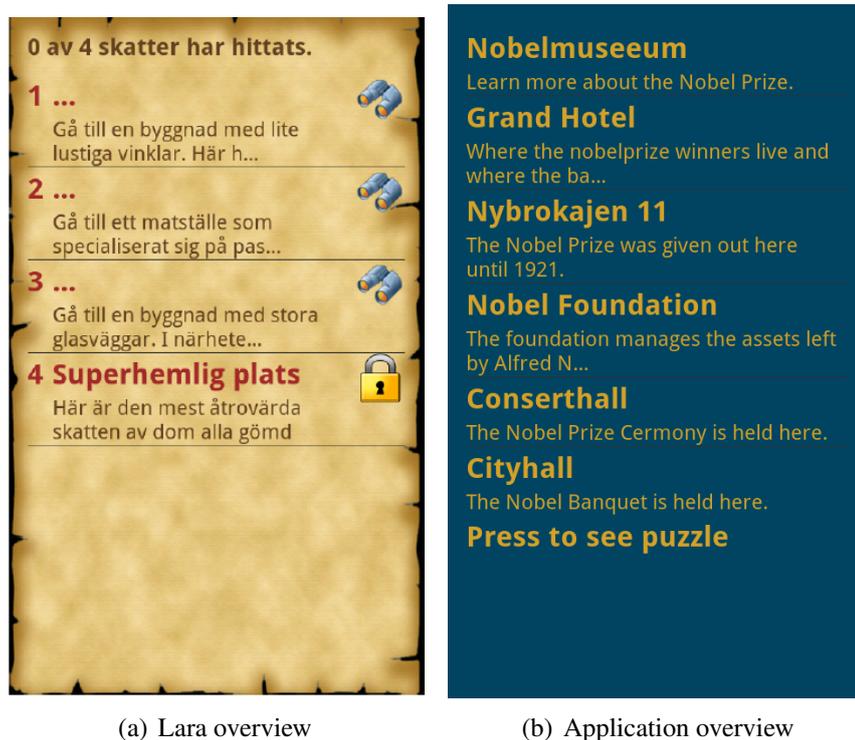
Figure 8: Changes made to the paper prototype



(a) Lara start screen

(b) Application start screen

Figure 9: Start screen comparison

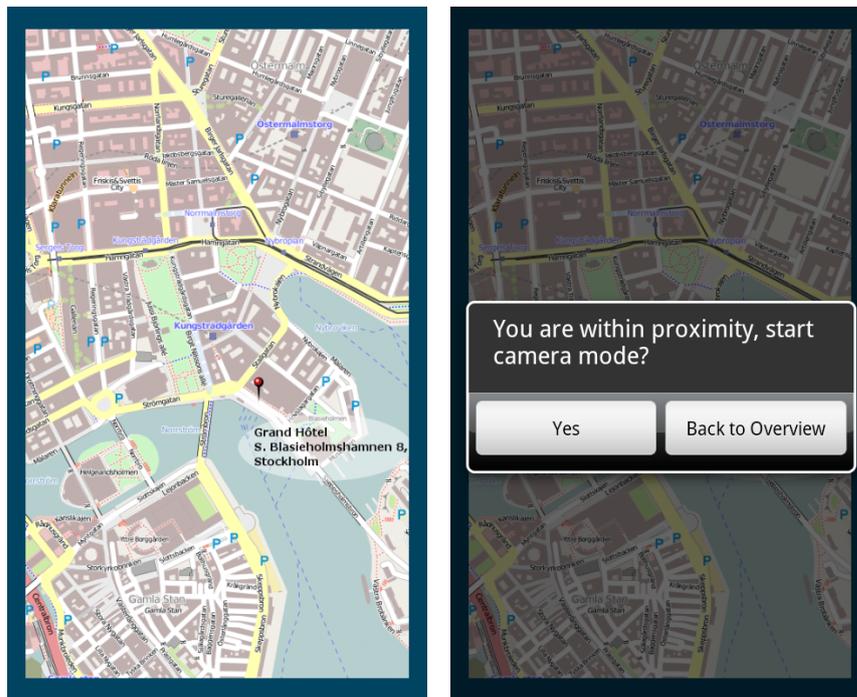


(a) Lara overview

(b) Application overview

Figure 10: Overview screen comparison

The OSM view was created as a static map which was edited and the appropriate POI was added for each view. In the map a small pin was added with address information that would help the user get to the correct place. The initial idea was to add the users location but due to time issues this idea was put on hold. See an illustrated OSM view for Grand Hôtel in Figure 11(a). When the user gets within a certain radius of the chosen coordinates the phone will start to vibrate to notify the user that they now can enter camera mode. This is illustrated in Figure 11(b). When the user presses on "Yes" the augmented reality application is initiated for Grand Hôtel.



(a) The OSM view for Grand Hôtel

(b) The OSM view when the user is notified of being in range of the location.

Figure 11: Navigation

The puzzle overview (shown in Figure 13) was created from scratch. This view was one of the vital parts of the application as all pieces could be located and linked to information i.e., the images that corresponded to each POI (with each piece linked to a specific POI that has been discovered). The image and information views are reachable from both the augmented reality screen and the puzzle overview. To reach these views from the puzzle overview a view containing the information from both the image and information was created. This view is called the info overview (see Figure 12).

A new augmented reality view was created with Unity 3D. Each time that the Unity part is initiated a splash screen is shown with a brief instruction to look for objects (illustrated in Figure 14(b)) and then automatically starts the correct scene for that location. When the scene for that location is loaded a more detailed instruction for what to look for is presented at the top of the screen. The augmented reality view was edited so that the information, image, and

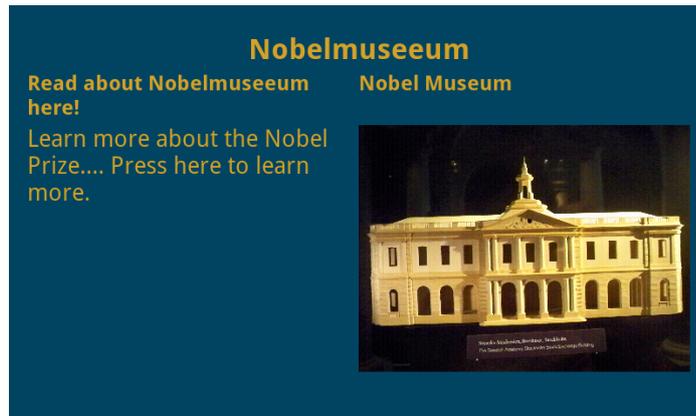


Figure 12: Info overview for the Nobel Museum

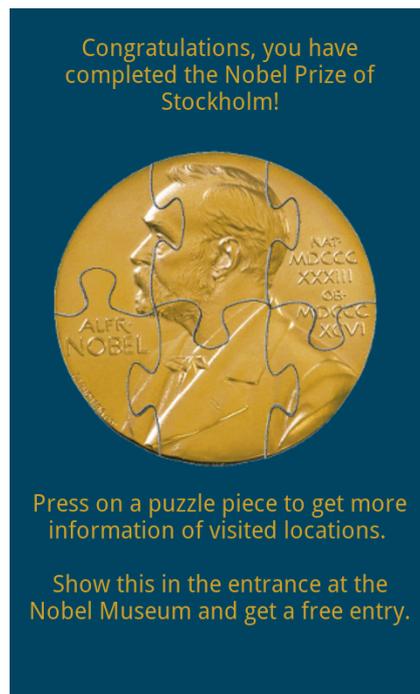


Figure 13: Puzzle overview - Completed

puzzle was added to different locations depending on each view (see an example of this in Figure 14(a)). Note that in Lara an object is always displayed in the center of the view, Lara also had problems with using textures (did not render very well) and did not use Unity 3D. Each item is a button that leads to the appropriate view.

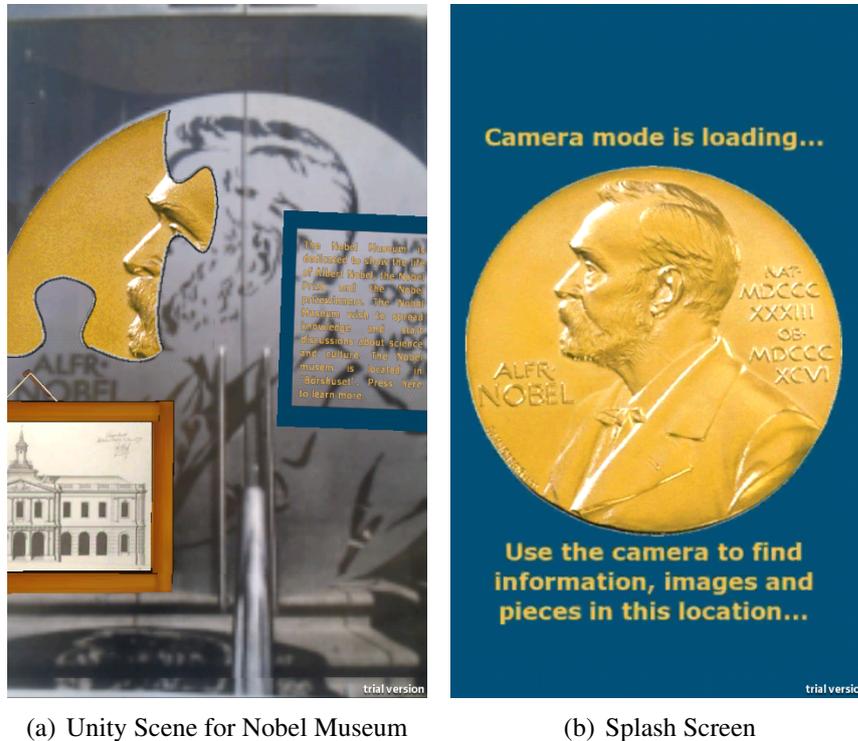


Figure 14: Unity

The Unity project communicated between several pages: getting & sending data, loading pages, checked if the puzzle piece had been located, and to load the correct scene. To accomplish this four C# scripts were created: `load_level` (showing the splash screen and loading the correct scene), `puzzle_found` (checks if the puzzle has been found, removes or creates the object), `puzzle_pressed` (identifies if a button has been pressed and loads the correct view), and `panel_menu` (how the panel buttons work). These four scripts communicate with a class called `UnityPlay` that starts and handles all requests to and from the Unity part of the application.

Known issues of the beta application before the usability test: sometimes the application crashes and restarts when initiating Unity 3D. Some buttons in Unity do not work when pressed for the different POI.

Some aspects that were with in the improved paper prototype (see Section 6.2) were not with in the high-fi prototype due to time issues: distance in overview to POI, direction of POI in overview, and adding the "Learn more" & "Find next piece" buttons in puzzle overview. Navigation was reduced to a map without the users position. These aspects were left for future work.

Changes in the flow of the application (from the paper prototype) consist of adding a control of GPS at the initiation of the application, adding a splash screen, and the view when the user is within distance to the POI. This is illustrated in Figure 15.

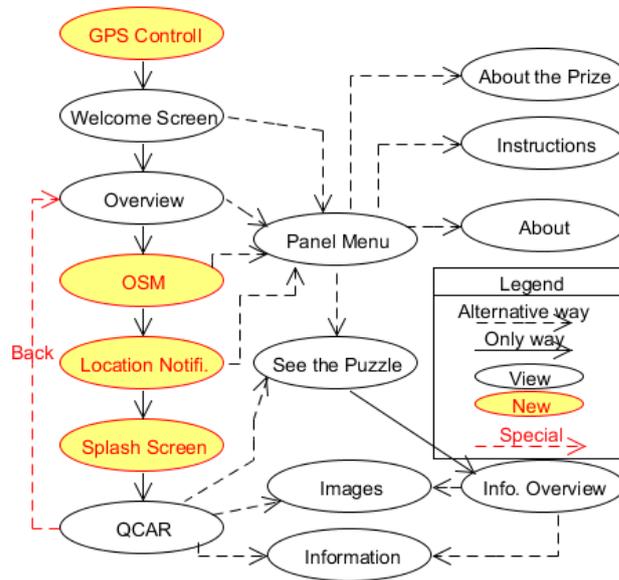


Figure 15: The high definition prototype flow chart (with the new states indicated in color)



---

## 7. Usability Testing

This chapter describes the user studies that were conducted throughout the project. The first user study was a paper prototype (a low-fi prototype). This was followed by a study with the high-fi prototype. For each user test there were several steps: preparing the test, conducting the user test, and evaluating the feedback from each test.

### 7.1. Paper Prototype

The paper prototype realizes a low-fi prototype (see in Section 4.1.2 on page 19). Further information about the paper prototype in Section 6.2 on page 31.

#### 7.1.1. User Study

Task driven user tests were used when working with the paper prototype. The user was given a few tasks in order to find problems with the design. Three user studies were conducted in the Nobel Museum. Users were recruited when visitors to the museum asked for directions to where the Nobel Prize was given out or where the Nobel Banquet took place. The other selection criteria was that the persons was between 20-35 years old and owned an Android phone. The test was done with two test subjects at a time. Two test subjects were selected at a time because it turned out to be problematic finding an individual tourist who was interested of participating. A total of six users participated as test subjects.

During the user test of the paper prototype additional information was required to instruct and explain the navigation and augmented reality views. Augmented reality had to be initially explained as a concept so that the user would have an understanding of the view that was (to be) provided. The navigation page required instructions to indicate that the view would automatically update depending on which direction the user looked, and where the user was located. The dynamic views (navigation and augmented reality) were problematic to illustrate and explain in a good way, this may have lead to problems getting accurate feedback from the test subjects regarding these aspects.

The tasks that the test subjects were assigned were:

1. Locate a puzzle piece at the concert hall.
2. Learn more about the concert hall.
3. Look at images regarding the concert hall.
4. Find out how many puzzle pieces were found.

A few questions were prepared before the tests. The questions that were asked were:

- What type of navigation would you prefer?
- What do you think should be improved about this prototype?
- Do you have any other opinions regarding the prototype?

Follow up questions were asked depending on the test subject's answers in order to clarify any open issues. Using the answers to these questions several important problems were identified, these are examined in the next section.

### **7.1.2. Evaluation**

After showing the test subjects the two different navigation views (map and radar) it became clear that the user trusted a map view more than a radar view. One test subject also complained about the accuracy of the GPS, indicated that they would be more comfortable and a map with an approximate location with a general direction in which to move, rather than turn-by-turn navigation. However, this opinion was not shared by all test subjects.

During testing it was clear that there were two distinct user groups that had to be satisfied. One group of users were interested in learning information while on vacation, while the other group were mainly interested in completing the puzzle. The later group users seemed to be interested in learning more about the location after they were back home.

There seemed to be some confusion about the use of the "Back" button used on some views. The Android phone has a built in back button that could also be used. One test subject said that he would not be confident when using the back button. This was because several applications that he had used crashed when the back button was pressed. However, they thought they would try it if there were no other back buttons available. The main problem was that there was an inconsistency in the use of the back button between the screens.

Some test subjects were also interested in learning more about the Nobel Prize, and they asked for some way of finding out additional information about the prizes.

A few test subjects thought that there should be some more information shown on the puzzle overview, and also some sort of indication of what puzzle piece represented which POI.

When looking at images the test subjects asked for a description of the image. Also some subjects would like to slide their finger across the screen to change image (as this is the means that windows are typically changed in Android).

## **7.2. High definition Prototype**

This section describes the high definition prototype. The creation of this version of the prototype was described thoroughly in Chapter 6. The high definition prototype was built based upon the improved paper prototype with smaller adjustments. Read more about the high-fi prototype in 6.3 on page 34.

### **7.2.1. User Study**

Three usability studies were conducted with three individuals who were acquainted with Stockholm and used a Android phone (and was part of the target user group). The tests were conducted with one test subject at a time. Two test subjects came from other parts of Sweden (Stockholm was not their home town) and one test subject came from England. One user study was conducted in the evening, while the other two were conducted during day time. The number of usability studies should preferably have been more extensive, but due to time

limitations testing had to be limited to three participants. Both doing more usability studies and looking deeper into how test subjects used the application to find additional issues.

The study was conducted as a task based test. Each test subject got a introduction to why they where there. This is to clarify that it is not the user doing something wrong if something does not work, but it is rather the system. The test subjects also got a short scenario description to understand what they where going to do:

"You just walked past the Nobel Museum and wish to find out if this is where the Nobel Prize is given out. You go in to ask and find out that the Nobel Prize is given out in the city hall. Luckily the staff tells you that they have an application for Android that you can use to find to it and learn more about different locations. The staff helps you install the application and you are of to explore the city with a perspective of the Nobel Prize."

The purpose of the scenario description is to make the test subject understand the purpose of the application. The test subject then got different tasks to complete (one at a time in chronological order):

1. Find information about the Nobel Museum.
2. Go to Grand Hôtel and find a puzzle piece.
3. Go to Nybrokajen 11 and look on pictures.
4. Find out how many puzzle pieces that are found.

Additionally questions where prepared to ask the test subjects after the tests were completed:

1. I know Android very well (answered using the Likert scale).
2. Was the application easy or hard to use?
3. Do you think that it is possible to move the museum experience from within its area with the help of this kind of technology?
4. What do you think was good with the application?
5. What do you think needs improvement?
6. I know how to find my way around in Stockholm (answered with the Likert scale).
7. Any other comments?

The study was limited to a part of the application (only half of the POIs) to see if there where any repeating problems at the different locations.

### 7.2.2. Evaluation

This part examines the result of the usability study conducted with the high-fi prototype. To determine the usability grade different aspects will be discussed: Measuring Usability, Problem Count, Opinion Poll, and Understanding the System. In the following section suggestions will be made for how to improve the usability problems that were discovered and discussed in this part.

The questions that were answered using the Likert scale with five options (where the lowest is Strongly disagree and highest Strongly agree), these five options are then transformed into a grade between 1 to 5. The three answers for the test subjects subjective opinion of their Android knowledge were: 2, 5, 3 with an average of 3.3. The result for how well the test subjects knew Stockholm was 2, 3, 1 with an average of 2. This makes it possible to draw the conclusions that the test subjects did not know Stockholm very well, and that they have about average knowledge about using an Android device.

In each of the usability studies some known bugs did appear (these are mentioned in the end of Section 6.3), where each of these bugs falls in to the category of critical issues. However, these bugs were not critical for what needed to be tested, so we will look past them from this point on.

During the user studies it was quite clear that there were differences in how well the application worked depending on whether it was evening or not. The test that was conducted in the evening showed that the application did not track very well and it was hard to get the augmented reality part of the application to work. After the usability study, this was a critical problem for the test conducted in the evening. This test subject was brought to an office to see if there were any issues with the augmented reality parts of the object.

The reference image for Grand Hôtel did not work very well. It did not work in the evening test, but worked in one of the tests during the day after some time. The Nobel Museum and Nybrokajen 11 both worked well for the daytime tests, but not at all for the evening test. During the second usability study the lens was wiped of and the tracking worked a lot better, this might have caused some of the problems that occurred during the evening test (that was conducted first).

At the Nobel Museum a glass door had been broken and the scene had changed, see Figure 16. This change actually worked for the day time tests, but not for the evening. I believe that this might have worked if the entire view was intact.

At Grand Hôtel and Nybrokajen 11 the test subjects had problems with standing in the correct location (a location which gives the best tracking), they were within range, but should have stood maybe a bit further away, or a little bit closer for optimal recognition.

All three test subjects had problems with the initial navigation. The map that was used did not display the test subject's current location, this is a critical problem that the test subjects finally overcame. Once they had navigated to the first location (Grand Hôtel) they knew where they were and could easily find their way to the next location (Nybrokajen 11). One test subject would have liked a more detailed map containing for example the location of the subway or some other POI. Also the text and pin used was too small, the test subject would have liked it to be larger. These things were considered to be small issues.

One test subject tried to zoom in on the augmented reality objects, and also thought that



Figure 16: Picture of the changed seen at the Nobel Museum

there should have been some sort of indication that the objects on the screen could be pressed (a pulsation or something similar).

The test subjects had no problems locating and checking how many pieces of puzzle they had located. They had already identified the piece of the puzzle when in the overview view for choosing locations.

Their answers to the questions are summarized in Table 4 (a complete formulation of the questions are given in Section 7.2.1).

Considering the problems described we learned that there are several different places in the system that a few critical issues need to be handled. The following critical issues reduced the usability of the application: if objects can not be pressed on, targets not identified, or if the system crashes. There were a few smaller issues, but not many occurred and the occurred infrequently that these issues should be classified as of medium importance.

Through the interviews it was clear that the users thought that it was possible to move the experience of the museum into the streets of a city. Their general thought seems to be that the test subject each thought that the application is easy to use, and that the only real usability issue that was a major critical issue was the navigation.

When looking at how each test subject used the system, they all had a general idea of where they were and had no major problem locating different items. The users easily changed from looking for different places to look at, and how many pieces that they had already found.

Combining all of these results gives us some measure of the application's usability. It is clear that there are some critical issues that need to be handled (bugs, navigation, and better reference images) before high usability can be reached (throughout the entire application).

Question	Summarized Answers
2: Ease of use	Yes, easy to understand except the navigation part.
3: Moving the Museum Experience	Yes, definitely. Especially as a virtual tour guide with an integrated with the GPS.
4: Positive with application	GPS integration, simplicity, offline mode, vibration, and the some test subjects thought that the augmented reality experience was fun.
5: Improvements	The overview was boring, night time mode, navigation with user position, something that indicates that object in the augmented reality view can be pressed, and try to get the native "look-and-feel" of an Android application.
7: Other comments	Fix the bugs (application crash and objects that could not be pressed). Some users thought it was a good tour of Stockholm.

Table 4: Summary of the test subjects' answers to the questions

The test subjects all found that the application was easy to use and was not too complex.

### 7.2.3. Suggested Improvements

To solve some of the important issues presented in the evaluation, suggestions will be made here about how to solve these issues.

To aid navigation (until Google Navigation is available in offline mode) some suggestions are: Add the distance and direction in the overview to help the user understand the general direction (this will probably make the overview less boring). This will aid the user in finding his or her own location on the map and make it easier to know what direction to go in. At each location make the pin and text larger so it is easier to see and read. Adding the users own location on the map is also an alternative (all thought this would probably take quite a long time to accomplish).

Crash handling will require some way to catch crashes in the application and taking the appropriate actions. It is important to fix the problem with objects in the augmented reality view that could not be pressed.

Tests should be conducted to see how well targets could be tracked in the evening, and try to take reference images that are appropriate (perhaps create an evening mode). Instructions should be added that objects in the view could be pressed when the splash screen is initiated (instead having a pulsation of all objects). Additional instructions should be provided through the panel menu when in the augmented reality view, enabling tips can be accessed, e.g. If the camera does not track very well, try wiping of the camera.

To solve problems with people standing in the wrong position, some sort of navigation suggestions should be added indicating where to stand. This would help the user move to the correct location and would make it easier find the correct reference image.

## 8. Conclusions and Future work

This chapter presents the conclusions and results from the thesis project. The chapter also describes what work that has been left undone and suggests appropriate future work.

### 8.1. Conclusions and Results

The user studies of the paper prototype (Section 7.1) were conducted with one interviewer, which could lead to important information being missed when the interviewer both was acting as both a log keeper and facilitator. A lot of information was learned from the way that the test subjects reacted to a new view and what they pressed when presented with a new screen. This will probably not have a big effect on the improved paper prototype, but smaller design flaws could have been missed.

Unfortunately, several of the suggestions brought up by the paper prototype were not implemented in the beta version (due to limited time and too large scope), where also problems that were later discovered. This showed that the low-fi and high-fi prototype both experienced the same issues (as was concluded in [57]), while it was easier to test the navigation and augmented reality parts with the high-fi prototype.

The project started out with learning how Lara worked, and understanding the programming. In the end not very many views were used and in the final prototype only some background functions, introduction, and overview were used (described in detail in Section 6.3). Everything else was written from scratch. In retrospect it would have been better to start the project as a new project rather than trying to improve an existing project. This would have saved time due to the fact that it took a lot of time to understand how Lara worked, what functions worked where, and what their dependencies were.

When choosing a reference image, it is really important to understand what problems might occur for the chosen target, how this view might be effected by weather or the time of day that the application is used (Section 7.2.2). Choosing a good reference image may make the difference between a working application and an application that probably will not work. The quality of the camera that was used for the reference image also effected how well the reference image worked (Section 5.3).

How well the phone tracks is also dependent on how good camera and processor the smartphone has. A short tour to the different locations trying out the application with a Samsung Galaxy S 2 and comparing it to the HTC Desire showed that the Galaxy worked in almost all locations while the HTC Desire only worked on almost half the tested locations.

The framework (QCAR in Section 2.2.1) that was used, might not have been the most appropriate. In the beginning of the project we wanted a framework with high trackability to add a layer of information on specific interesting objects e.g. "In this window have these Nobel price winners lived at Grand Hôtel". In a later stage of the project this aspect was not as important and the use of a framework that was based on geographical location could have been used. Such a solution would have been that the Wikitude SDK (see Section 2.2.4) was used where a general direction in the correct location would have been enough to get the augmented reality experience. Also it could be possible to combine these frameworks as was done in the Spike Island project (see Section 2.3.2). This would then also make it possible to

use the application later in the evening where problems occurred for the current solution (see Section 7.2.2).

When creating a beta version for a future prototype it might have been better to aim to keep it as close as possible to the intended end product as possible. In this case it is related to the navigation part where OSM was used instead for Google Navigation. It might have been better to do the usability study with Google Navigation, and used supported devices for the users to do the tests on. This would have contributed to a more realistic understanding of what the user really thought of the application as it was intended.

The problem definitions were divided into two parts (as noted in Section 1.2):

1. Designing and evaluating an augmented reality application that offers both high usability, and navigation within the city of Stockholm.
2. Specifically the project should demonstrate an Android application that extends the museum experience of a Nobel Museum visitor to outside the walls of the museum.

In the usability study for the high-fi prototype the test subjects found the application to be easy to use and that it had good usability except for some parts (navigation). This identified the part of the application that had high usability, while identifying some parts that did not have high usability related to navigation (see Section 7.2.1). This leads to an answer for our first defined problem the application had high usability, except for the part with navigation.

The second question asked if the experience of the Nobel Museum could be moved outside its walls. The usability study results with the limited number of test subjects showed that they were unanimous in saying that it could be done (Section 7.2.2). Hence the answer to the second defined problem is: Yes, potential museum visitors believe that it is possible to move the experience outside its walls with the aid of technology.

These conclusions are drawn from usability studies of only three participants (Section 7.2.1), who potentially might not represent the intended target group. This means that without more thorough usability studies, these conclusions might be faulty. However, since most of the participants' views were similar it could still represent the responses of the potential user group. Also two of the test subjects in these tests were from Sweden, which was not the main target group, but since they were not too familiar with Stockholm (Section 7.2.1) they were still seen as potential users. Throughout the project several users have been contacted (to gather more information about potential users (Section 6.1), usability study for low-fi prototype (Section 7.1, and in the high-fi usability study (Section 7.2)) and the usability results should thus represent the potential user group.

## 8.2. Future Work

This section describes what has remains to be finished. In this section the focus is on what could be interesting research topics and suggests some improvements to the application.

Further testing is needed to learn which Android phones can be supported by the application developed during this thesis project. A goal would be to increase the number of different types of Android phones that are supported. Adding support for iOS phones is an obvious next step for this project.

---

A more extensive user study on a broader and larger user group should be conducted. There are several issues with conducting user studies with augmented reality. These issues should be examined in a larger, and more extensive user study in order to find the problems with the specific application and more generally how these types of applications should be designed.

It could be interesting to make an application that offers multiple POIs for the users. Additionally, it could be interesting to try to create an augmented reality navigation with more POIs that could offer more to the user.

Tests should be conducted to see if a better camera would give better reference images or if higher quality images would be too detailed to be useful as reference images.

Due to the fact that the first time the application starts QCAR needs internet connectivity (to download and install drivers for the phone), this can be a problem if the user does not start the application at the Nobel Museum. A clear set of questions for the future are: How should the application be distributed to the phones and how should the installation of the application work (wireless, bluetooth etc.)? These are several important aspects that need to be solved if the application will be used by a tourist and as a marketing tool for the museum.

It would be quite easy to change the focus of the application to be an indoor application that the Nobel Museum could use within the museum and this change could be a possible future task.

In Section 7.2.3 several suggestions were made about how to improve the usability of the application. These should obviously be included in future work, if the project is to be developed any further. There are some features that could be added to increase both usability and functionality: Notification when close to a target (not the target you are going to). Adding zoom and wiping motions as ways of interacting, add possibility to share the results of the application on Facebook, and add an empty image of a puzzle in puzzle overview when no piece found yet. Add a puzzle instead of the text (directing the user to the puzzle overview) in the overview, and adding the possibility to see how many images are available for each location in an image view. Improve 3D objects in scenes, choose better targets for reference images (smaller in the size of a door works well), and add settings for users (e.g. use Google Navigation even when there is internet connectivity).



---

## References

- [1] Nobelmuseet AB. About the nobel museum(swedish). Last visited July 4 2011 from <http://www.nobelmuseum.se/sv/om-nobelmuseet>.
- [2] Nobel Media AB. About the nobel prizes. Last visited July 4 2011 from [http://nobelprize.org/nobel\\_prizes](http://nobelprize.org/nobel_prizes).
- [3] Stockholm Visitors Board. Facts about stockholm's tourism industry. 2010. [http://www.investstockholm.com/Global/Investment%20promotion/Dokument/Fakta\\_turism\\_ENG%20-11.pdf](http://www.investstockholm.com/Global/Investment%20promotion/Dokument/Fakta_turism_ENG%20-11.pdf).
- [4] J. Hederén. Kunskapsturism - en underhållande och kreativ resa för tillväxt. Not published, contact: [jan.hederen@homecom.se](mailto:jan.hederen@homecom.se).
- [5] R. T. Azuma. A survey of augmented reality. In *Presence - Teleoperators and Virtual Environments*, volume 6 (4), pages 355–385, Hughes Research Laboratories. Malibu, CA 90265, August 1997.
- [6] F. Zhou, H. Been-Lirn Duh, and M. Billinghurst. Trends in augmented reality tracking, interaction and display: A review of ten years of ismar. In *Proceedings of the 7th IEEE/ACM International Symposium on Mixed and Augmented Reality, ISMAR '08*, Washington, DC, USA, 2008. IEEE Computer Society, pages 193-202.
- [7] O. Bimber and R. Raskar. Modern approaches to augmented reality. In *ACM SIGGRAPH 2006 Courses, SIGGRAPH '06*, ACM. New York, NY, USA, 2006.
- [8] D. Schmalstieg and D. Wagner. A handheld augmented reality museum guide. volume IADIS International Conference Mobile Learning, A8010 Graz, 2005. Graz University of Technology.
- [9] L. Madden. *Professional Augmented Reality Browsers for Smartphones: Programming for Junaio, Layar and Wikitude*. John Wiley & Sons, 2011.
- [10] Soren Lauesen. *User Interface Design: A Software Engineering Perspective*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2005.
- [11] Anton Hedström and Patrik Ragnarsson. Lara : en skattjakt med mobil datorförstärkt verklighet. Master's thesis, 2011. <http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-70313>.
- [12] Android Developer. App install location. Last visited October 25 2011 from <http://developer.android.com/guide/appendix/install-location.html>.
- [13] I. E. Sutherland. Ultimate display. In *Proceedings of IFIP Congress, IFIP '65*, pages 506–508, 1965.

- 
- [14] D. Wagner and D. Schmalstieg. First steps towards handheld augmented reality. In *Wearable Computers, 2003. Proceedings. Seventh IEEE International Symposium on*, pages 127 – 135, oct. 2003.
- [15] W. Piekarski and B. Thomas. Arquake: The outdoor augmented reality gaming system. volume 45 of *Communications of the ACM*. Wearable Computer Lab, Januari 2002.
- [16] T. Asghar and A. A. Nauman. Augmented reality in museum environment - a case study at lund university historical museum. Master’s thesis, Lund University, Department of Design Sciences, Division of Ergonomics and Aerosol Technology, Lund University, August 2010.
- [17] Daniel Wagner, Thomas Pintaric, Florian Ledermann, and Dieter Schmalstieg. Towards massively multi-user augmented reality on handheld devices. In *In Third International Conference on Pervasive Computing*, May 9-10 2005. volume 3468/2005, Munich, Germany, [http://www.ims.tuwien.ac.at/media/documents/publications/wagner05\\_pervasive.pdf](http://www.ims.tuwien.ac.at/media/documents/publications/wagner05_pervasive.pdf).
- [18] S. Moore. Gartner identifies top ten disruptive technologies for 2008 to 2012. Gartner Inc. Last visited July 7 2011 from <http://www.gartner.com/it/page.jsp?id=681107>.
- [19] R. Azuma. Tracking requirements for augmented reality. *Commun. ACM*, 36:50–51, July 1993.
- [20] Aidong Sun, Yan Sun, and Caixing Liu. The qr-code reorganization in illegible snapshots taken by mobile phones. In *Computational Science and its Applications, 2007. ICCSA 2007. International Conference on*, pages 532 –538, aug. 2007.
- [21] Yue Liu, Ju Yang, and Mingjun Liu. Recognition of qr code with mobile phones. In *Control and Decision Conference, 2008. CCDC 2008. Chinese*, pages 203 –206, july 2008.
- [22] Qualcomm Austria Research Center GmbH. Developer guide, frame markers. [https://ar.qualcomm.at/qdevnet/developer\\_guide](https://ar.qualcomm.at/qdevnet/developer_guide) Press on Trackables and then Frame Markers. Last visited July 6 2011.
- [23] Qualcomm. Developer guide. Last visited July 18 2011 from [https://ar.qualcomm.at/qdevnet/developer\\_guide](https://ar.qualcomm.at/qdevnet/developer_guide).
- [24] Dave Shreiner. *OpenGL Reference Manual: The Official Reference Document to OpenGL, Version 1.2*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 3rd edition, 1999.
- [25] Qualcomm. Sdk. Last visited October 24 2011 from <https://ar.qualcomm.at/qdevnet/sdk>.

- 
- [26] String Labs Ltd. String is the fastest, easiest to use augmented reality sdk for ios. Last visited July 18 2011 from <http://www.poweredbystring.com/product>.
- [27] Metaio GmbH. Mobile sdk. Last visited July 13 2011 from <http://www.metaio.com/software/mobile-sdk/>.
- [28] Metaio GmbH. Unifeye sdk mobile. Last visited July 13 2011 from <http://docs.metaio.com/bin/view/Main/UnifeyeMobileSDK>.
- [29] Mobilizy GmbH. *Developing with the Wikitude SDK*. Ginzkeyplatz 11, 5020 Salzburg, Austria. [www.wikitude.com](http://www.wikitude.com).
- [30] Jean-Philippe Sak Lokham. Android application for augmented reality tour of irish tourist site (spike island) - including 3d modelling. Master's thesis, ENSEEIHT engineering school, made at the University College of Cork, March 21 2011.
- [31] metaio. Augmented reality museum experience. Last visited July 7 2011 from <http://www.youtube.com/metaioar#p/u/40/RxSb4tjdTPk>.
- [32] A. Moody. Total immersion brings augmented reality to arizona science center's new exhibit. Last visited July 19 2011 from <http://www.awn.com/news/visual-effects/total-immersion-brings-augmented-reality-arizona-science-centers-new-exhibit>.
- [33] Arizona Science Center. My digital world. 600 E. WASHINGTON ST. PHOENIX, AZ 85004. Last visited July 19 2011 from [http://www.azscience.org/my\\_digital\\_world.php](http://www.azscience.org/my_digital_world.php).
- [34] S. Pradhan, C. Brignone, J. Cui, A. McReynolds, and M.T. Smith. Websigns: hyperlinking physical locations to the web. *Computer*, 34(8):42–48, aug 2001.
- [35] J. Cui. A scalable and efficient design of websignsys, 2002. UCLA CSD TR # 020048.
- [36] M. Lechner and M. Tripp. Arml - an augmented reality standard. Technical report, Mobilizy GmbH, Jakob Haringer Strasse 5/IV 5020 Salzburg, Austria, 2010.
- [37] Layar. Development. Last visited July 16 2011 from <http://www.layar.com/development/>.
- [38] Layar. A beautiful, fun augmented reality app. Last visited July 13 2011 from <http://www.layar.com/>.
- [39] Mobilize GmbH. Wikitude augmented reality reinvented for android tablets. Last visited July 20 2011 from [http://www.wikitude.com/category/02\\_wikitude/world-browser](http://www.wikitude.com/category/02_wikitude/world-browser).
- [40] metaio GmbH. How junaio creates the augmented city... Last visited July 15 2011 from <http://www.junaio.com/augmentedcity>.

- [41] R. Jacob, P. Mooney, P. Corcoran, and A. C. Winstanley. Haptic-gis: exploring the possibilities. *SIGSPATIAL Special*, 2:13–18, November 2010.
- [42] S. L. Lau and K. David. Movement recognition using the accelerometer in smartphones. In *Future Network and Mobile Summit, 2010*, pages 1–9.
- [43] A. Serra, D. Carboni, and V. Marotto. Indoor pedestrian navigation system using a modern smartphone. In *Proceedings of the 12th international conference on Human computer interaction with mobile devices and services*, MobileHCI '10, pages 397–398, New York, NY, USA, 2010. ACM.
- [44] Nuno Crato and Nuno Crato. How gps works. In *Figuring It Out*, pages 49–52. Springer Berlin Heidelberg, 2010. 10.1007/978-3-642-04833-3\_12.
- [45] A. DeWitt and R. Bresin. Sound design for affective interaction. In Ana Paiva, Rui Prada, and Rosalind Picard, editors, *Affective Computing and Intelligent Interaction*, volume 4738 of *Lecture Notes in Computer Science*, pages 523–533. Springer Berlin / Heidelberg, 2007. 10.1007/978-3-540-74889-2\_46.
- [46] W. Huang C. Wu and T. We. A study on the design of voice navigation of car navigation system. In J. A. Jacko, editor, *Human-Computer Interaction. Ambient, Ubiquitous and Intelligent Interaction*, volume Part III of *Human-Computer Interaction*, Graduate School of Industrial Design, Tatung University, 3rd Section Taipei 10452 Taiwan, 2009. Springer-Verlag Berlin Heidelberg. 13th International Conference, HCI International 2009, ISBN-13 978-3-642-02579-2.
- [47] Google. Kml documentation introduction. Last visited July 7 2011 <http://code.google.com/intl/sv-SE/apis/kml/documentation/>.
- [48] Google. Kml tutorial - kml - google code. Last visited October 21 2011 from [http://code.google.com/apis/kml/documentation/kml\\_tut.html](http://code.google.com/apis/kml/documentation/kml_tut.html).
- [49] M. Tripp and M. Lechner. Arml specification for wiktitude 4. Last visited July 7 2011 from <http://www.openarml.org/wiktitude4.html>.
- [50] M. Lechner. Arml - augmented reality markup language. Ginzkeyplatz 11, 5020 Salzburg, Austria, October 2010.
- [51] Google. Google maps navigation (beta). Last visited October 21 2011 from <http://www.google.com/mobile/navigation/>.
- [52] Open Street Map. Open street map. Last visited October 24 2011 from <http://www.openstreetmap.org/>.
- [53] John Pruitt and Jonathan Grudin. Personas: practice and theory. In *Proceedings of the 2003 conference on Designing for user experiences*, DUX '03, pages 1–15, New York, NY, USA, 2003. ACM.

- 
- [54] Marco de Sá and Luís Carriço. Low-fi prototyping for mobile devices. In *CHI '06 extended abstracts on Human factors in computing systems*, CHI EA '06, pages 694–699, New York, NY, USA, 2006. ACM.
- [55] Stephen J. Andriole. *Storyboard prototyping: a new approach to user requirements analysis*. QED Information Sciences, Inc., Wellesley, MA, USA, 1989.
- [56] Midieum Shin, Byung-soo Kim, and Jun Park. Ar storyboard: An augmented reality based interactive storyboard authoring tool. In *Proceedings of the 4th IEEE/ACM International Symposium on Mixed and Augmented Reality*, ISMAR '05, pages 198–199, Washington, DC, USA, 2005. IEEE Computer Society.
- [57] Robert A. Virzi, Jeffrey L. Sokolov, and Demetrios Karis. Usability problem identification using both low- and high-fidelity prototypes. In *Proceedings of the SIGCHI conference on Human factors in computing systems: common ground*, CHI '96, pages 236–243, New York, NY, USA, 1996. ACM.
- [58] Jeff Sauro and James R. Lewis. Estimating completion rates from small samples using binomial confidence intervals: Comparisons and recommendations. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, HFES '05, pages 2100–2104. HFES, 2005.
- [59] *Applied imagination: Principles and procedures of creative problem-solving*. C. Scribner's Sons, 1957.
- [60] Henrik Kniberg. *Scrum and XP from the Trenches: Enterprise Software Development*. Lulu.com, 2007.
- [61] Eclipse Foundation. About the eclipse foundation. Last visited October 24 2011 from <http://www.eclipse.org/org/>.
- [62] Android. Adt plugin for eclipse. Last visited October 24 2011 from <http://developer.android.com/sdk/eclipse-adt.html>.
- [63] Qualcomm. Sdk - getting started - unity extension. Last visited October 24 2011 from <https://ar.qualcomm.at/qdevnet/sdk>, press on Getting Started - SDK, then on Step 1: Setting up the Development Environment.
- [64] Unity Technologies. Integrating unity with eclipse. Last visited October 24 2011 from <http://unity3d.com/support/documentation/Manual/Android-Integrating%20Unity%20With%20Eclipse.html>.
- [65] Unity Technologies. Unity3. Last visited October 24 2011 from <http://unity3d.com/unity/>.
- [66] MonoDevelop. Mono develop. Last visited October 25 2011 from <http://monodevelop.com/>.

- [67] SQLite. Sqlite - welcome. Last visited October 24 2011 from <http://www.sqlite.org/>.
- [68] Cygwin DLL. This is the home of the cygwin project. Last visited October 24 2011 from <http://www.cygwin.com/>.
- [69] Eastman Kodak Company. Kodak easyshare m1033/m2008 digital camera user's guide. Retrived from [http://resources.kodak.com/support/pdf/en/manuals/urg00846/M1033\\_M2008\\_Ex\\_UG\\_GLB\\_en.pdf](http://resources.kodak.com/support/pdf/en/manuals/urg00846/M1033_M2008_Ex_UG_GLB_en.pdf).
- [70] HTC Corporation. Your htc desire user guide. Retrived from [http://dl4.htc.com/web\\_materials/Manual/HTC\\_Desire/100629\\_Desire\\_HTC\\_UM\\_WWE.pdf](http://dl4.htc.com/web_materials/Manual/HTC_Desire/100629_Desire_HTC_UM_WWE.pdf).

## A. Reference Images

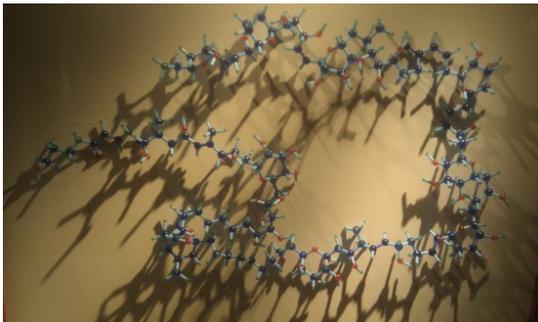
The reference images taken with HTC Desire and Kodak M1033 of three objects.



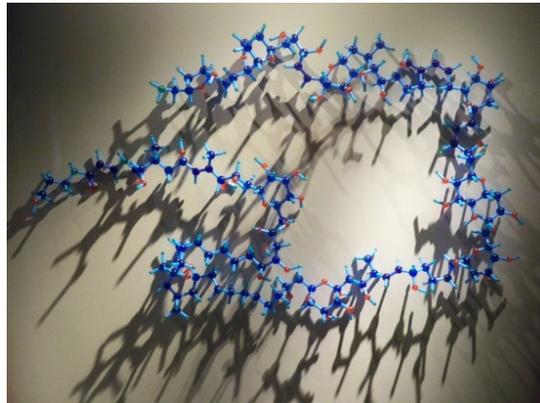
(a) HTC Desire: Börshuset



(b) Kodak: Börshuset



(c) HTC Desire: Palytoxin



(d) Kodak: Palytoxin



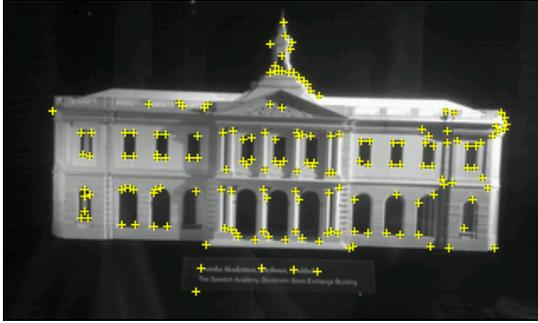
(e) HTC Desire: Entrance



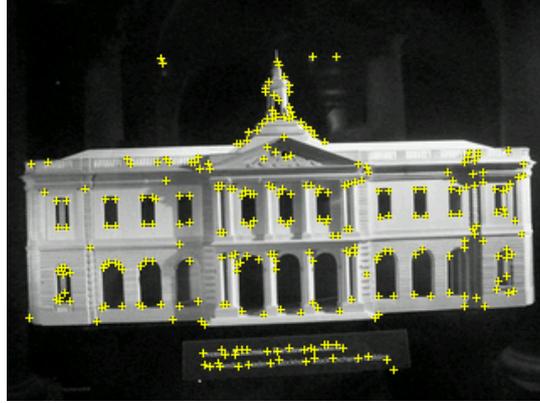
(f) Kodak: Entrance

## B. Reference Images with markings

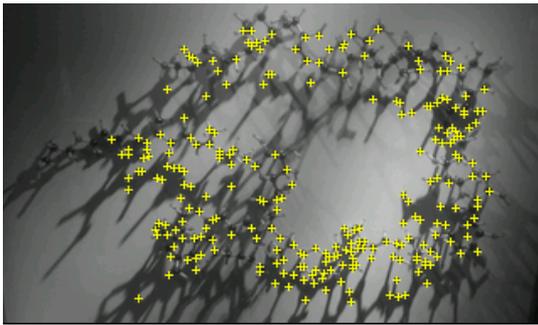
The reference images with markings taken with HTC Desire, and Kodak M1033 of two objects.



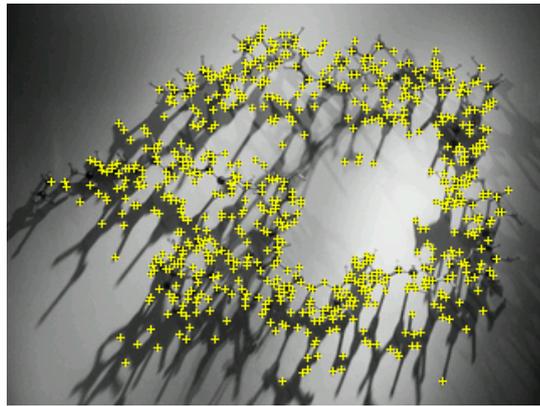
(g) HTC Desire: Börshuset



(h) Kodak: Börshuset



(i) HTC Desire: Palytoxin



(j) Kodak: Palytoxin

