Augmented Reality on Retail Store (Android App)

Submitted in partial fulfilment of the

A Final Project

SEWP ZG629T DISSERTATION

BY

Shaik Abdul Kaleem

(2010HW71443)

UNDER THE SUPERVISION OF

Sainath Varanasi.

BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI
RAJASTHAN – 333031, INDIA
November, 2014

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI

SEWP ZG629T DISSERTATION

Augmented Reality on Retail Store (Android App)

FINAL SEMESTER REPORT

Dissertation Proposal for partial fulfillment of the requirements of M.S. (Software Engineering)

By

Shaik Abdul Kaleem
(2010HW71443)

Under the supervision of

Sainath Varanasi

Project Lead, Wipro Technologies

Dissertation work carried out at

Wipro Technologies, HYDERABAD

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE
PILANI, RAJASTHAN – 333031, INDIA

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI

CERTIFICATE

This is to certify that the Dissertation entitled **Augmented Reality on Retail Store (Android App)** is submitted by **Shaik Abdul Kaleem** ID No. **2010HW71443** in partial fulfillment of the requirements of SEWP ZG629T Dissertation embodies the work done by him/her under my supervision.

Sainath.

Signature of the supervisor

Date: 22/8/14.

Name : Sainath Varanasi.

Designation: Project Lead.

ACKNOWLEDGEMENTS

I would like to thank our supervisor Sainath Varanasi continued guidance, support and encouragement. We would also like to thank Professor Vidya for his advice and guidance in implementing our **Augmented Reality on Retail Store (Android App)**. We would also like take this opportunity to thank Professor Y.V.K Ravi, Professor Sashikanth Akula and Our coordinator Santosh Sridhar, for their moral support.

Finally, yet importantly, I would like to express my heartfelt thanks to my beloved parents for their blessings, my friends/classmates for their help and wishes for the successful completion of this project.

Shaik Abdul Kaleem

2010HW71443

Abstract

With the mixing of real and virtual, Augmented Reality (AR) is a technology that has attracted lots of attention from the science community and is seen as a perfect way to visualize context related information. Computer generated graphics is presented to the user overlaid and registered with the real world and hence augmenting it. Promising intelligence amplification and higher productivity, AR has been intensively researched over several decades but has yet to reach a broad audience.

Technology has changed the way consumers shop. With the raise of online retail, a phenomenon known as "showrooming" has emerged. Shoppers visit brick and mortar stores to check out merchandise first-hand, then return home to make their purchase from an online retailer that offers the same goods at a lower price. This trend has taken a toll on brick and mortar retailers, creating a need to tap into new and more innovative ways to capture their shoppers' attention.

This Application presents efforts in bringing Augmented Reality to retail store domain and thus to the general public. Implementing technologies on limited devices, such as mobile phones, poses a number of challenges that differ from traditional research directions. These include: limited computational resources with little or no possibility to upgrade or add hardware, limited input and output capabilities for interactive 3D graphics. The Application presented will address these challenges:

1) How traditional retailing can be made more intuitive by combining with augmented reality and e-commerce.

2) Optimise the warehouse space by simplifying the navigation procedure.

3) Improves customer satisfaction by letting the customers "try" before they buy with a 3D product preview.

4) Displays additional information about products shown, enriches shopping experience.

5) Visualizes product catalogues.

6) Brings the customer to the store with a vivid and intuitive augmented experience.

7) Interactive marketing/ advertising.

8) Instant Purchase.

This application has been developed on the Android platform and supports all the latest android versions. The application is planned to be developed on Google Glass if Google Glass users increases.

TABLE OF CONTENTS

1. INTRODUCTION

1.1.Customer Profile

Reliance Digital is a consumer electronics company in India. The first Reliance Digital Store was opened on 24 April 2007 in Delhi. Currently there are around 170 Reliance Digital Stores in around 70 cities in India. The stores are spread across the various states of the country

Reliance Digital Stores are bigger in size than the other two formats Digital Xpress & Mini stores.

Recently, they also launched a range of premium stores called as Digital Xpress, where customers can test devices in their designated solution docks - environments created for custom experience. The Xpress stores house lifestyle products like smartphones, Ultra books, and speakers, Cameras, Tablets, Laptops and smart TVs among others. The stores offer buyers an opportunity to learn to do extra stuff with their gadgets such as setting up IP cameras at one's house to help monitor using a mobile device, or set up wireless backup solution and then access data using a mobile app.

Digital Xpress Mini Stores are relatively smaller in size than Reliance Digital & Digital Xpress Stores. These stores are about 250 square feet and mainly sell the company's telecom services, smartphones, tablets and also accessories of other brands. They are planning to open around 2,000 stores at the end of the fiscal year 2014-15.

ResQ is the service arm of Reliance Digital / Digital Xpress and Digital Xpress mini stores, which caters to customers for after sales service. ResQ is India's first Multi product, Multi brand, Multi-location service facility which offers service from 10am to 10pm, 365 days a year. The ResQ Care Plans offer scheduled preventive maintenance visits and Standby units in special cases.

Our application is targeted at taking the retail experience of Reliance Digital and ResQ stores to a whole different level by introducing the concept of augmented reality in their stores that would make their stores one of the first in the country to implement such a technology.

This would take the customer satisfaction to a whole new level with intuitive and interactive shopping and thus making the store one of the premier stores in the country.

1.2. Abstract

With the mixing of real and virtual, Augmented Reality (AR) is a technology that has attracted lots of attention from the science community and is seen as a perfect way to visualize context related information. Computer generated graphics is presented to the user overlaid and registered with the real world and hence augmenting it. Promising intelligence amplification and higher productivity, AR has been intensively researched over several decades but has yet to reach a broad audience.

Technology has changed the way consumers shop. With the raise of online retail, a phenomenon known as “showrooming” has emerged. Shoppers visit brick and mortar stores to check out merchandise first-hand, then return home to make their purchase from an online retailer that offers the same goods at a lower price. This trend has taken a toll on brick and mortar retailers, creating a need to tap into new and more innovative ways to capture their shoppers’ attention.

This Application presents efforts in bringing Augmented Reality to retail store domain and thus to the general public. Implementing technologies on limited devices, such as mobile phones, poses a number of challenges that differ from traditional research directions. These include: limited computational resources with little or no possibility to upgrade or add hardware, limited input and output capabilities for interactive 3D graphics. The Application presented will address these challenges:

9) How traditional retailing can be made more intuitive by combining with augmented reality and e-commerce.

10) Optimise the warehouse space by simplifying the navigation procedure.

11) Improves customer satisfaction by letting the customers “try” before they buy with a 3D product preview.

12) Displays additional information about products shown, enriches shopping experience.

13) Visualizes product catalogues.

14) Brings the customer to the store with a vivid and intuitive augmented experience.

15) Interactive marketing/ advertising.

16) Instant Purchase.

This application has been developed on the Android platform and supports all the latest android versions. The application is planned to be developed on Google Glass if Google Glass users increases.

2. BUSINESS OBJECTIVE

Technology has changed the way consumers shop. With the raise of online retail, a phenomenon known as “showrooming” has emerged. Shoppers visit brick and mortar stores to check out merchandise first-hand, then return home to make their purchase from an online retailer that offers the same goods at a lower price. This trend has taken a toll on brick and mortar retailers, creating a need to tap into new and more innovative ways to capture their shoppers’ attention.

This Application presents efforts in bringing Augmented Reality to retail store domain and thus to the general public. Implementing technologies on limited devices, such as mobile phones, poses a number of challenges that differ from traditional research directions. These include: limited computational resources with little or no possibility to upgrade or add hardware, limited input and output capabilities for interactive 3D graphics. The Application presented will address these challenges:

- How traditional retailing can be made more intuitive by combining with augmented reality and e-commerce.
- Optimise the warehouse space by simplifying the navigation procedure.
- Improves customer satisfaction by letting the customers “try” before they buy with a 3D product preview.
- Displays additional information about products shown, enriches shopping experience.
- Visualizes product catalogues.
- Brings the customer to the store with a vivid and intuitive augmented experience.
- Interactive marketing/ advertising.

- Instant Purchase.

3. PRE-STUDY

This section will introduce us to the many different aspects of our pre-study. For a project of this size, one very often has to get familiar with new concepts, techniques and methodologies, of both technical and managerial art. In our case, we were to develop an application for the Android operating system, using the Scrum development framework, programming for augmented reality and so on. The majority of the group had never done any programming for the Android from before, no one knew Scrum very well and no one had any experience with developing applications for augmented reality. From this, it was pretty clear that we had to do a lot of studying on both the technical parts as well as the managerial parts of the project. We will try our best to introduce these concepts and describe the result of our extensive studies.

Purpose

The purpose of the pre-study is to introduce us to the many aspects of augmented reality, and especially concerning augmented reality on smart phones. In addition it serves as an introduction to the different methodologies and various technologies that are relevant to this project.

Scope

The pre-study is a part that can be possibly huge, and it was necessary to limit our focus area to the parts we thought the customer wanted us to have a closer look at. That is why we have brought an emphasis on the augmented reality on smart phones, and on a few image recognition algorithms that had a suitable license that we could use. We also had a look at different methodologies so that we could make an informed choice between them for this project.

3.1. Augmented Reality

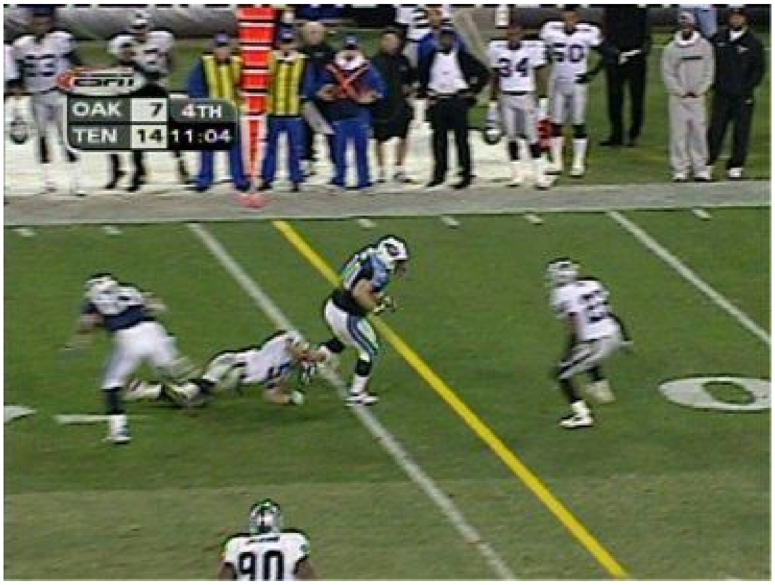
Augmented reality is the set of techniques that allow virtual elements to be fixed into reality. These virtual elements can use the five human senses, the most common application using images, and sometimes sound. Augmented reality has to combine these three features:

- Combining real and virtual reality
- Interactive in real time
- Registered in 3D

The beginning of augmented reality

The first applications of augmented reality we can find, are in the army. In the cockpit of the plane, or in the helmet of the pilot, he can see several pieces of information, overlaid on his current view. Then, some medical applications appeared, helping doctors to understand some phenomena, or helping with high risk and hard surgery operations.

Another application that is quite old and famous, is the overlaying of line and information onto the sport grounds, such as in American soccer or ice hockey.



One example of the first applications of augmented reality

Different augmented reality degrees

We can distinguish between three very different ways of doing augmented reality:

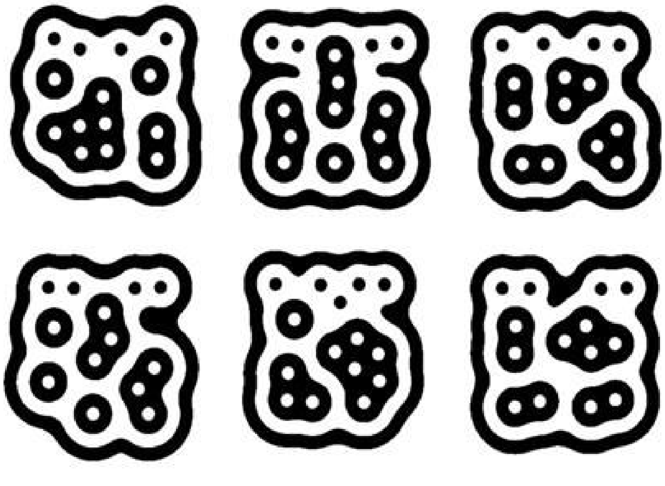
1. **Using positioning information**

The application mixes information relative to where the user is or at the place the user is looking at. It overlays the information it can retrieve on the screen, concerning his position. See below for a screenshot of the Wikitude World Browser, which allows some web search from this kind of information. Using this technique, it is hard to make the reality and the virtual elements match, the application has no idea of where exactly the pointed object or building is on the user's screen. We can also argue that it is hard to find the users location accurately, for example with the GPS system, the accuracy is about 15 m. Of course, depending on the application, this could be sufficient. However, the time constraint is well respected with these solutions, so we can speak of real time applications.

2. **Using markers**

(a) Fiduciary markers

A fiduciary marker is a simple shape, strongly contrasted. Usually, one can use a simple black and white figure. Squares are often used, but not exclusively.

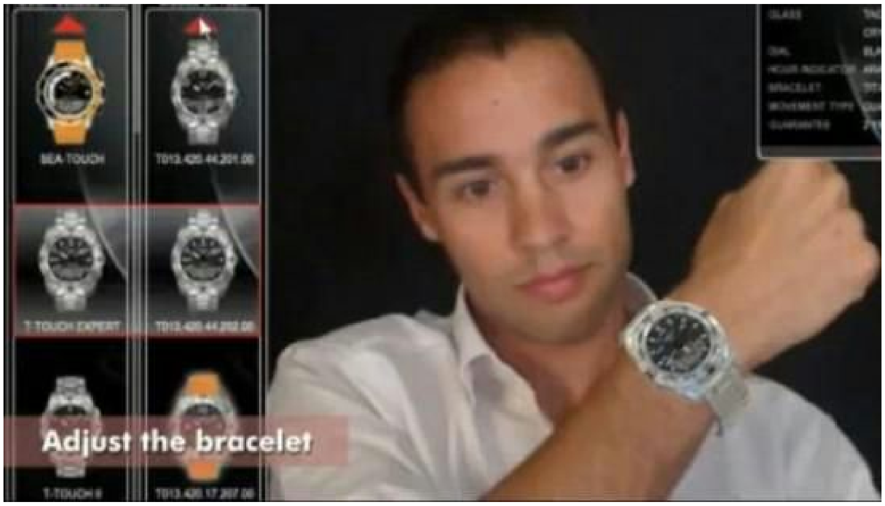


Example of fiduciary markers

These markers are easily and quickly recognized by a computer and a camera. Here, the applications can localize precisely tracked objects, and then integrate the information on it, in real time. The only problem here is that you cannot put such a marker on each building of a city, for instance. This marking process can be long and inappropriate, for example in some museums on would like to help visitors without vitiating the exhibit.

b) Other markers

One way to get round this scaling problem is to use the reality as a marker, which is possible in some situations. For instance, look at this application, used by Tissot to sell watches:



Tissot use augmented reality to let the customer to try the watch

In this application, the user just put a wristband which is easily recognized by the computer, compared to fiducial markers. Then he can try all the brand's watches, quickly and easily. Another application has been realized for toys' advertising, the marker being the toy box, which customers show in front of a camera, to see on a screen what the toy looks like in reality. This kind of markers is a good solution to combine time efficiency and ease of use, but we cannot use them with objects such as buildings, or exhibits.

3. **Using image recognition**

The limit between markers and image recognition is a bit fuzzy. For example, a lot of facial detection applications use the eyes as markers. It is very easy to recognize eyes on somebodies face because it is two zones darker. Image recognition use a camera and a computer to extract the main features from a picture, and compare it to some other data that we acquired earlier. Of course, this way one can know precisely where the object is, without having marked it before. The problem here is the time required for a computer to recognize an object. Further, this recognition can be harder depending on the camera angle, or the light conditions.

Augmented reality using smart-phones

With the development and democratization of smart-phones, mobile based application of augmented reality became more and more present. The latest smart-phones include GPS, camera and compass, so it is easy to get basic global positioning.

Lake Helen, Florida 20.0 miles

For the high-altitude lake in Lassen Volcanic National Park, see Lake Helen (Lassen Peak)Lake Helen is a city in Volusia County, Florida, United

Wikitude World Browser, on one of the most famous smart-phones, iPhone 3GS, using GPS and solid state compass

Smart-phones are complete devices, on which one can use all the techniques seen above. That is one of the reasons why augmented reality is expanding very quickly. But there are some limitations to the use of a smart-phone, you have to keep your phone/camera in front of you while walking, which is not very natural, and it does not allow you to use your hands. You are focused on your screen and have to choose between looking at the reality directly or through the screen. Other solutions exist and they might be considered, such as the Sixth-Sense project, which use a mini-projector to project additional information either on you (for instance a hand) or on the walls or objects in front of you. The cam-era is situated around your neck, so you don't have to keep it elevated all the time.

The smart-phone solution must not hide all the other ones, because it is not as natural as it seems to use its camera.

What is and what is not augmented reality

All the ways of doing augmented reality have a common point: The mix of reality and virtual. The most known way to do that is on a picture, but it is not the only one, and there is still some new features to develop in applications. At the same time, using the techniques presented above does not mean doing augmented reality. It is a fashion name nowadays, and lots of project are mentioned to be augmented reality although they are not. For example, some cities provide QR Codes, which redirect to a website if you point at it with a smartphone. In these projects, QR Codes are placed on famous buildings and places, and the website you are redirected to explains you the building history, and some other extra information. This kind of project is not augmented reality, because it does not mix this extra information with the reality, it just links it. Again, in augmented reality applications, reality and virtual have to be mixed. The more integrated the extra information will be in reality representation, the closer we get to perfect augmented reality.

3.1.4 Limitations

As we already mentioned, one problem is that it is not that natural to walk with your phone camera pointing everywhere. Some tools exists to avoid this problem, such as augmented reality glasses (e.g. 3D Visors), but these solutions are not used by a representative part of people. Augmented reality should not physically constrain the user, just to add some virtual information. It must simplify his life and not complicate it. Another problem that is rising in people mind, is the freedom to look. When you visit an exhibit, you might want to discover it by yourself, maybe you will be interested in some special information, not in the main subject. Looking through your camera, following the instruction, you become more passive, you have less choice concerning the information you want to look at. This problem has to be solved developing original application, integrating the user's wishes, and giving him the choice to be active. Concerning these applications helping you walking through a city and showing you the best stores, the danger is that information about a big store might be hiding information about little ones. If we consider this kind of system based on advertising, we understand this problem better, how can we be sure that the application answers our wishes only, without considering the stores' ones ?

3.1.5 Conclusion

Along this state of art, we saw that we have a long way to go before we have under-stood everything that augmented reality can offer. If we examine the techniques rst, there are some improvements to make concerning the image recognition response time and the precision of the localization tools. Concerning the support, the smartphone is not yet popular enough to see the impact of some application on the population's habits, we will have to wait a little for the smartphone democratization. Further, it is not as natural as it seems to use it, but there are some other support to provide augmented reality. Concerning the applications themselves, they have to involve the final user, to make him active, taking his mind and habits into account.

3.2.Object recognition

Object recognition is a sub-field of the more general computer vision field. The task of object recognition is, given an image of an object, recognize this object in another image, possibly rotated or scaled. The general case of object recognition is still a hard task, requiring lots of computation power and time. In our project we have focused our attention on two main object/image recognizing algorithms / descriptors, SIFT and SURF.

SIFT

Scale-invariant feature transform (SIFT) is an algorithm to detect and describe local features in images. This algorithm is used in computer vision and for applications that include object recognition, image stitching, 3D modeling, gesture recognition, robotic mapping and navigation, match moving, and video tracking. It was published by David Lowe in 1999.

The SIFT transforms the image into some local feature vectors. These vectors have a different features that convert in a distinctive and invariant to any scaling, rotation or translation of the image. This algorithm can also be used to find the correspondences between different views of the same scene. These local features are stored in descriptors that try to describe important areas of certain image variables. This algorithm has a big computational cost but a successful implementation can be used in a real-time application because the most high cost operations apply only to certain function as those have passed an initial screening test. SIFT is divided into four distinct phases: Scale-space extrema detection, Key-point localization, Orientation assignment, and Key-point descriptor.

Scale-space extrema detection

This phase is responsible of finding the first set of points of interest (key-points) in the image. These key-points have to be filtered because some of these will not meet certain requirements.

Key-point localization

This is the second SIFT phase. It focuses on storing all the information available for each key-point. For each key-point, this phase saves the position in the scale pyramid where it was found and its position on that scale. With this information it will exclude key-points according to two criteria, because these points would be very sensitive to them. The rst criteria is that low contrast points are discarded. The second criteria is if it is located along the edges, since responses from edge tends to be very high, the point will not be robust.

Orientation assignment

This phase will focus on calculating the key-point orientations. Once one has them, one can build the descriptors in a way which is rotation invariant. Then one store the following information in every image region: location, octave, scale and main orientations.

Key point descriptor

The parameters we obtained above form a 2D coordinate system. This system describes the local information about each image region and for this reason pro-vides invariance to the same parameters. The next step is to obtain a descriptor for each interest area that will also provide robustness to the system. One must perform a series of modifications to achieve greater robustness against possible changes in lighting. The goal is to be invariant along three types of light variation: brightness, contrast and nonlinear variation.

SURF

SURF: Speeded Up Robust Features is a scale- and rotation invariant interest point detector and descriptor. It was first presented in a paper by Herbert Bay, Tinne Tuytelaars and Luc Van Gool in 2006. SURF substantially outperforms other similar methods of image feature detectors/descriptors e.g. SIFT, with regards to repeatability, robustness, distinctiveness, and speed.

The basic features of the interest point detection part of SURF is the integral image, the hessian matrix, the scale space construction and the interest point detection. The detected points are then given a reproducible orientation, and a 64/128 dimensional vector with descriptors.

Key point detection

This is the part of SURF where the SURF points are detected, using a basic Hessian-matrix approximation. This is used in combination with a data structure called integral image, or sum area table. This is just an intermediate image representation, for fast calculation of pixel intensities in certain rectangles. Given an input image I and a point (x, y), the value of the integral image at point (x, y) is then calculated by the sum of the values between the point and the origin.

i x j y

I (x; y) = I(x; y)

=0 j=0

P X_i X

Getting the pixel intensity sum in any sized rectangle is now just an easy constant time calculation, by getting the four points of a rectangle, A, B, C, D. Then the sum of pixel intensities in the rectangle is:

A + D (C + B)

SURF uses integral images to perform fast image convolution and this is one of the reasons SURF is faster than SIFT.

Only points with a stronger hessian response than a provided threshold will be considered for further description.

Key point description

The SURF point description is the other part of SURF. The first step of this part is to give the points a reproducible orientation. First Haar wavelet responses around the detected point is produced. Then a sliding window of size $_3$ is rotated around each point. Haar wavelet responses are then summed up, this creates a vector which is the dominate orientation for that point.

Then the actual descriptor of the point can be found. The 16 squares are constructed around the SURF point, then the Haar wavelet response in horizontal, and vertical direction is computed, and stored in the description vector. The absolute sum of the values of the responses are also added to the description vector. This results in a vector of length 64.

The usefulness and robustness of this detector and descriptor was demon-started by Bay et al in their paper Speeded-Up Robust Features(SURF).In this paper they show how SURF performs with respect to other image recognition schemes.

3.2.3 Why we chose SURF

The decision to use SURF was done for a couple of reasons. The first one is the limited processing power we have available on mobile phone. This is a good reason to choose not to use SIFT. As was shown by Bay et al in. The SURF detector and descriptor runs significantly faster than the SIFT counterpart and is more robust. This constitutes the greatest reason to choose SURF over SIFT. The second reason is that the customer wants a general image recognition module, and that is why we have chosen to look at SURF and SIFT primarily. There are other good image recognition algorithms which can be used when the case is not as general. Haar cascade codes, contour detection, and corner detection, are a couple of useful schemes that comes to mind.

3.3.QR codes

QR Code is a barcode consisting of a matrix of black and white cells. These cells are usually called “modules”, and the module configuration as well as the number of modules is determined by the version of the QR Code. The barcode can carry various amount of data, ranging from 152 to 23 648 data bits with version 1and 40 respectively. A higher version number is not necessarily better, but a code consisting of more modules. This equates with a spatially more extensive barcode if kept at the same scale. All QR Codes are square, and the specific configuration of rows/columns is called a version.

A standards-compliant QR Code also provides ECC capability, with a choice between four levels. This is accomplished by adding Reed-Solomon code to the existing data. The different levels defines the number of code words dedicated to ECC.

The information encoded in the code could be both text and data. For ourpurpose, encoding of a short text or a numeric string should be sufficient. Storing enough information in the bar code itself, may be possible in some situations, but encoding a key string is more advantageous with regards to the flexibility of updating information. A solution using bar codes of any kind, should only do so in order to grab the “actual” data from some kind of database. That way the service provider has a single point of update, and the propagation is immediate unless data is cached on the client side. Even if this is the case, it is trivial to limit the database queries to “Do I need to update the information in my cache with timestamp xxxx”, and defer from receiving updates unless the cached information is stale.

QR Code is standardized by ISO and JIS, among other standards bodies. However the ISO-specification is not available for free, although the standard itself is.

A possible challenge is to find a good compromise between the size of the bar code and the required maximum distance between the camera and the bar code. This depends on the amount of noise in the picture as well as the CCD-resolution. Regardless of that, the camera has to be positioned relatively close to the barcode – depending on the size of the code.

The use of QR Codes is widespread and there is a wealth of processing software for most platform, including Android. There is certainly not a lack of off-the-shelf software, and through the use of specific features in the Android system 2 makes it easy to plug existing bar code reader software into any application if needed.

Data matrix is a competing standard for matrix barcodes. Unlike QR Codes, the bar code does not have to be a square. However for our application it won't make any difference. They both serve the same purpose, and the same technical restrictions apply to both of them.
Because there's a wealth of QR code-reader software available for Android, it makes perfectly sense to call on QR Code reader software.

3.4.Target platform/The choice of platform

It is a given that the client side part of the software stack, is going to run on some kind of mobile platform. Exactly what kind of platform that would be, was an open issue at the beginning of the project.

These days, mobile phones are omnipresent, and advanced features like high-resolution displays, high-performance graphics processors, fast CPU's and dedicated signal processors for audio and video decoding have becoming increasingly common. The feature gap between the most feature-rich- and the least expensive models has coined the term \smart phone". Our focus will be on such phones, but there are a variety of options to choose between.

3.4.1 Choice of software platforms

The choice of what software platform what will be the target of this (client-side) product is naturally characterized by the team members own knowledge and impression of contemporary systems. In reality, the following platforms were considered:

Symbian Android

iOS [3]

The team never considered other alternatives, although the systems are by no way exclusive to the mobile environment.

What follows, is a brief description of the platforms. Their relative merits and disadvantages are in no way evaluated at depth, but the discussion serves as a description of the alternatives from the team-members' standpoint.

Symbian

The Symbian operating system (and framework) has its roots in the EPOC-operating system. Initially developed in the latter part of the 1980s by Psion Corporation, it grew in popularity throughout the 90's, and it was adopted in product made by major corporations like Nokia, Sony-Ericsson and Motorola. The \standard" nature, and advantage of having one OS on a multitude of products from different manufacturers is likely to be one of the reasons why

Symbian became increasingly popular.

Symbian is still a significant player in the market, but has met fierce com-petition from Android and iOS. Even Motorola - a manufacturer that has used Symbian on their products - have released at least three phones with Android.

It was hard for the team to assess the advantages of developing for Symbian, because no members have any experience developing for it. However the use of C++ and the QT framework makes it a strong contender feature-wise.

iOS

Since the introduction of the original iPhone in 2007, it has had an increasing influence in the smart phone market. Developers of iOS applications enjoys relative homogeneity in the hardware. While this is generally true, there is a difference in hardware between the first generation iPhone and the fourth generation. The screen resolutions spans from 320x480 to 960x640 on iPhone, and 1024x768 on iPad. Apple delivers a closely integrated product when combined with the iTunes application. iTunes is the main, and only way for end-users to purchase and download applications onto their phones and iPods when at their PC. It is also possible to do the same through dedicated software on the mobile hardware.

With respect to how familiar the team is with the iOS SDK, no members have any experience with developing for the iPhone, iPod Touch or iPad. Moreover Objective C is the primary language if one are to develop iOS-applications. While a programming language itself is nothing but a tool (thus no reason to get to passionate about it), the truth of the matter is that Objective C is not widely used outside the \Apple environment". It is possible to compile Objective C code with mature and decent compilers like Clang and GCC -the latter being the primary choice at this moment[4]. However, the libraries that give the applications the iOS/Mac-"feel" are both proprietary and only available for OS X and iOS.
The iOS SDK is only available for OS X, and Apple's own IDE - XCode - provides a variety of tools necessary for iOS development. The availability is a problem. As it is today, OS X market shares are insignificant in the desktop market. In addition Apple does not approve the installation of OS X on non-Apple products. In fact, it is disallowed by the EULA. In any case, the upfront costs are high unless the developer already has access to Apple PCs.

As for the cost, it is possible to download the SDK for free. But it is not possible to distribute applications without enrolling in Apple's development program, with the price of 99USD per year. But then there's no extra charge in order to sell the applications through iTunes App Store. Unfortunately any application distributed through the App Store, has to go through a review by Apple. It is generally held to be true, that the rules that decides what gets approved, is not transparent. But the guidelines are in fact available

Android

The last major contender in the mobile market is backed up by Google. While by no means very comforting in itself, a large company like Google has the influence to funnel a huge amount of resources into the maintenance and continued improvement of the Android OS.

Android applications are written in the Java language, and is compiled to byte code as is familiar to any Java developer. However a Sun Java compliant JVM is not able to run the resulting class- les. Instead Android uses a different JVM altogether, called Dalvik. This is not a major hindrance in itself, but serves as a reminder that a given program is not able to run on an ordinary PC outside the Android emulator.

All the members of the team are quite familiar with Java and the standard Sun Java class library. This is an advantage that is not easy to neglect when choosing target platform. In addition, the Android platform is open. Not only open as in \open source" (but with limitations on \approved hardware" for access to Android market); the SDK is available for all major platforms, including Windows, OS X and GNU/Linux. With an appropriate plugin, the integration with the Eclipse Java IDE is unsurpassed-when compared to the alternatives.

Due to Android's openness, it is available on a variety of hardware. This means that the functionality of Android devices differs significantly. This is not a problem in this project, because the target is Android hardware with very specific hardware capabilities. But the differences in screen sizes, resolution and densities has to be taken into account when developing any Android application.

Unlike iOS, it is possible to distribute Android applications outside the official channels. The user can install Android applications, much like they can install arbitrary software on a PC, by running an installation package or executable.

3.4.2 Conclusion

Choosing Android as the target platform is a natural choice: Three members of the group have an Android phone, and one more member is going to get one in the nearest future. One member have a fair amount of experience with programming on Android and two more members have some introductory experience. But no one on the group have ever worked with an iPhone. Also the fact that iPhone uses its own programming language while Android uses Java, that all the group members have experience with, counts heavily in the decision. In a time-constrained project like this, familiarity with required tools and language is a nice addition to Android's other benefits, like availability of SDK and easy distribution of software. All these arguments, as well as guidelines from the customer, are the reasons why we favor the chosen platform.

3.5.State of the art

As a part of the initial description given to us by the customer, we were given the task to \ a) Present state-of-the-art for object recognition and augmented reality for mobile phones (Android and iPhone), including available commercial and open source solutions"[19]

We have spent quite some time on this, discovered some nice applications and algorithms that already exists. The scope of this task was a narrowed down in the very beginning of the project, as we agreed with the customer (see 3.4.2) to only focus on making an Android program.

3.5.1 Android Market

On the Google Android system, the main place to download new programs is through Android Market. This is a market place intended for every program made for Android, free as well as priced ones. Until recently, there were only nine countries in which the Android users could buy an application through the Market, while the free apps are available to a lot of countries, as listed in the previous link. Of course, this may (and probably will) change, even though only a few months ago it did not seem as if Google would focus on making users in ,for example Norway, able to buy apps from the Market[22]. This rapidly changed in September-early October, and in the first half of October, Google introduced paid apps in Norway and 17 other countries.

There exist some alternatives to the Android Market, for example Adroia[24]. Adroia is, according to themselves, “en norskutviklet markedsplass for Android-applikasjoner med betalingsl sning.", or in English: a Norwegian-developed market place including a payment solution for Android applications. It is made to make it possible for Norwegian developers to sell their applications. SlideMe.org is another alternative market place for Android apps, which also contains a payment solution.[26] In addition, there are some available hacks which allows the customer to simulate being in a country opened for buying apps on Android Market, but this is a bit edgy and definately out of our scope.

Bearing this in mind, we have considered mainly applications that are available for free, of which most are some kind of open source. The programs we have had a special look at are QR-code readers (\barcode scanners"), Layar, Google Goggles, Google Sky Map, Google Maps, Wikitude, ARToolKit and Surf.

3.5.2 Licenses

An important aspect to know about and bear in mind when talking about specific program and frameworks available. Here, we give a quick introduction to the most common licensing paradigms for software. First, an important distinction between proprietary and open software. In proprietary software, the source code of the program is secret and not shared outside the company or equivalent. In opposition to this, in open source software, the source code is available for whoever interested to read and study it.

There are several approaches to open source software, and also a significant discussion about terms, names and so on, which we will not dig into here. Instead, we will describe the most common open source licenses in use. By this we mean the GNU General public license, the Apache Software License and the BSD license. All of these are free software, meaning that they give the user four essential freedoms:

1. Freedom to run the program regardless of the purpose.
2. Freedom to study how the program works and adapt it to own needs.
3. Freedom to distribute copies.
4. Freedom to release one's own improvements to public domain.

Another central concept here is copy left. If a license implies copy left, every distributed program that uses this licensed program must apply the same license itself, or an equivalent license. GNU GPL is seen as the most typical copy left license, and also the most commonly used one. On the other hand, neither BSD nor Apache license uses copy left. From this, everyone can use parts of code licensed under one of these licenses in his commercial program not open source itself, and thus, we take a special look at the available software licensed under this type of license.

3.5.3 Google Goggles

Google Goggles lets the user take a picture of something, and then it analyses the image and uses the result of this analysis to search the web (of course using the Google search engine) for it. They state themselves that “Google Goggles works better with certain types of queries. Try taking pictures of books & DVDs, landmarks, logos, contact info, artwork, businesses, products, barcodes, or text. Currently, it's not so good when taking pictures of animals, plants, cars, furniture, or apparel." Unfortunately, Google Goggles is proprietary and not applicable for non-Google developers.

3.5.4 Google Sky Map

Google Sky Map uses GPS, the clock and the compass of the phone to show which stars and planets are in the sight of the phone.Of course, this updates as the user moves the phone around and looks in different angles et cetera. Also, the program supports several different layers, allowing the user to specify what of the information he is interested in. Google Sky Map, as Goggles, is proprietary.

3.5.5 Google Maps

Google Maps works in general much the same way as Google Sky Map, but focusing on the earth and not the skies. In addition, Google Maps can use mobile phone tower triangulation to locate the user, in case GPS is not activated. This means that it uses the footprint from each mobile phone tower to find an approximate location for where you are.[37] Also, Google Maps is synchronized with a database of shops and enterprises nearby, as well as other services from Google and several different map layers.

3.5.6 Wikitude

Wikitude works in much the same way as Google Maps does. It uses GPS and compass to locate the user, and then searches through a user defined set of databases (so called “wikitude world" for hits in those near the user. Among these “worlds" are well-known services as Wikipedia, Flickr, Foursquare, Youtube, Twitter and many quite unknown ones such as \Irish pubs nearby" or “1800TravelBooking.com". The results from the search may be shown on a map (for this, it actually uses Google Maps), in a simple list or by using the phone's camera. When using the camera, the discovered items from the search which is located in front of the user is shown on the screen. This way, what hits are dis-played to the user depends on the direction the user is holding the phone. Either way, the user is shown what is found and in which wikitude world it originated. She can also easily get information like how far away she is from the place and often a picture as well.

3.5.7 ZXing

Later in this chapter, (3.3) we discuss QR codes, including what a QR code is and how it works. The QR codes are well documented and standardized, and there are many Android applications that read QR codes and return the information encoded in them. One example of such is Barcode Scanner, which as of 6th of October is the sixth most popular application on the entire Android Market (number one is the earlier mentioned Google Maps). Barcode Scanner is licensed under an Apache license, which states that “the Apache License allows the user of the software the freedom to use the software for any purpose, to distribute it, to modify it, and to distribute modified versions of the software, under the terms of the license.

The Apache License, like BSD licenses, does not require modified versions of the software to be distributed using the same license (in contrast to copy left licenses) Taking this into account, we may take use of the source code of this project for our purposes, and adapt it to our application.

3.5.8 Layar

Layar is an “augmented reality browser", available for both iPhone and Android. One can say that Layar is the available application that is closest to what our final application will be, if we bear in mind our product backlog (Chapter 6) and project description (Chapter 1.6), and notice that the constructors of Layar describe their application this way: \The idea is simple: Layar works by using a combination of the mobile phone's camera, compass and GPS data to identify the user's location and field of view, retrieve data based on those geographical coordinates, and overlay that data over the camera view. "[41]. One can see that this is quite similar to the requested features of our project, but on the other hand, the application area differs significantly from ours. Layar is meant for outside use, finding local cafes, restaurants, shops et cetera, while our library will serve as the platform for pervasive games. Nevertheless, it could be quite useful to adapt at least parts of the Layar code for our application, though we are not able to do so: the terms and conditions of Layar clearly states that \You agree not to modify, copy, or create derivative works based on the Services. You will refrain from selling, trading, reselling, leasing, renting, loaning and distributing the Services for any purposes."

3.5.9 ARToolKit

ARToolKit is a software library that provides the ability for the user to see a 3D overlay on a physical marker. This is usually a paper with some black and white squares on it, and ARToolKit can then handle both orientation and moving of the physical object. We do not see a clear link between the scope of this project and the use of ARToolKit, but we recommend the customer to have a closer look at it while developing new products based on our library. The ARToolKit can be used at a later moment for providing some nice extra features, and here, we notice that it has been ported to the Android platform, in a project called AndAR. This project has a GPL license, meaning that the use of it will require either releasing the actual application under the same open source license, or making a specific deal with the developer. The ARToolKit itself is also defined under this license, but on their web page, they explicitly state "ARToolKit is made available freely for non-commercial use under the GNU General Public License. Commercial licenses to a professional implementation of ARToolKit are available for users for whom the GPL is not suitable, or who require a higher level of support, or who require customization or other specialist modifications. Commercial licenses are administered by ARToolworks, Inc., Seattle, WA, USA.Click here for more information on ARToolKit licenses and your usage options."

3.6.Android NDK and JNI Bindings

JNI

The Java Native Interface is a programming framework that allows Java code running in a Java Virtual Machine (JVM) to call and to be called by native applications (programs specific to a hardware and operating system platform) and libraries written in other languages, such as C, C++and assembly.

Android NDK

The NDK is the native development kit. It allows programmers to compile native C/C++ code to a shared library le. This shared library's functions can be called from java by the use of JNI. The use of the NDK together with JNI can significantly increase the performance of an android program. But this does not mean that the whole android program can just be written in C to make it faster. Transferring variables from java to native is a slow operation. So the best use of native code is when the computational cost is significantly larger than the amount of data exchanged between the layers by JNI. In Android version 2.2 JIT, just in time, compiling was introduced. JIT compiles java byte code to native code before running it. This JIT compiling has been proven to significantly increase the speed of android programs. This could reduce the amount of native code, one would have to write to get adequate speed.

3.7.Server-side technologies

Both as a consequence of the requests from the customer as well as our own understanding of the problem, we ended up using a client-server architecture, considering the phone as the client and the server being an external computer. This is for the key point matching described in the section about SURF, and also for the database containing the additional information of each object of interest, marked with an identifier such as a QR code. This latter part could have been done on the client without using too much battery power nor time, but this would require the user to update the application every time a new object is added or the description of an existing object is changed, leading the user into frustration and being bad usability.

For the server side of this database part, our customer requested us to make them able to use their existing infrastructure; Linux, Apache, MySQL, PHP - LAMP. This is a very commonly used program combination on database servers, and also the solution we would have proposed ourselves if there was a need. Linux is an operating system kernel, Apache is an HTTP web server, MySQL a database management system and PHP is a server-side programming language. The AMP-part is very quick to set up on a computer that already runs Linux, and PHP is a very simple language. This, in addition to the close connection between PHP and MySQL making the communication between these two very straight-forward, makes this part easy to create, even though the group had very little experience with PHP in front of the project.

Regarding the key points matching, the goal of executing it on the server was to speed it up. This goal made obvious the choice of C++. To improve the matching speed, we made some research, described in the appendix H. This has lead us to use a KDTree representation, that is a binary tree allowing to store and compare K points in a high D dimension. To select the relevant matching pairs we use the ratio between the two nearest neighbors of the received feature and compare it to a threshold, as described in the same appendix.

4. WORK PROPOSAL

4.1.Pre-Study Analysis & Approach

The proposed solution to fulfill the Business Objectives is based on the following analysis:

- ✓ This Application should be easily ported and integrated to Next generation devices (like Google Glass) in future.
- ✓ The augmented images should be properly integrated. The application should be light and should be free of anomalies.
- ✓ Discuss and come to a common conclusion on the interfaces to be used (input and output) during the initial phases of the project.

- ✓ Multiple solution approach, dependencies and risks are identified and are reviewed with Customer. Based on this the delivery plan and schedule is worked upon by both parties.
- ✓ Usage of third party tools such as Wikitude Studio.
- ✓ Existing software architecture can be re-used by building on top of it.
- ✓ Pure object oriented design methodology to be followed for implementation.

4.2. Proposed Solution

The proposed solution should be compatible with all the Android devices and can later be migrated to other platforms or devices. There should minimal space utilization and service resource usage.

This solution addresses the retail store shopping experience by utilizing the concept of augmented reality on a retail store products (ex: Laptop). The aspects of intuitive product description, product demo, warehouse space navigation etc. have all been proposed in the solutions.

The solution would be completed in 3 or 4 phases with each phase as an improvement to the latter phase. The solution is targeted at making the shopping experience of the customer an unforgettable one thus increasing the productivity and popularity of the retailer.

The solution would utilize third party tools such as **Android studio**, **Wikitude Studio** and **Zxing library** that would make the app function better and stand out among the other applications that are available in the market.

4.3.Technologies

- ✓ Android Studio
- ✓ Android SDK
- ✓ Programming Language - Java
- ✓ Wikitude Studio
- ✓ Zxing Library for QR code and Bar code

5. Requirement Analysis and System Specifications

5.1. Feasibility Study:

Depending on the results of the initial investigation, the survey is expanded to a more detailed feasibility study. Feasibility study is a test of system proposal ac- cording to its work ability, impact on the organization, ability to meet user needs, and effective use of resources. The objective for this phase is not to solve the problem but to acquire a sense of scope. During the study, the problem definition is crystallized and aspects of the problem to be included in the system are deter- mined.

Mobile Application Development Systems are capital investments because resources are being spent currently in order to achieve benefits to be received over a period of time following completion. There should be a careful assessment of each project before it is begun in terms of economic justification, technical feasibility, operational impact and adherence to the master development plan. We started the project by listing the possible queries that the user might want to be satisfied. And on these lines we guided the project further.

The three main points, kept in mind at the time of project, are:

- Possible (To build it with the given technology and resources)
- Affordable (given the time and cost constraints of the organization)
- Acceptable (for use by the eventual users of the system)

The three major areas to be considered while determining the feasibility of a project are:

1. **Technical Feasibility:** The technical issue usually raised during the feasibility stage of

the investigation includes the following :

Does the necessary technology exist to do what is suggested?
Do the proposed equipment have the technical capacity to hold the data required to use the new system?

Will the proposed system provide adequate response to inquiries, regardless of the number or location of users?

Can the system be upgraded if developed?

Are there technical guarantees of accuracy, reliability, ease of access and data security?

Earlier no system existed to cater to the needs of Secure Infrastructure Implementation System. The current system developed is technically feasible. It is a web based user interface. Thus it provides an easy access to the users. The databases purpose is to create, establish and maintain a work- flow among various entities in order to facilitate all concerned users in their various capacities or roles. Permission to the users would be granted based on the roles specified. Therefore, it provides the technical guarantee of accuracy, reliability and security. The software and hardware requirements for the development of this project are not many and are already available as free as open source. The work for the project is done with the current equipment and existing software technology. Necessary bandwidth exists for providing a fast feed- back to the users irrespective of the number of users using the system.

2. **Operational Feasibility:** Under this category of service we conduct a study to analysis and deter-mine whether your need can be fulfilled by using a proposed solution. The result of our operational feasibility Study will clearly outline that the solution proposed for your

business is operationally workable and conveniently solves your problems under consideration after the proposal is implemented. We would precisely describe how the system will interact with the systems and persons around. Our feasibility report would provide results of interest to all stakeholders. It will do as per the needs of the business requirements.

3. **Timeline Feasibility:** It is important to understand that a need must be fulfilled when it has to be. Some otherwise feasible and highly desirable projects can become non-feasible due to very restrictive timeline constraints. This fact makes it imperative that milestones are clearly linked to the timeline and projects are well conceived with safe unforeseen margins. We make sure that we strictly follow what has been stated above.

5.2.Software Requirement Specification Document:

Data Requirements:

Data requirement is meant to be the data that will be used in our application. Data required in this project is Product QR codes and Product marker or Product is used in this application. So two main requirements are:

- Product QR codes and
- Product marker or Product

Functional Requirements:

In order to make this application functional, we require the following:

- **Download mobile application:**

  A user should be able to download the mobile an application through either an application store or similar service on the mobile phone. The application should be free to download.

- **Internet in Mobile application:**

  The device must need to have internet to work fully.

- **Navigation in Application:**

  The user should be able to navigate back to and fro and it is properly implemented

Performance Requirements:

The requirements in this section provide a detailed specification of the user interaction with the software and measurements placed on the system performance.

- **Response Time:**

  The response time should not be more than 5 seconds to scan the QR code if scan is in correct process.

- **Fault Tolerance:**

  The fault tolerance of the system should be very good. If the system loses the connection to the Internet or the system gets some strange input, the user should be informed.

System Dependability:

Following are the requirements that an application require from the device/mobile on which it is installed.

- **Internet Permission:**

  Application developed, require full internet permissions of mobile so that it can fetch the augmented content and overlay on original content.

- **System Tools:**

  This application require various system tools to be used. For example, it requires Camera of mobile.

- **Hardware Control:**

  It uses Camera.

- **Account Info:**

  It also fetches your google account information in order to get the user registered with Google Cloud Messaging.

Maintainability Requirements:

Following are the maintainability requirement:

- **Application extendibility:**

  The application should be easy to extend. The code should be written in a way

that it favors implementation of new functions. It is requires in order for future functions to be implemented easily to the application.

- **Application testability:**

Test environments should be built for the application to allow testing of the applications different functions.

Look and Feel Requirements:
Regarding look and feel, our client is straight forward. They believe in simplicity. So these are their requirements:

- **Simple and Light:**

The user interface should be simple and lightly colored. It should give relaxing effect on looking at its GUI. No bright colors should be used while designing the UI of this application.

- **Easy to Use:**

The application should be easy to use. If any user is doing something wrong, he/she should be informed correctly, what is going wrong behind the scene. There should be proper instructions for the user to use this application.

- **Soft Sound Notification**:

The sound for notification should be very soft. It should not disturb the peers with a loud note. Everything should be sober in this application.

5.3.Validations:

Any application is useless without validation. There should be a way to validate the user input first before sending the user request to the server. Following are the validations implemented in proposed system:

QR code Validation:

The application must scan the product bar or QR code the application must check that the code is valid which is not empty or don’t have information which is unrelated.

5.4.Expected Hurdles:

The main hurdles that can come in the notice are as follow:

- **Device SD Card:**

  The application will be requiring access to the SD card of the user mobile. It may be possible that SD Card is full or missing SD Card. So the user will not be able to install application.

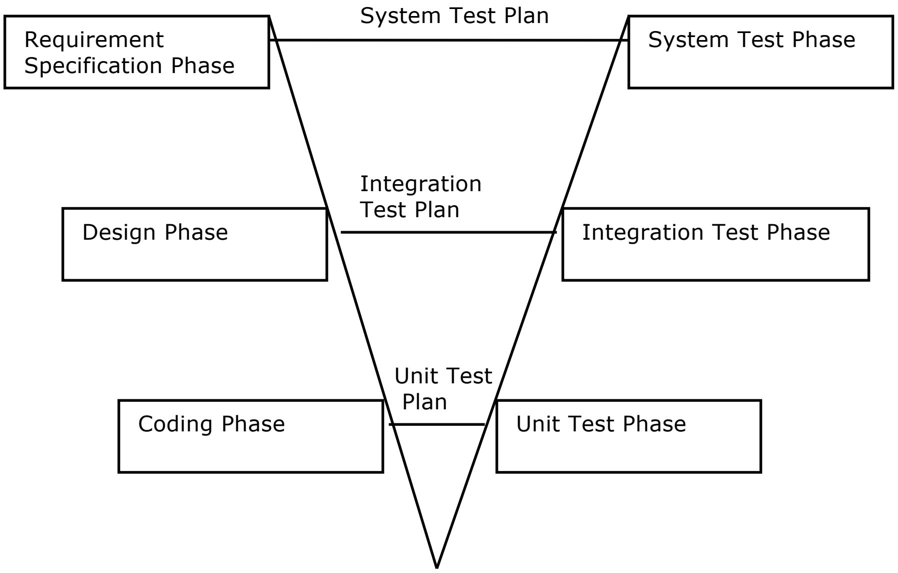
- **Google Play Services:**

  Registering to Google Play services on device installed. Other-wise user will not be able to install application or need to have the access to website from where the application download.

5.5.SDLC Model Used:

The development for this project follows the **V-Process Model**. The model is suitably customized according to the needs of this project.

Following are the different phases of the model for the proposed project:



- Requirement Specification Phase
- Design Phase
- Coding Phase
- Unit Test Phase
- System Test Phase
- Integration Test Phase

Requirement Specification Phase: The goal of this phase is to produce requirement specification document of the proposed system. Identify and document all the functional and non-functional requirements completely and unambiguously. Also, a system test plan is prepared explaining the approach for testing the features and functionality of the project. The system test cases are prepared based on the test plan.

Design Phase: The goal of this phase is to produce a design document. Both, a high level and a detailed design document will be prepared. An appropriate design methodology suitable for the project is decided and a detailed design document is prepared. An integration test plan is prepared explaining the approach for testing the interfaces between various units of the project.

Coding Phase: The main activity of this phase would be developing source code based on the detailed design document. A unit test plan is prepared explaining the approach for testing the implementation of features and functionality of each component or each module of the project. Unit test cases are prepared based on the Unit Test Plan.

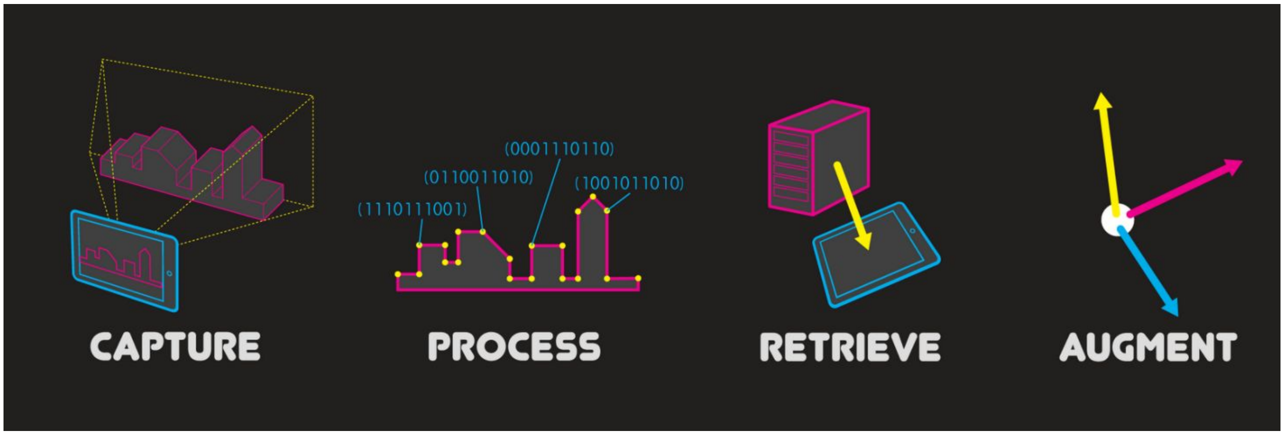
Unit Test Phase: The individual components of the project are tested according to the procedure defined in the Unit Test Plan. The unit test cases are executed and their results are recorded.

System Test Phase: The system is tested, according to the test procedures defined in the System Test Plan. Moreover, the test cases are executed and the results are recorded.

Integration Test Phase: The system is tested, according to the test procedures defined in the Integration Test Plan.

5.6. Functional Architecture

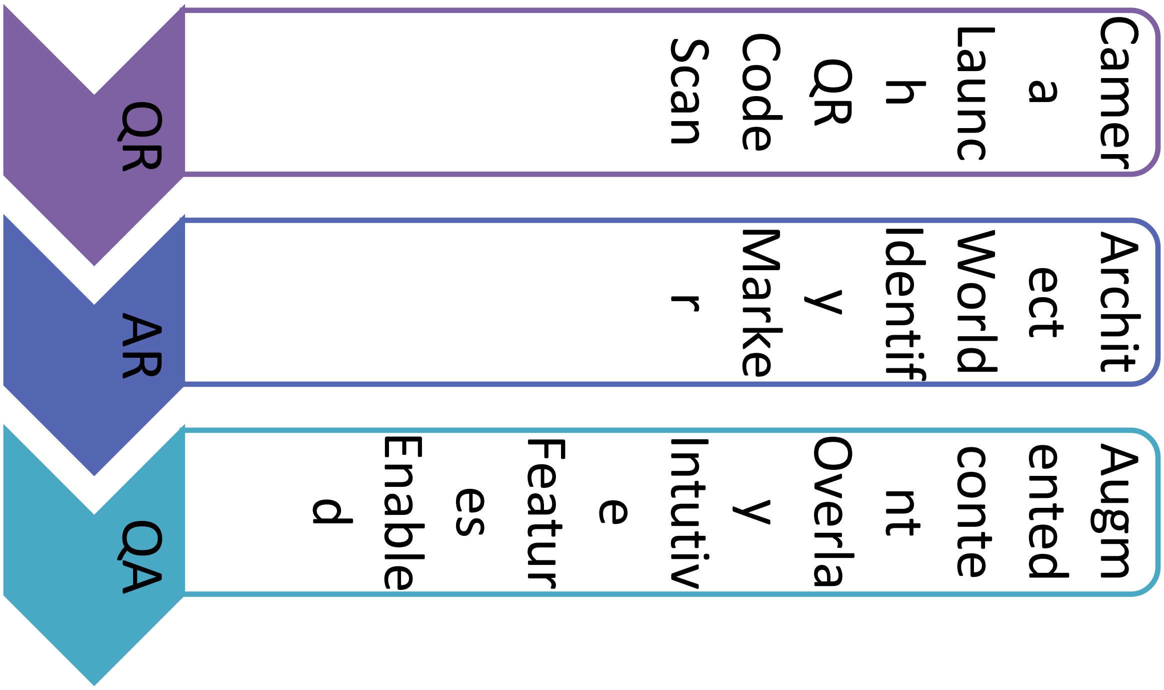
Scan the Product code Bar or QR code



Process of Augmentation

Augmentation on Smart Phone

5.7. Technical Architecture


QR
Camera Launch QR Code Scan
AR
Architect World Identify Marker
QA
Augmented content Overlay Intutive Features Enabled

QR – Quick Response

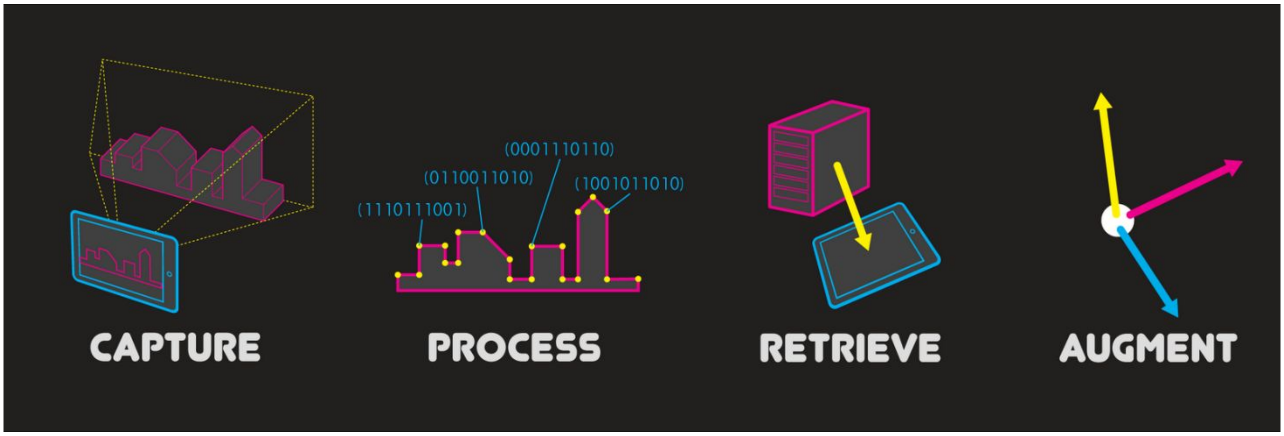
Based on the pre-study analysis using Zxing code library to scan the QR codes.
Process of scanning – The application launch the camera and scan QR code and decode the content and identify product

AR – Augmented Reality

Based on the pre-study analysis using Wikitude a Web based augmented reality library.
The decode code from QR is used and open the Architect View to get the exact context

QA – Quick Augment

The Application Architect View with context must identify marker and compare marker image with hosted marker in Wikitude studio server or our own server if matched display the augmented content overlay on the marker .The overlay content is intuitive in nature we can touch and navigate through the content



QA Platfrom on Reatil Store

Capture - Capturing the QR code of the product by invoking the camera from zxing library
Process - Processing the QR code of the product and analyzing the type of product and sending that information to the Architect view
Retrieve – Architect view will identify the Product and compare product image with hosted marker in wikitude studio or own server
Augment- if matching entry is found overlay augmented content on the Product which is intuitive in nature.

6. Implementation, Testing and Maintenance:

Introduction to Languages, IDE's, Tools and Technologies Used for Implementation

Java

Java is a very popular programming language developed by Sun Microsystems (now owned by Oracle). Developed long after C and C++, Java incorporates many of the powerful features of those powerful languages while addressing some of their drawbacks. Still, programming languages are only as powerful as their libraries. These libraries exist to help developers build applications.

Some of the Javas important core features are:

- It is easy to learn and understand.
- It is designed to be platform-independent and secure, using VM.
- It is object-oriented.

Android relies heavily on these Java fundamentals. The Android SDK includes many standard Java libraries (data structure libraries, math libraries, graphics libraries, networking libraries and everything else you could want) as well as special Android libraries that will help you develop awesome Android applications.

Platform Independence Importance

With many programming languages, you need to use a compiler to reduce your code down into machine language that the device can understand. While this is well and good, different devices use different machine languages. This means that you might need to compile your applications for each different device or machine language in other words, your code isn't very portable. This is not the case with Java. The Java compilers convert your code from human readable Java source files to something called byte code in the Java world. These are interpreted by a Java Virtual Machine, which operates much like a physical CPU might operate on machine code, to actually execute the compiled code.

Although it might seem like this is inefficient, much effort has been put into making this process very fast and efficient. These efforts have paid off in that Java performance in generally second only to C/C++ in common language performance comparisons.

Android applications run in a special virtual machine called the Dalvik VM. While the details of this VM are unimportant to the average developer, it can be helpful to think of the Dalvik VM as a bubble in which your Android application runs, allowing you to not have to worry about whether the device is a Motorola Droid, an HTC Evo, or the latest toaster running Android. You don't care so long as the device is Dalvik VM friendly and that's the device manufacturers job to implement, not yours.

Why is Java Secure?

Let's take this bubble idea a bit further. Because Java applications run within the bubble that is a virtual machine, they are isolated from the underlying device hardware. Therefore, a virtual machine can encapsulate, contain, and manage code execution in a safe manner compared to languages that operate in machine code directly. The Android platform takes things a step further. Each Android application runs on the (Linux- based) operating system using a different user account and in its own instance of the Dalvik VM. Android applications are closely monitored by the operating system and shut down if they don't play nice (e.g. use too much processing power, become unresponsive, waste resources, etc.).

Therefore, it's important to develop applications that are stable and responsive. Applications can communicate with one another using well- defined protocol.

Android Development Tools

Android SDK:

The Android Software Development Kit (Android SDK) contains the necessary tools to create, compile and package Android applications. Most of these tools are command line based. The primary way to develop Android applications is based on the Java programming language.

Android debug bridge (adb):

The Android SDK contains the Android debug bridge (adb), which is a tool that allows you to connect to a virtual or real Android device, for the purpose of managing the device or debugging your application.

Android Developer Tools and Android Studio:

Google provides two integrated development environments (IDEs) to develop new applications. The Android Developer Tools (ADT) are based on the Eclipse IDE. ADT is a set of components (plug-ins), which extend the Eclipse IDE with Android development capabilities. Google also supports an IDE called Android Studio for creating Android applications. This IDE is based on the IntelliJ IDE.

Both IDEs contain all required functionality to create, compile, debug and deploy Android applications. They also allow the developer to create and start virtual Android devices for testing. Both tools provide specialized editors for Android specific files. Most of Android's configuration files are based on XML. In this case these editors allow you to switch between the XML representation of the file and a structured user interface for entering the data. Dalvik Virtual Machine The Android system uses a special virtual machine, i.e., the Dalvik Virtual Machine (Dalvik) to run Java based applications. Dalvik uses a custom bytecode format which is different from Java bytecode.

Therefore you cannot run Java class files on Android directly; they need to be converted into the Dalvik bytecode format.

Android RunTime (ART):

With Android 4.4, Google introduced the Android RunTime (ART) as optional runtime for Android 4.4. It is expected that versions after 4.4 will use ART as default runtime. ART uses

Ahead of Time compilation. During the deployment process of an application on an Android device, the application code is translated into machine code. This results in approx. 30% larger compile code, but allows faster execution from the beginning of the application.

Security and Permission Concept in Android

Security Concept in Android:

The Android system installs every Android application with a unique user and group ID. Each application file is private to this generated user, e.g., other applications cannot access these files. In addition each Android application is started in its own process.

Therefore, by means of the underlying Linux kernel, every Android application is isolated from other running applications. If data should be shared, the application must do this explicitly via an Android component which handles the sharing of the data, e.g., via a service or a content provider.

Permission concept in Android:

Android contains a permission system and predefines permissions for certain tasks. Every application can request required permissions and also define new permissions. For example, an application may declare that it requires access to the Internet.

Permissions have different levels. Some permissions are automatically granted by the Android sys-tem, some are automatically rejected. In most cases the requested permissions are presented to the user before installing the application. The user needs to decide if these permissions shall be given to the application.

If the user denies a required permission, the related application cannot be installed. The check of the permission is only performed during installation, permissions cannot be denied or granted after the installation.

An Android application declares the required permissions in itsAndroidManifest.xml configuration file. It can also define additional permissions which it can use to restrict access to certain components.

Coding Standards of Language Used

Coding Standards of Java are used in the whole project. This standard include the following:

- No wildcard imports.
- Overloads appear sequentially.
- Braces are used even when the body is empty or contains a single statement. Two space indentation.
- Column limit can be 80 or 100 characters. No C-style array declarations.
- Default statements required in switch statements.
- Modifiers appear in the order recommended by the Java Language Specification.
- Constants use CONSTANT CASE. Note that every constant is a static final field, but not all static final fields are constants.
- Class name should start with uppercase letter.
- Function name should start with lowercase letter.

Project Scheduling

Project scheduling is concerned with the techniques that can be employed to manage the activities that need to be undertaken during the development of a project. Scheduling is carried out in advance of the project commencing and involves:

- Identifying the tasks that need to be carried out.
- Estimating how long they will take.

- Allocating resources (mainly personnel).
- Scheduling when the tasks will occur.

Once the project is underway control needs to be exerted to ensure that the plan continues to represent the best prediction of what will occur in the future based on what occurs during the development and often necessitates revision of the plan. Effective project planning will help to ensure that the systems are delivered:

- Within cost;
- Within the time constraint;

To a specific standard of quality.

Two project scheduling techniques will be presented, the Milestone Chart and Gantt Chart.

1. Milestone Charts:

   Milestones mark significant events in the life of a project, usually critical activities which must be achieved on time to avoid delay in the project. Milestones should be truely significant and be reasonable in terms of deadlines (avoid using intermediate stages).

   Examples include:
   Installation of equipment.
   Completion of phases.
   File conversion.
   Cut over to the new system

2. Gantt Charts:

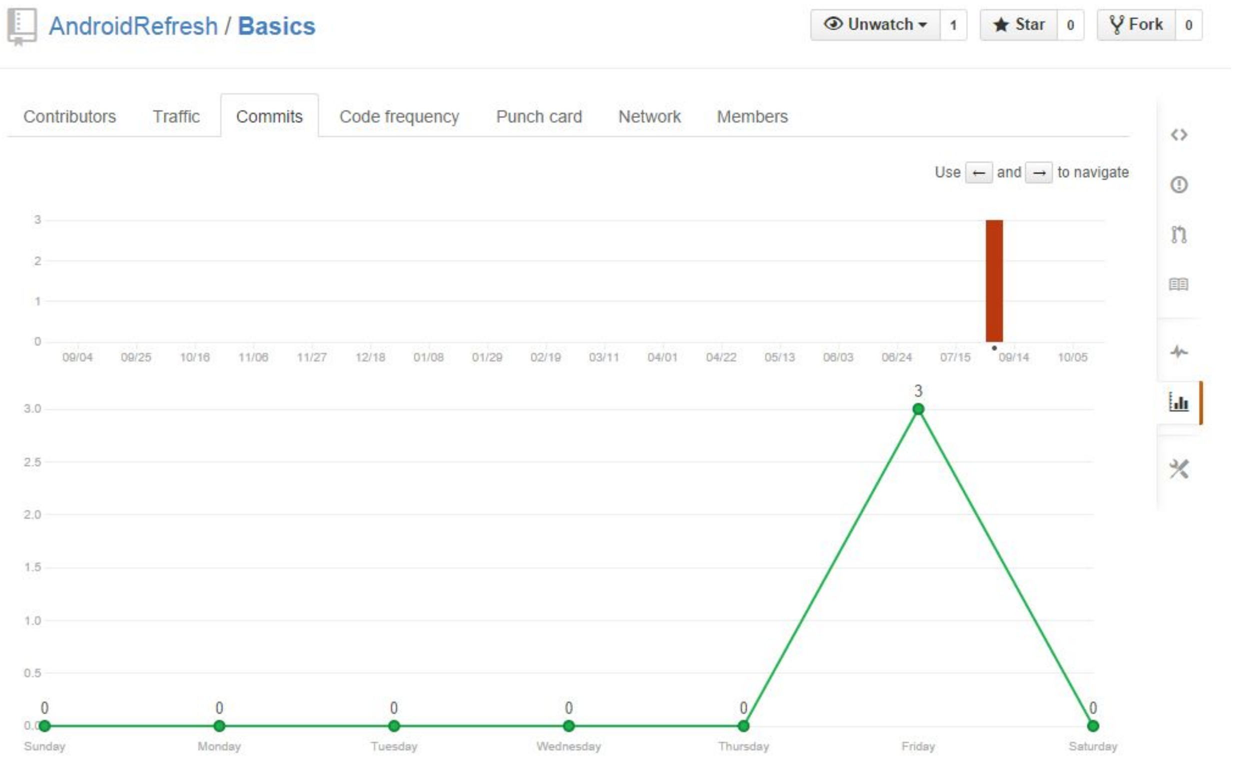
   A Gantt chart is a horizontal bar or line chart which will commonly include the following features:
   Activities identified on the left hand side.
   Time scale is drawn on the top (or bottom) of the chart.
   A horizontal open oblong or a line is drawn against each activity indicating estimated duration.

Dependencies between activities are shown.

At a review point the oblongs are shaded to represent the actual time spent (an alternative is to represent actual and estimated by 2 separate lines).

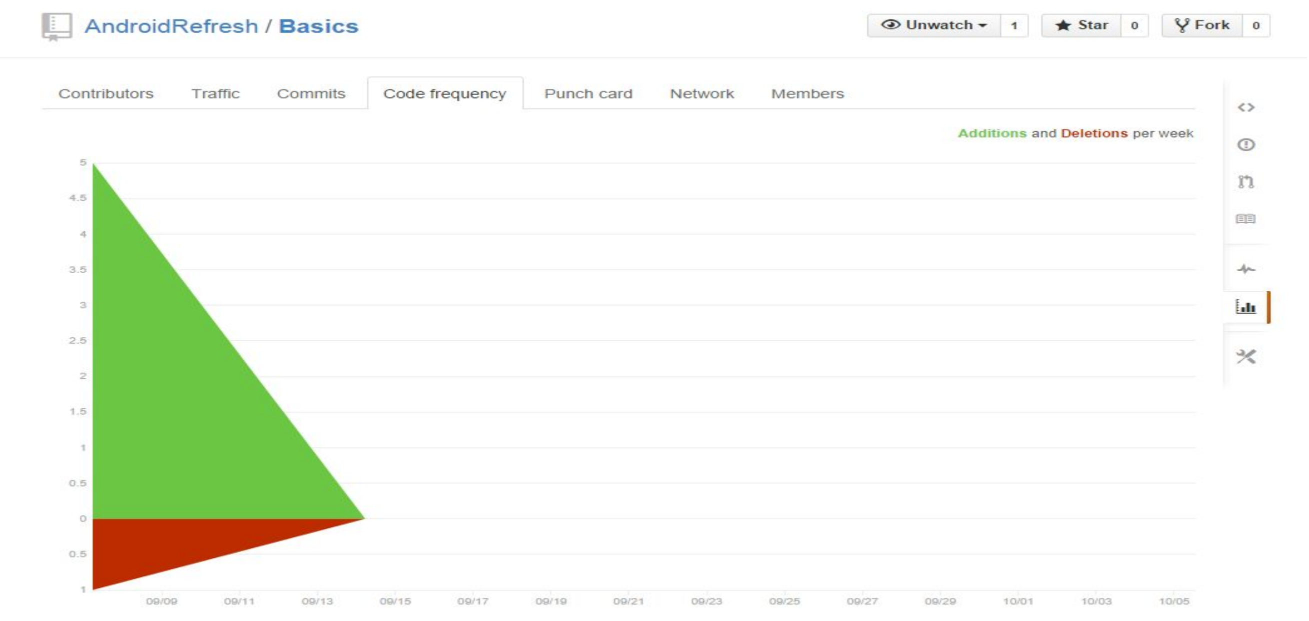
A vertical cursor (such as a transparent ruler) placed at the review point makes it possible to establish activities which are behind or ahead of schedule.

Following is the chart of various commits on github, while developing this application:



Sample Graph of Commits on GitHub

This graph is showing commit of September. It depicts how many times, changes are pushed on to GitHub after testing.



Sample Code Frequency Graph on GitHub

This graph depicts the frequency of code that is pushed on github on respective dates.

Test Plan and Test Activities

Test Plan

A test plan can be defined as a document describing the scope, approach, resources, and schedule of intended testing activities. It identifies test items, the features to be tested, the testing tasks, who will do each task, and any risks requiring contingency planning. In software testing, a test plan gives detailed testing information regarding an upcoming testing effort, including

- Scope of testing
- Schedule
- Test Deliverables Release Criteria
- Risks and Contingencies

It is also be described as a detail of how the testing will proceed, who will do the testing, what will be tested, in how much time the test will take place, and to what quality level the test will be performed.

The process of defining a test project so that it can be properly measured and controlled. The test planning process generates a high level test plan document that identifies the software items to be tested, the degree of tester independence, the test environment, the test case design and test measurement techniques to be used, and the rationale for their choice.

A testing plan is a methodological and systematic approach to testing a system such as a machine or software. It can be effective in finding errors and flaws in a system. In order to find relevant results, the plan typically contains experiments with a range of operations and values, including an understanding of what the eventual workflow will be.

Test plan is a document which includes, introduction, assumptions, list of test cases, and list of features to be tested, approach, deliverables, resources, risks and scheduling. A test plan is a systematic approach to testing a system such as a machine or software. The plan typically contains a detailed understanding of what the eventual workflow will be. A record of the test planning process detailing the degree of tester independence, the test environment, the test case design techniques and test measurement techniques to be used, and the rationale for their choice.

Test Activities

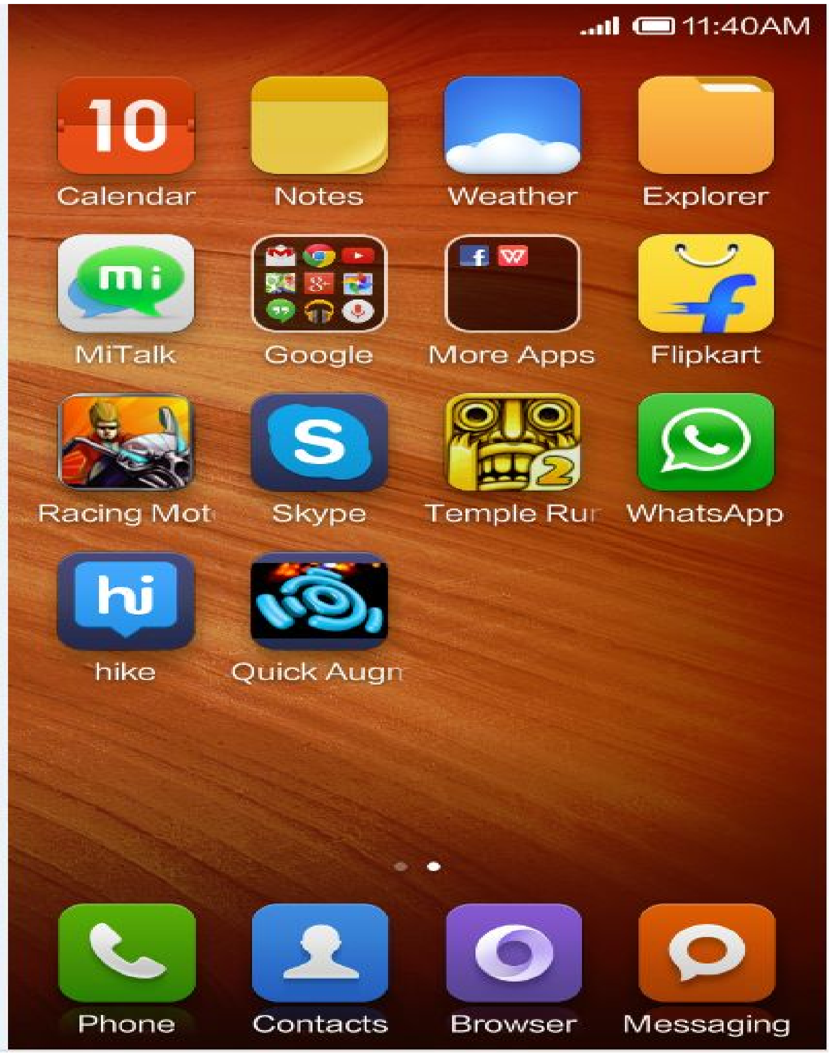
Various Testing Activities are as follow:

1. Black box testing internal system design is not considered in this type of testing. Tests are based on requirements and functionality.
2. White box testing: This testing is based on knowledge of the internal logic of an applications code. Also known as Glass box Testing. Internal software and code working should be known for this type of testing. Tests are based on coverage of code statements, branches, paths, conditions.
3. Unit testing: testing of individual software components or modules. Typically done by the programmer and not by testers, as it requires detailed knowledge of the internal program design and code. May require developing test driver modules or test harnesses.

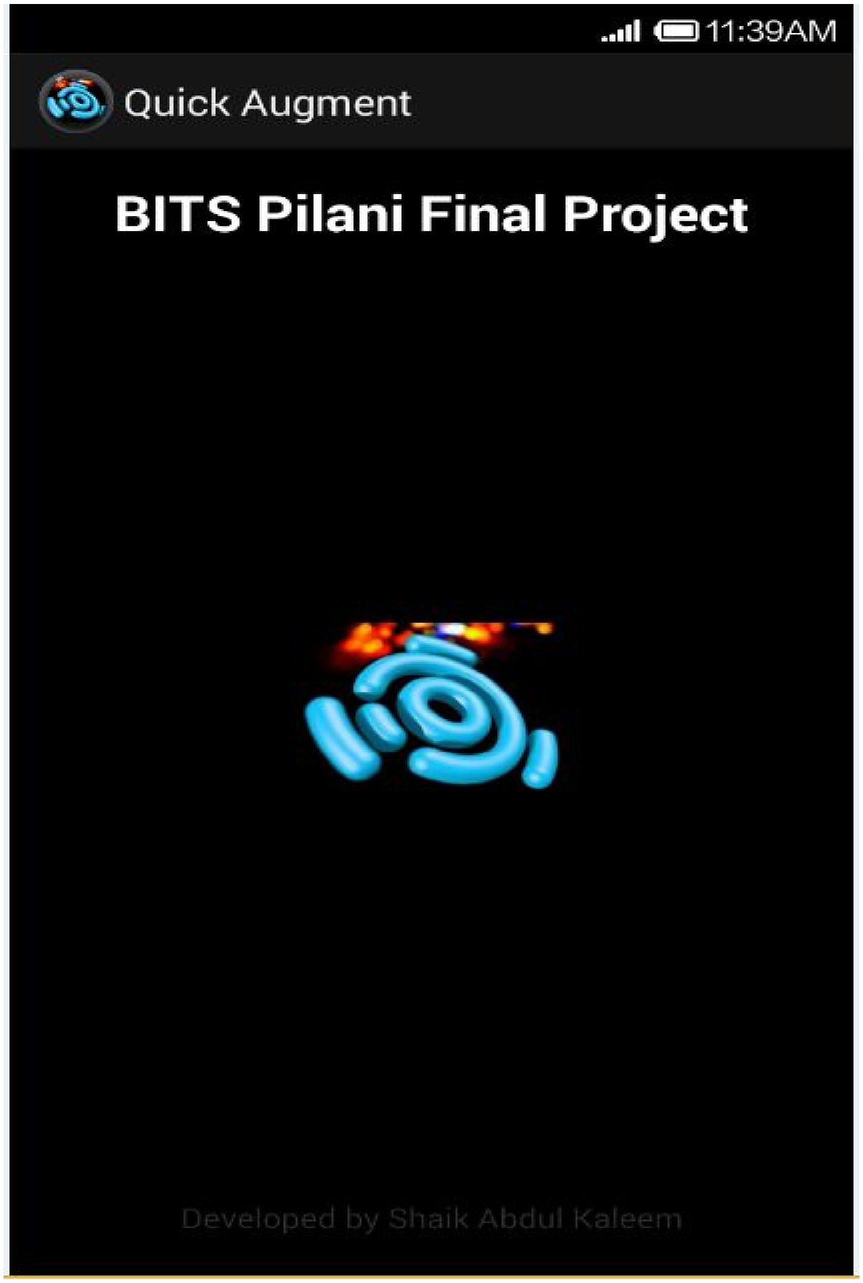
4. Incremental integration testing: Bottom up approach for testing i.e continuous testing of an application as new functionality is added; Application functionality and modules should be independent enough to test separately. Done by programmers or by testers.
5. Integration testing: Testing of integrated modules to verify combined functionality after integration. Modules are typically code modules, individual applications, client and server applications on a network, etc. This type of testing is especially relevant to client/server and distributed systems.
6. Functional testing: This type of testing ignores the internal parts and focus on the output is as per requirement or not. Black-box type testing geared to functional requirements of an application.
7. System testing: Entire system is tested as per the requirements. Black-box type testing that is based on overall requirements specifications, covers all combined parts of a system.
8. End-to-end testing Similar to system testing, involves testing of a complete application environment in a situation that mimics real-world use, such as interacting with a database, using network communications, or interacting with other hardware, applications, or systems if appropriate.
9. Acceptance testing - Normally this type of testing is done to verify if system meets the customer specified requirements. User or customer do this testing to determine whether to accept application.
10. Usability testing User-friendliness check. Application flow is tested, Can new user understand the application easily, Proper help documented whenever user stuck at any point. Basically system navigation is checked in this testing.

6.1.Snapshots of Application.

Home Page



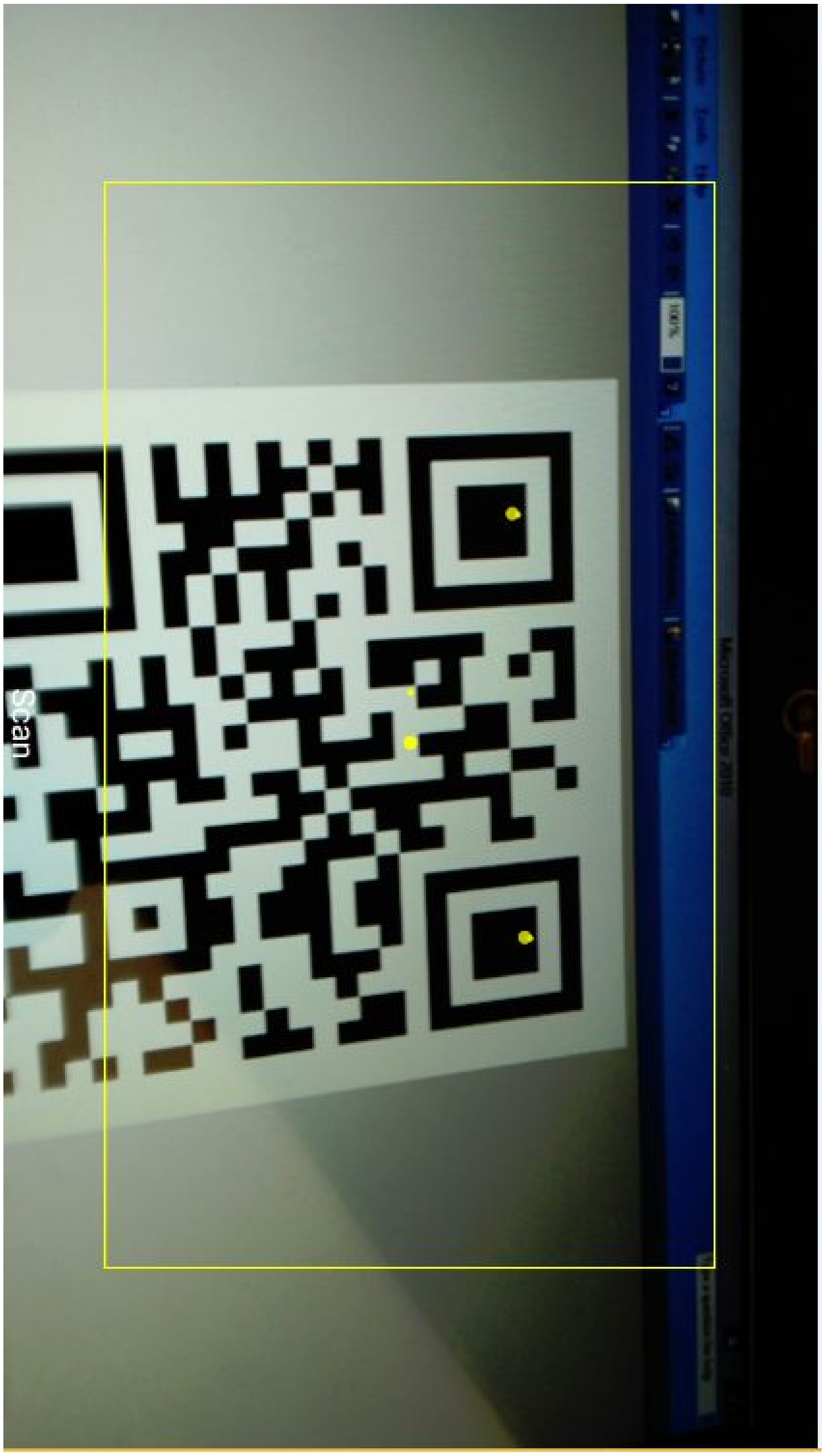
Splash Screen:



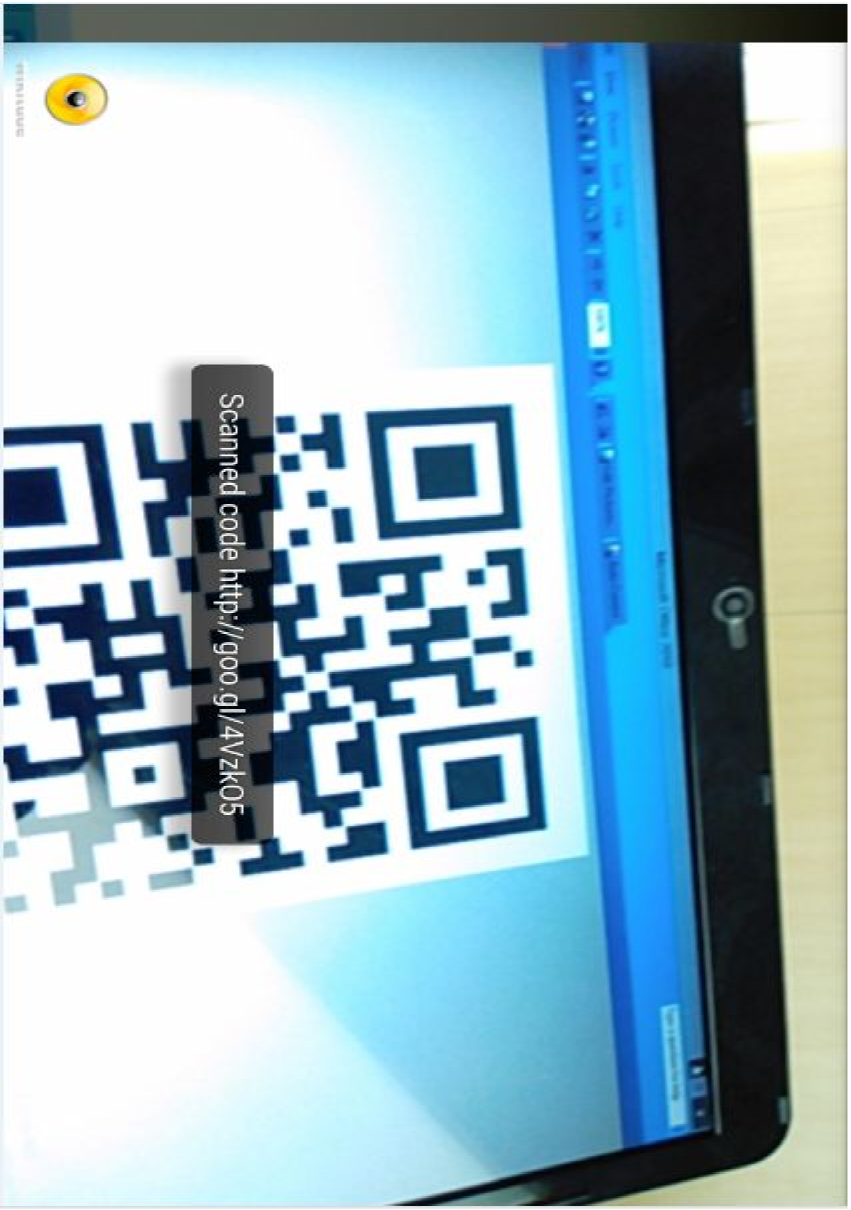
BITS Pilani Final Project

Developed by Shaik Abdul Kaleem

Scanning QR code of Product:

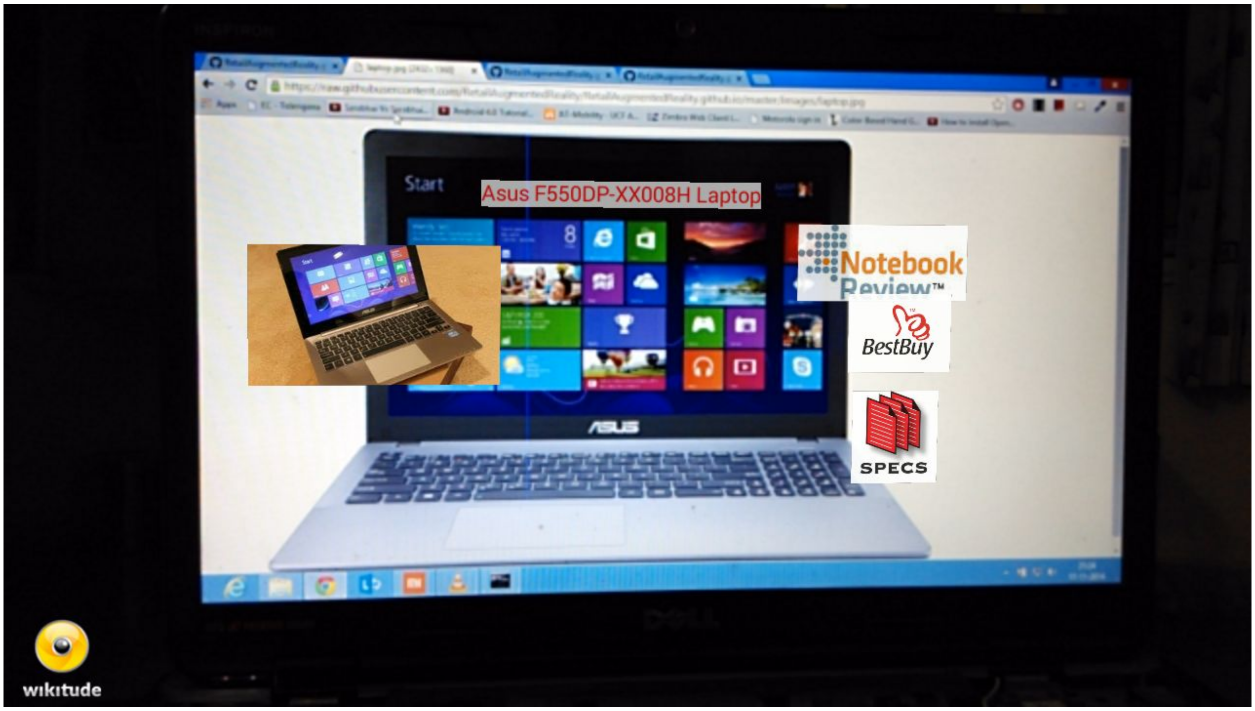


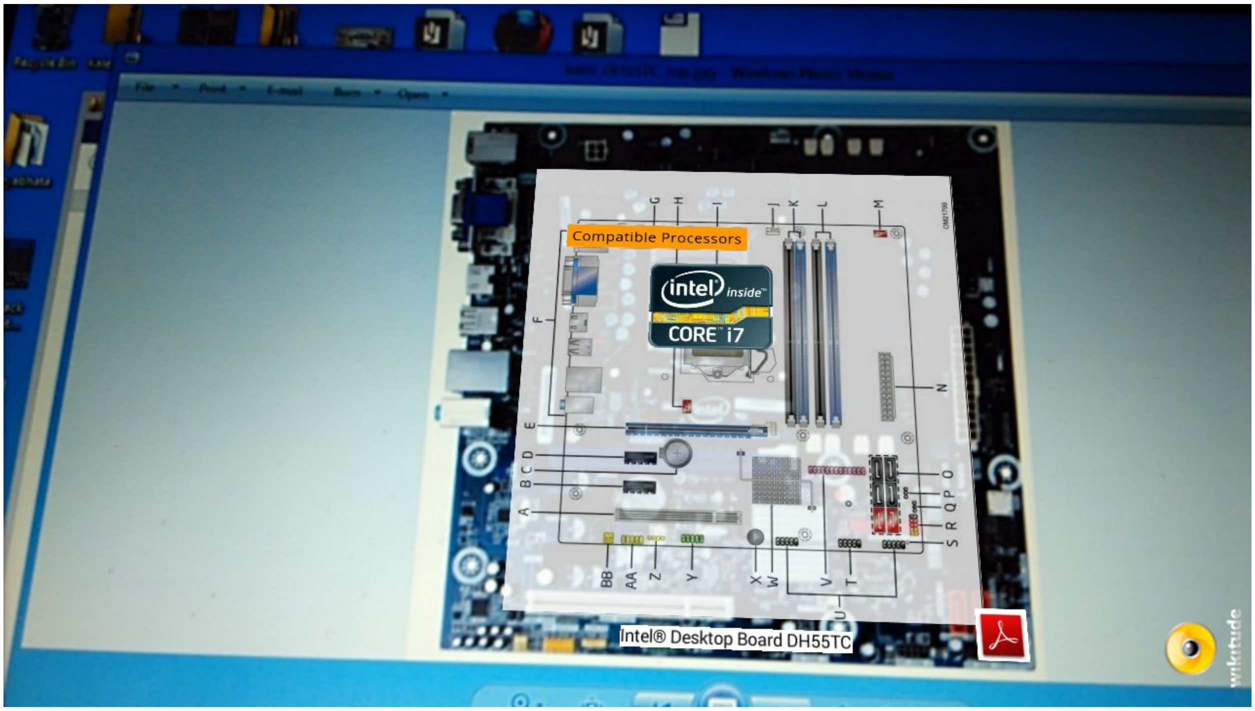
Decode and Architect view:

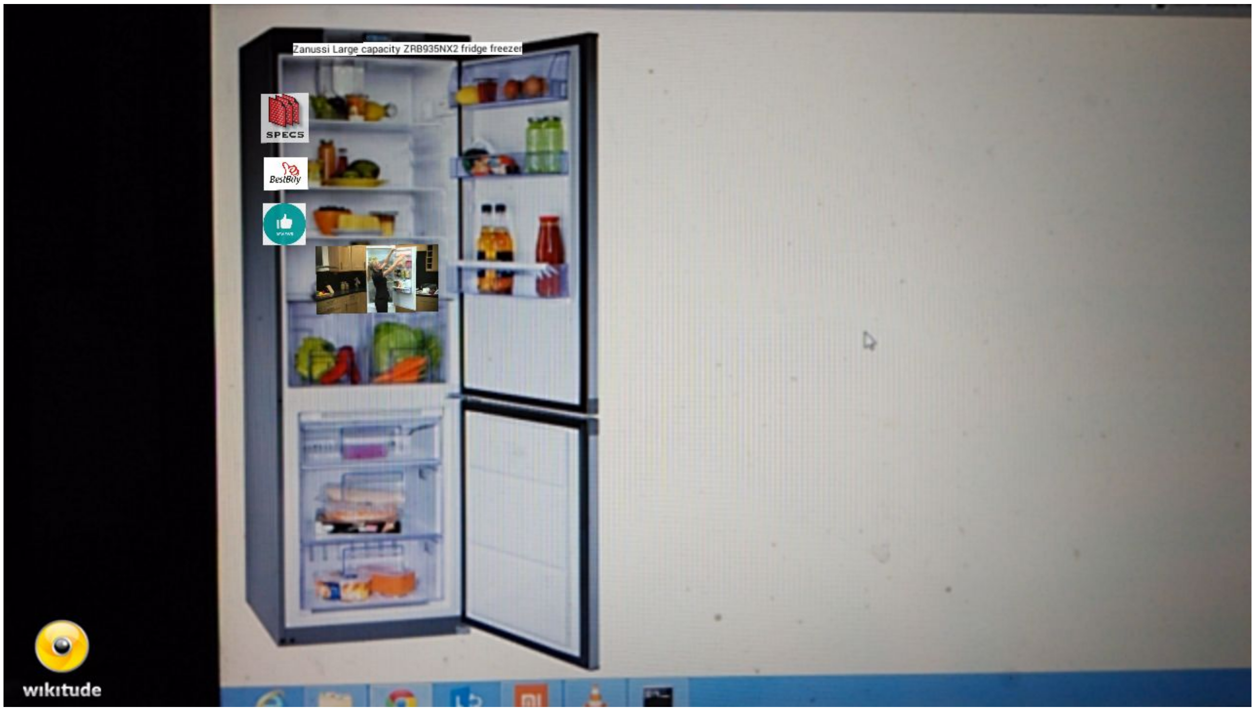


Scanned code http://goo.gl/4VzkO5

Augmented on Product


Asus F550DP-XX008H Laptop
Notebook Review™
BestBuy
SPECS
wikitude


Compatible Processors
intel inside
CORE i7
Intel® Desktop Board DH55TC
wikitude

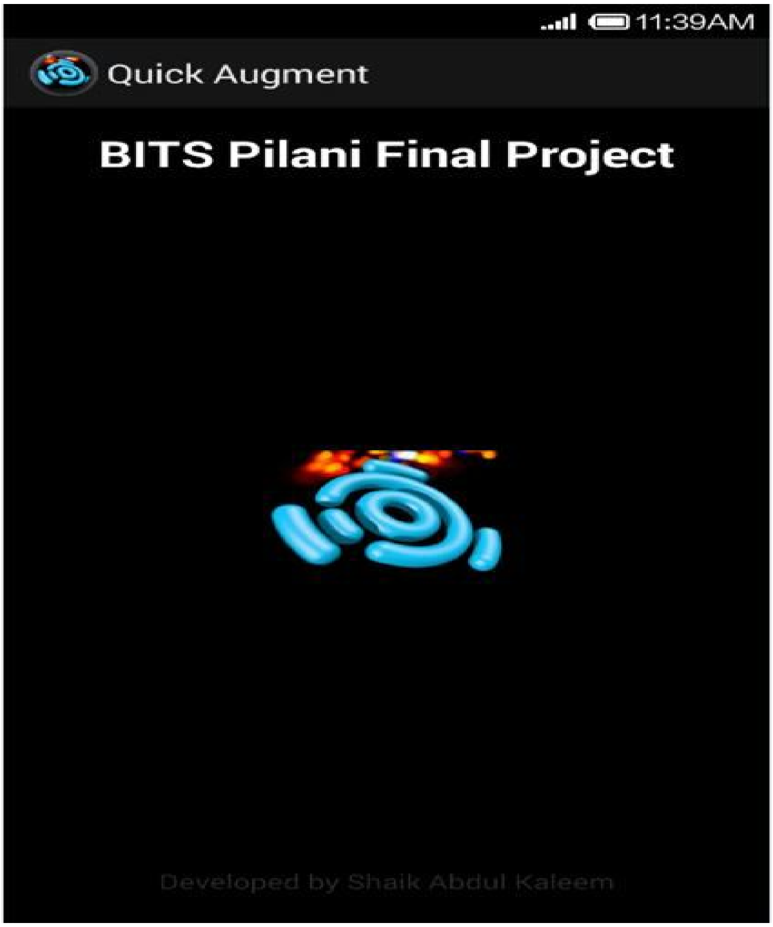


6.2.Assumptions and Constraints

Sl. No.	Assumptions / Constraints
1.	Smartphone with Internet connectivity is required
2.	Internet connectivity errors to be considered in design.

7. Step by Step Explanation of Application

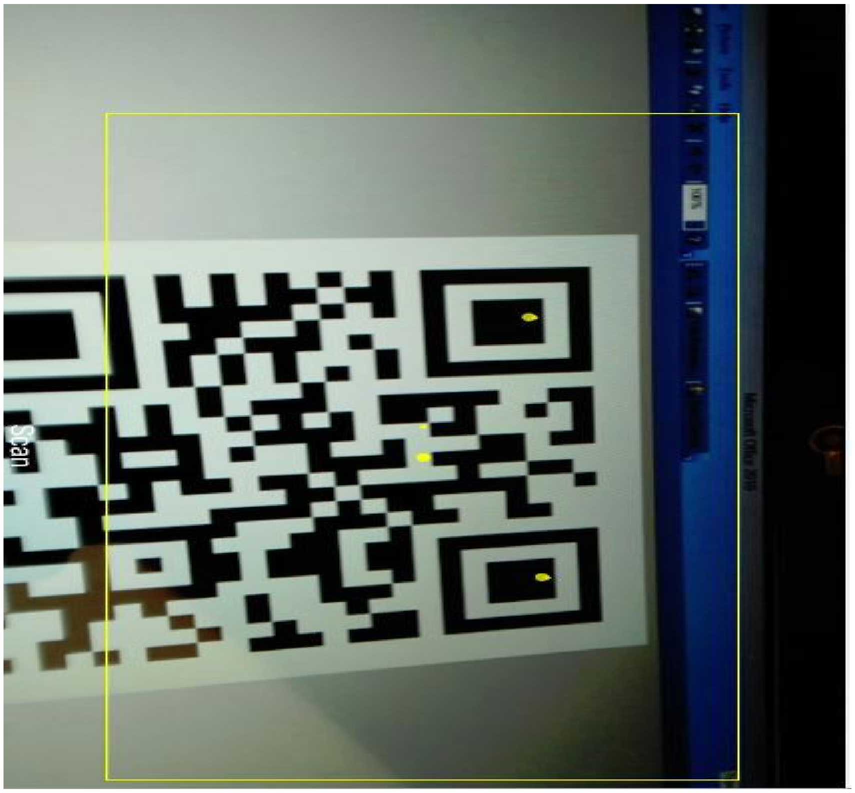
Before preceding you must have enabled internet in your phone and follow steps explained below
Step 1:
Install the APK from
https://github.com/RetailAugmentedReality/RetailAugmentedReality.github.io/tree/master/Android_APK
Note: if you are unable to install go to settings -> security in your Phone Check unknown source is enabled or not if not, enable it and install it.
Step 2:
After installation Open the App, you get splash screen of a developer for a second and app will start
Please see below screen shot



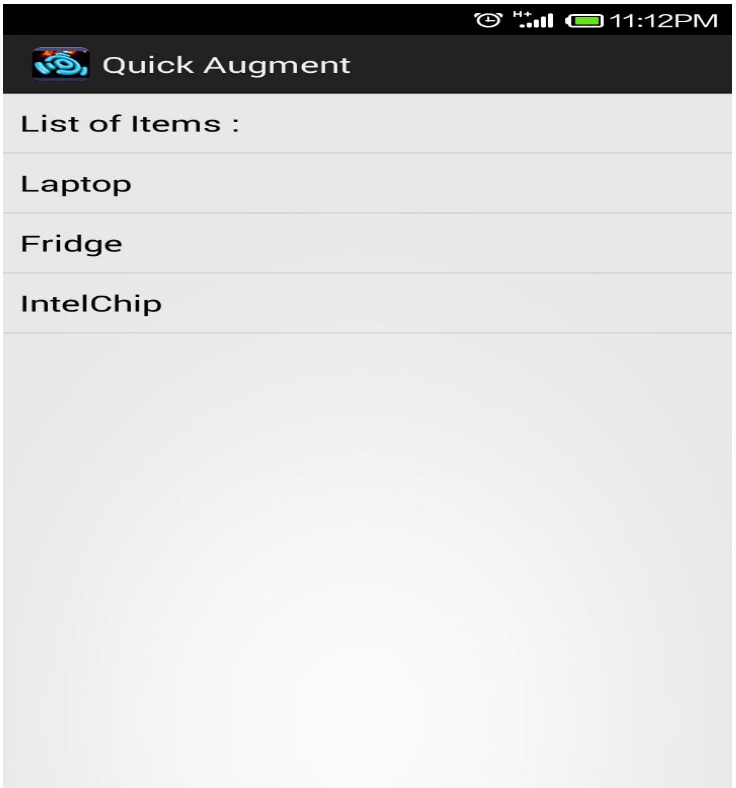
BITS Pilani Final Project

Developed by Shaik Abdul Kaleem

Step 2:
App will open with QR code reader, they are two sections of app, one just knows you about individual product in Shop and other to know the list of items in shop and augment on top of it.
a) If you want to know only about specific product in the shop just scan the QR code of the Product as shown below:



b) If you want to check what are the products in Shop just scan the shop QR code which is in front of shop you will get list of items as shown below screenshot:



Quick Augment

List of Items :

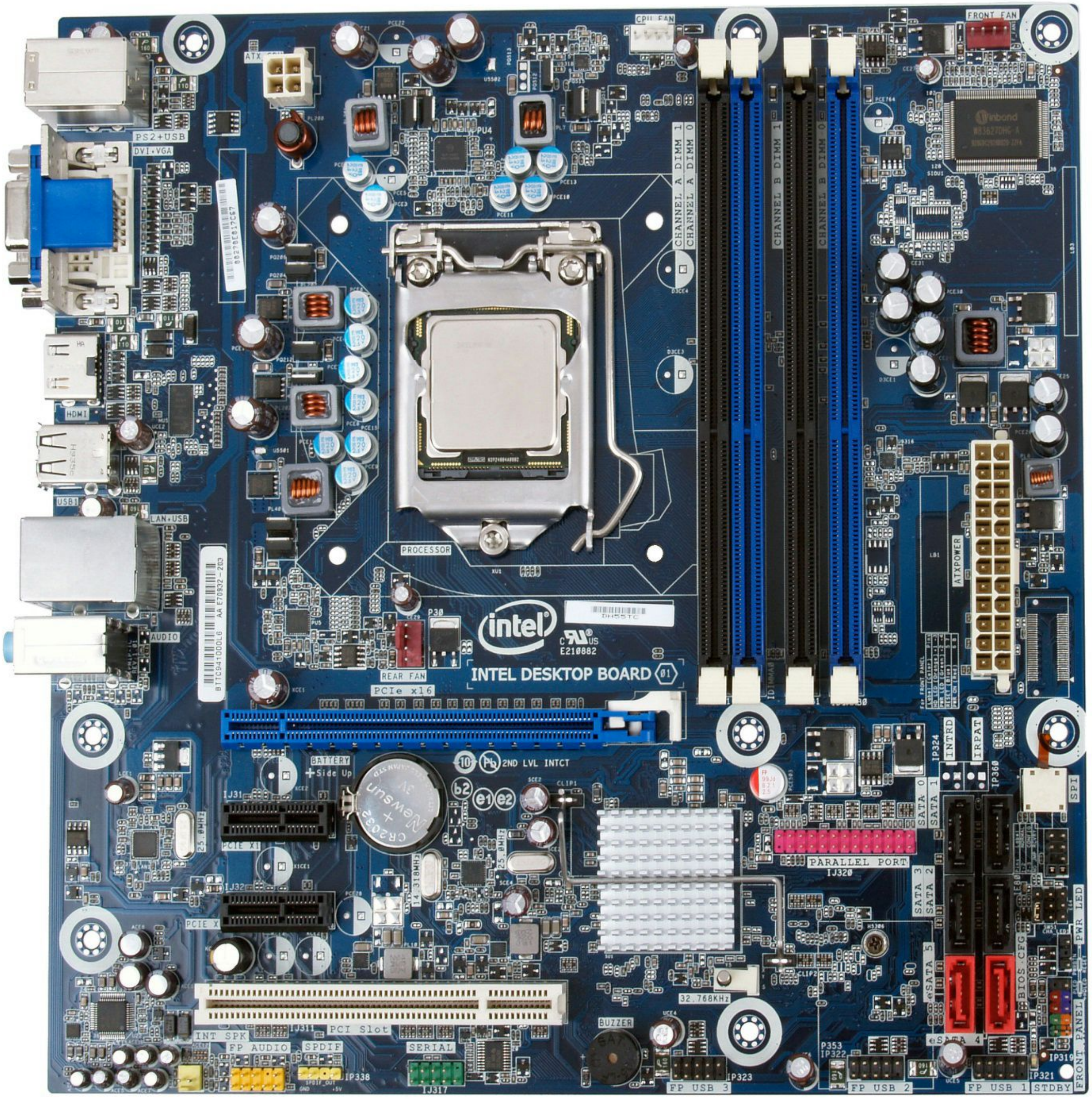
Laptop

Fridge

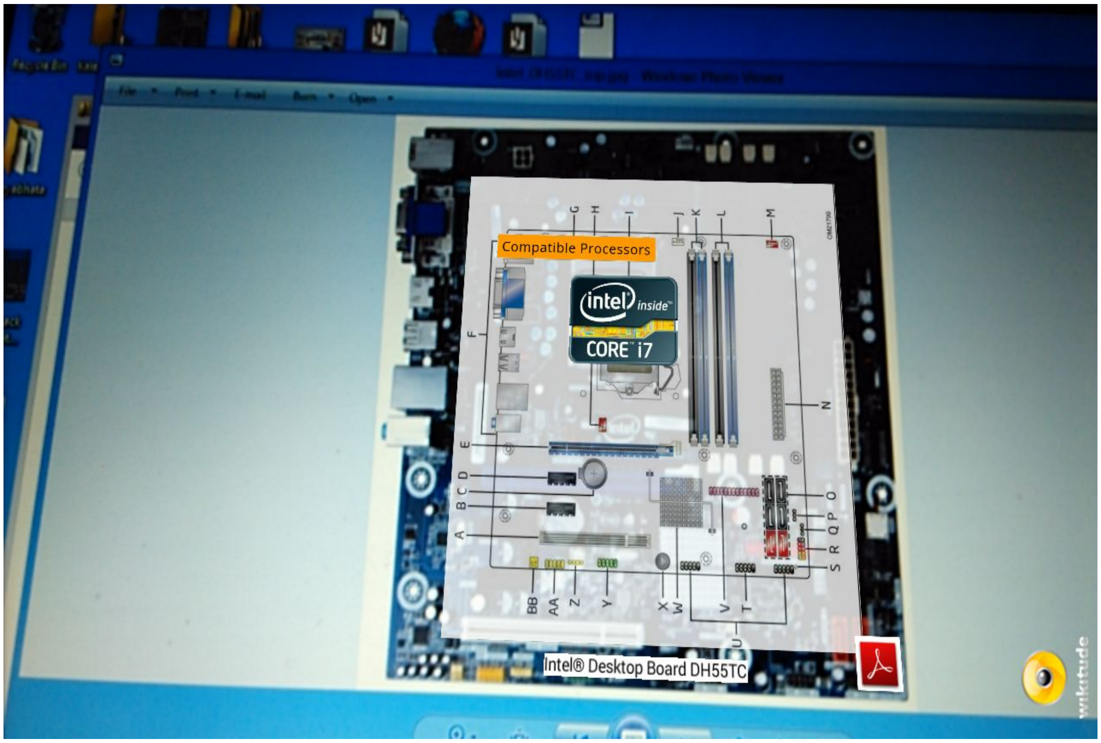
IntelChip

Step 3:
Sample example demo :
If you are following Step 2 a) section then scan this code

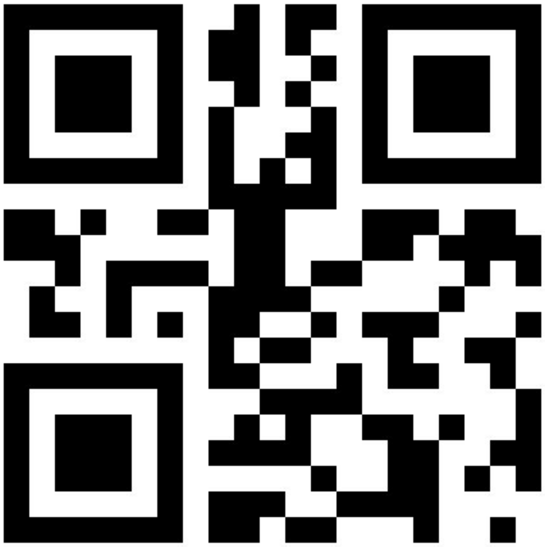
QR will be decoded and your Architect view will open (Camera will open) and see the below product (image) from camera



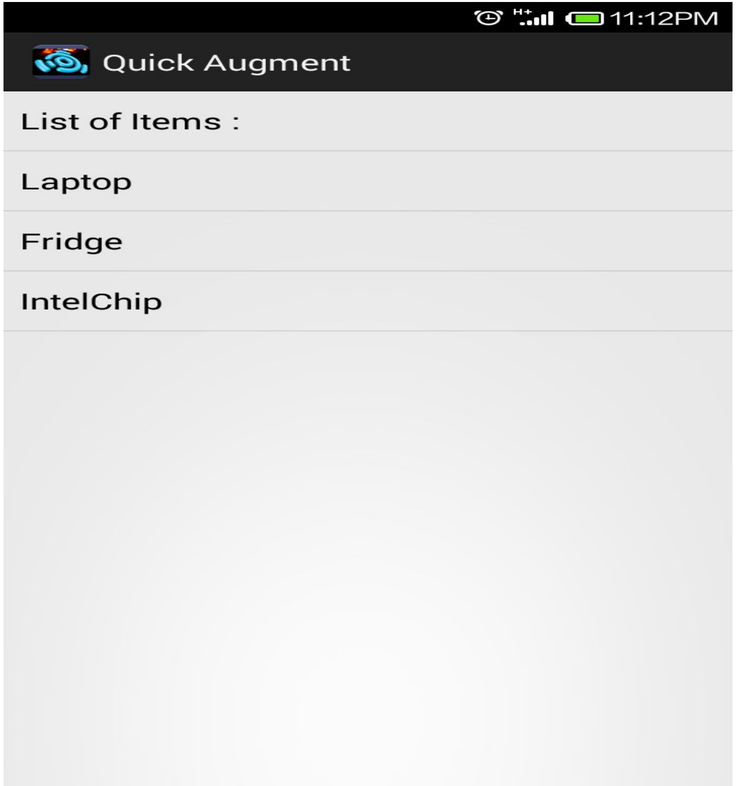
The App will start augmenting and will overlay on real image with some virtual content on it. The augment which is overlaid is intuitive you can click on it and get the relevant information then and there itself see below figure:



If you are following Step 2 b) section then scan this code



After scanning this code it will get you products list
See below screenshot will appear



List of Items :

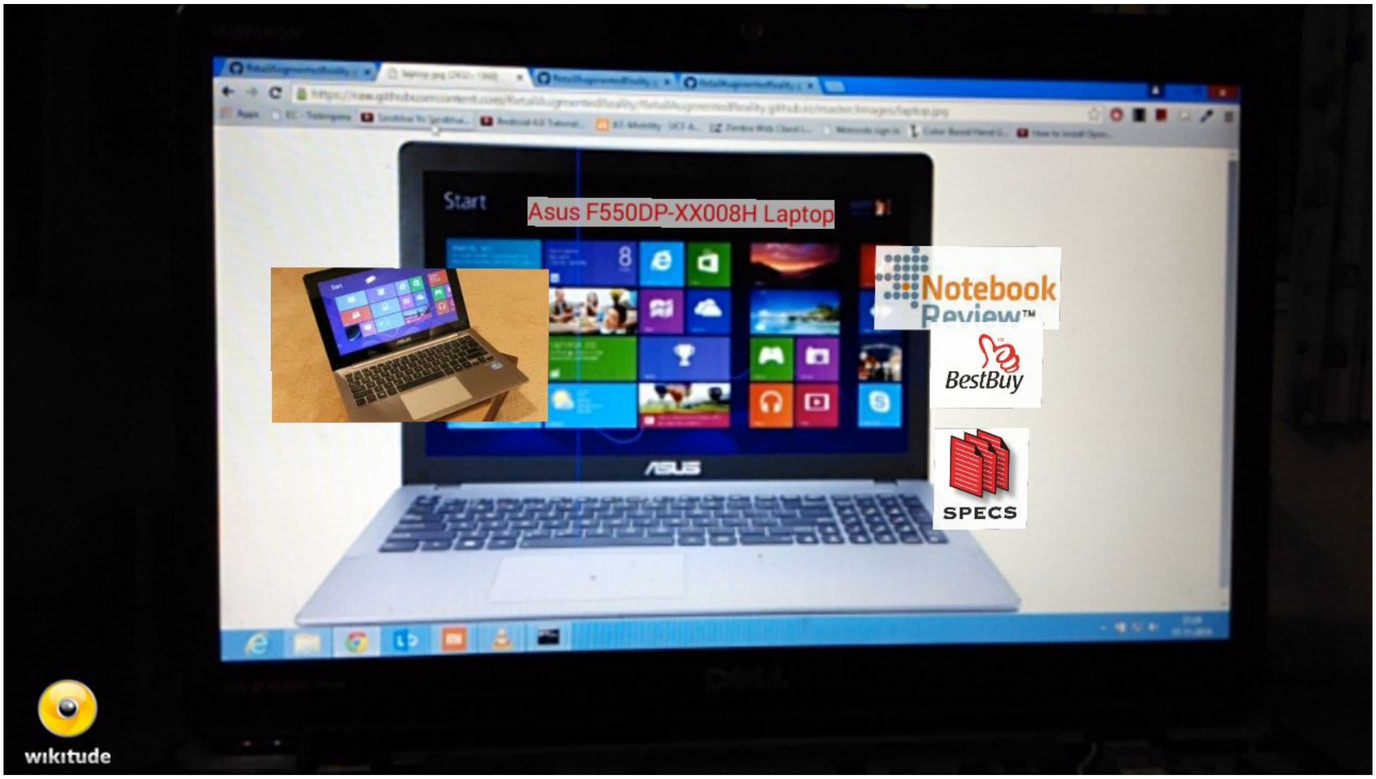
Laptop

Fridge

IntelChip

Choose any one of them and see augmentation on it for example I selected laptop and it opens camera and see the laptop from camera
See on below image

It will augment on laptop see below image



The augment will overlay on it and the augment which is overlaid is intuitive you can click it and get the relevant information then and there itself

To experiment on other products in the list I have placed all data required for project in Github:
Please find below link to the project:
https://github.com/RetailAugmentedReality/RetailAugmentedReality.github.io

8. VERSION CONTROL SYSTEM (GITHUB)

I am using GitHub which is open source VCS

I have created “RetailAugmentedReality” organization and created “RetailAugmentedReality.github.io” repository

You can access entire project from below link
https://github.com/RetailAugmentedReality

9. CUSTOMER BENEFITS

- ✓ More accurate data regarding product.
- ✓ How traditional retailing can be made more intuitive by combining with augmented reality and e-commerce.
- ✓ Visualizes product catalogues.
- ✓ Interactive marketing/ advertising.
- ✓ Brings the customer to the store with a vivid and intuitive augmented experience.
- ✓ Providing competitive edge in market.
- ✓ Improves customer satisfaction by letting the customers "try" before they buy with a 3D product preview.
- ✓ Displays additional information about products shown, enriches shopping experience.

10. DETAILED PLAN OF WORK

The following section gives the list of tasks with their estimated timelines and list of deliverables at every milestone of the project.

Task ID	Tasks to be Done	Planned efforts (in weeks)	Deliverables	Remarks
1.	Capture High level Business requirement	1-2	Requirement Specification document	Done
2.	Analysis of the Business requirement and preparation of Software Requirement			Done

	Specification			
3.	Technical Design of the Application Architecture.	3-4	High Level Design Document	Done
4.	Preparation of High Level Design Document	5-6		Done
5.	Test Case preparation for Unit Testing and System Integration Testing	7	Unit and System test case documents	Done
6.	CUT Activities	8-14	Unit tested Code and UTC Results	Done
7.	Execution of System Integration Test Cases	15-16	System Integration test cases (SIT)	Done
8.	Documentation	16	Final Dissertation Report	Done
9.				

You can access entire project from below link
https://github.com/RetailAugmentedReality

11. Recommendations/Direction for future work

You can develop similar application for different domains

E-Commerce / M-commerce offer a virtual fitting room where apparel can be tried on live. The main access to this revolutionary application is through websites, resulting in lowered return rates. TryLive by Total Immersion is radically transforming e-commerce by making virtual try-on and product visualization a reality. Whether online or in-store TryLive applications are available for prescription glasses and sunglasses, jewelry, footwear, clothes and furniture.

Digital Marketing of your product or a brand through ground-breaking Augmented Reality games. Augmented Reality is recognized as an enhanced marketing application that customers can engage with at home, in store, or on the go. Key digital aspects: Augmented packaging, on street marketing, geolocalized apps, advergaming and interactive consumer products.

Geolocation and its real time display of enhanced maps. Acting as a GPS, Augmented Reality provides classified and suggestive information to the user (restaurants, bars, grocery stores, fashion outlets, etc.)

Educational resources are emphasized by Augmented Reality systems and can be used to re-create historical events, activate regular books into 3D images, or even present structures of the galaxy; all superimposed in real-time. Augmented Reality is extremely useful for educators in classroom settings or during presentations and allows students to gain a deeper understanding on the topic at hand. By merging content to media the reading experience is enhanced and the reader is fully engaged. The text and images are on a page as usual, but Augmented Reality allows you to see dynamic, 3D computer graphics "hovering" over it.

Industrial, military and medical applications concerning the validation of designs or plans are a specialization of Augmented Reality. When soldiers need information on their surroundings they can receive detailed 3D maps. If a doctor is performing surgery, a live image of a human subject is accessible. Important occupations in need of crucial information use Augmented Reality tools to visually superimpose their solutions.

12. REFERENCES

- ✓ https://www.youtube.com/watch?v=z1aeWUVsuKE
- ✓ http://developer.android.com/index.html
- ✓ http://www.wikitude.com/
- ✓ https://www.youtube.com/watch?v=vyTvIufrdPE
- ✓ https://www.youtube.com/watch?v=NGOm8Ke-LhM

13. ACRONYMS

QR:	Quick Response
API:	Application Programming Interface
AR:	Augmented Reality
CPU:	Central Processing Unit
CVS:	Concurrent Version System
ECC:	Error Correcting
GPS:	Global Positioning System
IPC:	Inter-Process Communication
ISO:	International Organization for Standardization
JNI:	Java Native Interface
LAMP:	Linux, Apache, MySQL, PHP
MVC:	Model View Controller
MVP:	Model View Presenter
NDK:	Native Development Kit
OS:	Operating System
SDK:	Software Development Kit
SIFT:	Scale-Invariant Feature Transform
SRS:	Software Requirements Specification
SURF:	Speeded-Up Robust Features
SVN:	Subversion
QA:	Quick Augment
App:	Application

14. Check List of items for final report

Check list of items for final report (with Yes or No marked, as applicable)

a) Is the Cover page in proper format? [✓ Y] / N

b) Is the Title page in proper format? [✓ Y] / N

c) Is the Certificate from the Supervisor in proper format? Has it been signed? [✓ Y] / N

d) Is Abstract included in the Report? Is it properly written? [✓ Y] / N

e) Does the Table of Contents' page include chapter page numbers? [✓ Y] / N

f) Is Introduction included in the report? Is it properly written? Y/N

g) Are the Pages numbered properly? [✓ Y] / N

h) Are the Figures numbered properly? [✓ Y] / N

i) Are the Tables numbered properly? [✓ Y] / N

j) Are the Captions for the Figures and Tables proper? [✓ Y] / N

k) Are the Appendices numbered? [✓ Y] / N

l) Does the Report have Conclusions/ Recommendations of the work? [✓ Y] / N

m) Are References/ Bibliography given in the Report? [✓ Y] / N

n) Have the References been cited in the Report? [✓ Y] / N

o) Is the citation of References/ Bibliography in proper format? [✓ Y] / N