ADBIZ-ANDROID APPLICATION TO MANAGE THE ADVERTISING BUSINESS

A Thesis
Presented to the
Faculty of
San Diego State University

In Partial Fulfillment
of the Requirements for the Degree
Master of Science
in
Computer Science

by
Saurin M. Shah
Summer 2012
SAN DIEGO STATE UNIVERSITY

The Undersigned Faculty Committee Approves the

Thesis of Saurin M. Shah:

AdBiz-Android Application to Manage the Advertising Business

Joseph Lewis, Chair
Department of Computer Science

Alan Riggins
Department of Computer Science

Joel Davis
School of Journalism and Media Studies

5/23/2012
Approval Date
Copyright © 2012
by
Saurin M. Shah
All Rights Reserved
DEDICATION

I would like to dedicate this work to my family members who have been supporting me through entire graduation. Also, I would like to dedicate this work to my professors for their undisputed guidance. And last but not the least, special thanks to all my friends who helped me in each step during my study.
ABSTRACT OF THE THESIS

AdBiz-Android Application to Manage the Advertising Business
by
Saurin M. Shah
Master of Science in Computer Science
San Diego State University, 2012

A smartphone is a high-end mobile phone that combines the functions of a personal digital assistant(PDA) and a mobile phone. While most feature phones are able to run applications based on platforms such as Java ME, a smartphone allows the user to install and run more advanced applications based on a specific platform. A smartphone can be considered as a mobile phone which works like a personal pocket computer since these devices are much smaller than the regular computers.

The term smartphone is usually used to describe phones with more advanced computing ability and connectivity than a contemporary feature phone, although the distinction can be vague and there is no official definition for what constitutes the difference between them. Smartphones run third-party applications using advanced application programming interfaces (APIs), which can allow those applications to have better integration with the phone's OS and hardware than is typical with feature phones.

In today’s world, the basic idea to run a successful business is an advertisement. Earlier, the advertisement was displayed on the newspaper, books, and magazine. Then radio and television came in our life. So, people started to advertise their business on radio and televisions. As time passes, new technology came into our life. Now, you can see an advertisement on big digital billboards, outside the buses, inside the buses, on taxi cabs, at the airport, in the malls etc.

Currently, there is no such system available through which people can advertise efficiently on different places. This smartphone application is an innovative geo-dynamic media campaign management system. This application is mainly developed specifically for Android OS installed smartphones. Through this application, people can establish target markets as a media management professional providing better value. In pursuing contracts with business entities with established needs for asset tracking, this application will achieve reliable distribution channels for its advertising platform. A company which has similar functionality to this application is also available. RVUE is the company which does exactly the same thing. But, the main advantage of this application compared to RVUE company is that it gives the control to the user to select the area for an advertisement.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABSTRACT .................................................................................................................. v</td>
</tr>
<tr>
<td>LIST OF FIGURES ....................................................................................................... viii</td>
</tr>
<tr>
<td>ACKNOWLEDGEMENTS .................................................................................................. ix</td>
</tr>
<tr>
<td>CHAPTER</td>
</tr>
<tr>
<td>1 INTRODUCTION ......................................................................................................... 1</td>
</tr>
<tr>
<td>1.1 Need for an Application ......................................................................................... 1</td>
</tr>
<tr>
<td>1.2 Android Framework Overview ............................................................................... 3</td>
</tr>
<tr>
<td>2 RESEARCH BACKGROUND AND LITERATURE ......................................................... 5</td>
</tr>
<tr>
<td>2.1 What is Advertisement? ......................................................................................... 5</td>
</tr>
<tr>
<td>2.2 History of Advertisement ..................................................................................... 5</td>
</tr>
<tr>
<td>2.2.1 First Generation of Advertisement .................................................................. 5</td>
</tr>
<tr>
<td>2.2.2 Second Generation of Advertisement ............................................................... 6</td>
</tr>
<tr>
<td>2.2.3 Third Generation of Advertisement .................................................................. 6</td>
</tr>
<tr>
<td>2.2.4 Fourth Generation of Advertisement ................................................................ 9</td>
</tr>
<tr>
<td>2.3 Outdoor Advertisement ....................................................................................... 10</td>
</tr>
<tr>
<td>2.3.1 Digital Out of Home - DOOH ........................................................................ 10</td>
</tr>
<tr>
<td>2.3.2 Types of Outdoor Advertising ........................................................................ 11</td>
</tr>
<tr>
<td>2.3.2.1 Billboard Advertising ................................................................................. 11</td>
</tr>
<tr>
<td>2.3.2.2 Mobile Billboard Advertising ....................................................................... 11</td>
</tr>
<tr>
<td>2.4 Google’s Engine ................................................................................................... 11</td>
</tr>
<tr>
<td>2.4.1 Mobile Internet ................................................................................................. 12</td>
</tr>
<tr>
<td>2.4.2 What is Google Android? .................................................................................. 13</td>
</tr>
<tr>
<td>3 TECHNOLOGY ........................................................................................................ 14</td>
</tr>
<tr>
<td>3.1 Requirements ....................................................................................................... 14</td>
</tr>
<tr>
<td>3.2 Java ...................................................................................................................... 14</td>
</tr>
<tr>
<td>3.2.1 Robustness ....................................................................................................... 16</td>
</tr>
<tr>
<td>3.2.2 Multithreading ................................................................................................. 17</td>
</tr>
<tr>
<td>Section</td>
</tr>
<tr>
<td>------------------------------------------------------------------------</td>
</tr>
<tr>
<td>3.2.3 Simplicity</td>
</tr>
<tr>
<td>3.3 Android</td>
</tr>
<tr>
<td>3.3.1 Features</td>
</tr>
<tr>
<td>3.3.2 Android Architecture</td>
</tr>
<tr>
<td>3.3.2.1 Applications</td>
</tr>
<tr>
<td>3.3.2.2 Application Framework</td>
</tr>
<tr>
<td>3.3.2.3 Libraries</td>
</tr>
<tr>
<td>3.3.2.4 Android Runtime</td>
</tr>
<tr>
<td>3.3.2.5 Linux Kernel</td>
</tr>
<tr>
<td>3.3.2.6 The Manifest File</td>
</tr>
<tr>
<td>3.3.3 Development in Eclipse with ADT</td>
</tr>
<tr>
<td>3.4 SQLite</td>
</tr>
<tr>
<td>3.5 Callback</td>
</tr>
<tr>
<td>3.6 Protocol</td>
</tr>
<tr>
<td>4 IMPLEMENTATION AND RESULT</td>
</tr>
<tr>
<td>4.1 How to Use the Application</td>
</tr>
<tr>
<td>4.2 Hardware Required</td>
</tr>
<tr>
<td>4.3 Application on Home Screen</td>
</tr>
<tr>
<td>4.4 Zoom Controls on Map</td>
</tr>
<tr>
<td>4.5 Selecting an Area on Map</td>
</tr>
<tr>
<td>4.6 Show Screens Inside Selected Area</td>
</tr>
<tr>
<td>4.7 Uploading an Image for an Advertisement</td>
</tr>
<tr>
<td>4.8 Result</td>
</tr>
<tr>
<td>4.9 Portability</td>
</tr>
<tr>
<td>5 CONCLUSION</td>
</tr>
<tr>
<td>6 FUTURE PLAN</td>
</tr>
<tr>
<td>REFERENCES</td>
</tr>
<tr>
<td>Works Cited</td>
</tr>
<tr>
<td>Works Consulted</td>
</tr>
</tbody>
</table>
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Market share in mobile internet browsing as of October 2011.</td>
<td>12</td>
</tr>
<tr>
<td>3.1</td>
<td>Activity transfer to another activity using intent.</td>
<td>22</td>
</tr>
<tr>
<td>3.2</td>
<td>Android architecture.</td>
<td>23</td>
</tr>
<tr>
<td>3.3</td>
<td>Conversion from .java to .dex file.</td>
<td>24</td>
</tr>
<tr>
<td>4.1</td>
<td>Information about the phone in phone settings.</td>
<td>30</td>
</tr>
<tr>
<td>4.2</td>
<td>Wi-Fi settings of wireless network in the phone settings.</td>
<td>31</td>
</tr>
<tr>
<td>4.3</td>
<td>3G network in the phone settings.</td>
<td>32</td>
</tr>
<tr>
<td>4.4</td>
<td>Home page screen of the phone.</td>
<td>33</td>
</tr>
<tr>
<td>4.5</td>
<td>Zoom controls.</td>
<td>34</td>
</tr>
<tr>
<td>4.6</td>
<td>Map screen of the application.</td>
<td>35</td>
</tr>
<tr>
<td>4.7</td>
<td>Selecting a point on map.</td>
<td>37</td>
</tr>
<tr>
<td>4.8</td>
<td>Selecting multiple points on map.</td>
<td>38</td>
</tr>
<tr>
<td>4.9</td>
<td>Selecting a polygon on map.</td>
<td>39</td>
</tr>
<tr>
<td>4.10</td>
<td>Screens from different network providers.</td>
<td>41</td>
</tr>
<tr>
<td>4.11</td>
<td>Image select screen before selecting an image.</td>
<td>42</td>
</tr>
<tr>
<td>4.12</td>
<td>Image select screen after selecting an image.</td>
<td>43</td>
</tr>
</tbody>
</table>
ACKNOWLEDGEMENTS

This thesis study would not have been completed without the help of my advisors Dr. Joseph Lewis, Professor Alan Riggins, Professor Joel Davis. Dr. Joseph Lewis, who offered me a lot of encouragement, guidance, and continued supervision. Dr. Joseph Lewis and I have worked on various different strategies to deal with various problems that came in my way while developing this application. He has been constantly encouraging and discussing with me the contents of my thesis, in addition to the response and advises he was giving me within unexpectedly short time intervals. Dr. Joseph Lewis has also been helping me a lot in my academic program. Professor Davis also helped me a lot. Without his precious feedback, I was not able to cover all the aspect related to this thesis. Therefore, I would like to thank all of my advisors a lot.
CHAPTER 1

INTRODUCTION

The main objective of this thesis is to identify the design and implementation issues for this android smartphone application. This is an Android based application for business owners through which they can display the advertisement with easy real-time customization of time and location. Currently, advertisers are facing a difficulty of customizing content for time and location. And, fleet managers are also facing the problem of high capital costs for fleet management systems and difficult to track ROI of these systems. Through this system, fleet managers can generate additional cash flow supported by advertising revenue and offsetting the high costs of fleet management.

Since, the application is developed for Android based smartphones, it has to be designed efficiently both in terms of user interface (UI) and in terms of source code.

Development and testing of the application performed on Android based platform. Android framework has tons of functionalities among this application mainly uses location based framework for position and locating screens and buildings. Smartphones are very essential and more important. They come under the category of latest technology of this new technical era.

1.1 NEED FOR AN APPLICATION

Now-a-days mobile applications serve an integral part of the huge mobile market. Now, people need smartphone to do some daily life activities like play games, to check emails, to check the news, to check the social network updates, entertainment, movies, search local business etc.

This application is particular for business owners. Currently, there are some systems available in the market but they do not use the real time customization of time and location efficiently. There are few companies which are doing similar stuff. The company called “RVUE” has a web based application in which they allow user to select any particular area and the size of radius which they want to select. What it does is after selecting those criteria,
this application will create a circle around that area. And then, whichever screen comes under that selected area, will display the advertisement which the business owner wants show. But, this method has some drawbacks like if the business owner wants display his/her advertisement on some screens which are located in near to each other but in a manner in which they cannot be covered in a small amount circle. So, if the business owner wants to display the advertisements in all these screens, then he/she has to select big area which means he/she has to pay more.

There is another approach is done in this area. A company called “TITAN” located in all parts of the United States is doing the similar stuff. We provide sales, marketing, creative, research and maintenance of creative outdoor advertising on bus, rail, bulletins, telephone kiosks and street banners. Titan also leads the way in the development and successful introduction of market leading digital outdoor platforms. What this company does is that they post a video on moving screen like buses or fixed bill board. So, the advertisement will be shown in the fix route which is allocated to the bus. So, they are not giving a control of selecting the area to the user.

There is another company called “TRIPINK Media” which is located in Miami, Florida. This is also doing similar thing to what “TITAN” does. This company has its own set of vehicles. So, they drive their own small truck like vehicles in the urban area and they have attached one big screen on the back of the vehicle onto which they display the advertisement. But, this is not an efficient way to display the advertisement.

So, all this companies have developed the system but it does not have the real time customization of time and location. The way “TRIPINK Media” is running the business, is not an efficient way. It cost more money and man power to both customers as well as the company. “TITAN” displays the advertisements on the bus which follows the fix route. So, if a person wants to display the advertisement on the screen which does not come into the fix route of the bus, then this system useless. “RVUE” comes close when it comes to the efficient use of time and location. It gives the control to the user to select the area by drawing circles but if there is a scenario when the user wants to display ad on the nearby screen, then they have to increase the radius to cover that screen which adds the other unwanted area also, which may cost little higher for the business owner.
So, what I want to build is the similar application of what “RVUE” does. The application through which business owners can select the time and location criteria as well as they can select the random geographical area onto which they want to display the advertisement. We can detect the location of moving as well as stationary screens through GPS co-ordinate, which is a key element. We can add bidding algorithm to this application as well. What it does is it gives users benefit of displaying the advertisement at lowest price. This bidding algorithm gives benefit to business owners as well. We can set the minimum price for an advertisement and once the bidding starts, the price will be increased by the customers (just like ebay).

1.2 ANDROID FRAMEWORK OVERVIEW

Android is a software stack for mobile devices that includes an operating system, middleware and key applications. The Android SDK provides the tools and APIs necessary to begin developing applications on the Android platform using the Java programming language. Android is built on the open Linux Kernel. Furthermore, it utilizes a custom virtual machine that was designed to optimize memory and hardware resources in a mobile environment. Android is open source; it can be liberally extended to incorporate new cutting edge technologies as they emerge. The developers have free access to the mobile phones resources through the OS and can perform and create any top level applications that make use of the underlying hardware.

By providing an open development platform, Android offers developers the ability to build extremely rich and innovative applications. Developers are free to take advantage of the device hardware, access location information, run background services, set alarms, add notifications to the status bar, and much, much more.

Developers have full access to the same framework APIs used by the core applications. The application architecture is designed to simplify the reuse of components; any application can publish its capabilities and any other application may then make use of those capabilities (subject to security constraints enforced by the framework). This same mechanism allows components to be replaced by the user. A rich and extensible set of views and that can be used to build an application including lists, grids, textboxes, buttons and even an embeddable web browser. Android relies on Linux for core system services such as
security, memory management, process management, network stack and driver model. The kernel also acts as an abstraction layer between the hardware and the rest of the software stack.

An application usually consists of multiple activities that are loosely bound to each other and they have a lifecycle. At first, it has onCreate() method to create an activity, then onStart() method, which starts an activity. It also has methods such as onPause() and onResume() which handles the tasks when the control from that particular activity is gone and when the control to that activity is returned. Typically, one activity in an application is specified as the "main" activity, which is presented to the user when launching the application for the first time. Each activity can then start another activity in order to perform different actions. Each time a new activity starts, the previous activity is stopped, but the system preserves the activity in a stack.
CHAPTER 2

RESEARCH BACKGROUND AND LITERATURE

2.1 WHAT IS ADVERTISEMENT?

Advertising is a form of communication which includes the name of a product or service and how that product or service could benefit the consumer, to persuade a target market to purchase or to consume that particular brand. These messages are usually paid for by sponsors and viewed via various media. Advertising can also serve to communicate an idea to a large number of people in an attempt to convince them to take a certain action. [1]

2.2 HISTORY OF ADVERTISEMENT

The nineteenth century saw the skills of the advertiser come to the forefront, as ads began to mix images and words, and adopt the techniques of language and layout that we are familiar with today. With the proliferation of goods and services in this century, it became recognized that advertising was an important part of business, and should be dealt with by experts in the field. Most of the advertising agencies that dominate the global markets today were founded in the latter part of the nineteenth century.

2.2.1 First Generation of Advertisement

As education became an apparent need and reading, as well as printing, developed advertising expanded to include handbills. In the 17th century advertisements started to appear in weekly newspapers in England [2]. These early print advertisements were used mainly to promote books and newspapers, which became increasingly affordable with advances in the printing press; and medicines, which were increasingly sought after as disease ravaged Europe. However, false advertising and so-called "quack" advertisements became a problem, which ushered in the regulation of advertising content [2]. As the economy expanded during the 19th century, advertising grew alongside. In the United States, the success of this advertising format eventually led to the growth of mail-order advertising.

In June 1836, French newspaper "La Presse" was the first to include paid advertising in its pages, allowing it to lower its price, extend its readership and increase its profitability.
and the formula was soon copied by all titles [2]. Around 1840, Volney B. Palmer established a predecessor to advertising agencies in Boston [2]. Around the same time, in France, Charles-Louis Havas extended the services of his news agency, Havas to include advertisement brokerage, making it the first French group to organize [2]. At first, agencies were brokers for advertisement space in newspapers. N. W. Ayer & Son was the first full-service agency to assume responsibility for advertising content. N.W. Ayer opened in 1869, and was located in Philadelphia. [2]

### 2.2.2 Second Generation of Advertisement

In the early 1920s, the first radio stations were established by radio equipment manufacturers and retailers who offered programs in order to sell more radios to consumers [2]. As time passed, many non-profit organizations followed suit in setting up their own radio stations, and included: schools, clubs and civic groups. When the practice of sponsoring programs was popularised, each individual radio program was usually sponsored by a single business in exchange for a brief mention of the business' name at the beginning and end of the sponsored shows. However, radio station owners soon realized they could earn more money by selling sponsorship rights in small time allocations to multiple businesses throughout their radio station's broadcasts, rather than selling the sponsorship rights to single businesses per show.

Advertising was a large, well established industry in 1914 and it continued to expand after the First World War. Psychology was growing in stature as a science during this period, and advertisers were quick to latch on to key ideas in order to reach their audience. As new ways of reaching a mass audience became technologically available (cinema, radio) advertising was quick to latch on to new media and became an important way for broadcasters to help fund their programming. Radio was an especially successful way to reach audiences in the 1920s. The term 'soap opera' as we know it came into being as soap manufacturers sponsored domestic radio dramas in return for frequent plugs for their product.

### 2.2.3 Third Generation of Advertisement

After the Second World War there was both a boom in consumer spending AND a new medium: television. Millions of dollars annually were poured into the coffers of advertising agencies as manufacturers sought to inform newly prosperous consumers of the
dazzling array of new goods they could purchase (washing machines, vacuum cleaners, food mixers, TV sets...) that they never previously knew they needed. With the combined impact of image AND sound, it soon became clear that brand recognition was much greater with TV than with radio and soon the airwaves were full of programmes as advertisers scrambled to access this new wave of consumers. This is sometimes referred to as the single-sponsor era, as the tv programmes were actually produced by the advertising agencies and paid for by a single sponsor. The 1950s were bonanza years for advertising, and it was at this stage that the men behind the ads became media stars and shot into the league of the super-highly paid.

In the early 1950s, the DuMont Television Network began the modern practice of selling advertisement time to multiple sponsors [2]. Previously, DuMont had trouble finding sponsors for many of their programs and compensated by selling smaller blocks of advertising time to several businesses. This eventually became the standard for the commercial television industry in the United States. However, it was still a common practice to have single sponsor shows, such as The United States Steel Hour. In some instances the sponsors exercised great control over the content of the show—up to and including having one's advertising agency actually writing the show. The single sponsor model is much less prevalent now, a notable exception being the Hallmark Hall of Fame.

As the 1960s began, the networks wanted more control over the content and style of programming, and as TV became more sophisticated and production costs rose, single sponsors began to struggle. NBC executive Sylvester Weaver came up with the idea of selling not whole shows to advertisers, but separate, small blocks of broadcast time. [3] Several different advertisers could buy time within one show, and therefore the content of the show would move out of the control of a single advertiser - rather like a print magazine. This became known as the magazine concept, or participation advertising, as it allowed a whole variety of advertisers to access the audience of a single TV show. Thus the 'commercial break' as we know it was born.

The 1960s saw advertising transform into a modern approach in which creativity was allowed to shine, producing unexpected messages that made advertisements more tempting to consumers' eyes.

In 1970s, an act of Congress forced cigarette commercials off television. That move did little to hinder the tobacco industry’s advertising efforts; it just spent more on other
media. Ironically, cigarette sales increased dramatically in 1971, the first year of the cigarette-ad ban, probably because anti-smoking commercials, which had indeed caused a decline in cigarette consumption could no longer be aired for free.

The 80s proposed, for the first time in years, that you could have a booming market economy and also masses of what AA Gill has called "cultural cute" [4]. For the first and last time, advertising was absolutely at the centre of things. Inevitably, all that 80s ad magic comes back to Saatchi & Saatchi.[5] They didn't make all the brilliant British ads of the 80s - though they did make a creditable fair few; nor invent all the big developments - though they did, significantly, bundle all their media-buying in one separate company, Zenith Media. In one sense, S&S was a proxy for the whole ad industry: [5] it made the creative industries' "intangibles" really valuable for the first time, and their producers really rich. The creativity thing came first, of course. By the end of the 70s, a second generation of UK advertising agencies were producing commercials such as the insanely glamorous Benson & Hedges "Iguana" cinema ad (1978) [5]. At one dizzy point, S&S was the world's largest "communications" business. Their work for BA epitomised 80s advertising - brilliant, big-budget commercials focused on national pride in a very shiny, postmodernist way.

Over the course of the 90s, the business model of traditional "full-service" advertising agencies - the ones with the smart account men and the clever creatives - collapsed and they were humbled. These agencies had lived on commission: broadly, 15% of the cost of the media they bought for clients. After the 90s recession, media-buying departments were floated off as separate businesses: clients wanted to control cash flow. Media were becoming increasingly complex with the advent of multichannel TV, a huge expansion in print and, by the end of the decade, online advertising.

The late 1980s and early 1990s saw the introduction of cable television and particularly MTV. Pioneering the concept of the music video, MTV ushered in a new type of advertising: the consumer tunes in for the advertising message, rather than it being a by-product or afterthought. As cable and satellite television became increasingly prevalent, specialty channels emerged, including channels entirely devoted to advertising, such as QVC, Home Shopping Network, and ShopTV Canada. [2]
2.2.4 Fourth Generation of Advertisement

The late-90s advertising bonanza came courtesy of the dot.com boom, with most agencies delivering overblown new age fantasies that usually failed to explain the basic proposition or set up the target market. Over the last 10 years, advertising has been under increasing pressure. Pressure on the gorgeous, indulgent, big lunch culture of the 80s. Pressure for accountability, for measurable effectiveness. Pressure to align itself to the real back-office world its clients inhabit, and to globalisation.

Marketing through the Internet opened new frontiers for advertisers and contributed to the "dot-com" boom of the 1990s [2]. Entire corporations operated solely on advertising revenue, offering everything from coupons to free Internet access. At the turn of the 21st century, a number of websites including the search engine Google, started a change in online advertising by emphasizing contextually relevant, unobtrusive ads intended to help, rather than inundate, users. This has led to a plethora of similar efforts and an increasing trend of interactive advertising.

The share of advertising spending relative to GDP has changed little across large changes in media. For example, in the US in 1925, the main advertising media were newspapers, magazines, signs on streetcars, and outdoor posters. Advertising spending as a share of GDP was about 2.9 percent. By 1998, television and radio had become major advertising media. Nonetheless, advertising spending as a share of GDP was slightly lower—about 2.4 percent. [2]

A recent advertising innovation is "guerrilla marketing", which involve unusual approaches such as staged encounters in public places, giveaways of products such as cars that are covered with brand messages, and interactive advertising where the viewer can respond to become part of the advertising message. Guerrilla advertising is becoming increasingly more popular with a lot of companies. This type of advertising is unpredictable and innovative, which causes consumers to buy the product or idea. This reflects an increasing trend of interactive and "embedded" ads, such as via product placement, having consumers vote through text messages, and various innovations utilizing social network services such as Facebook.
2.3 Outdoor Advertisement

Outdoor advertising is essentially any type of advertising that reaches the consumer while he or she is outside the home. This is in contrast with broadcast, print, and Internet advertising. Out of home advertising, therefore, is focused on marketing to consumers when they are "on the go" in public places, in transit, waiting (such as in a medical office), and/or in specific commercial locations (such as in a retail venue). Outdoor advertising formats fall into four main categories: billboards, street furniture, transit, and alternative.

2.3.1 Digital Out of Home - DOOH

Digital out-of-home refers to dynamic media distributed across placed-based networks in venues including but not limited to cafes, bars, restaurants, health clubs, colleges, arenas, gas stations and public spaces. DOOH networks typically feature independently addressable screens, kiosks, jukeboxes and/or jumbotrons. DOOH media benefits location owners and advertisers alike in being able to engage customers and/or audiences and extend the reach and effectiveness of marketing messages. It is also referred to as Digital Signage.

The reason that this category is growing so rapidly is because busy people are typically busy at home and with the introduction and acceptance of digital video recorders, it has diluted the frequency with which traditional television commercials are viewed. Every day more TV viewers are skipping past commercials with their DVRs which in turn has made out-of-home advertising all the more appealing. A Nielsen media research study in 2009 showed that 91% of DVR owners skipped commercials [6]. As a result, traditional TV advertisers are hungry for an effective substitute, and digital out-of-home ads appear to be one of the solutions.

DOOH also includes stand-alone screens, kiosks, and interactive media found in public places. The availability of inexpensive LCD screens with built-in media players has opened the door for companies to add interactive video messages in Point of Purchase (POP) Displays. The displays allow consumers to get additional information at the moment of decision on a product or service. Growth in the DOOH industry has been increasing in 2009, with more POP manufacturers, advertisers, and content developers moving to digital. [7]
2.3.2 Types of Outdoor Advertising

There are mainly two types of outdoor advertising. Billboard advertising and mobile billboard advertising.

2.3.2.1 Billboard Advertising

Billboards are large structures located in public places which display advertisements to passing pedestrians and motorists. Most often, they are located on main roads with a large amount of passing motor and pedestrian traffic; however, they can be placed in any location with large amounts of viewers, such as on mass transit vehicles and in stations, in shopping malls or office buildings, and in stadiums. [2]

2.3.2.2 Mobile Billboard Advertising

Mobile billboards are generally vehicle mounted billboards or digital screens. These can be on dedicated vehicles built solely for carrying advertisements along routes preselected by clients, they can also be specially equipped cargo trucks or, in some cases, large banners strewn from planes. The billboards are often lighted; some being backlit, and others employing spotlights. Some billboard displays are static, while others change; for example, continuously or periodically rotating among a set of advertisements. Mobile displays are used for various situations in metropolitan areas throughout the world, including: Target advertising, One-day, and long-term campaigns, Conventions, Sporting events, Store openings and similar promotional events, and Big advertisements from smaller companies. [2]

2.4 Google’s Engine

Google is known for search engine which helps to connect the world together. The main purpose of Google Search is to hunt for text in Web pages, as opposed to other data, such as with Google Image Search. Its vision is to make a search engine so robust and powerful that it can understand the entire world. Google App Engine lets you run web applications on Google's infrastructure. Its goal is to provide much higher level of services to all those who seek information, whether they are at home, office, businesses or in travel. App Engine applications are easy to build, easy to maintain, and easy to scale as your traffic and data storage needs grow. With App Engine, there are no servers to maintain: You just upload
your application, and it's ready to serve your users. Google search is not only limited to the personal computer world but it has also set foot in the mobile internet world with their Android OS. [8]

### 2.4.1 Mobile Internet

The Mobile Internet refers to the use of Internet-connected applications, or browser-based access to the Internet from a mobile device, such as a smartphone or tablet computer, connected to a wireless network. The Web is becoming more accessible by portable and wireless devices. The shift to mobile Web access has been accelerating with the rise since 2007 of larger multi touch smartphones (see Figure 2.1).

#### Figure 2.1. Market share in mobile internet browsing as of October 2011.

The distinction between mobile Web applications and native applications is anticipated to become increasingly blurred, as mobile browsers gain direct access to the hardware of mobile devices and the speed and abilities of browser-based applications improve. Mobile internet is growing rapidly. More than 50% of mobile subscribers use mobile internet these days and according to ad Mob report, there are 8 billion requests for the mobile ads worldwide at the end of October 2011. [8]
2.4.2 What is Google Android?

Android is an operating system for mobile devices such as smartphones and tablet computers. It is developed by the Open Handset Alliance led by Google. Android consists of a kernel based on the Linux kernel, with middleware libraries and APIs written in C and application software running on an application framework which includes Java-compatible libraries based on Apache Harmony. The Open Handset Alliance’s common goal is to foster and develop a new breed of innovation for mobile devices allowing a far better user experience than today’s current mobile platforms. Android was listed as the best-selling smartphone platform worldwide in Q4 2010 by Canalys with over 190 million Android devices in use by October 2011. [9]
CHAPTER 3

TECHNOLOGY

This chapter focuses on the technology used to develop the software and briefly discusses about the supporting software, tools and integrated development environments (IDE).

3.1 REQUIREMENTS

The prime objective of building this application is to be able to advertise the business from mobile phones. The easiest way to start developing Android applications is to download the Android SDK and the Eclipse IDE. Android development can take place on Microsoft® Windows®, Mac OS X, or Linux. The requirements gathered are enlisted below.

- Android is a software stack for mobile devices that includes an operating system, middleware and key applications. The Android SDK provides the tools and APIs necessary to begin developing applications on the Android platform using the Java programming language.
- Android supports creating of databases based on SQLite db. Each database is private to the applications that creates it. SQLite reads and writes directly to ordinary disk files.
- Nowadays, every android device comes with google map installed in it. So, on map, we can select a particular area and then select the screen.

3.2 JAVA

JAVA is used to build this application and Java development kit version 6 is used for the programming. Eclipse is an open source platform for development which is used as the integrated programming environment for this application. The reasons behind choosing Java over other software languages include the following and these points are elaborated below:

- Java was designed to be easy to use and is therefore easy to write, compile, debug, and learn than other programming languages.
- Java is object-oriented: Java is object-oriented because programming in Java is centered on creating objects, manipulating objects, and making objects work together. This allows you to create modular programs and reusable code.
Java is distributed: Distributed computing involves several computers on a network working together. Java is designed to make distributed computing easy with the networking capability that is inherently integrated into it.

Java is secure: Java is one of the first programming languages to consider security as part of its design. The Java language, compiler, interpreter, and runtime environment were each developed with security in mind.

Most of the android applications are developed on Java language which runs on top of google’s Dalvik Java virtual machine.

Java is robust: Robust means reliable and no programming language can really assure reliability. Java puts a lot of emphasis on early checking for possible errors, as Java compilers are able to detect many problems that would first show up during execution time in other languages.

Java is multithreaded: Multithreaded is the capability for a program to perform several tasks simultaneously within a program. In Java, multithreaded programming has been smoothly integrated into it, while in other languages, operating system-specific procedures have to be called in order to enable multithreading. Multithreading is a necessity in visual and network programming.

Java is platform-independent: One of the most significant advantages of Java is its ability to move easily from one computer system to another.

Java is interpreted: An interpreter is needed in order to run Java programs. The programs are compiled into Java Virtual Machine code called bytecode.

Writing network programs in Java is like sending and receiving data to and from a file. For example, the diagram below shows three programs running on three different systems, communicating with each other to perform a joint task.

Java is an object oriented programming language. Object oriented language is a type of programming in which programmers define not only the data type of a data structure, but also the types of operations that can be applied to the data structure. In this way, the data structure becomes an object that includes both data and functions. In addition, programmers can create relationships between one object and another. Java is object oriented because it focuses on creating objects and making them work together. Java features all of the object oriented concepts mentioned below:

- **Data Encapsulation**: The encapsulation is the inclusion within a program object of all the resources need for the object to function - basically, the methods and the data. In OOP the encapsulation is mainly achieved by creating classes, the classes expose public methods and properties. The class is kind of a container or capsule or a cell, which encapsulate the set of methods, attribute and properties to provide its indented functionalities to other classes. In that sense, encapsulation also allows a class to change its internal implementation without hurting the overall functioning of the
system. That idea of encapsulation is to hide how a class does it but to allow requesting what to do.

- **Interface**: The Interface separates the implementation and defines the structure, and this concept is very useful in cases where you need the implementation to be interchangeable. Apart from that an interface is very useful when the implementation changes frequently. Some say you should define all classes in terms of interfaces, but I think recommendation seems a bit extreme. Interface can be used to define a generic template and then one or more abstract classes to define partial implementations of the interface.

- **Inheritance**: An important concept in object-oriented programming is inheritance. It provides a way for objects to define relationships with each other. As the name inheritance suggests an object is able to inherit characteristics from another object. In more concrete terms, an object is able to pass on its state and behaviors to its children. For inheritance to work the objects need to have characteristics in common with each other. Just like abstraction is closely related with generalization, the inheritance is closely related with specialization.

   In my application, I have created one object in the class and in its child class, I have used inherited object which inherits all the characteristics from the parent class. Thus, I don’t have to define any extra object which saves memory and time also.

- **Polymorphism**: Polymorphisms is a generic term that means 'many shapes'. More precisely *Polymorphisms* means the ability to request that the same operations be performed by a wide range of different types of things. At times, I used to think that understanding Object Oriented Programming concepts have made it difficult since they have grouped under four main concepts, while each concept is closely related with one another. Hence one has to be extremely careful to correctly understand each concept separately, while understanding the way each related with other concepts. In OOP the polymorphisms is achieved by using many different techniques named method overloading, operator overloading and method overriding.

   Things like these make programming very loosely coupled so that the complexity is reduced and the programming models become highly independent and modular.

### 3.2.1 Robustness

A robust program is a program that will accept junk input and not crash. Java is robust because it is highly supported language, meaning that unlike C you cannot crash your computer with a bad program. Also, another factor in its robustness is its portability across many Operating systems, with is supported by the Java Virtual Machine. Java is one of the most robust languages I've ever seen. Automatic memory management and garbage collection is the biggest contributor here. Strong type checking also helps. Bugs, especially
system crashing bugs, are rare. Still, they're not quite non-existent. In particular, you can overtax Java's thread system by creating many compute intensive threads. The exact details vary from virtual machine to virtual machine but 2000 threads each calculating the Mandelbrot set will hang or crash pretty much any virtual machine.

I have tried to write the code as easy as possible. But, even in some scenarios, malfunctioning may happen on the application due to some bad program. But since, java is a robust language, device will not crash due to that.

3.2.2 Multithreading

Multithreading as a widespread programming and execution model allows multiple threads to exist within the context of a single process. These threads share the process' resources but are able to execute independently. The threaded programming model provides developers with a useful abstraction of concurrent execution. However, perhaps the most interesting application of the technology is when it is applied to a single process to enable parallel execution on a multiprocessor system. In Java, the Java Virtual Machine (JVM) allows an application to have multiple threads of execution running concurrently. It allows a program to be more responsible to the user. When a program contains multiple threads then the CPU can switch between the two threads to execute them at the same time.

On my application, when I select the area on the map, program will call three different methods through which user will be able to draw a polygon. These methods will run in background independent to each other, which will make the process faster. Thus, multithreading is very important.

3.2.3 Simplicity

Java compiler automatically creates the Java compiled classes into machine readable byte-code. The most important feature of Java which makes it very simple is its ability to handle automatic memory management. Java uses automatic garbage collection when an object is destroyed to release the memory unlike C++ where programmer is responsible for freeing the memory associated with the deleted object. It make life easier for the programmer when he/she uses java language. [10-11]
3.3 Android

Android is a Linux-based operating system for mobile devices such as smartphones and tablet computers. It is developed by the Open Handset Alliance led by Google. The unveiling of the Android distribution in 2007 was announced with the founding of the Open Handset Alliance. Google releases the Android code as open-source, under the Apache License. The Android Open Source Project (AOSP) is tasked with the maintenance and further development of Android. Developers write primarily in a customized version of Java. Android was listed as the best-selling smartphone platform worldwide in Q4 2010 by Canalys with over 300 million Android devices in use by February 2012. According to Google’s Andy Rubin, as of February 2012 there are over 850,000 Android devices activated every day.

Android powers millions of phones, tablets, and other devices and brings the power of Google and the web into your hands. The Android Open Source Project (AOSP) is led by Google, and is tasked with the maintenance and development of Android. According to the project "The goal of the Android Open Source Project is to create a successful real-world product that improves the mobile experience for end users." AOSP also maintains the Android Compatibility Program, defining an "Android compatible" device "as one that can run any application written by third-party developers using the Android SDK and NDK", to prevent incompatible Android implementations. The compatibility program is also optional and free of charge, with the Compatibility Test Suite also free and open-source.

The Android SDK provides the tools and APIs necessary to begin developing applications on the Android platform using the Java programming language. Android was built from the ground-up to enable developers to create compelling mobile applications that take full advantage of all a handset has to offer. It was built to be truly open. For example, an application can call upon any of the phone’s core functionality such as making calls, sending text messages, or using the camera, allowing developers to create richer and more cohesive experiences for users. Android is built on the open Linux Kernel. Furthermore, it utilizes a custom virtual machine that was designed to optimize memory and hardware resources in a mobile environment.
Android applications written in Java, compiled Java code that is packaged into an Android package known as an .apk file. Bundling application data in this way allows 21 applications to be easily distributed for installation on mobile devices. Indeed, the .apk file is the file that users are required to download and all the code in an .apk is essentially one application. [12]

3.3.1 Features

- **Refined, evolved UI**: Focused on bringing the power of Android to the surface, Android makes common actions more visible and lets people navigate with simple, intuitive gestures. Refined animations and feedback throughout the system make interactions engaging and interesting. An entirely new typeface optimized for high-resolution screens improves readability and brings a polished, modern feel to the user interface.

Rich and interactive notifications let you keep in constant touch with incoming messages, play music tracks, see real-time updates from apps, and much more. On smaller-screen devices, notifications appear at the top of the screen, while on larger-screen devices they appear in the System Bar.

- **Home screen folders and favorites tray**: New home screen folders offer a new way for you to group your apps and shortcuts logically, just by dragging one onto another. Also, in All Apps launcher, you can now simply drag an app to get information about it or immediately uninstall it, or disable a pre-installed app. On smaller-screen devices, the home screen now includes a customizable favorites tray visible from all home screens. You can drag apps, shortcuts, folders, and other priority items in or out of the favorites tray for instant access from any home screen.

- **Resizable widgets**: Home screens in Android are designed to be content-rich and customizable. You can do much more than add shortcuts — you can embed live application content directly through interactive widgets. Widgets let you check email, flip through a calendar, play music, check social streams, and more — right from the home screen, without having to launch apps. Widgets are resizable, so you can expand them to show more content or shrink them to save space.

- **New lock screen actions**: The lock screens now let you do more without unlocking. From the slide lock screen, you can jump directly to the camera for a picture or pull down the notifications window to check for messages. When listening to music, you can even manage music tracks and see album art.

- **Quick responses for incoming calls**: When an incoming call arrives, you can now quickly respond by text message, without needing to pick up the call or unlock the device. On the incoming call screen, you simply slide a control to see a list of text responses and then tap to send and end the call. You can add your own responses and manage the list from the Settings app.
• **Swipe to dismiss notifications, tasks, and browser tabs:** Android 4.0 makes managing notifications, recent apps, and browser tabs even easier. You can now dismiss individual notifications, apps from the Recent Apps list, and browser tabs with a simple swipe of a finger.

• **Improved text input and spell-checking:** The soft keyboard in Android 4.0 makes text input even faster and more accurate. Error correction and word suggestion are improved through a new set of default dictionaries and more accurate heuristics for handling cases such as double-typed characters, skipped letters, and omitted spaces. Word suggestion is also improved and the suggestion strip is simplified to show only three words at a time.

• **Control over network data:** In the Settings app, colorful charts show the total data usage on each network type (mobile or Wi-Fi), as well as amount of data used by each running application. Based on your data plan, you can optionally set warning levels or hard limits on data usage or disable mobile data altogether. You can also manage the background data used by individual applications as needed.

• **Designed for accessibility:** A variety of new features greatly enhance the accessibility of Android 4.0 for blind or visually impaired users. Most important is a new explore-by-touch mode that lets you navigate without having to see the screen. Touching the screen once triggers audible feedback that identifies the UI component below; a second touch in the same component activates it with a full touch event. The new mode is especially important to support users on new devices that use virtual buttons in the System Bar, rather than dedicated hardware buttons or trackballs. Also, standard apps are updated to offer an improved accessibility experience. The Browser supports a script-based screen reader for reading favorite web content and navigating sites. For improved readability, you can also increase the default font size used across the system.

• **Rich and versatile camera capabilities:** The Camera app includes many new features that let you capture special moments with great photos and videos. After capturing images, you can edit and share them easily with friends.

• **Powerful web browsing:** The Android Browser offers an experience that’s as rich and convenient as a desktop browser. It lets you instantly sync and manage Google Chrome bookmarks from all of your accounts, jump to your favorite content faster, and even save it for reading later in case there’s no network available. To get the most out of web content, you can now request full desktop versions of web sites, rather than their mobile versions. You can set your preference for web sites separately for each browser tab. For longer content, you can save a copy for offline reading. To find and open saved pages, you can browse a visual list that’s included with browser bookmarks and history. For better readability and accessibility, you can increase the browser’s zoom levels and override the system default text sizes.

• **Android Beam for NFC-based sharing:** Android Beam is an innovative, convenient feature for sharing across two NFC-enabled devices. It lets people instantly exchange favorite apps, contacts, music, videos — almost anything. It’s incredibly simple and convenient to use — there’s no menu to open, application to launch, or pairing
needed. Just touch one Android-powered phone to another, then tap to send. For sharing apps, Android Beam pushes a link to the app's details page on Google Play. On the other device, the Google Play app launches and loads the details page, for easy downloading of the app. Individual apps can build on Android Beam to add other types of interactions, such as passing game scores, initiating a multiplayer game or chat, and more.

- **Wi-Fi Direct and Bluetooth HDP**: Support for Wi-Fi Direct lets you connect directly to nearby peer devices over Wi-Fi, for more reliable, higher-speed communication. No internet connection or tethering is needed. Through third-party apps, you can connect to compatible devices to take advantage of new features such as instant sharing of files, photos, or other media; streaming video or audio from another device; or connecting to compatible printers or other devices. Android 4.0 also introduces built-in support for connecting to Bluetooth Health Device Profile (HDP) devices. With support from third-party apps, you can connect to wireless medical devices and sensors in hospitals, fitness centers, homes, and elsewhere. [9, 13]

### 3.3.2 Android Architecture

The following diagram shows the major components of the Android operating system.

#### 3.3.2.1 Applications

It makes the top most layer of android architecture. Built in applications like sms, mms, contacts, browser are part of this layer. Also all the user application code lays in this section. These user applications, data and resource files make an android package. One of the main features of Android is that the basic elements of applications can be shared. If permissions are granted for these elements, it is possible to use already developed features of other applications.

This means that Android has been designed to allow application processes to be started when any part of it is needed. In other words, while most systems and applications have a single entry point, or main function, Android applications rely on central components that are instantiated and run when they are needed. There are four major types of Android components: Activities, Services, Broadcast receivers, and Content providers. Applications are comprised of components. Components communicate using Intent messages. Recipient components assert their desire to receive Intent messages by defining Intent filters (see Figure 3.1).
There are four types of components used to construct applications:

1. Activities in the system are managed in an activity stack. When a new activity is launched it becomes the top of the stack. Any previous activity will be below it and won’t come to the top until the new one exists. Android runs each activity in a separate process each of which hosts a separate virtual machine. The activity can be in one of four states:
   - **Active**: The activity started, is running and is in the foreground.
   - **Paused**: The activity is running and visible, but another activity is running on the top. A paused activity maintains state and member information but can be killed by the system in low memory situations.
   - **Stopped**: It is similar to paused but invisible.
   - **Dead**: Either the activity is not started or it was in pause or stop state and was terminated by the system to free some memory or by asking the user to do so.

2. Service components provide for background processing when an application’s activity leaves focus and another GUI application comes in the foreground.

3. Broadcast receiver components provide a general mechanism for asynchronous event.

4. Content provider is for sharing of data between Applications.

### 3.3.2.2 Application Framework

By providing an open development platform, Android offers developers the ability to build extremely rich and innovative applications. Developers are free to take advantage of the device hardware, access location information, run background services, set alarms, add notifications to the status bar, and much, much more.

Developers have full access to the same framework APIs used by the core applications. The application architecture is designed to simplify the reuse of components; any application can publish its capabilities and any other application may then make use of those capabilities (subject to security constraints enforced by the framework). This same mechanism allows components to be replaced by the user. Underlying all applications is a set of services and systems, including:
• A rich and extensible set of Views that can be used to build an application, including lists, grids, text boxes, buttons, and even an embeddable web browser.

• Content Providers that enable applications to access data from other applications (such as Contacts), or to share their own data.

• A Resource Manager, providing access to non-code resources such as localized strings, graphics, and layout files.

• A Notification Manager that enables all applications to display custom alerts in the status bar.

• An Activity Manager that manages the lifecycle of applications and provides a common navigation back stack (see Figure 3.2).

Figure 3.2. Android architecture.

3.3.2.3 LIBRARIES

Android includes a set of C/C++ libraries used by various components of the Android system. These capabilities are exposed to developers through the Android application framework. Some of the core libraries are listed below:
- System C library - a BSD-derived implementation of the standard C system library (libc), tuned for embedded Linux-based devices.
- Media Libraries - based on Packet Video’s OpenCORE; the libraries support playback and recording of many popular audio and video formats, as well as static image files, including MPEG4, H.264, MP3, AAC, AMR, JPG, and PNG.
- Surface Manager - manages access to the display subsystem and seamlessly composites 2D and 3D graphic layers from multiple applications.
- LibWebCore - a modern web browser engine which powers both the Android browser and an embeddable web view.
- SGL - the underlying 2D graphics engine
- D libraries - an implementation based on OpenGL ES 1.0 APIs; the libraries use either hardware 3D acceleration (where available) or the included, highly optimized 3D software rasterizer
- FreeType - bitmap and vector font rendering
- SQLite - a powerful and lightweight relational database engine available to all applications. [14, 15]

3.3.2.4 ANDROID RUNTIME

Every Android application runs in its own process, with its own instance of the Dalvik virtual machine. Dalvik has been written so that a device can run multiple VMs efficiently. The Dalvik VM executes files in the Dalvik Executable (.dex) format. It was designed specifically for Android running in limited environment, where the limited battery, 25 CPU, memory and data storage are the main issues. Android gives an integrated tool “dx”, which converts generated byte code from .jar to .dex file, after this byte code becomes much more efficient to run on the small processors. See Figure 3.3.

![Conversion from .java to .dex file.](image)

3.3.2.5 LINUX KERNEL

This is the bottom layer of android architecture. It is used for memory management, process management. You will never have access to this layer. This provides better
performance, better scalability, improved threading support, networking features. The kernel also acts as an abstraction layer between the hardware and the rest of the software stack.

### 3.3.2.6 The Manifest File

Every application must have an AndroidManifest.xml file (with precisely that name) in its root directory. The manifest presents essential information about the application to the Android system, information the system must have before it can run any of the application's code. Among other things, the manifest does the following:

- It names the Java package for the application. The package name serves as a unique identifier for the application.
- It describes the components of the application — the activities, services, broadcast receivers, and content providers that the application is composed of. It names the classes that implement each of the components and publishes their capabilities (for example, which Intent messages they can handle). These declarations let the Android system know what the components are and under what conditions they can be launched.
- It determines which processes will host application components.
- It declares which permissions the application must have in order to access protected parts of the API and interact with other applications.
- It also declares the permissions that others are required to have in order to interact with the application's components.
- It lists the Instrumentation classes that provide profiling and other information as the application is running. These declarations are present in the manifest only while the application is being developed and tested; they're removed before the application is published.
- It declares the minimum level of the Android API that the application requires.
- It lists the libraries that the application must be linked against.

The manifest is a structured XML file and is always named AndroidManifest.xml (this is a meta data file generated automatically with every Android project) for all applications. It does a number of things in addition to declaring the application’s components, such as naming any libraries the application needs to be linked against (besides the default Android library) and identifying any permission the application expects to be granted. But the principal task of the manifest is to inform Android about the application’s components. [15]
3.3.3 Development in Eclipse with ADT

Eclipse and the ADT plugin provide GUIs and wizards to create all three types of projects (Android project, Library project, and Test project):

- An Android project contains all of the files and resources that are needed to build a project into an .apk file for installation. You need to create an Android project for any application that you want to eventually install on a device.
- You can also designate an Android project as a library project, which allows it to be shared with other projects that depend on it. Once an Android project is designated as a library project, it cannot be installed onto a device.
- Test projects extend JUnit test functionality to include Android specific functionality. For more information on creating a test project, see Testing from Eclipse with ADT.

3.4. SQLITE

SQLite is an Open Source Database which is embedded into Android. SQLite supports standard relational database features like SQL syntax, transactions and prepared statements. In addition it requires only little memory at runtime (approx. 250 KByte). SQLite supports the data types TEXT (similar to String in Java), INTEGER (similar to long in Java) and REAL (similar to double in Java). All other types must be converted into one of these fields before saving them in the database. SQLite itself does not validate if the types written to the columns are actually of the defined type, e.g. you can write an integer into a string column and vice versa.

SQLite is available on every Android device. Using an SQLite database in Android does not require any database setup or administration. You only have to define the SQL statements for creating and updating the database. Afterwards the database is automatically managed for you by the Android platform. Access to an SQLite database involves accessing the filesystem. This can be slow. Therefore it is recommended to perform database operations asynchronously, for example via the AsyncTask class.

If your application creates a database, this database is saved in the directory DATA/data/APP_NAME/databases/FILENAME.

The parts of the above directory are constructed based on the following rules. DATA is the path which the Environment.getDataDirectory() method returns. APP_NAME is your application name. FILENAME is the name you specify in your application code for the database.
Android.database.sqlite contains the SQLite database management classes that an application would use to manage its own private database. Applications use these classes to manage private databases. If creating a content provider, you will probably have to use these classes to create and manage your own database to store content.

3.5 CALLBACK

Java does not have any procedure to implement function pointers unlike other procedural languages like C and partially object oriented language C++. Callbacks are a way of passing functions as parameters. The callee can call the function passed to it as a parameter as often as it wants, with whatever parameters it wants.

We need callback to decouple UI from actual implementation and also providing API structure so that any other developer can used application as an API to develop its own graphical user interface on top of the API.

Callback makes debugging easier and code reusable. It adds object oriented feature to the actual development.

3.6 PROTOCOL

HTTP functions as a request-response protocol in the client-server computing model. In HTTP, a web browser, for example, acts as a client, while an application running on a computer hosting a web site functions as a server.

The client submits an HTTP request message to the server. The server, which stores content, or provides resources, such as HTML files and images, or generates such content as required, or performs other functions on behalf of the client, returns a response message to the client. A response contains completion status information about the request and may contain any content requested by the client in its message body. [6]

- Request Methods: HTTP defines nine methods (sometimes referred to as “verbs”) indicating the desired action to be performed on the identified resource. What this resource represents, whether pre-existing data or data that is generated dynamically, depends on the implementation of the server. Often, the resource corresponds to a file or the output of an executable residing on the server.

- HEAD: Asks for the response identical to the one that would correspond to a GET request, but without the response body. This is useful for retrieving meta-information written in response headers, without having to transport the entire content.
• GET: Requests a representation of the specified resource. Requests using GET (and a few other HTTP methods) SHOULD NOT have the significance of taking an action other than retrieval.

• POST: Submits data to be processed (e.g., from an HTML form) to the identified resource. The data is included in the body of the request. This may result in the creation of a new resource or the updates of existing resources or both.

• PUT: Uploads a representation of the specified resource.

• DELETE: Deletes the specified resource.

• TRACE: Echoes back the received request, so that a client can see what (if any) changes or additions have been made by intermediate servers.

• OPTIONS: Returns the HTTP methods that the server supports for specified URL. This can be used to check the functionality of a web server by requesting ‘*’ instead of a specific resource.

• CONNECT: Converts the request connection to a transparent TCP/IP tunnel, usually to facilitate SSL-encrypted communication (HTTPS) through an unencrypted HTTP proxy.

• PATCH: Is used to apply partial modifications to a resource. HTTP servers are required to implement at least the GET and HEAD methods and, whenever possible, also the OPTIONS method.
CHAPTER 4

IMPLEMENTATION AND RESULT

4.1 HOW TO USE THE APPLICATION

Let’s start with a story. “Mary” is a business owner a small restaurant and she wants to expand her business. For that she has to advertise her business. So, she has to contact the advertising agencies. Thus, she is advertising her business. But this is very time consuming process and also it cost more money. One day she was out in the market and she saw an advertisement about managing advertising business through smart phone. She can advertise her business more efficiently by selecting more crowded area where maximum people can see her advertisement. So, at this moment this application comes in handy. First user has to select the area on the map which he/she thinks is more crowded. It means many people can see his/her advertisement. After selecting the area, user can select the screens on which he/she can display the advertisement. After selecting the area and screens, user just has to upload an image which he/she wants to display on the big billboards. This is very simple, easy to understand and time and money saving process.

4.2 HARDWARE REQUIRED

Any device installed with Android OS 1.5 or later version having Google Map API would be able to run this application. Currently, Application screen shots shown in this chapter is run on Android OS 2.3.6 (Gingerbread ) as shown in Figure 4.1.

In order to run this application we require Wi-Fi connection or 3G connection and Location Based Services. All android phones has built in Wi-Fi Manager that is available to connect with 33 available Wi-Fi hot spot or with their data networks and thus with Internet. As shown in Figure 4.2 how we can connect to Wi-Fi browsing through Menu->Settings->Wireless & networks ->Wi-Fi Settings. Also shown in Figure 4.3 how we can connect to 3G browsing through Menu->Settings->Wireless & networks ->Mobile networks.
Figure 4.1. Information about the phone in phone settings. Source: Picture taken by author with a Samsung, Inc. Galaxy S2 phone.

4.3 APPLICATION ON HOME SCREEN

To start an Application go to home screen and select application icon. Application name is based on google maps because it is based on google maps. As shown in Figure 4.4 an Icon and name of an Application on Home Screen of the phone.
4.4 ZOOM CONTROLS ON MAP

User can zoom in and zoom out of the map. He/she just have to double click the touch screen and then zoom in and zoom out buttons will appear on the bottom of the screen. By clicking on any one of them, user can go to any area on the map. Figure 4.5 shows the zoom controls on map.

4.5 SELECTING AN AREA ON MAP

So, when a user open this application, it will directly lead the user to google map screen. So, this screen has three different buttons as well on top of the map. Figure 4.6 shows
how map screen will look like. If user wants to select any area on the map, he/she will have to touch the screen only once. User will have to touch the screen multiple times to create polygon which will select the area on the map. Once, the user touches the screen one time, it will display a pinpoint on the map which confirms it as one point. Then he/she has to touch on any other point on the map. When user will select another point on map, this application will automatically draw a line between those two points. After selecting multiple points on
Figure 4.4. Home page screen of the phone. Source: Picture taken by author with a Samsung, Inc. Galaxy S2 phone.
Figure 4.5. Zoom controls. Source: Picture taken by author with a Samsung, Inc. Galaxy S2 phone.
the map, when user wants to close the polygon, he/she just have to click on showarea button, which will automatically form a polygon on the map with selected points.

When the user selects any point, this application will get the latitude and longitudes of that particular point by the touch and then it will display a point on map according to those latitude and longitudes. Here is the example for that process:

Figure 4.6. Map screen of the application. Source: Picture taken by author with a Samsung, Inc. Galaxy S2 phone.
String coordinates14[] = {"32.764796", ",-117.046836"};
double lat14 = Double.parseDouble(coordinates14[0]);
double lng14 = Double.parseDouble(coordinates14[1]);

p14 = new GeoPoint(
    (int) (lat14 * 1E6),
    (int) (lng14 * 1E6));

OverlayItem overlayItem14 = new OverlayItem(gp, null, null);
mapItemizedOverlay.addOverlay(overlayItem14);

Once, user has selected multiple points on the map, an algorithm will draw a line between the point which you select and previously selected point. here is an example of that algorithm:

Paint mPaint = new Paint();
mPaint.setDither(true);
mPaint.setColor(Color.BLUE);
mPaint.setStyle(Paint.Style.FILL_AND_STROKE);
mPaint.setStrokeJoin(Paint.Join.ROUND);
mPaint.setStrokeCap(Paint.Cap.ROUND);
mPaint.setStrokeWidth(2);

Point gp1 = new Point();
Point gp2 = new Point();

Path path = new Path();
Projection projection = mapv.getProjection();
projection.toPixels(p1, gp1);
projection.toPixels(p2, gp2);
path.moveTo(gp1.x,gp1.y);
path.lineTo(gp2.x,gp2.y);

Figure 4.7 to 4.9 shows the whole procedure of selecting the an area on the map.
Figure 4.7. Selecting a point on map. Source: Picture taken by author with a Samsung, Inc. Galaxy S2 phone.
Figure 4.8. Selecting multiple points on map. Source: Picture taken by author with a Samsung, Inc. Galaxy S2 phone.
Figure 4.9. Selecting a polygon on map. Source: Picture taken by author with a Samsung, Inc. Galaxy S2 phone.
4.6 Show Screens Inside Selected Area

After selecting the area, when user clicks on the screen button, all the available screens within that selected area will appear on the map. It is shown in Figure 4.10. Those screens can be from different network providers. Figure 4.10 shows different screens from two different network providers.

4.7 Uploading an Image for an Advertisement

When user clicks on the “adselect” button from the map screen, it will move to the new screen on which user has to select an image for the advertisement. By clicking on the browse button, it will direct to the sdcard content of the device. From here, user can select an image to which he/she wants to display it on screen. After selecting the image, user just have to hit the upload button, which will upload that image on the server and from the server, it can be uploaded on those selected screens. Figure 4.11 and Figure 4.12 shows the image select screen before and after selecting an image respectively.

4.8 Result

Start an Application by clicking icon on home screen, application will start loading and display first screen which is google map. Figure 4.6 shows first screen after application loaded successfully. Buttons on the top side of the map are the part of map overlay. Application can be run on 3G as well as wifi.

So first, user has to select any area on the map. So, he/she will have to select multiple points on map and when he/she wants to close the polygon, he/she just have to click on the “showarea” button. After creating the polygon, system has to display the available screens inside the selected area. So, for that user has to click on the “screen” button. So, once the user clicks on the “screen” button, system will display all the available screens inside that selected area. So, now area and screens are selected. After that, user has to upload an image which he/she wants to display on the selected screens. For that he/she has to click on the “adselect” button. When user clicks on the “adselect” button, it will go to the new screen which is imageupload screen. Here user has to click on “browse” button to select an image. User can select any image from his/her device, once he/she clicks on the “browse” button. After selecting an image, user has to upload that image onto server. For that, user has to click on
Figure 4.10. Screens from different network providers. Source: Picture taken by author with a Samsung, Inc. Galaxy S2 phone.
Figure 4.11. Image select screen before selecting an image. Source: Picture taken by author with a Samsung, Inc. Galaxy S2 phone.
Figure 4.12. Image select screen after selecting an image. Source: Picture taken by author with a Samsung, Inc. Galaxy S2 phone.
“upload” button. Once, user clicks the “upload” button, the image will be uploaded on the server and from the server it can be shown on those selected screens.

4.9 **PORTABILITY**

This application is using Google map APIs. So, project taken for the purpose of changes should have the Google map APIs feature on the Android platform.
CHAPTER 5

CONCLUSION

There had been number of challenges while developing this application on android framework. First, I would like to talk about the design and user interface of this application. I have tried to make this application as easy as possible. So that any user can use this application very easily. This application was made from end user perspectives. Next thing is the way of writing a code. Since, this application is for smart phones, it will use some internal memory of the device. I have tried to write the code in such a way that it occupies very less amount of internal memory.

There were many issues while developing this application like less knowledge about the advertising business, outdoor advertising, mobile advertising, flow of developing this application, different functionalities and different methods for those functionalities, how to write a code which is easy to understand for any user. But, I would like mention one problem which is really difficult.

I wanted to give freedom of selecting an area by drawing randomly to the user. But since, this functionality is not available in android, user has to select different points to draw an area. I researched a lot about this functionality and I found out that similar type of functionality is available for the web based program, where user can open the google map and then draw any random area on the map. Libraries which are required to support this functionality are available for web based program. But, for android, user has to touch the screen to draw the area. Android does not have all the libraries which support google map and touch event for drawing. Some libraries are not available in the market yet. So, this functionality is not possible in android. So, instead of giving the freedom to user of drawing random area, I created the functionality of creating a polygon on the map. It is as easy as drawing a random area on the map.
CHAPTER 6

FUTURE PLAN

Currently, this application is made in WiMAX. But currently LTE has just started to cover the market which the next generation of the wireless network. There are many benefits for LTE over WiMAX like LTE supports handover and roaming with the 3GPP mobile networks. However, these services are not easy to achieve with WiMAX. So, this new generation will be a major part in future and it will stay there for many years. This application is capable to run on LTE network as well.

The main reason why I chose WiMAX compared to LTE is because currently it is widely used in United States. Plus, LTE is fairly new in United States. Even today WiMAX is also fairly new in some countries like some Asian countries. In these countries LTE hasn’t been introduce and it may take a decade to introduce LTE in these countries. So, WiMAX is more popular in those countries. So, I choose WiMAX over LTE.

We can provide a separate module to keep track of screens and users. For example, we can provide the login module for the user and then we can keep track of each user. We can provide history of each user, how many times he/she has used this system. We can store all the images which are uploaded by the user. Also, we can assign unique id to each screen and with that we can keep track of each screen as well. Like how many screens are booked and how many screens are available at a given time within selected area on google map. Also, if the screen is currently booked, when will it become free etc. If the screen is booked for longer duration like a month or more than that, then we can display some kind of symbol which indicates that the screen is booked for longer duration.

We can add bidding algorithm to this application in future, which can be very useful module. To understand it better, let’s consider the following example. If different user wants to display the advertisement on the same screen in same area for same time, then we can apply the bidding algorithm. So whoever wins that bid, can display his/her advertisement on the screen.
We can also provide one kind of algorithm to this application, through which a user can display special scheme for the advertisement, if there is any event occurring inside the selected area.

We can also provide date and time criteria for uploading an image. Through which a user can buy a screen using this system and then show a breakfast ad in the morning, lunch ad during the day, and dinner ad throughout the evening.

We can also add some algorithm to this application to restrict the users from uploading unacceptable images on the screens. This project improvises Google API for maps. We can also add other emerging technologies if available like Google goggle API that has inbuilt ability to do image processing. Google goggle API will help in finding out details of the famous landmark to select the area.

Currently, we don’t have any feature to keep track of screens and we can keep track of each screen in future, which I have already mentioned above. Having said that, there are two ways to keep track of those screens. One is, the administrator himself/herself can do this or we can get the statistics of those screens from the screen owner as well. These statistics contains information like currently how many screens are free or how many are busy, in future how many screens will free and how many will be busy. If a screen is already booked for future advertisement, then how long it will be booked or if the screen is how long it is free. It is easier to understand the current market flow as well occupancy of different screens, which makes the decision easier for the business owner for which screen to select for their advertisement.

Currently, there is one drawback of the system where user can only upload the image which he/she has already in his/her android device. If a user wants to upload an image from the browser, then he cannot do this. So, in future we can improve this functionality and give the user the freedom of selecting an image from the browser as well. If the business owner wants to create an special image for his/her advertisement, he can directly go to an artist to create that image also. So, instead of providing a softcopy of an image, we can provide an option for the business owners to directly contact the graphic designer to create an advertisement, which can make extra money for the application owner as well as the graphic designer and it can save some money to the business owner also.
We can also provide this version of application to some users for beta testing. By saying some users, I mean some advertisers, some business owners and some screens owners also. So, once we can get the feedback from them and then we can add more functionalities to the system or even modify the original functions also. For example, Let say after selecting the screens, user goes to the upload page. After reaching the upload page, he thought that he selected wrong screens and he wants to go back make some changes in the selected screens. So he/she will hit the back button. But once the user hits the back button, all the data from the previous activity will be lost. So, at that moment user has to start it from the scratch. And, I got this feedback from my thesis adviser. So, we can make this type of changes also.

Another thing is we don’t have any system to upload contact the screen owner’s server directly to upload an image. Even, we don’t know how their algorithm works. So, in this application, I have just assumed that the flow of this application will work in this direction. But, it can be a wrong direction. So, after getting the feedback from the screen owners, advertisers and some other users, we can decide proper algorithm to connect the screen owner’s server. Also, there is not a proper payment system in this application. So, we can also add that in future after getting the feedback.

There are many possibilities and functionalities which we can add in the future work for this application. So, for all these things, first we have to give it for beta testing to some users. And from the feedback, we can decide the next step. This is a core application which has some basic functionalities. So, it has functionalities but does not have reaction. So, feedback is must.
REFERENCES

WORKS CITED


WORKS CONSULTED

