Mobile Voting using Android

Masters Research Project

Rajesh Vedala
Rimon Gargho
Wachiravute Lackchaloemporn (Axe)
MS in Wireless Communications

Laith Salih Hadi
MS in Computer Science

Under guidance of

Professors
Dr Mohammad Amin
Dr BhaskarRaj Sinha
Dr Pradip Peter Dey

Jan 26th 2013

School of Engineering, Technology, and Media
National University
San Diego, California
Mobile Voting using Android

I have examined the final copy of this project report. I have accepted this project in partial fulfillment of the requirements for the degree of Master of Science in Wireless Communications.

------------------------------------------
Dr. Mohammad Amin
Professor
DEDICATATION

To Our Family, Parents, Friends, and Relatives
ACKNOWLEDGEMENTS

All praise is due to God, the most gracious, and the most merciful

Firstly our heartfelt thanks to all family members for their love and support in encouraging pursuing Grad studies which lead to cherish a successful life long career and like to thank Professor Dr Mohammed N Amin and Dr Raj B Sinha for their meticulous support during entire period of the project, without which this work would not been possible and who guided me throughout my grad program and has contributed significantly in shaping me to a wireless telecom expert and a researcher. Also, special thanks to my advisor for expertise guidance and friendly attitude, and would like to thank friends who helped in precise way towards grad studies and this project. We unfeignedly hope we are able to be as good to the people who contributed to our growth as a person, engineer and good human and good to us.
ABSTRACT

Mobile computing has sprung up considerably in Holocene epoch, with an increase number of mobile apps being evolved to work on next generation networks. In current scenario SW is designed to necessitate and drive mobile components interacting Packet switch network leading to further simplication tasks to a greater extent. Design level mostly into development of database, web server, domain hosting along with crypting all interacting to a physical devices running on Android OS in direction of Mobile voting system along with further unique identity using NFC tag.

Keywords: Android, Mobile App, RFID, Mobile Voting, Web server and GPS Address.
# TABLE OF CONTENTS

1.0 Introduction 10  
1.1 Triangulation of Mobile Voting 11  
1.2 Android 12-33  
1.3 Web Hosting 34-49  
1.4 RFID/NFC 50-53  
1.5 MySQL 54-68

2 Project Description / Statement of project 69-71

3 Application Design and Block diagram 72-73

4.1 Android App porting / installation 74-75  
4.2 RFID / NFC 76  
4.3 Voting Process / Results 77-81  
4.4 Webhosting 82-84  
4.5 Database 85

5 Results 86-87

6 Future Developments 88

7 Conclusion 89

8 Programming instruction for the developed code 90-96

9.0 Mobile Voting Source Code 97  
9.1 ANDRIOD 97-129  
9.2 Web hosting 130-141  
9.3 Andriod and webhosting linking files 142-149  
9.4 Test Cases 150-154  
9.5 MySQL Database 155-157

10 List of Acronyme 158-159

11 References 160-166
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Mobile Voting Android App</td>
<td>10</td>
</tr>
<tr>
<td>2</td>
<td>Architecture of an Android OS</td>
<td>15</td>
</tr>
<tr>
<td>3</td>
<td>Dalvik Virtual Machine</td>
<td>19</td>
</tr>
<tr>
<td>4</td>
<td>Mobile Handset with Jelly Bean</td>
<td>25</td>
</tr>
<tr>
<td>5</td>
<td>Android devices partake towards various versions</td>
<td>26</td>
</tr>
<tr>
<td>6</td>
<td>RFID reader Block Diagram</td>
<td>51</td>
</tr>
<tr>
<td>7</td>
<td>Logical Architecture</td>
<td>60</td>
</tr>
<tr>
<td>8</td>
<td>Mobile Voting three modules integration</td>
<td>70</td>
</tr>
<tr>
<td>9</td>
<td>Mobile Voting Block Diagram</td>
<td>72</td>
</tr>
<tr>
<td>10</td>
<td>Passive RFID Block Diagram</td>
<td>72</td>
</tr>
<tr>
<td>11</td>
<td>Android Core Blocks</td>
<td>73</td>
</tr>
<tr>
<td>12</td>
<td>Android Blocks</td>
<td>73</td>
</tr>
<tr>
<td>13</td>
<td>Android Mobile Voting using RFID</td>
<td>76</td>
</tr>
<tr>
<td>14</td>
<td>Mobile Voting App in Android Device</td>
<td>77</td>
</tr>
<tr>
<td>15</td>
<td>login screen</td>
<td>77</td>
</tr>
<tr>
<td>16</td>
<td>login screen with keypad</td>
<td>77</td>
</tr>
<tr>
<td>17</td>
<td>login screen with RFID, Authentication</td>
<td>78</td>
</tr>
<tr>
<td>18</td>
<td>Login Successful</td>
<td>78</td>
</tr>
<tr>
<td>19</td>
<td>Voting options</td>
<td>78</td>
</tr>
<tr>
<td>20</td>
<td>Voting with selectable options</td>
<td>79</td>
</tr>
<tr>
<td>21</td>
<td>Option selection</td>
<td>79</td>
</tr>
<tr>
<td>Figure 22</td>
<td>Vote selection Locked</td>
<td>79</td>
</tr>
<tr>
<td>Figure 23</td>
<td>Vote confirmation</td>
<td>79</td>
</tr>
<tr>
<td>Figure 24</td>
<td>Vote Submission</td>
<td>79</td>
</tr>
<tr>
<td>Figure 25</td>
<td>Network Signal settings</td>
<td>80</td>
</tr>
<tr>
<td>Figure 26</td>
<td>NFC module option</td>
<td>80</td>
</tr>
<tr>
<td>Figure 27</td>
<td>NFC module turn on selection</td>
<td>80</td>
</tr>
<tr>
<td>Figure 28</td>
<td>RFID authentication with user name and password</td>
<td>80</td>
</tr>
<tr>
<td>Figure 29</td>
<td>Vote Submission</td>
<td>80</td>
</tr>
<tr>
<td>Figure 30</td>
<td>Wrong login details</td>
<td>80</td>
</tr>
<tr>
<td>Figure 31</td>
<td>Administrator mode for web hosting</td>
<td>83</td>
</tr>
<tr>
<td>Figure 32</td>
<td>System engineer for web hosting</td>
<td>83</td>
</tr>
<tr>
<td>Figure 33</td>
<td>Login screen for web hosting</td>
<td>84</td>
</tr>
<tr>
<td>Figure 34</td>
<td>Vote submission result in Android</td>
<td>86</td>
</tr>
<tr>
<td>Figure 35</td>
<td>System engineer for html post towards results</td>
<td>86</td>
</tr>
<tr>
<td>Figure 36</td>
<td>Administrator mode for html post towards results</td>
<td>86</td>
</tr>
<tr>
<td>Figure 37</td>
<td>GPS Google maps results</td>
<td>87</td>
</tr>
<tr>
<td>Figure 38</td>
<td>Web Hosting Source Coding files</td>
<td>91</td>
</tr>
<tr>
<td>Figure 39</td>
<td>Logout button</td>
<td>91</td>
</tr>
<tr>
<td>Figure 40</td>
<td>Mobile voting logo</td>
<td>91</td>
</tr>
<tr>
<td>Figure 41</td>
<td>Home page mobile voting</td>
<td>91</td>
</tr>
<tr>
<td>Figure 42</td>
<td>Database graphical programming main screen</td>
<td>92</td>
</tr>
<tr>
<td>Figure 43</td>
<td>User login details graphical programming</td>
<td>93</td>
</tr>
<tr>
<td>Figure 44</td>
<td>Voting questions/option, its graphical programming</td>
<td>93</td>
</tr>
</tbody>
</table>
Figure 45  Vote casted, updated lat long, vote time, Q1 its graphical programming  94
Figure 46  Vote casted, updated lat long, vote time, Q2 its graphical programming  94
Figure 47  Vote casted, updated lat long, vote time, Q3 its graphical programming  95
Figure 48  Vote casted, updated lat long, vote time, Q4 its graphical programming  95
Figure 49  Vote casted, updated lat long, vote time, Q5 its graphical programming  96
1.0 INTRODUCTION

In the present, mobile phone is very useful to communications, entertainments, education, game, business and socializations. Thus many applications were created in order to support their requirements of user. Our project Mobile Voting System using Android Application, The purpose of this project is convenience of admins and voter, and accuracy to count score of voting. The Mobile Voting System using basically operation system based on android phone that is the great deal to develop it to iPhone application or internet application in the future. The designed android application has several pages which consist of home page, login page, login error page, voting page, and last page fulfilling the requirement and supporting great accessibilities figure – 1.

figure – 1 Mobile voting (Android App)
1.1 TRIANGULATION OF MOBILE VOTING

There’s a huge utilization of computerized information towards public administration from the emerging moments of internet period from 1990’s and played a vital role for government organizations towards its operations for fruitful task executions in very less time. However on time it got useable to private organizations and citizen due to its contrivance and convenience, and so on of its versatilely it set forth almost on everywhere, some of which in context to this paper details as administrative tasks for governments such as Electoral Voting, Official addressing, government operations, government services and so on [1] [2]. This provided flexibleness towards physical approachability of voting stations allowing cushy task of entire voting system [2].

Development of online voting is of two types in generic, 1) Voting Superintended by the physical presence of electoral authorities, 2) Voter’s physical presence is at this solo cognitive factor by utilizing vote either through online voting, mobile phones, iptv etc., Withal to the voting system it contingent’s to three main degrees as 1) Information acquisition, 2) Opinion formation and 3) Decision itself – [3]. When it comes to practical implementation it staggeringly depends of three divisions as (1) Law, (2) Socialization and (3) Technology – [3]. Despite all these mobile voting and internet voting are on a peak of popularity, where as mobile voting can lead to overcome from the voting station scarceness particularly in rural areas, when it comes to internet voting more globally convenient as mobile phones still in emerging emplacement, of these it may be vulnerable to possible breakdowns in terms of misuse but still provides the easier access and high voting ratio.
1.2 ANDROID

Android, every week there’s hundreds of millions of mobile devices with operating system as android, developed by Google introducing in more than 190 countries round the world. Android is the largest operating systems for mobile computing devices as a platform and growing fast with every upcoming day, there is a addition of another million user’s who turn on their computing devices with this Android powerful OS for the first time to explore for apps, games, and other digital content [4].

- Providing a world class platform for application development and as well as games for its users everywhere, while offering as an open marketplace for distributing them instantly [4].

- The term has been derived from the Greek word andr-, which means “man or male” and the suffix - eides, used to mean “alike or of the species”. This together means nothing but as “being human” [5].

- This is a software stack for the mobile devices which includes operating system, middleware along with other key applications towards its complete modular installation package. Further this is platform is based upon the Linux operating system which was developed by Google and the Open Handset Alliance (O H A).

- Google Incorporation purchased the initial developer of the Android software i.e. Android Incorporation in August 2005 making Android Incorporation. Android operating system is fully based on a modified version of the Linux kernel system, which are most notable examples of free and open source software. Google and other members of Open Handset Alliance cooperatively organized on the Android's development and release. The Android Open Source Project which also popularly known as A O S P is tasked with the maintenance and further development of Android development [6].
Open Handset Alliance (O H A)

On the November 5, 2007 the Open Handset Alliance, a consortium of several companies like Google, HTC, LG and many more unveiled itself. On the same day, the Open Handset Alliance also realised their first product called Android, a mobile device platform based upon the modified Linux kernel [7].

FEATURES of the Android System:

1. Dalvik Virtual Machine

   It is the core integral module of the Android based on extremely low-memory virtual machine functioning at software level for computing systems working on embedded systems providing faster performance ideally tuned towards CPU attributes creating a unique file format called .DEX through build time post processing with a Conversion option from Java classes and .DEX format by using “dx” tool [8].

2. INTEGRATED Web Browser

   Android operating system uses open source web browser as on Web Kit which is of into two pass layout and frame flattening, this two pass layout loads the requested data content with absence of holding block elements, like external CSS file or external JavaScript with a spell of providing all the resources in porting to the device. Frame flattening are block of individual single loads embedded into rafts for processing block which in turn loads into web browser leading to higher browsing speed with an entire sophistication of cache for faster processing towards browsing [9].
3. **SQLite**

   This is similar to SQL database with a lower version requirement towards its functioning while providing the faster and efficient performance, of which it is integrated a one of the core module of entire android operating system which is into typical a 500kb enormously belittled sizing and completely grounded on function calls and a single file of which digital content is stored [9].

4. **Connectivity**

   Beside its advance colligate with the sub modules of the operating system, it supports a efficacious connection towards telecom technologies as such GSM, CDMA, Bluetooth, GPRS, NFC, 4G technologies etc., [9].

5. **Messaging**

   As such with above mentions it does offer additional features as SMS, MMS, and other communication protocol connection systems along with its sophistication [9].

6. **Media Support**

   In aspect of the operating system performance it do provide additional application packages towards media system [9].

7. **ADDITIONAL Hardware Support**
In context to its operating system along with its software sub modules, it do support all the GPS, touch screen, accelerometers, HDMI, graphics, and so on with its drivers and its software packages [10].

8. MARKET

The Android operating system does have an application package installation where in which a vast variety of applications can be installed of which it is into entertainments, songs, games, etc., [11].

9. MULTITASKING

This is thread based operating system supporting much higher level multitasking for a several co-occurent functioning along with faster performance and does have a competitive CPU, GPU, Memory modules and so on with its architecture as figure – 2,[12] .

ANDROID ARCHITECTURE

![Architecture of an Android operating system figure – 2.](image-url)
Applications

This operating system do have the minimal applications installed as a core such as messaging, email, calendar, browser, maps and so on and do support installation of several other applications written using java programming language [13].

Application Framework

It is a multi threading building blocks for a sophisticated frame functioning towards a multi tasking faster processing and all these are preinstalled with the provided OS with a extension of supporting own components / applications etc [13].

**The most important parts of the framework are as follows:**

**Activity Manager:**

Controlling the life cycles of application and maintaining user navigation cache of back stack [13].

1) **Content providers:**

The data encapsulated for functioning and sharing applications, such as apk extension files and some of few as messaging. [13].

2) **Resource manager:**
All software applications towards its performance level for its functioning rather than its realtime code development and is used for managing various installed applications.

3) **Location manager:**

It maintains several realtime GPS measurements towards geolocation with an option of turn on and off for the GPS module, network triangulation of which this module is responsible towards it.

4) **TELEPHONY MANAGER:**

Provides core telephoning functionalities

5) **LIBRARIES**

Kernel layer and Android Native libraries form a sophisticated operating system library files for multitasking while simultaneously operating several hardware modules at complex architecture which is developed by device manufacturers in collaboration with android OS developers [12].

Some of the most important native libraries are the following:
- **SURFACE MANAGER**: Responsible for operating and maintaining display system including 3D graphics for the multiple applications along with multitasking of it.

- **MEDIA LIBRARIES**: Providing support to playback and recording of multiple audio / video formats.

- **SQLite**: It is approx 500kb enormously belittled sizing and completely grounded on function calls and a single file of which digital content is stored[9].

- **FREE TYPE**: Font rendering used for rasterization of digital content and its operations [6].

- **SGL**: A dedicated GPU engine towards 2 dimension processing [6].

- **LibC**: A BSD – Used for embedded devices running on Linux as operating system with the implementation on the standard C library [6].

**Android Runtime**

Based on the core libraries towards system functionality is designed and included by a set of system core libraries using java programming language as a development platform [13].

Android uses a key module as Dalvik virtual machine which is powerful module for high processing towards cache together with other modules making a powerful OS as android which inturn runs as a high speed processing running on its own processor making it to efficiently run multi tasking and this follows a (.dex) file format with working functionality completely as register based and executes files for the Dalvik Executables leaving a minimal memory footprint and includes a tool as “dx” for its file transformation [8] further more this is a linux kernel functionality in multi threading with a low level memory management while processing higher files systems [8].
Dalvik Virtual Machine

This is java interpreter machine compiling the code line by line completely optimized towards android developed to run at low level memory computing devices with prominent aspects with capability to run along with a application compilation enhancing runtime performance, however this consists a special format and is in byte level code by “.dex” as extension executable files and doesn’t really depend as functioning on java and delivers a good performance at multilevel runtime environment with a ultra low power consumption, below as its functioning process figure – 3.

Dalvik Virtual Machine figure – 3.

This does own and use Apache Harmony Java implementation libraries and its more value reliable since its low power consumption with a high performance with multithreading and do have a vast libraries with a non fragmentary applications used for interfacing all modules and relates to the high performance functioning even at high scale multimedia application and do
support higher application easily with ultra low processing requirements and uses java virtual execution environment which was developed by Google.

The Java Standard Edition (Java SE) and the Java Mobile Edition (Java ME) along with core java libraries run by android which do have high amount of convergence [12].

**Linux Kernel**

It kernel interfaces as security, memory management, process management, network stack, etc., with a driver model as core system services with abstraction layer between hardware and software stacks for entire new embedded systems platform. [13].

- **Power Management**

  Android runs with ultra low power with a functioning of multithreading while providing high performance with multi tasking with zero power consumption of its CPU in case of no applications running on its platform [15].

- **BINDER (IPC) DRIVER**

  It does support multi drivers towards its various modules with interfacing options towards wide level applications for various usages with centralized servers for its realtime updates, and offer shared memory with high performance for synchronous calls [15].
Anatomy of an Android Application:

Entire android system block is into four stages embedded as a single OS

Activity:
User interface component, which corresponds to the one screen at a time. It means that for the simple application such as Address Book, the developer should have only one activity for displaying contacts and another activity component for displaying more detailed information for choosing name and etc. [16].

Intent Receiver:
It wakes up a predefined action through the external event [16].

Service:
A service is a task, which is done in the background process. It means that the user can start an application from the window and keep the service active, while browsing other applications [16].

Content Provider:
It allows sharing some of the data with other processes and applications. It is the best way to communicate the applications between application and the server [16].

DEVELOPMENT TOOLS
The Android SDK includes a variety of custom tools that helps the user to develop mobile applications on the Android platform. The three most significant tools are as follows:

- Full java IDE
- Graphical UI binders
o On-device developer options
o Develop on hardware device
o Develop on virtual machines
o Powerful debugging
o Testing
o Native development [17].

**Different versions of Android operating systems**

**Android history:**

The first release of Android beta was in November 2007 for the first time. Its commercial version 1.0 is released in September 2008. Android went many ongoing developments by Google in collaboration with (OHA), and had numerous updates towards core OS since initial development which made to offer plenty of new features to it systems, all together as of 2012 it have over 500 million devices running successfully all over the world with various versions as Donut, Froyo, Gingerbread, Ice Cream Sandwich, Jelly Bean etc., [18].

**Pre-commercial release versions (2007–2008)**

**Android alpha**

- The second of releases was R2-D2 code-named. Dan Morrill created the first mascot logos, with green color logo as current version is by Irina Blok.
- With credit to Ryan Gibson toward majority of the public releases from cupcake in april 2009 and on to various versions in naming schemes [18].

**Android beta**

- The main releases as nov 5th and 12th towards beta and its SDK [18].
a) **Android 1.0 (API level 1)**

First commercial version was into access as on 23 September 2008 by its developers[19].

b) **Android 1.1 (API level 2)**

This is initially came into access for HTC mobile corporation towards Dream only with a internal naming as “petit four” on Feb 9th 2009.

c) **Android 1.5 Cupcake (API level 3)**

This is fully onto a linux kernel system with 2.6.27 v, includes a new module as UI amendments for all releases there with release on April 30th 2009. [20] [21].

d) **Android 1.6 Donut (API level 4)**

With numerous updates and plenty of new key features based on kernel 2.6.29 it release this SDK under “dubbed Donut” on Sep 15th 2009 [20] [22].

e) **Android 2.0 Eclair (API level 5)**

Updates with some new features its SDK with same version as Donut there is a new release under “Éclair” on Oct 26th 2009 [20].

f) **Android 2.0.1 Eclair (API level 6) and Android 2.1 Eclair (API level 7)**

Bugs were fixed along with frame work changes and made minor amendments towards API on Dec 3rd 2009 and Jan 12th 2010 respectively.

g) **Android 2.2–2.2.3 Froyo (API level 8)**

With kernel update to 2.6.32 its named Froyo which is a frozen yogurt on May 20th 2010 [23].

h) **Android 2.3–2.3.2 Gingerbread (API level 9)**

With similar updates to 2.635 for the kernel it named Gingerbread for its SDK and released on Dec 6th 2010 [24].
i) **Android 2.3.3–2.3.7 Gingerbread (API level 10)**

With support for some new applications along with new hardware support as camera enhancement, battery enhancement, Bluetooth, animation and some bugs fixing with API improvements released on various stages ranging from Feb 9th, April 28th, July 25th, Sep 2nd, Sep 21st 2011 respectively.

j) **Android 3.0 Honeycomb (API level 11)**

With developed SDK towards tablets running fully on linux kernel with 2.6.36 version focused on several features as multitasking usage, redesign of keyboard, camera exposure, hardware acceleration, multi core processor, user data encryption ability, improved HTTS stack with SNI and several features and its optimization towards tablet and its support with holographic user interface along new virtual made release on Feb 24th 2011 [25].

k) **Android 3.1 Honeycomb (API level 12)**

Made supports for plenty applications joysticks, audio playback, http proxy and some features with a release on May 10th 2011.

l) **Android 3.2 Honeycomb (API level 13)**

Advancement in several features along with multi bug fixes along with enabling Google TV with hardware improvement support made 6 release from mid 2011 and early 2012[20] [26].

m) **Android 4.0–4.0.2 Ice Cream Sandwich (API level 14)**

It provided some new options as well as some new features like improvised copy paste functions, keyboard corrections, hardware acceleration, wifi direct, full HD video recording, sync, music player controls VPN framework for internet and so on released as ice cream sandwich on linux kernel 3.0.1 with SDK 4.0.1 on Oct 19th 2011[18] [27].
n) Android 4.0.3–4.0.4 Ice Cream Sandwich (API level level 15)

Made a major changes towards multimedia advancement like camera performance, video stabilization, graphics, Bluetooth functionality, database, along with fixing several bugs with two releases on Dec 16th 2011 and Mar 29th 2012 respectively along with a support on Adobe Systems Flash Player.

o) Android 4.1 Jelly Bean (API level 16)

Implemented several new features with competitive specs as such with improved camera app, google wallet, multichannel audio, usd audio, offline voice dictation, smoother user interface, gestures, new feature for swipe keyboard with some bug fixes on three difference times as July 9th 23rd and Oct 9th 2012 with SDK as 4.1 Jelly Bean on kernel 3.0.31[18].

p) Android 4.2 Jelly Bean (API level 17)

This the latest at most and the advanced as of now currently which runs with a advancement on always-on VPN, premium SMS confirmation, group texting, power controls notification, wireless display support, gestures towards keyboard, panorama photos, Bluetooth gamepads support and its advancement of new features with some bugs fixing towards performance enhancements with SDK as 4.2 on Nov 13th 2012 with updates as 4.2, 4.2.1, 4.2.2 on 27th Nov 2012 and 11th Feb 2013 respectively[16][28] with its main screen as figure – 4.

Mobile Handset with Jelly Bean figure – 4
**Future versions**

With bruits of Key Lime Pie as successor to the jelly bean besides the official confirmation there was a accidental leaked document on Feb 2013 from Qualcomm mentioning about it as release version on second quarter on 2013 mostly [29].

![Graphical representation of android devices partake figure – 5](image)

**Graphical representation of android devices partake figure – 5**

**Current Distribution**

The above pie chart in figure – 5 and below table is based on the number of Android devices that have accessed Google Play within a 14-day period ending on the data collection dated March 4th 2013 [30].

<table>
<thead>
<tr>
<th>Version</th>
<th>Codename</th>
<th>API</th>
<th>Distribution</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.6</td>
<td>Donut</td>
<td>4</td>
<td>0.1%</td>
</tr>
<tr>
<td>2.1</td>
<td>Éclair</td>
<td>7</td>
<td>1.5%</td>
</tr>
<tr>
<td>2.2</td>
<td>Froyo</td>
<td>8</td>
<td>3.2%</td>
</tr>
<tr>
<td>2.3 - 2.3.2</td>
<td>Gingerbread</td>
<td>9</td>
<td>0.1%</td>
</tr>
</tbody>
</table>
Verifying Back-End Calls from Android Applications:

- Application are hugely virtually run based on server-side back end along with the user name, passwords and some other personal content towards its specific application.
- These data are required for registration and user uniqueness and genuinity, hence it is on server side and is accessed through HTTP endpoints with client applications by POST request among the internet world which is a packet switch instead of running the mobile phones or communicating devices which uses circuit switch.

PROFESSIONAL Android Sensor Programming

METHODS USED TO DETERMINE LOCATION

Being Andriod as the advanced OS it uses network BTS/Node B triangulation method to determine the approximate geographical location instead of accessing GPS module which
consumes a lot of power from its battery, further more there too have certain pros and cons on the other aspect which is concerned with accuracy, tracking, etc.,

**GPS Provider**

Current mobile phones, tablets do actually have A-GPS instead of direct GPS module which requires lot of power to transreceives signal directly to the satellites, hence it depends on the servers of the satellites for providing the information of the nomadic device location and possible on simultaneous GPS [S-GPS] to standardize the accuracy.

**Defining device’s latitude longitude**

Classes:
- Lat Long Manager
- Lat Long Provider
- Lat Long
- Criteria
  - Defining Location Sources
    - Geo location provider
    - Server based location provider
    - Passive location provider

**Tracking Devices**

**COLLECTING LOCATION DATA**

- Receiver based updates with a Broadcast Receivers

In order for the lat long updates with the receiver a developer have to have as below
- Broadcast Receiver band – its level.
- Registering of android with broadcast receiver.
- Android receives the lat long info from the registered broadcast receiver along with the location manager.
- Broadcast receiver functionality extending.

- Registering the BroadcastReceiver with Android
- Attesting the registration
- Broadcast receiver registration manually
- Location registering updates with Pending Intent
- Multiple Receivers with one Intent responder

- VIEWING THE TRACKING DATA
- Tracking the location activity and displaying its data in graphical interface with Google maps accordingly.

- Google Maps Library Components
  Displaying Google maps with data, Track Location Activity uses as below from the Google Maps with external library which is embedded in the Google maps:
  - Map View
  - Overlay Item
  - Itemized Overlay
  - Map Activity
Android Open Accessory

A SHORT HISTORY OF AOA

It is almost a new feature of Android SDK, which is only announced by Google, with this is supports in advance level for external hardware as USB, NFC and some extension peripherals, however during release version it is still in early stages of development of the APIs and help in some idiosyncrasies as power requirements etc. the interest towards this also focused on ADK microcontroller working on Arduino hardware platform.

USB Host versus USB Accessory

This AOA uses USB host towards a USB transfer protocol for data tranreceiving through various USB ports especially in context to USB hub.

Electrical Power Requirements

The android device too works at regular specification in accordance with USB standards example 0.5A, 5V and further some sophistication devices do charge the Andriod device battery through USB and some do have have power USB instead regular DC charging ports

Supported Android Devices

Most advanced versions of these android do support this AOA and mostly these are known to link these file for functionality based on the updates for its support.
THE ANDROID DEVELOPMENT KIT (ADK)

ADK compatible versions do have the AOA mode embedded with the android kernel

DETERMINING DEVICE ORIENTATION

With the sophistications, devices do come with multiple new hardware features and some of which are as below used for various application, hence the android being a platform does support their drivers and best determination of its usage for an improvised utilization.

Various kind of sensors

- Gravity sensor
- Accelerometer and magnetometer
- Gravity sensor and magnetometer
- Rotation sensor

TYPE_ORIENTATION sensor may not be used in example application because it is deprecated [31].

Image-Processing Techniques

THE STRUCTURE OF IMAGE-PROCESSING PROGRAMS

Below is the general structured processing program for image processing.

Common Image-Processing Operations

1. Image-to-Image Operations

   Every pixel resulted from a mathematical operation for better productive performance towards point operations.
Thresholding:
It lies in the specified band to differentiate one from the other specific value for easier and smoother usage without much processing.

Histogram equalization:
Frequency count of pixel values resulting image contrast along from a small portion of pixel image value by means manipulating its data.

Conversion:
It is a point operation of special kind with image type from one to another without rendition of image pixels content in any other ways.

Reduction:
Using Invariant factor for abridging the size of image.

Spatial transformation:
Rearrange pixels positions without changing their values as such for stretching or rotating of the image.

Filtering:
Filtrate the undesirable artifact with context to the features of interest.

Grouping:
Process of diminution of image file and small assemblage features towards it with still producing output as same.

2. **Image-to-Object Operations**

   Count pixel value frequency and operation of computing histogram.

3. **Jon’s Java Imaging Library (JJIL)**

   It is important library which provides all most all of the image processing types its optimization for running ar low level memory processing devices as android [31].
1.3 WEB HOSTING

**Web server:**

Combination of both hardware and software towards computer device along with a software application to deliver the web content which is accessed through internet are called to be web servers.

Hardware:

Device which holds the websites content usually in html as a collection is known as website hardware, typically servers.

Software:

A software program which runs on internet whose main purport is to serve the client by responding with appropriate data block for the specific request web page, this process is also known as client-server interaction [33].

**Web browser:**

A software program used to retrieve the digital files from the servers for accessing irrespective of time, location on the web is known as web browser, on the other aspect the URI, Uniform Resource Identifier identifies the web pages, images, video or other content in the web servers.

Tags the content of text in the code form with the angler brackets such as <head>, <body>, <h3> and so on gets interprets as the browser executes HTML files, which enables browser about the structure and html text. These tags enable the browser to determine the text, images to be displayed as well hidden of other content on web pages [34].

**Opening Web page in a browser:**
Using browser, Tab “Open File...” under the File menu option, navigate to “file_name.html” file. choose it and tab “Open” option.

**Introduction to HTML:**

Physicist Tim Berners-Lee proposed internet based hypertext and stated HTML towards browser software for accessing information from server – client systems in 1990 and collaboration with a system engineer in CERN [35].

The Text form of the web page i.e., program line written in HTML is actually a HTML tags and these HTML files are known as servers which details the specific design of its web page style and display’s accordingly. This code is used mostly towards development of only web pages and is been incessantly evolving from very first day of internet creation to constant development.

**An HTML into of two sections:**

This contains head and body sections which are the entire content of HTML code, hence these are into control and content texts, i.e body contains information to be displayed on webpage and the hidden one towards its programming structure which is head, both encapsulated in html element containing title element enabling the browser to display its text in title bar.

**The <head>**

The <head> element just collects all other header elements in a document. It is the first thing to appear after the opening <html> tag and it must close after the <title> tag.

Each <head> element contains a <title> element indicating the title of the document which display’s in the title bar above the window, although it may also contain any combination of the following elements, in any order[36]:

- <base>, <object>, which is designed to include images, JavaScript objects, Flash animations, MP3 files, QuickTime movies, and other types of files in a page. [37]
<link> to link to an external file, such as a style sheet or JavaScript file [38].

<ruby> to link to an external file, such as a style sheet or JavaScript file [39].

<ruby> for including script in the html document [40].

<meta> which includes information about the document such as keywords and a description, which are particularly helpful for search applications [41].

**The <body> element:**

The <body> tag specifies document's body.

The <body> includes all objects of HTML document as text, images, hyperlinks, lists, tables, etc [42].

The <body> element which mostly carry all the attributes from the attribute group is the key part of the webpage which shows the information on the webpage is usually after the <head> element and referred as body content of the web page.

**Core Elements and Attributes:**

Basic structure of general html file:

<html>, <head>, <title>, and <body>.

These are the common elements on which every webpage is developed using with.

**<html>**

Every HTML file do have a opening element as <html> tag and a closing by </html> tag which indicates the root of its specific file and this is after the optional xml declaration and DOCTYPE declaration[6].

**The <title> Element:**
We should specify a title for each and every page that we write using the `<title>` element. It is presented and used in several ways:

- Title of the content for the page visited is displayed on the top of browser window.
- Title of the page visited is considered as the default name while saving to the favourite as bookmarks, which are same in every browser almost as IE, Safari, Firefox.
- Search engines that use its content to help finding index pages [44].

**Attribute Groups:**

There are three groups of attributes that many of the XHTML elements can carry:

- Core attributes: The class, id, and title attributes.
- Internationalization attributes: The dir, lang, and xml:lang attributes.
- Common scripting events: Attributes associated with events such as onclick, ondoubleclick, onmousedown, onmouseup, onmouseover, onmousemove, onmouseout, onkeypress, onkeydown, and onkeyup [45].

**Tags and Elements:**

- The two angular brackets and all of the characters between them are known as a **tag**.
- There are opening tags and closing tags. The closing tag is always slightly different from the opening tag in that it has a forward slash after the first angled bracket:

```html
</html>
```
- `<html>` and `</html>` relates to opening and closing tag which determines its heading accordingly along with its content it is known as element.

- Tags are of with angles brackets on both sides along with its letters and numbers in between them, while the elements are any content with each tag at opening and closing, hence two tags.
Entire html code is with `<html>` and `</html>` tags in which text embedded in between `<h3>` and `</h3>` is towards heading and content between `<p>` and `</p>` is the paragraphs of the information to be shown in web page.

**Separating Heads from Bodies:**

The entire html code is by `<html>` and `</html>` tags with elements in side these are as below:

1. The `<head>` element: Content which controls the code of the webpage.
2. The `<body>` element: Content which shows in the webpage with its code content again as `<body>`, `</body>` towards opening and closing.

**Adding Hyperlink:**

If wanted the link to open in a new window, add a target attribute to the opening `<a>` tag as well, and give it a value of `_blank`:

```
<a href="http://www.Google.com" target="_blank"> </a>
```

**Different elements we can use to describe the structure of text:**

- The six levels of headings: `<h1>`, `<h2>`, `<h3>`, `<h4>`, `<h5>`, and `<h6>`. [48] [49]
- All three below towards text formatting elements as
  - Paragraph contents as `<p>`, preformatted sections `<pre>`, line breaks `<br>` , and addresses `<address>` [50].
  - Presentational elements: `<b>`, `<i>`, `<u>`, `<s>`, `<tt>`, `<sup>`, `<sub>`, `<strike>`, `<big>`, `<small>`, and `<hr>`.
  - Phrase elements such as `<em>`, `<strong>`, `<abbr>`, `<acronym>`, `<dfn>`, `<blockquote>`, `<q>`, `<cite>`, `<code>`, `<kbd>`, `<var>`, `<samp>`, and `<address>` [51].
• Lists such as ordered, unordered and less common lists using < ul > and < li >, ordered lists using < ol > and < li >, and definition lists using < dl >, < dt >, and < dd > [52].

Basic text formatting elements:

• < h 1 >, < h2 >, < h3 >, < h4 >, < h5 >, < h6 >.

• < p >, < br />, < pre > [50].

The align Attribute:

The align attribute indicates whether the heading appears to the left, centre, or right of the page.

Example:

< h1 align="left" >Left-Aligned Heading < /h1 >

< p > This heading uses the align attribute with a value of left aligned paragraph. < /p >

< h1 align="center" >Centred Heading < /h1 >

< p > This heading uses the align attribute with a value of center aligned paragraph. < /p >

< h1 align="right" >Right-Aligned Heading < /h1 >

< p > This heading uses the align attribute with a value of right aligned paragraph. < /p > [53].

Creating Paragraphs Using the < p > Element:

The < p > element offers another way to structure the text. Each paragraph of text should be in between an opening < p > and closing < /p > tag.

Creating Line Breaks Using the <br/> Element:

• Using the < br /> element to display contents for stating in the next line. This is an example of empty element; which don’t need to be a open and close tags, as there’s nothing additional required to place in between them [54].

Lists
Used to display our text in some order.

we can create three types of lists in HTML:

1. **Unordered lists**, which are like lists of bullet points, or some other unordered points.
2. **Ordered lists**, which use a sequence of numbers, letters or roman letters instead of bullet points
3. **Definition lists**, which allow to specify a term and its definition.[44]

**Using the `<ul>` Element to Create Unordered Lists**

If we want to make a list of bullet points, we must write the list within the `<ul>` element here `ul` stands for unordered list. Each bullet point or line we want to write should then be contained between opening `<li>` tag and closing `</li>` tag here `li` stands for *list item*.

If we want to create a bulleted list, we can do so like this.

```html
<ul>
  <li> Bullet point number one </li>
  <li> Bullet point number two </li>
</ul>
```

Displays bullet list for above.

**Ordered Lists:**

If we want list the lines to be ordered. In an ordered list, we can use either numbers (1, 2, 3 ...), letters (A, B, C ...), or Roman numerals (i, ii, iii ...) to prefix the list item.

An ordered list is contained inside the `<ol>` element here `ol` refers to the ordered list. Each item in the list should then be entered inside the `<ol>` element and contained between opening `<li>` and closing `</li>` tags [44].

**Definition Lists:**
It is a special kind of list for providing terms followed by a short text definition or description for them. Definition lists are contained inside the <dl> element. The <dl> element then contains alternating <dt> and <dd> elements. The content of the <dt> element is the term you will be defining. The <dd> element contains the definition of the previous <dt> element [55].

**Editing Text:**

There are two elements specifically designed for revising and editing text:

- The <ins> element, which is used to add text in middle of the words/characters (usually shown underlined in a browser).
- The <del> element, which is used to delete some text (usually shown crossed out in a browser) [56].

**Basic Links:**

A link specified using <a> element, hence any content between this <a>, </a> tags is part of the entire content of the html which is towards webpage for browsing.

**Linking to Other Web Pages**

Linking a webpage under a main web page is by connecting the opening tag <a> and carry an attribute called href; hence value of href attribute is name of the file linking to.

Example the <body> of the page.html, which is in the code. This page contains a link to a second page called index.html:

```html
<body>
<p>Return to the <a href="index.html" >home page</a>.</p>
</body>
```

As long as index.html is in the same folder as page.html, and executes “home page,” the index.html page will be loaded into the same window, replacing the current page.html page [57].
**Linking to E-mail Addresses:**

To create a link to an e-mail address, we need to use the following syntax with the `<a>` element:

```html
<a href="mailto:name@example.com">name@example.com</a>
```

**Adding Images Using the `<img>` Element:**

Images are added to a site using the `<img>` element, which has to carry at least two attributes: the `src` attribute, indicating the source of the image, and an `alt` attribute, which provides a description of the image.

For example, the following line would add the image called logo.gif into the page, the image lives in a directory called images.

```html
<a href="http://www.google.com" target="blank">
<img src="google.gif"/>
</a>
```

**The `src` Attribute:**

The `src` attribute determines browser to find image and it determined only with URL, which is similar to linking another webpage and this URL can either a absolute URL or a relative URL.

```
<img src="image.gif"/>
```

**The `alt` Attribute:**

This value is always determines only in form of text description of the image and appears on each and every `<img>` element.

```
<img src="image.gif" alt="image"/>
```

Referred to as alt text since it is important as value of this attribute really describes image, this is so in order to display at least text if unable to load the image.
**Height and Width Attributes:**

The values of these attributes are shown in pixels which specify its further details as below along with choosing right image format.

```
< img src="sunset.gif" width=160 height=110 alt="image"/> [62].
```

**Using Images as a Links:**

Image link is simply embedded inside the tags and specify its links, and this is much easier to access rather than placing text between `<a> </a>`. These are used mostly towards graphical buttons or links to other pages as below

```
<a href="www.google.com">
<img src="google.gif" alt="google" width="32" height="32"></a></p> [63].
```

**Tables:**

**Basic Table Elements and Attributes:**

The `<table>` Element Creates a Table these defines a a block of data [64].

The `<table>` element defines

**The `<tr>` Element:**

The `<tr>` element is used to contain data in each new row in a table. Anything appearing within a `<tr>` element should appear on the same row and this is usually a child element of the table [64].

**The `<td>` and `<th>` Elements:**

Every cell in a table is represented either with `<td>` or `<th>` elements for cells containing table data or table headings [64].

**Adding a `<caption>` to a Table:**
The `<caption>` element provides a visual caption for the table

`<caption> Opening hours </caption>`

`<tr> [64].

**Forms:**

**Creating a Form with the `<form>` Element**

Every `<form>` element should carry at least two attributes:

- **action method**
  
  A `<form>` element may also carry all of the universal attributes, the User Interface (UI) event attributes, and the following attributes:

  `enctype accept accept-charset onsubmit, onreset.`

**Text Inputs:**

There are actually three types of text input used on forms they are as follows:

- **Single - line text input controls.**
- **Password input controls.**
- **Multi - line text input controls.**

**Single Line Text Input Controls**

Single line text input controls the text that are created using an `<input>` element whose `type` attribute has a value of `text` it can able to display a single line only not even more than that.

`<input type="text" name="txtSearch" value="Search for" size="20" maxlength="64" />

**Password Input Controls**
If we want to collect sensitive data such as passwords and credit card information, we can use the password input. This password input masks the characters of the user types on the screen by replacing them with either a dot (.) or asterisk (*).

Password:

```html
< input type="password" name="pwdPassword" value="" size="20" maxlength="20" />
```

**Multiple-Line Text Input Controls**

If we want to allow a visitor to a site to enter more than one line of text, we should provide a multiple-line text input control using `<textarea>` element.

```html
< textarea name="txtFeedback" rows="10" cols="50">
```

**Buttons**

Buttons are commonly used to submit a form, even though they are sometimes used to clear or reset a form and even to trigger a client-side scripts.

**Creating Buttons Using the `<input>` Element**

When we use the `<input>` element to create a button, the type of button that we create is specified using the `type` attribute of the element.

```html
< input type="submit" name="btnVoteRed" value="Vote for reds" />
```

**Using Images for Buttons**

We can use an image for a button rather than using the standard button that are provided by a browser renders. Creating an image button is similar to creating any other button, but the type of attribute has a value of image:

```html
< input type="image" src="download.jpg" alt="download" name="btnImage" />
```

**Checkboxes:**
They are just like the little boxes on paper forms in which we can place a cross or tick on the paper without any limitations. They can be either on or off. When they are checked they are on the user can simply toggle between on and off positions by clicking the particular box called checkbox.

<input type="checkbox" name="Skills" value="xhtml" />X HTML <br />

**Radio Buttons**

They are similar to the checkboxes in that they can be either on or off, but there are two key differences as follows:

- When we have a group of radio buttons in that group only one of them can be selected. Once one of the radio button has been selected, if the user clicks another option, the new option will be selected whereas the old one is deselected.
- We should not use the radio buttons for a single form control where the control indicates on or off, because once a lone radio button has been selected it cannot be deselected again.

<input type="radio" name="Class" value="First" />First class <br />

**Select Boxes**

A drop down select box allows the users to select one option from a drop down menu. Drop-down select boxes can take up far less space than a group of radio buttons.

<select name="setColor">
  <option selected="selected" value="" > Select color </option>
  <option value="red" > Black </option>
  <option value="green" > Aqua </option>
  <option value="blue" > Brown </option>
</select>
Sending Form Data to the Server:

As the browser requests the server for data content of the specific WebPages, which in turn server responds to browser of the client system and forwards the data packets, thus one of this process can be done by using hyper text transfer protocol (HTTP). This process is client and server data request response protocol.

There are two methods for transreceives of the data client server request response protocol is by for HTTP get and HTTP post. [67].

**HTTP get**

When we send form data to the server using HTTP get method, this form data is appended to the URL that are specified in the action attribute of the `<form>` element [52].

For example, take the following login form, which we saw before when the password form control was introduced:

```html
<form action="http://www.example.com/login.aspx" method="get">
    Username:  <input type="text" name="txtUsername" value="" size="20" maxlength="20" />
    Password: <input type="password" name="pwdPassword" value="" size="20" maxlength="20">
    <input type="submit" />
</form>.
```

**HTTP post**

When we send data from a form to the server using the HTTP post method, this form data are sent transparently through the HTTP headers. Meanwhile we are not able to see these headers; they are not, strictly speaking because they are secure on their own. If we are sending sensitive
information such as Bank Account details, the data must be sent under a Secure Sockets Layer, or SSL, and they should be in encrypted because the data should be kept securely [67].

Example:

User-agent: MSIE 7
Content-Type: application/x-www-form-urlencoded Content-length: 35

...other headers go here...

txtUserName=Bob & pwdPassword=LetMeIn

<form_action="http://www.example.com/newsletter.asp?action=subscribe" method="post">

Frames:

Frames are the one which divides a browser window into two or more separate parts or panes, with each pane containing a separate web page.

The <frame> Element

The <frame> element should indicate what is going to happen in each frame of the frameset. It is always an empty element and therefore it should not have any content between them, although each <frame> element should always carry one attribute which is called “src” to indicate the page that should be represent in that particular frame [68].

The <noframes> Element

If a browser does not support frames then the contents of the <noframes> element should be displayed to the user instead of unsupported frames.

<noframes>

<body>This site requires frames.</body>

</noframes>[68].

Creating Links Between the Frames:
To link different frames following code is used

```html
<frameset cols="200, *">
  <frame src="frames/linksNav.html"/>
  <frame src="frames/linksMain.html" name="main_page"/>
</frameset> [69].
```
1.4 RFID/NFC

**RFID Systems:**

RFID stands for Radio Frequency Identification which transfers the digital information by means of electromagnetic radio waves and use of these waves at NFC is a unlicensed band [71]. These RFID tags are as similar to smart cards where the data is stored either in magnetic or semiconductor devices powered by radio waves whose charge is stored in a small coil as capacitance which is embedded in it for every passive chip, further more these RFID tag transponds and starts tranreceiving the digital data as it gets into a galvanic contact with the RFID reader which is operating at same frequency as of the tag, this technique by its intelligent operation is adopted from fields of radio and RADAR engineering [72].

**Modules:**

These systems are mainly made up with two types as below:

- A Transceiver block embedded in the tag.
- A Transceiver block very mostly active model dedicated as a reader to detect and read all the tags.

Reader consisting a RF transmitter and receiver along with a control unit interfaced the central controlling system with a database connected to it which could be by RS232, USB, SPI, I2C interfaces for transreceives of the data from computer/embedded system to the RFID reader and there on galvanic interface with tag [72].

The RFID tag is again another transrecive module mostly passive devices running at low power with a low data rate transfers where in which the devices power on by means of receiving the
electromagnetic power from the radio waves which are usually transmitted by RFID reader [72] refer figure – 6.

![Diagram](image)

**figure – 6**

These systems are developed by numerous varieties of manufacturers with usage in wide applications and operate in two modes, full duplex, half duplex communication systems.

RFID systems run right from KHz range to GHz band which entirely depends on the technology sophistication and intended usage, however the working functionality remains same irrespective of the band it is used and only matters as active or passive and communication type used, on the other aspect if it is a active tag its transmitter distance can be much longer than the passive tag and depends on the transmit power for the distance range, along with this it also depends on the communication speed [73]. Further more security specifications are really much under development stage and it is more or less vulnerable for breaching unless it is not into much high level encrypting for a precision usage, along with all these the regular rfid tags to offer memory of much low level and it is approx 10 bit or 12 bit capacity for general and depends on products sophistication and technology for accessing higher capacities [73].
**NFC:**

NFC is propinquity based communication system used for low rate data communication while consuming ultra low power of performance usage which is considered the one of the next generation communication system slightly alternative to Bluetooth despite it usage from electronic communication systems in world war times [70]. This Device is slightly similar to barcode reader but with its unique distinguish on Radio Waves instead of optical imaging processing. The chip currently used in our project is based on 13.56MHz signal operation at 424kbit/s with 0.1 seconds towards setup time, despite the available bands of the RFID this 13.56MHz signal is choose accordingly based on the reader’s availability option [71], this is so only on the basis of its operational usage as environmental favors like high frequency signals at metal structures, low frequency signals for basic structures and access, however in this project a passive tag is choose such that to take advantage of battery free operation which is especially chosen for short distance communication for low data rates.

**NFC application with RFID**

NFC communication protocol operate at band of 13.56MHz as said above using the radio waves which transmit at this band out of wide band of its operations. Further more in details RFID is towards identification of a tag, operate at any given frequency band ex: 1 KHz to 1GHz, where as NFC operate at 13.56MHz which lies in the said 1KHz to 1GHz.

Well as of the RFID reader it can operate at any several multiple frequencies simultaneously of which it is completely a dedicated embedded system, where as the NFC module is embedded in mobile devices/tablets and variety of computing devices which are multi
applications based intended to read and write the tags with its module embedded into Android device as example.
1.5  MY SQL

**Databases:**

Two types of databases are now in real world they are,

i. Flat databases

ii. Relational databases

A relational databases stores data as a text files in a table and acts like a human brain to operate operations relationally [74].

**MySQL Database:**

My SQL, source not restricted (open) RDBMS running as server provides multiuser access towards huge database as storage under GNU terms [75].

It is a powerful database package developed much towards offering the open source for every individual to access along with handling better performance. Further it is much closer functionality towards PHP, it do support large databases upto 50 million rows or higher in tables along with a theoretical limit with extension upto 8 Million TB. [76].

**MySQL Versions:**

<table>
<thead>
<tr>
<th>Versions</th>
<th>Developed by</th>
<th>Year and month</th>
</tr>
</thead>
<tbody>
<tr>
<td>first version</td>
<td>MySQLAB</td>
<td>On 23(^{rd}) may 1995</td>
</tr>
</tbody>
</table>
MySQL AB was purchased by Sun Microsystems which in turn purchased by Oracle in 2008 and 27th Jan 2010 respectively [75].

The MySQL interface
In MySQL using of interface is simple and easy.

Entering MySQL interface following code may be used:

```
unixprompt> mysql -u wbyeats -p
```

Enter password:

```
mysql>
```

The “-u” tag = the username, and
the “-p” tag = the password prompt [77].

**Basic MySQL administration tools**

**Users, passwords, and privileges**

Created database doesn’t contain any user id and passwords, hence towards creation

```
mysql> GRANT USAGE ON tissueinfo.* to wbyeats@localhost
IDENTIFIED BY 'ode2maud';
```

user_name as “wbyeats”, password as “ode2maud”.

In case of granting limited access towards user for viewing data, add data in a table herewith below as

```
mysql> GRANT SELECT, INSERT ON tissueinfo.* to wbyeats@localhost
IDENTIFIED BY 'ode2maud';
```

In case of full administration rights herewith as follows

```
mysql> GRANT ALL ON tissueinfo.* to wbyeats@localhost
```
IDENTIFIED BY 'ode2maud';

In case of storing the login info during interface access, `.my.cnf` in home directory saving file can lead to cache particular user login info, example UNIX – SQL.

```
[client]
user=user name
host=localhost
password=password [77].
```

**Data types used in creation of database:**

<table>
<thead>
<tr>
<th>Data types</th>
<th>Textual description</th>
</tr>
</thead>
<tbody>
<tr>
<td>INT</td>
<td>Integer</td>
</tr>
<tr>
<td>FLOAT</td>
<td>Small floating point number</td>
</tr>
<tr>
<td>DOUBLE</td>
<td>Double precision floating point number</td>
</tr>
<tr>
<td>CHAR(N)</td>
<td>N characters long (N=1..255)</td>
</tr>
<tr>
<td>VARCHAR(N)</td>
<td>Variable length text up to N characters long</td>
</tr>
<tr>
<td>TEXT</td>
<td>up to 65535 characters long</td>
</tr>
<tr>
<td>LONGTEXT</td>
<td>to 4294967295 characters long</td>
</tr>
</tbody>
</table>

[77]
MySQL operations as:

<table>
<thead>
<tr>
<th>Operation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Execute MySQL Script</td>
<td>SQL script executed in MySQL database.</td>
</tr>
<tr>
<td>MySQL BinLog</td>
<td>View binary log and its conversion to text.</td>
</tr>
<tr>
<td>MySQL Check</td>
<td>Analysis and optimization for specified tables in MySQL database.</td>
</tr>
<tr>
<td>MySQL Dump</td>
<td>Backs up specified MySQL database tables.</td>
</tr>
<tr>
<td>MySQL Import</td>
<td>Importing data into MySQL database table from a text file.</td>
</tr>
</tbody>
</table>

Below with details performing the operation as

1. Create a database
2. Create a table
3. Load data into the table
4. Retrieve data from the table in various ways

1) Creating and Selecting a Database:

Creating database where menagerie uses current database is with following code

```sql
mysql> CREATE DATABASE menagerie;
```

2) Create a table

Below with is used to specify layout of a table:

```sql
mysql> CREATE TABLE table name (string VARCHAR(20), owner VARCHAR(20),
--> Species VARCHAR(20), sex CHAR(1), birth DATE, death DATE);
```
3) **Loading Data into a Table**

Below with is used to load the data in a table:

```sql
mysql> LOAD DATA LOCAL INFILE '/path/file.txt' INTO TABLE table_name;
mysql> INSERT INTO table_name
-> VALUES ('Puffball','Diane','hamster','f','1999-03-30',NULL);
```

*Note:* MySQL uses date format as 'YYYY-MM-DD', Insert statement is optimal when compare to LOAD DATA.

4) **Retrieving Information from a Table:**

For retrieving data from a table SELECT statement is used.

```sql
SELECT what_to_select
FROM which_table
WHERE conditions_to_satisfy; [79].
```

**MySQL so Popular:**

1. Size and speed.
2. Ease of installation.
3. Attention to standards.
4. Responsiveness to community.
5. Easy interface to other software [80]

**MySQL’s Logical Architecture:**
The layer of this architecture as services topmost, Query parsing, Optimization, Caching and all other inbuilt functions, third towards storage engines, Logical Architecture as in figure – 7.

![Logical Architecture Diagram](image)

**figure – 7**

**Four aspects of using MySQL:**

1. **MySQL server:**
   
   Servers dealing installation and configuration of the database server.

2. **SQL:**
   
   Developing database.

3. **Programming languages:**
   
   Automatically creation and execution in MySQL server programming languages such as PHP or PEARL are used.

4. **Web database applications:**
Creating dynamical web applications towards publishing the information content on the internet provided by users [80].

**Single Table SELECTs**

To read the local data from all rows and columns in a table SELECT statement is used.

Start the system and choose the database:

```sql
mysql> use database_name;
```

Database changed

To retrieve all of the data in the table:

```sql
mysql> SELECT * FROM table name;
```

A simple SELECT statement has four components:

1. **SELECT** keyword.
2. **`*`** (asterisk) symbol is used to display all columns in a relation.
3. **FROM** keyword.
4. The table name [80].

**Choosing Columns:**

For example, if we want only a single column in a table, we can use:

```sql
mysql> SELECT column name FROM table name;
```

To retrieve more than one column from a relation in such cases we need to use following piece of code:

```sql
mysql> SELECT column name1, column name2 FROM table name;
```

To display data from the table in a another database without changing the database

```sql
mysql> SELECT column FROM database.table;
```
The database.table component after the FROM keyword specifies the database name and table [80].

**WHERE Clauses:**

If required to filter the results from database we can use a conditional clause as WHERE clause. Following syntax may be used:

SELECT field1, field2,...fieldN table_name1, , table_name2...

[WHERE condition1 [AND][OR]] condition2....

WHERE Clause is not a mandatory part of a SELECT statement by using this it can filter data, by help of some condition and is also valid when it uses DELETE or UPDATE commands.

It is similar to the “if” condition in a programming language.

**Operators used in the WHERE Clause:**

=      Equals.
!=     Not Equals.
<      Less Than.
<      Greater Than.
>=     Less Than or Equals.
<=     Greater Than or Equals [81].

**Combining conditions OR, AND, XOR and NOT:**

Boolean operators like AND, OR, NOT and XOR for conflating two or more conditions in a table. The following example explains that if displays the columns title greater than H but less than M.

```sql
mysql> SELECT columns FROM album WHERE column name > "H" AND column name < "M";
```
The AND operation is invalid in the WHERE clause in which it meets several conditions. While OR operator used to satisfies at least one of several conditions.

To list all columns that have a title beginning with M, T, or F. We can use AND and OR in condition.

```sql
mysql> SELECT columns FROM table WHERE
  -> Column LIKE "M%" OR
  -> Column LIKE "T%" AND
  -> Column LIKE "%F";
```

If we want to list all columns list, columns except 1 or 3 then following code may be used however unary NOT negates Boolean statement.

```sql
mysql> SELECT * FROM table WHERE NOT (column = 1 OR column= 3);
```

By using ORDER BY Clauses we can return the values in some particular order that we specifies.

Consider following code for the example of ORDER BY Clauses

```sql
mysql> SELECT * FROM table ORDER BY attribute; [80].
```

**The LIMIT Clause:**

In a nonstandard SQL tool rows are onput that allows to control and limit clause is used to limit about some specified condition.

```sql
mysql>SELECT columns FROM table LIMIT 10;
```

The LIMIT clause in this example restricts the output to the first 10 rows [80].

**Joining Two Tables:**

To join two or more tables we will use join statement to join tables here we use inner join for joining tables using some specific condition.
mysql> SELECT column1,column2 FROM table INNER JOIN album -> USING (condition); [80].

The INSERT Statement:
To add new values to a table INSERT statement must be used

mysql>INSERT INTO table VALUES((SELECT 1+MAX(column) FROM table), "attribute");

This INSERT function do operate stop on first duplicate key. It’s optional to add IGNORE clause towards error prevention [80].

The DELETE Statement:
For removing one or more rows in a table this statement can be used.

mysql>DELETE row_name FROM table;

This code results in removing the row name in a relation [80].

Removing All Rows with TRUNCATE:
Generally DELETE statement is used to remove rows in a table; however this statement removes only a single entry in table. Hence to remove the whole structure of a table and re-creating the table DELETE statement is complex, in order to overcome this problem a simple and easy shortcut is used which is TRUNCATE statement.

Example:

mysql>TRUNCATE TABLE table_name;

This statement drops the structure of the table [80].

The UPDATE Statement:
This statement is used to update (Alter) the row of a single table, for this UPDATE keyword is used.
mysql> UPDATE artist SET artist_name = UPPER(artist_name);

The above code changes the artist name to a uppercase letters from a relation artist [80].

**Indexing for High Performance:**

Finding rows quickly in table, a special data structures such as indexes are used which are called as keys.

**Types**

- B-Tree indexes.
- Hash indexes.
- Spatial (R-Tree) indexes.
- Full-text indexes.
- Other types of index [80].

**Benefits of Indexes**

1. Effort and time saving.
2. Avoids searching and sorting problem in table.
3. Random I/O converted into sequential I/O leads slow processing [80].

**Features of MySQL:**

A. Version Control Integration
B. Macro Record and Playback
C. Database Browser.
D. Code Snippet Editor
E. Security Manager
F. SQL Editor
G. Fast, Multi-Tabbed Schema Browser

H. Import/Export Utility

I. DB Extract, Compare-and-Search Utility [82].

**Top 10 tools for MySQL development and administration of sandbox**

1) **Workbench**

The standard edition of this work bench can improve and adds additional features of the Community Edition (free licence)

- Option of representing database model in graphical form with editing option in table.
- Recovery options towards table structure and relationships from earlier one to current stores one in the database server.

- SQL Query editor and several other functions [83].

2) **Navicat**

- Application tool towards administration and development of databases running ofr any server of MySQL.

- Visual Query designer
- SSH and HTTP tunnel
- Importing, exporting and backup of data
- Migration and synchronization of data and structures[83].

3) **PHPMyAdmin**

- This tool is used by server administrator, designed for accessing web-based interface which can run commands and view content inside the web through browser [83].

4) **dbForge Studio for MySQL**
This tool is for developers to automate routine tasks and save time, which is available in three editions as a). Express, b). Standard, c). Professional

Visual profiler queries, administering user privileges, database designer towards visual diagrams [83].

5) HeidiSQL

A database management tool for managing data and an alternative for PhpMyAdmin, with feature as create, view, triggers, edit tables, procedures, and editing its data [83].

Connecting the server via command line.

Formatting disordered SQL.

Batch optimization and recovery tables.

Synchronizing tables between various databases and other key features.

6) SQL Maestro for MySQL

This tool uses the “user friendly graphical interface” for backup, managing access and mainly used for management development and administration

Database designer availability

Visual Query

Editing, grouping, filtering and sorting

SSH and HTTP tunnel [83].

7) EMS SQL Manager for MySQL

Tool towards database, administration and development supporting various functions with support of all MySQL previous versions. Enabling visually edit, import and export of database, manage user privileges, run scripts SQL and visually design databases MySQL

SSH and HTTP tunnel
Data support UTF8
Text and graphical tools to query [83].

8) **SQLyog**

SQLyog tool combines the capabilities of MySQL Administrator, PhpMyAdmin and other tools towards its administration, database development. Compatible with almost every operating system
HTTP, HTTPS, SSH tunnel
Unicode support
Synchronizing data [83].

9) **DBTools Manager**

A application data management tools with integrated tool support various data bases as MySQL, PostgreSQL, MSSQL server, MS Access.
Wizards for creation of form ad reports
Designer diagrams and more
Query editor [83].

10) **MyDB Studio**

Application tool towards create, edit, delete records of spreadsheets and databases fot managing MySQL with compatibility for windows OS.
SSH channel
Connection towards limitless databases
Transfer, export, restore, backup of databases and other key features [83].
2 PROJECT DESCRIPTION

This developed project is a careful study on the latest development of RFID functioning with Android application programming along with accessing the latitude longitude information captured from cellular / internet networks rather than GPS module which saves power consumption, more over this application is developed to enable the android devices and to install the designed application for casting the vote from a nomadic device through either a cellular network or by a wifi connected to the internet world, in this scenario the developments are in four stages, 1\textsuperscript{st} on the android application development which awaits the input as username, password and a next level security as RFID tag, 2\textsuperscript{nd} on the web hosting which provides the access to the internet world for TX and RX of the data packets from and to My SQL data base as well as the android app, 3\textsuperscript{rd} on My SQL database to link the centralized data to the internet world with the web hosting and to android app, this means the data of the website content, well on the final fourth stage complete integration of all three packages into a single full-fledged platform to productive usage to meet the requirements for achieving the things in a easier form, refer figure – 8 for graphical understanding.
In details, the user logs in to the mobile voting app in the android device with username, password and rfid tag, here on the android devices reads the username, password, rfid details and sends it to the My SQL database using cellular / wifi internet networks, this is where it confirms the data on the both sides from android as well as My SQL database and there on transreceives the voting information and proceeds further in updating preciousely, on the other end for system administration a dedicated web hosting is developed and used after interfacing with My SQL database where the administrator can access the up to date necessary information from database. Well in the administration level, the web hosting is of two modes, with a difference in only limitation of information access levels for security concerns. This application updates preciousely and avoids double voting or so on, maintains the vote confirmation and casting level to avoid any errors while experiencing the cellular / internet signals loss, confirmation of user input details,
rfid additional security level, also provides a option to let the administrator know the exact geographical location where the voting is done, and so on additional features.

And precious advantage is of capturing the lat long information from the phone which doesn’t have GPS module in built, this lat long is used for security concern.
3 APPLICATION DESIGN AND BLOCK DIAGRAM

figure – 9 Final product block diagram

figure – 10 Passive RFID Block
figure – 11 Android Core Blocks

figure – 12 Android Blocks
4.1 ANDROID APP PORTING / INSTALLATION

The procedure of installing the customized designed app in two ways, downloading from Google play with which it is previously registered and confirmed the security and various other terms with Google. Other way is simply porting the app from a computer to the device working on android platform. However when concerned with the operation and device compatibility this needs the requirement of android OS, when it is concerned with the device models of the smart phones it needs to have a access to RF reader module with the phone with which It can load the designed database and connects to the web server only after passing through the rfid unique pass code.

Well for the application downloading through Google play it is much easier and it is a procedure of opening the file in Google play and tabbing the download option. However on the other aspect for porting the application from computer/laptop to the android device for the applications which are usually not published in the Google play database their follows certain procedure for installing in android device, herewith as follows with a requirement of android SDK software along with proper installation of its USB drivers towards android device. [84]

1st install the USB drivers by connect android device to the computer with USB cable wherein which windows do detect the new device and launch plug and play for procedure towards hardware wizard, hence follow this in installing

2nd provide the path for the software package and install its drivers in this wizard

3rd open the browse option, provide the folder path and there on give load the APK file from the computer and then port it to the android device.
4th if it asks for any administrative option to install or for interface device procedure please install the same as directed with which it remembers the device and the standard procedure for this specific device which makes easier to install the APK every time.
4.2 RFID METHODOLOGY / OPERATION IN THIS APPLICATION

The designed application uses NFC module of the android device to read the RFID tag, capture its data and there on proceed the same to the android application which transmits the data to the MySQL database where which it verifies the genuinity and sends acknowledgment which allows the android application to either to login successfully or to indicate login error message based on the acknowledgment from MySQL. The RFID tag used for this application comprises of unique 10bit data which is allocated to the tag by the administrator along with a match to the username password with the database of the entire application system. Herewith as an example of reading the information from RFID tag and capturing its content is as figure – 13 for a successful login of with correct username, password along with RFID.

figure – 13
4.3 VOTING PROCESS / RESULTS

The designed application does support for Android OS 4.2 version, this means it can support not just for mobile phones, but also for tablets and other computing devices which runs on Android OS. The designed application is developed using various software’s for the complete integrated product application usage. This application involves casting the vote from the application installed in the android OS and sends it casted details to the centralized database and there on the same database is interfaced with web hosting for viewing the results and these are two stages with a difference in limitation of accessibility.

The process of casting the vote from a android device is as follows,

1\textsuperscript{st} install the Mobile voting application in the android device, this application is provided as attached CD (1) which contains all the software and coding details.

2\textsuperscript{nd} open the application in the android device as figure – 14.

3\textsuperscript{rd} enter the username, password as figure – 15.

4\textsuperscript{th} figure – 16 shows the username and password as entered.

4\textsuperscript{th} figure – 16 shows the username and password as entered.
5th Tab the RFID of the logged in user, once it is verified with moves further for login, the application shows as figure – 17 which indicates as authentication accepted, once after logging in successfully it also shows the login successful information as in figure – 18.

6th Once the login is successful along with authentication, the application directs to the voting options for question 1. (This procedure explains only for question 1 / option 1, since the procedure is same for the rest questions/options), as in figure – 18.

7th After successful login it directs to the main screen which provides the various voting options/questions, as in figure – 19.

8th once selecting the question/opti on 1 from the procedure 7th as from figure – 19, it provides the question 1 and its voting options as in figure – 20.

9th it allows various options for nominating its answer for the question as in figure – 20.

10th selection is made from the options provided for the question 1 as in figure – 21.

11th once after confirming the selection from procedure 10th as in figure – 21 it directs to next level as in figure – 22 which freezes the option.
12\textsuperscript{th} once after locking/freezing the vote option, it asks for confirmation along with option yes to move further or no to repeat the selection process, hence when it is moving ahead by tabbing yes it confirms the nominated vote option as in figure – 23 and updates the database figure – 24 which says vote submitted.

13\textsuperscript{th} during the entire process if there is any cellular / internet signal error it directs to the android devices signal connection options as in figure – 25.
14th If the NFC module for RFID detection and data accessing is not turned on, the mobile voting application automatically directs to the NFC module option in the android device as in figure – 26 and turning on the NFC module as in figure – 27 for casting vote.

![figure – 25](image1)
![figure – 26](image2)
![figure – 27](image3)

14th In the entire voting process if the login information or the RFID data accessing had any problem it directs as figure – 28 which shows please authenticate.

![figure – 28](image4)
![figure – 29](image5)
![figure – 30](image6)

15th In any aspect of double voting or so on it directs to as in figure – 29 which says submitted vote, this is only after confirming with database whether the vote is utilized or not.
16\textsuperscript{th} well on the aspect of results from the user/voter side the designed application shows submitted vote if it is submitted successfully as in figure – 29.

17\textsuperscript{th} In any aspect if the username / password results in login error it directs a error message as wrong username/password as in figure – 30.
4.4 WEBHOSTING

The Website is developed with HTML using PHP script after registering for a member with domain [http://www.000webhost.com/](http://www.000webhost.com/). The entire webpage is created on basis of the above said to provide relevant user name info and its corresponding details, however as the webpage application is concerned there are two modes as login, one relates to the system level which contains the basic information of the nominee details along with total voted results and the other one administrative level which details the information of all user names, passwords, voting details like option chosen for the nominee along with the voting time and location, this is only in terms of security aspects for limitation of accessibility. Here with below are the two modes of the its information accessibility limitation levels as below figures – 31 and 32 and these WebPages can be accessed by logging to the URL [http://votingappresult.net23.net/](http://votingappresult.net23.net/) figure – 33 and there on providing the user name and password, well the username and password leads to its appropriate login either system level or administration level, This is where the entire mobile voting system controls are along with the display of full-fledged results of system.

Refer html webhosting source coding for more details towards its development as in CD
### Administrator

<table>
<thead>
<tr>
<th>ID Number</th>
<th>Username</th>
<th>Password</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>stephen</td>
<td>vote</td>
</tr>
<tr>
<td>2</td>
<td>dan</td>
<td>interesting</td>
</tr>
<tr>
<td>3</td>
<td>timon</td>
<td>1985</td>
</tr>
<tr>
<td>4</td>
<td>raj</td>
<td>jef</td>
</tr>
<tr>
<td>5</td>
<td>balb</td>
<td>1214</td>
</tr>
<tr>
<td>6</td>
<td>amn</td>
<td>4321</td>
</tr>
<tr>
<td>7</td>
<td>amn</td>
<td>amn</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Question</th>
<th>Username</th>
<th>Answer</th>
<th>Latitude</th>
<th>Longitude</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>amn</td>
<td>1</td>
<td>32823566</td>
<td>-117144380</td>
<td>2013-03-12-22:33:30</td>
</tr>
<tr>
<td>2</td>
<td>raj</td>
<td>3</td>
<td>32894117</td>
<td>-117228325</td>
<td>2013-02-05-09:02:11</td>
</tr>
</tbody>
</table>

### System Engineer

<table>
<thead>
<tr>
<th>Question</th>
<th>Option 1</th>
<th>Option 2</th>
<th>Option 3</th>
<th>Option 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Question 1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Question 2</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Question 3</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>Question 4</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>Question 5</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Mobile Voting System

Welcome To Mobile V

figure – 33
4.5 DATABASE MY SQL

The database is preciously developed with MySql platform with a GUI programming which is a advanced 4th generation programming avoiding absolute human errors in the development scenario in the terms of the server – database – webpage integration, hence the database is created with 7 core stacks of which each contains users/administrator login information relating to its unique name / password / rfid serial no information and its relevant ID, Voting options each containing its relevant sub-options as choice, the voter’s chosen option details (sub-options) along with the time and geolocation of the vote option utilized. This database is operated / managed using the PhPMyAdmin with registration as a member from the domain http://www.000webhost.com/ towards phpMyAdmin Access. This developed database is towards providing the voting options questions/answers results to the android devices and to display its content in the website, hence it is in general the mediate interface between the android device and website. Furthermore it is towards development and administration level only and not for the front end user availability.

Refer My SQL data source coding for more details towards its development as in CD
5 RESULTS

The entire system results are at two level of which one is on the android device level as figure – 34 which just provides information if it the vote is used and submitted to the centralized database for counting and its effective utilization. On the other side, next one is at website level for administration and this is again into two modes as system engineer as figure – 35 and administrator level figure – 36 with a difference in limitation of information accessibility. The administrative level result do include the lat long information which can be plotted in Google maps for further information on voted location as figure – 37. This feature is also accessible for a device which really doesn’t use the internal GPS module.

figure – 34

figure – 35

figure – 36.
figure – 37
6 FUTURE DEVELOPMENTS

Despite the surmount of designed application performance there still need to develop a strong encrypting / decryption system used for paging the packets right from the mobile station there on to cellular network and to the database and webhosting / server locations with vice versa along RFID crypting techniques, however it can really be a predominately utilized if the designed application is available in wide platforms which leads to proper usage of the application throughout the world. More over the development of Doppler shift which supporting android devices for capturing lat long do provide best accuracy for user voted location.
CONCLUSION

The proposed system can only be successfully implemented based on the law and political society with context to the governing authority along with the technology aspects, and is well interesting technology product which reduces efforts of voting physically, time consideration, leading to easier online with much less time to utilize the vote hence this voting system using mobile phones / tablets based on android operating system, thus leading to a powerful outcome of entire voting system which rules extremely unique when meets the standard of art and technology in facet to security concerned issues. Furthermore the RFID tags thusly provide certain level of unique detection preventing dupery when its security system sounds advantageously.
This entire application is developed using various softwares of which webhosting is done by purchasing an account with [www.000webhost.com](http://www.000webhost.com) and there on programming using php scripts towards html website.

Another part database using My SQL query and its is developed by graphical programming which is connected in between android and webhosting.

Open the website

[www.000webhost.com](http://www.000webhost.com)

Open below URL

http://members.000webhost.com/login.php

Enter username as mccluredanr@gmail.com

Password 1forgotit.

After successful login it directs to below URL

http://members.000webhost.com/index.php?login_hash=wuP8imDc5MgGnTwB

>>Here for viewing the website, open the votingappresult.net23.net which is under Domain

>>For programming and controlling the domain click the Go to CPanel option which is under action, this directs to the main screen of the control options

>>Click Files option and click the file manager, this directs to


enter the user name as a8586955

password as 1forgotit

The control version of source coding


>>Click the public.html which contains all the webhosting source coding files as figure – 38

This contains as below
Web Hosting Source Coding files figure – 38

Here files with form.css, index.html, login.php, styles.css are the source code files towards webhosting files. (Refer these files in source code)

The files with backhome.png, mobilevotingsyslogonew1.jpg, logouticon.jpg are the image files used in the webhosting. (Refer these as figure – 39, 40 and 41.

Folder with name android is the directory which contains all the linking files as scripts for connecting android and web hosting. This contains source code scripts as answer.php, db.php, logout.php, question.php, verify.php, vote.php. (Refer these files in source code, also all the required content in attached CD)

Another directory with name as testing is towards the test cases and its source code files as randomnumbergen.php, test.html, test2.html (Refer these files in source code)

Here open the file which is in the name as form.css (a cascading style sheet file for the program)
For database towards phpMyAdmin

Open the URL
http://members.000webhost.com/cpanel.php?accountID=12508022&login_hash=65HfAVzEzOAJ0Q2F

>>Procedure

http://members.000webhost.com/login.php

>>Enter username as mccluredanr@gmail.com

>>Password Iforgotit.

Now click the phpMyAdmin which is under Software/services

>>Click Enter phpMyAdmin which is under phpMyAdmin

>>This directs to the main database of the voting options questions, login details as username, passwords and rfid tag info and some other info

Herewith below is details of its main screen and their on to its sub options as graphical programming as figure – 42

Database graphical programming main screen as figure – 42
User login details graphical programming as figure – 43

Voting questions/option details along with its updated/casted answers as opted, its graphical programming as figure – 44
Details of vote casted and its info as updated lat long, vote time, user voted for question 1, voted options as answers, its graphical programming as figure – 45

figure – 45

Details of vote casted and its info as updated lat long, vote time, user voted for question 2, voted options as answers, its graphical programming as figure – 46

figure – 46
Details of vote casted and its info as updated lat long, vote time, user voted for question 3, voted options as answers, its graphical programming as figure – 47

figure – 47

Details of vote casted and its info as updated lat long, vote time, user voted for question 4, voted options as answers, its graphical programming as figure – 48

figure – 48
Details of vote casted and its info as updated lat long, vote time, user voted for question 5, voted options as answers, its graphical programming as figure – 49

figure – 49
9.0 MOBILE VOTING SYSTEM SOURCE CODE

9.1 ANDRIOD APPLICATION SOURCE CODE FILE

package efthyvoulos.votingapp;

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.nio.charset.Charset;
import java.util.ArrayList;
import java.util.List;
import org.apache.http.NameValuePair;
import org.apache.http.client.entity.UrlEncodedFormEntity;
import org.apache.http.client.entity.UrlEncodedFormEntity;
import org.apache.http.message.BasicNameValuePair;
import org.apache.http.client.entity.UrlEncodedFormEntity;
import org.apache.http.message.BasicNameValuePair;
import android.location.Location;
import android.location.LocationListener;
import android.location.LocationManager;
import android.nfc.NfcAdapter;
import android.nfc.Tag;
import android.nfc.tech.MifareClassic;
import android.nfc.tech.NfcA;
import android.nfc.tech.NfcB;
import android.nfc.tech.NfcF;
import android.nfc.tech.NfcV;
import android.os.Bundle;
import android.provider.Settings;
import android.app.Activity;
import android.app.AlertDialog;
import android.app.PendingIntent;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.IntentFilter;
import android.util.Log;
import android.view.Gravity;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.view.WindowManager;
import android.view.inputmethod.InputMethodManager;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemSelectedListener;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.CheckBox;
import android.widget.CompoundButton;
import android.widget.EditText;
import android.widget.ListView;
import android.widget.TextView;
import android.widget.Toast;
import android.widget.ViewFlipper;
import android.widget.ViewSwitcher;
import android.widget.CompoundButton.OnCheckedChangeListener;
import android.support.v4.app.NavUtils;

public class VotingMain extends Activity {

    //GLOBAL VARIABLES
    private String Current_Username = "INVALID";
    private String NFC_Code = null;
    private Tag tag;
    private int Current_Vote_Selection = -1;
    private int Question_ID;
    private Context context;

private Location current_location;

private Boolean login = false;
private String Current_Vote_String = "";

public LocationManager locationManager;
public LocationListener loclistener;

private NfcAdapter mAdapter;
private PendingIntent mPendingIntent;
private IntentFilter[] mFilters;
private String[][] mTechLists;

//Views
private ViewFlipper switcher;
private CheckBox check_1;
private CheckBox check_2;
private CheckBox check_3;
private CheckBox check_4;

@Override
protected void onResume() {
    // TODO Auto-generated method stub
    super.onResume();
    mAdapter.enableForegroundDispatch(this, mPendingIntent, mFilters, mTechLists);
}
@Override
public void onPause() {
    super.onPause();
    mAdapter.disableForegroundDispatch(this);
    locationManager.removeUpdates(loclistener);
}

@Override
protected void onStop() {
    // TODO Auto-generated method stub
    super.onStop();
    finish();
}

@Override
public void onNewIntent(Intent intent) {
    // fetch the tag from the intent
    tag = (Tag)intent.getParcelableExtra(NfcAdapter.EXTRA_TAG);
    String[] tlist = tag.getTechList();
    android.util.Log.v("NFC", "Discovered tag ["+tlist[0]+"]["+tag+"] with intent: " + intent);
    android.util.Log.v("NFC", "{"+tag+"}");
    new ReadNFC().start();
@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_voting_main);

    context=this;

    switcher = (ViewFlipper) findViewById(R.id.profileSwitcher);
    EditText edittext_username = (EditText) findViewById(R.id.edittext_username);
    EditText edittext_password = (EditText) findViewById(R.id.edittext_password);

    // Network Location
    locationManager = (LocationManager) getSystemService(Context.LOCATION_SERVICE);
    loclistener = new MyLocationListener();
    locationManager.requestLocationUpdates(LocationManager.NETWORK_PROVIDER, 0, 0, loclistener);

    // NFC Code
    mAdapter = NfcAdapter.getDefaultAdapter(this);

    if (mAdapter == null) {

// adapter does not exists

Toast.makeText(getApplicationContext(), "Your Phone Does Not Support NFC", Toast.LENGTH_SHORT).show();

finish();
}

if (!mAdapter.isEnabled() ) {

// adapter is not turned on

Toast.makeText(getApplicationContext(), "Please turn ON NFC", Toast.LENGTH_LONG).show();

startActivity(new Intent(android.provider.Settings.ACTION_WIRELESS_SETTINGS));

finish();
}

mPendingIntent = PendingIntent.getActivity(
    this, 0, new Intent(this, getClass()).addFlags(Intent.FLAG_ACTIVITY_SINGLE_TOP), 0);

IntentFilter ndef = new IntentFilter(NfcAdapter.ACTION_NDEF_DISCOVERED);
try {
    //ndef.addDataType("*/*");    // Handles all MIME based dispatches.
    ndef.addDataType("text/plain");    // You should specify only the ones that you need.
}

catch (MalformedMimeTypeException e) {

throw new RuntimeException("fail", e);
}

mFilters = new IntentFilter[] {ndef, }

mTechLists = new String[][] { new String[] {
    NfcV.class.getName(),
    NfcF.class.getName(),
    NfcA.class.getName(),
    NfcB.class.getName()
} }

// LOGIN PAGE

// Submits the login information to the server

Button SubmitLogin = (Button) findViewById(R.id.button_login);
SubmitLogin.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        //GET RFID

        //POST!
        if(NFC_Code==null)
        {
            Toast toast = Toast.makeText(getApplicationContext(), "Please Authenticate", Toast.LENGTH_SHORT);
            toast.setGravity(Gravity.CENTER, 0, 0);
            toast.show();
        }
else {
    new LoginSend().start();
}

// QUESTIONS PAGE

new GetQuestions().start();

ListView listQuestions = (ListView) findViewById(R.id.list_questions);

listQuestions.setClickable(true);

listQuestions.setOnItemClickListener(new AdapterView.OnItemClickListener() {
    public void onItemClick(AdapterView<?> arg0, View arg1, int position, long arg3) {
        // TODO Auto-generated method stub
        ListView listQuestions = (ListView) findViewById(R.id.list_questions);
        TextView text_question = (TextView) findViewById(R.id.text_question);
        text_question.setText(listQuestions.getAdapter().getItem(position).toString());
    }
});
text_question.setText(listQuestions.getItemAtPosition(position).toString());

    Question_ID = position+1;
    ResetAnswers();
    new Get_Specific_Questions().start();
});

// Specific Questions Page
// confirm Vote
Button ConfirmButton = (Button) findViewById(R.id.button_confirm);
ConfirmButton.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        //POST!
        Check_Current_Vote();
        if(Current_Vote_Selection==-1)
        {
            Toast toast = Toast.makeText(getApplicationContext(), "Please Select An Answer", Toast.LENGTH_SHORT);
            toast.setGravity(Gravity.CENTER, 0, 0);
            toast.show();
        }
    }
});
TextView AnswerConfirm = (TextView) findViewById(R.id.text_submitanswer);

AnswerConfirm.setText(Current_Vote_String);
switcher.showNext();
}
}
});

Button BackButton = (Button) findViewById(R.id.button_goback);
BackButton.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        //POST!
        switcher.showPrevious();
    }
});

check_1 = (CheckBox) findViewById(R.id.check_option1);
check_1.setOnCheckedChangeListener(new OnCheckedChangeListener() {
    public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) {
        if (isChecked) {
            Current_Vote_Selection = 1;
        }
    }
});
Current_Vote_String = check_1.getText().toString();

CloseChecksExcept(1);}
else Current_Vote_Selection = -1;
}
});

check_2 = (CheckBox) findViewById(R.id.check_option2);
check_2.setOnCheckedChangeListener(new OnCheckedChangeListener()
{
    public void onCheckedChanged(CompoundButton buttonView, boolean isChecked)
    {
        if (isChecked)
        {
            Current_Vote_Selection = 2;
            Current_Vote_String = check_2.getText().toString();
            CloseChecksExcept(2);
        }
        else Current_Vote_Selection = -1;
    }
});

check_3 = (CheckBox) findViewById(R.id.check_option3);
check_3.setOnCheckedChangeListener(new OnCheckedChangeListener()
{
    public void onCheckedChanged(CompoundButton buttonView, boolean isChecked)
    {
        if (isChecked)
        {
            Current_Vote_Selection = 3;
            Current_Vote_String = check_3.getText().toString();
            CloseChecksExcept(3);
        }
    }
});
else Current_Vote_Selection = -1;
}
});

check_4 = (CheckBox) findViewById(R.id.check_option4);
check_4.setOnCheckedChangeListener(new OnCheckedChangeListener()
{
    public void onCheckedChanged(CompoundButton buttonView, boolean isChecked)
    {
        if ( isChecked )
        {
            Current_Vote_Selection = 4;
            Current_Vote_String = check_4.getText().toString();
            CloseChecksExcept(4);
        }
        else Current_Vote_Selection = -1;
    }
});

// Submit Vote
Button VoteButton = (Button) findViewById(R.id.button_submitvote);
VoteButton.setOnClickListener(new View.OnClickListener()
{
    public void onClick(View v)
    {
      //POST!
      Check_Current_Vote();
      if(Current_Vote_Selection===-1)
      {

Toast toast = Toast.makeText(getApplicationContext(), "Please Select An Answer", Toast.LENGTH_SHORT);

toast.setGravity(Gravity.CENTER, 0, 0);
toast.show();

} else {

final AlertDialog.Builder builder = new AlertDialog.Builder(context);

builder.setMessage("Are you sure you want to vote for ".concat(Current_Vote_String.concat("?")))
 .setCancelable(false)
 .setPositiveButton("Yes", new DialogInterface.OnClickListener() {
     public void onClick(@SuppressWarnings("unused") final DialogInterface dialog,
                     @SuppressWarnings("unused") final int id) {

     new SubmitAnswer().start();
     //if(post_result)
     switcher.showPrevious();
     switcher.showPrevious();

     }
 })
 .setNegativeButton("No", new DialogInterface.OnClickListener() {
     public void onClick(final DialogInterface dialog, @SuppressWarnings("unused") final int id) {

     dialog.cancel();
     switcher.showPrevious();

     }
  })

final AlertDialog alert = builder.create();
alert.show();

// Submit Vote
Button DontVoteButton = (Button) findViewById(R.id.button_dontsubmitvote);
DontVoteButton.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        //POST!
        switcher.showPrevious();
    }
});

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.activity_voting_main, menu);
    return true;
private void CloseChecksExcept(int i) {

    switch (i) {
    case 1:
        check_2.setChecked(false);
        check_3.setChecked(false);
        check_4.setChecked(false);
        break;
    case 2:
        check_1.setChecked(false);
        check_3.setChecked(false);
        check_4.setChecked(false);
        break;
    case 3:
        check_1.setChecked(false);
        check_2.setChecked(false);
        check_4.setChecked(false);
        break;
    case 4:
        check_1.setChecked(false);
        check_2.setChecked(false);
        check_3.setChecked(false);
        break;
    }
private void ResetAnswers() {

    Current_Vote_Selection=-1;
    check_1.setChecked(false);
    check_2.setChecked(false);
    check_3.setChecked(false);
    check_4.setChecked(false);
}

private void Check_Current_Vote()
{
    if(check_1.isChecked())

    {
        Current_Vote_Selection=1;
    }
    if(check_2.isChecked())

    {
        Current_Vote_Selection=2;
    }
    if(check_3.isChecked())

    {
        Current_Vote_Selection=3;
    }
if(check_4.isChecked())
{
    Current_Vote_Selection=4;
}

private class LoginSend extends Thread {

    @Override
    public void run() {

        if(login==false)
        {

        EditText edittext_username = (EditText) findViewById(R.id.edittext_username);

        EditText edittext_password = (EditText) findViewById(R.id.edittext_password);

        // Create a new HttpClient and Post Header
        HttpClient httpclient = new DefaultHttpClient();

        HttpPost httppost = new HttpPost("http://votingappresult.net23.net/android/verify.php");

        try {
            
        }
// Add your data
List<NameValuePair> nameValuePairs = new
ArrayList<NameValuePair>(2);

nameValuePairs.add(new BasicNameValuePair("username",
edittext_username.getText().toString()));

nameValuePairs.add(new BasicNameValuePair("password",
edittext_password.getText().toString()));

nameValuePairs.add(new BasicNameValuePair("rfid",
NFC_Code));

HttpPost.setEntity(new UrlEncodedFormEntity(nameValuePairs));

// Execute HTTP Post Request
HttpResponse response = httpclient.execute(httppost);

HttpEntity response_entity = response.getEntity();

InputStream is = response_entity.getContent();

BufferedReader r = new BufferedReader(new
InputStreamReader(is));

String test = r.readLine();

if(test.equals("true"))
{
    Current_Username=edittext_username.getText().toString();
    runOnUiThread(new Runnable () {
        public void run () {
            check_1.requestFocus();
        }
    });
}
Toast toast = Toast.makeText(getApplicationContext(), "Login Successful", Toast.LENGTH_SHORT);
    toast.setGravity(Gravity.CENTER, 0, 0);
    toast.show();

    login = true;

    switcher.showNext();
    InputMethodManager imm = (InputMethodManager) getSystemService(Context.INPUT_METHOD_SERVICE);

    imm.hideSoftInputFromWindow(switcher.getApplicationWindowToken(), 0);
    }
    }
} else{
    runOnUiThread(new Runnable() {
        public void run() {
            Toast toast = Toast.makeText(getApplicationContext(), "Wrong Username or Password", Toast.LENGTH_SHORT);
            toast.setGravity(Gravity.CENTER, 0, 0);
            toast.show();
        }
    });
}

} catch (ClientProtocolException e) {
    // TODO Auto-generated catch block
} catch (IOException e) {

private class GetQuestions extends Thread {

    @Override
    public void run() {

        // Create a new HttpClient and Post Header
        HttpClient httpclient = new DefaultHttpClient();

        HttpPost httppost = new
        HttpPost("http://votingappresult.net23.net/android/question.php");

        try {

            // Add your data
            List<NameValuePair> nameValuePairs = new
            ArrayList<NameValuePair>(2);

            nameValuePairs.add(new BasicNameValuePair("please", "thanks"));
            httppost.setEntity(new UrlEncodedFormEntity(nameValuePairs));
// Execute HTTP Post Request
HttpResponse response = httpClient.execute(httpPost);
HttpEntity response_entity = response.getEntity();

InputStream is = response_entity.getContent();

BufferedReader r = new BufferedReader(new InputStreamReader(is));
String test = r.readLine();
int numberQs = Integer.parseInt(test);

String[] tempQuestionsArray = new String[numberQs];
for(int i=0;i<numberQs;i++)
{
    test = r.readLine();
    tempQuestionsArray[i]=test;
}

final String[] QuestionsArray = tempQuestionsArray;

runOnUiThread(new Runnable() {
    public void run() {

        ListView listView = (ListView) findViewById(R.id.list_questions);

        ArrayAdapter<String> adapter = new ArrayAdapter<String>(context,android.R.layout.simple_list_item_1, QuestionsArray);

    }
});
// Assign adapter to ListView
listView.setAdapter(adapter);

private class Get_Specific_Questions extends Thread {

    @Override
    public void run() {

        // Create a new HttpClient and Post Header
        HttpClient httpclient = new DefaultHttpClient();
       HttpPost httppost = new HttpPost("http://votingappresult.net23.net/android/answer.php");

        // TODO Auto-generated catch block
        } catch (IOException e) {
            // TODO Auto-generated catch block
        }
try {

    // Add your data
    List<NameValuePair> nameValuePairs = new ArrayList<NameValuePair>(2);
    nameValuePairs.add(new BasicNameValuePair("qid", String.valueOf(Question_ID)));
    nameValuePairs.add(new BasicNameValuePair("username", Current_Username));
    httppost.setEntity(new UrlEncodedFormEntity(nameValuePairs));

    // Execute HTTP Post Request
    HttpResponse response = httpclient.execute(httppost);
    HttpEntity response_entity = response.getEntity();

    InputStream is = response_entity.getContent();

    BufferedReader r = new BufferedReader(new InputStreamReader(is));
    String worked = r.readLine();

    if(worked.equals("accepted"))
    {
        final String one = r.readLine();
        final String two = r.readLine();
        final String three = r.readLine();
        final String four = r.readLine();
runOnUiThread(new Runnable() {
    public void run() {

        findViewByld(R.id.check_option1);
        CheckBox check_1 = (CheckBox)

        findViewByld(R.id.check_option2);
        CheckBox check_2 = (CheckBox)

        findViewByld(R.id.check_option3);
        CheckBox check_3 = (CheckBox)

        findViewByld(R.id.check_option4);
        CheckBox check_4 = (CheckBox)

        check_1.setText(one);
        check_2.setText(two);
        check_3.setText(three);
        check_4.setText(four);

        switcher.showNext();
    }
});

} else {
    final String answer = r.readLine();

    runOnUiThread(new Runnable() {
        public void run() {

            } });
}
Toast toast = Toast.makeText(getApplicationContext(), "You Have Already Voted For " + answer, Toast.LENGTH_SHORT);
    toast.setGravity(Gravity.CENTER, 0, 0);
    toast.show();

} });

} }

} catch (ClientProtocolException e) {
    // TODO Auto-generated catch block
} catch (IOException e) {
    // TODO Auto-generated catch block
}

private class SubmitAnswer extends Thread {

    @Override
    public void run() {

        // Create a new HttpClient and Post Header
        HttpClient httpclient = new DefaultHttpClient();
        HttpPost httppost = new HttpPost("http://votingappresult.net23.net/android/vote.php");

        httppost.setEntity(new UrlEncodedFormEntity(new NameValuePair[] {
            new BasicNameValuePair("answer", answer)
        }));

    }

}
Location

LastKnown = locationManager.getLastKnownLocation(LocationManager.PASSIVE_PROVIDER);

String Latitude = String.valueOf((int)(LastKnown.getLatitude() * 1000000));

String Longitude = String.valueOf((int)(LastKnown.getLongitude() * 1000000));

try {

    // Add your data

    List<NameValuePair> nameValuePairs = new ArrayList<NameValuePair>(2);

    nameValuePairs.add(new BasicNameValuePair("qid", String.valueOf(Question_ID)));

    nameValuePairs.add(new BasicNameValuePair("answer", String.valueOf(Current_Vote_Selection)));

    nameValuePairs.add(new BasicNameValuePair("username", Current_Username));

    nameValuePairs.add(new BasicNameValuePair("longitude", Longitude));

    nameValuePairs.add(new BasicNameValuePair("latitude", Latitude));

    httppost.setEntity(new UrlEncodedFormEntity(nameValuePairs));

    // Execute HTTP Post Request

    HttpResponse response = httpclient.execute(httppost);

    HttpEntity response_entity = response.getEntity();

    InputStream is = response_entity.getContent();
BufferedReader r = new BufferedReader(new InputStreamReader(is));
String test = r.readLine();
if(test.equals("accepted"))
{
    runOnUiThread(new Runnable() {
        public void run() {

            check_1.requestFocus();

            Toast toast = Toast.makeText(getApplicationContext(), "Submitted Vote", Toast.LENGTH_SHORT);
            toast.setGravity(Gravity.CENTER, 0, 0);
            toast.show();

        }
    });
}
else{
    runOnUiThread(new Runnable() {
        public void run() {

            Toast toast = Toast.makeText(getApplicationContext(), "Submission Failed", Toast.LENGTH_SHORT);
            toast.setGravity(Gravity.CENTER, 0, 0);
            toast.show();

        }
    });
}
} catch (ClientProtocolException e) {
private class ReadNFC extends Thread {

    @Override
    public void run() {

        MifareClassic mifare = MifareClassic.get(tag);
        try {
            mifare.connect();
            mifare.authenticateSectorWithKeyA(1, MifareClassic.KEY_NFC_FORUM);
            int index = mifare.sectorToBlock(1);
            NFC_Code = mifare.readBlock(index).toString();

            byte[] payload0 = mifare.readBlock(index);
            String temp0 = new String(payload0, Charset.forName("US-ASCII"));
            byte[] payload1 = mifare.readBlock(index+1);
            String temp1 = new String(payload1, Charset.forName("US-ASCII"));
            NFC_Code = temp0.substring(temp0.length()-5);
            NFC_Code = NFC_Code.concat(temp1.substring(0, 5));
        }
    }
}
Log.e(NFC_SERVICE,NFC_Code);

runOnUiThread(new Runnable() {
    public void run() {
        Toast toast = Toast.makeText(getApplicationContext(), "Authentication Accepted", Toast.LENGTH_SHORT);
        toast.setGravity(Gravity.CENTER, 0, 0);
        toast.show();
    }
});

//byte[] payload = mifare.readBlock(index);
//Log.e(NFC_SERVICE,new String(payload, Charset.forName("US-ASCII")));)
//byte[] payload = mifare.readPages(4);
//return new String(payload, Charset.forName("US-ASCII"));
}
} catch (IOException e) {
    Log.e(NFC_SERVICE, "IOException while reading MifareClassic message...", e);
}
} finally {
    if (mifare != null) {
        try {
            mifare.close();
        }
    }
}
    catch (IOException e) {
        Log.e(NFC_SERVICE, "Error closing tag...", e);
    }
}


private final class MyLocationListener implements LocationListener {

    public void onLocationChanged(Location locFromNetwork) {
        // called when the listener is notified with a location update from the GPS
        current_location=locFromNetwork;

        //Toast.makeText(getBaseContext(), String.valueOf(current_location.getLatitude())+"",
        "+String.valueOf(current_location.getLongitude()), Toast.LENGTH_SHORT).show();
    }

    public void onProviderDisabled(String provider) {
        // called when the GPS provider is turned off (user turning off the GPS on the phone)
    }

    public void onProviderEnabled(String provider) {
        // called when the GPS provider is turned on (user turning on the GPS on the phone)
    }

    public void onStatusChanged(String provider, int status, Bundle extras) {
        // called when the status of the GPS provider changes
    }

}
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
package="efthyvoulos.votingapp"
android:versionCode="1"
android:versionName="1.0" >

<uses-sdk
    android:minSdkVersion="10"
    android:targetSdkVersion="15" />

<uses-permission android:name="android.permission.INTERNET"/>
<uses-permission android:name="android.permission.NFC"/>
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />

<application
    android:icon="@drawable/ic_launcher"
    android:label="@string/app_name"
    android:theme="@style/AppTheme" >
<activity
    android:name=".VotingMain"
    android:label="@string/title_activity_voting_main"
    android:screenOrientation="portrait">

<intent-filter>
    <action android:name="android.intent.action.MAIN" />
</intent-filter>
</activity>
</application>

128
<category android:name="android.intent.category.LAUNCHER" />

</intent-filter>

</activity>

</application>

</manifest>
9.2 WEB HOSTING SOURCE CODE FILE

FORM1 CSS

/* Stylesheet */

/* Imports */

@import url/animate.css);

/* Selections */

::selection {
    color: #fff;
    text-shadow: none;
    background: #444;
}

::-moz-selection {
    color: #fff;
    text-shadow: none;
    background: #444;
}

/* Basics */

html, body {
    width: 100%;
    height: 100%;
    font-family: "Helvetica Neue", Helvetica, sans-serif;
    color: #444;
    -webkit-font-smoothing: antialiased;
    background: #F0F0F0;
}

#container {
    position: fixed;
    width: 340px;
    height: 280px;
    top: 50%;
    left: 50%;
    margin-top: -140px;
    margin-left: -170px;
color: #555;
}

input {
    font-family: "Helvetica Neue", Helvetica, sans-serif;
    font-size: 12px;
    outline: none;
}

input[type=name],
input[type=password] {
    color: #777;
    padding-left: 10px;
    margin: 10px;
    margin-top: 12px;
    margin-left: 18px;
    width: 290px;
    height: 35px;
    border: 1px solid #c7d0d2;
    border-radius: 2px;
    box-shadow: inset 0 1.5px 3px rgba(190, 190, 190, .4), 0 0 0 5px #f5f7f8;
    -webkit-transition: all .4s ease;
    -moz-transition: all .4s ease;
    transition: all .4s ease;
}

input[type=name]:hover,
input[type=password]:hover {
    border: 1px solid #b6bfc0;
    box-shadow: inset 0 1.5px 3px rgba(190, 190, 190, .7), 0 0 0 5px #f5f7f8;
}

input[type=name]:focus,
input[type=password]:focus {
    border: 1px solid #a8c9e4;
    box-shadow: inset 0 1.5px 3px rgba(190, 190, 190, .4), 0 0 0 5px #e6f2f9;
}

#lower {
    background: #ecf2f5;
    width: 100%;
    height: 69px;
    margin-top: 20px;
    box-shadow: inset 0 1px 1px #fff;
    border-top: 1px solid #ccc;
    border-bottom-right-radius: 3px;
border-bottom-left-radius: 3px;
}

input[type=checkbox] {
    margin-left: 20px;
    margin-top: 30px;
}

.check {
    margin-left: 3px;
    font-size: 11px;
    color: #444;
    text-shadow: 0 1px 0 #fff;
}

input[type=submit] {
    float: right;
    margin-right: 20px;
    margin-top: 20px;
    width: 80px;
    height: 30px;
    font-size: 14px;
    font-weight: bold;
    color: #fff;
    background-image: -webkit-gradient(linear, left top, left bottom, from(#acd6ef),
    to(#6ec2e8));
    background-image: -moz-linear-gradient(top left 90deg, #acd6ef 0%, #6ec2e8 100%);
    background-image: linear-gradient(top left 90deg, #acd6ef 0%, #6ec2e8 100%);
    border-radius: 30px;
    border: 1px solid #66add6;
    box-shadow: 0 1px 2px rgba(0, 0, 0, .3), inset 0 1px 0 rgba(255, 255, 255, .5);
    cursor: pointer;
}

input[type=submit]:hover {
    background-image: -webkit-gradient(linear, left top, left bottom, from(#b6e2ff),
    to(#6ec2e8));
    background-image: -moz-linear-gradient(top left 90deg, #b6e2ff 0%, #6ec2e8 100%);
    background-image: linear-gradient(top left 90deg, #b6e2ff 0%, #6ec2e8 100%);
}

input[type=submit]:active {
    background-image: -webkit-gradient(linear, left top, left bottom, from(#6ec2e8),
    to(#b6e2ff));
    background-image: -moz-linear-gradient(top left 90deg, #6ec2e8 0%, #b6e2ff 100%);
    background-image: linear-gradient(top left 90deg, #6ec2e8 0%, #b6e2ff 100%);
}
INDEX.HTML

1.  
2.  &lt;br&gt;&lt;br&gt;&lt;br&gt;&lt;center&gt;&lt;img 
src="http://votingappresult.net23.net/MobileVotingSysLogoNew1.jpg" style="max-width:100%" align="middle" /&gt;&lt;/center&gt; 
3.  
4.  &lt;marquee&gt;Welcome To Mobile Voting System&lt;/marquee&gt; 
5.  
6.  
7.  &lt;link href="form.css" rel="stylesheet" type="text/css" /&gt; 
8.  
9.  &lt;form id='login' action='login.php' method='post' accept-charset='UTF-8'&gt; 
10.  
11.  &lt;legend&gt;&lt;b&gt;Login&lt;/b&gt;&lt;/legend&gt; 
12.  &lt;input type='hidden' name='submitted' id='submitted' value='1'/&gt; 
13.  
14.  &lt;label for='username'&gt;Username*:&lt;/label&gt; 
15.  &lt;input type='text' name='username' id='username' maxlength='50' /&gt; 
16.  
17.  &lt;label for='password'&gt;Password*:&lt;/label&gt; 
18.  &lt;input type='password' name='password' id='password' maxlength='50' /&gt; 
19.  
20.  &lt;input type='submit' name='Submit' value='Submit' /&gt; 
21.  
22.  &lt;/center&gt; 
23.  
24.  &lt;/form&gt; 
25.  &lt;/div&gt;
Login.php

```php
<?php

$DBUsername = "a8586955_vote";
$DBPassword = "1vote23";
$DBHost = "mysql4.000webhost.com";
$DBname = "a8586955_votedb";
$DBConnect = mysql_connect($DBHost, $DBUsername, $DBPassword) or die('Could not connect: '.mysql_error());
$DBSelect = mysql_select_db($DBname) or die('Could not select database');
$Num_questions1 = mysql_query('SELECT COUNT(*) FROM information_schema.tables WHERE table_schema = "a8586955_votedb"');
$Num_questions2 = mysql_fetch_row($Num_questions1);
$Num_questions = $num_questions2[0]-2;
if(empty($_POST['username']))
    
    echo "<br><br><br><br><br><br><br><br><br><br><br><br><br><br><br><br><br><br><center><br><br><br>Username is empty<br><br>Please! try again.</br></center>";
return false;

if(empty($_POST['password']))
    
    echo "<br><br><br><br><br><br><br><br><br><br><br><br><br><br><br><br><br><br><center><br><br><br>Password is empty.<br><br>Please! try again.</br></center>";
return false;

$username = trim($_POST['username']);
$password = trim($_POST['password']);
if($username=="rimon"&&$password=="vote")
```
34. {
35. 36. 37. //.SYSTEM ADMINISTRATOR /////////////
38.  echo "<center>";
39.  echo '}<form><img src="http://votingappresult.net23.net/logoutIcon.jpg" style="max-width:100%" title="Logout" align="middle" <input type=button value="Logout" onClick="parent.location='index.html""></form>;
40.  echo '<p style="font-size:4em;">Administrator</p>'
41.  $result = mysql_query("SELECT * FROM login");
42.  if (mysql_num_rows($result) > 0) {
43.  // yes
44.  // print them one after another
45.  echo "<p>";
46.  echo "<table cellpadding=10 border=1>";
47.  echo "<tr>";
48.  echo "<td>"."ID Number"."</td>";
49.  echo "<td>"."Username"."</td>";
50.  echo "<td>"."Password"."</td>";
51.  echo "</tr>";
52.  $row = mysql_fetch_row($result)) {
53.  while($row = mysql_fetch_row($result)) {
54.  echo "<tr>";
55.  echo "<td>".$row[0]."</td>";
56.  echo "<td>".$row[1]."</td>";
57.  echo "<td>".$row[2]."</td>";
58.  echo "</tr>";
59.  }
60.  echo "</table>";
61.  echo "</p>";
62.  }
63.  else {
64.  // no
65.  // print status message
66.  echo "No Voters found!";
67.  }
68.  for ($i = 1; $i<=$num_questions; $i++) {
69.  echo "\n";
70.  $result = mysql_query("SELECT * FROM question".$i);
71.  if (mysql_num_rows($result) > 0) {
72.  // yes
73.  // print them one after another
mysql_select_db("a8586955_votedb",mysql_connect($DBHost, $DBUsername, $DBPassword));

$cmd="select*from question where id='$i'";

$result1=mysql_query($cmd,mysql_connect($DBHost, $DBUsername, $DBPassword));

echo "<p>";

while ($rr=mysql_fetch_row($result1)) {

    echo "<tr>";
    echo "<td>"."Question "$i."</td>";
    echo "<td>"."Username"."</td>";
    echo "<td>"."Answer"."</td>";
    echo "<td>"."Latitude"."</td>";
    echo "<td>"."Longitude"."</td>";
    echo "<td>"."Time"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";

    echo "$rr[0]."."</td>";
    echo "$rr[1]."."</td>";
    echo "$rr[2]."."</td>";
    echo "$rr[3]."."</td>";
    echo "$rr[4]."."</td>";
    echo "$rr[5]."."</td>";
    echo "$rr[6]."."</td>";
    echo "$rr[7]."."</td>";
    echo "$rr[8]."."</td>";
    echo "$rr[9]."."</td>";

    echo "</tr>";
}

while($row = mysql_fetch_row($result)) {

    echo "<tr>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "<td>"."</td>";
    echo "</tr>";
}

else {
    // no
    // print status message
    echo "No Data for question ".$i."\n";
}

echo "</center>";

} elseif($username=="raj"&&$password=="jar") {

    //////////////////////////////////////////////////////////
    SYSTEM ENGINEER //////////////////////////////////////////////////

    echo "<center>";

} elseif($username=="raj"&&$password=="jar") {
<form><img src="http://votingappresult.net23.net/logoutIcon.jpg" style="max-width:100%" title="Logout" align="middle" type=button value="Logout" onClick="parent.location='index.html'"></form>

<p style="font-size:4em;">System Engineer</p>

<p>
<table cellpadding=10 border=1>
<tr>
<td>Question</td>
<td>Summary of Option 1</td>
<td>Summary of Option 2</td>
<td>Summary of Option 3</td>
<td>Summary of Option 4</td>
</tr>
</table>
</p>

for ($i = 1; $i<=$num_questions; $i++)
{
    $option1=0;
    $option2=0;
    $option3=0;
    $option4=0;

    $result = mysql_query("SELECT * FROM question".$i);
    if (mysql_num_rows($result) > 0)
    {
        while($row = mysql_fetch_row($result))
        {
            switch($row[1])
            {
                case 1:
                    $option1=$option1+1;
                    break;
                break;
                case 2:
                    $option2=$option2+1;
                    break;
                case 3:
                    $option3=$option3+1;
                    break;
                case 4:
                    $option4=$option4+1;
                    break;
            }
        }
    }
}
165. } } 
166. mysql_select_db("a8586955_votedb", mysql_connect($DBHost, $DBUsername, $DBPassword)); 
167. $cmd="select*from question where id='$i'"; 
168. $result1=mysql_query($cmd,mysql_connect($DBHost, $DBUsername, $DBPassword)); 
169. echo "<tr>"; 
170. $rr=mysql_fetch_row($result1); 
171. //echo "<td>"."Question "."$r[1]"."</td>"; 
172. echo "<td>"."$i"." Srr[1]"."</td>"; 
173. echo "<td>"."<center>".Soption1."</center>"."</td>"; 
174. echo "<td>"."<center>".Soption2."</center>"."</td>"; 
175. echo "<td>"."<center>".Soption3."</center>"."</td>"; 
176. echo "<td>"."<center>".Soption4."</center>"."</td>"; 
177. echo "</tr>"; 
178. } 
179. } } } } } } } } } } } 
180. echo "</table>"; 
181. echo "</p>"; 
182. echo "</center>"; 
183. echo "</center>"; 
184. echo "</center>"; 
185. echo "</center>"; 
186. } } } } } } else { 
187. echo "<br><br><br><br><br><br><br><br><br><br><br><br><br><br><br><br><br><br>Incorrect Username or Password.<br><br>Please! try again.<br><br>"; 
188. echo "<center><form><img src="http://votingappresult.net23.net/BackHome.png" style="max-width:100%" title="Go Back" align="middle" <input type=button value="Logout" onClick="parent.location='index.html'"/>></form></center>';
189. } 
190. 
191. 
192. 
193. ?>

**STYLES CSS**

```css
body {
  font: normal 11px auto "Trebuchet MS", Verdana, Arial, Helvetica, sans-serif;
  color: #E61818;
  background: #7FFFD4;
}
```
9.3 ANDROID AND WEBHOSTING LINKING FILES SOURCE CODE

ANSWER.PHP

<?php
require('db.php');

$username = $_POST['username'];
$qid = $_POST['qid'];

$questionspecific = "question" . $qid;

$myanswer = mysql_query("SELECT answer FROM $questionspecific WHERE username= "Username" ");

//$rowanswer = mysql_fetch_row($myanswer);

if (!$rowanswer[0]) {
    if (!mysql_num_rows($myanswer) > 0) {
        $result = mysql_query("SELECT answer1, answer2, answer3, answer4 FROM question WHERE ID = $qid");
        $row = mysql_fetch_row($result);
        echo "accepted\n";
        for ($i = 0; $i <= 3; $i++)
            echo "$row[$i]\n";
    }

} else {
    $rowanswer = mysql_fetch_row($myanswer);
    $myanswer = "$answer" . $rowanswer[0];
    // $myanswernumber = "$answer" . $myanswer;
    $myanswerstring = mysql_query("SELECT $myanswer FROM question WHERE ID = $qid");
    $rowresult = mysql_fetch_row($myanswerstring);
<?php
$DBUsername = "a8586955_vote";
$DBPassword = "1vote23";
?>
$DBHost = "mysql4.000webhost.com";
$DBname = "a8586955_votedb";

mysql_connect($DBHost, $DBUsername, $DBPassword) or die('Could not connect: '.mysql_error());

mysql_select_db($DBname) or die('Could not select database');

?>

LOGOUT.PHP

<?php
    session_start();
    session_unset();
session_destroy();
?>

<?php
require('db.php');

$result = mysql_query("SELECT question FROM question");
$num_rows = mysql_num_rows($result);
echo "$num_rows\n";

for ($i = 0; $i<=$num_rows; $i++) {
    $row = mysql_fetch_row($result);
    echo "$row[0]\n";
}
?>
$_SESSION['verify']=FALSE;

$username = $_POST['username'];
$password = $_POST['password'];
$rfid = $_POST['rfid'];

$sql = "SELECT * FROM login WHERE username = '$username' AND password = '$password'
AND rfid = '$rfid"; 
$result = mysql_query($sql);
if (!$result) {
    error('A database error occurred while checking your '.
'login details.'.
'If this error persists, please '.
'contact mccluredanr@gmail.com.');
}

if (mysql_num_rows($result) == 0) {
    echo "false";
    $_SESSION['verify']=FALSE;
} else {
    echo "true";
    $_SESSION['verify']=TRUE;
    $_SESSION['username']=$username;
VOTE.PHP

<?php
require('db.php');
if ($_SESSION['verify']==TRUE) { } ?>
echo "You are not logged in";
else{
    $username = $_POST['username'];
    $answer = $_POST['answer'];
    $latitude = $_POST['latitude'];
    $longitude = $_POST['longitude'];
    $qid = $_POST['qid'];
    $tablename = "question" . $qid;
    date_default_timezone_set('US/Pacific');
    $datetime = date("Y-m-d H:i:s");
    $insert_sql = "INSERT INTO $tablename (username, answer, longitude, latitude, time)
VALUES ('$username', '$answer', '$longitude', '$latitude', '$datetime');
$run_sql = mysql_query($insert_sql);

if($run_sql)
    echo "accepted";
else
    echo "failed";
}

9.4 TEST CASE FILES

Randomnumbergen.php
<?php
    echo rand(5, 1500) . "\n";
?>

<!DOCTYPE html>
<html>

Test.Html

<!DOCTYPE html>
<html>

150
<script>
$.ajaxSetup({ cache: false });
$(document).ready(function(){
    $('button').click(function(){
        $.get("randomnumbergen.php",function(data){
            document.write("<br/>Random Number: "+ data);
        });
        $.get("randomnumbergen.php",function(data){
            document.write("<br/>Random Number: "+ data);
        });
        $.get("randomnumbergen.php",function(data){
            document.write("<br/>Random Number: "+ data);
        });
    });
});
</script>
</head>
<body>
<button>3 Random Numbers</button>
</body>
</html>
Test2.Html

<!DOCTYPE html>
<html>
function writeMyPath(id) {
    var counter = 0;
    var path = [];
    var el = document.getElementById(id);

    function siblingCounter(el) {
        if (el.previousSibling !== null && el.previousSibling.nodeType === 1) {
            counter++;
            siblingCounter(el.previousSibling);
        } else if (el.previousSibling !== null) {
            siblingCounter(el.previousSibling);
        } else {
            return counter;
        }
    }

    function pathWriter(el) {
        if (el.parentNode.nodeType !== 9 && el.nodeType === 1) {
            counter = 0;
            siblingCounter(el);
        }
    }
}
path.unshift(el.nodeName + counter);

pathWriter(el.parentNode);

}

else if(el.parentNode.nodeType !== 9 && el.nodeType !== 1){

pathWriter(el.parentNode);

}

else {

path.unshift(document.documentElement + counter);

return path;

}

pathWriter(el);

alert(path);

</script>

</head>

<body><div><div><img id="img1" src="image1.gif" onclick="writeMyPath(this.id)" /><a id="anchor1" href="link.html" onclick="writeMyPath(this.id)">Click me!</a><img id="img2" src="image2.jpg" onclick="writeMyPath(this.id)" /></div> <div><img id="img3" src="image3.png" onclick="writeMyPath(this.id)" /></div></div></body>

</html>

9.5 MY SQL DATABASE QUERY

Login>> username
SELECT COUNT( * ) AS `Rows` , `username`  
FROM `login`  
GROUP BY `username`  
ORDER BY `username`  
LIMIT 0 , 30

Login>> password

SELECT COUNT( * ) AS `Rows` , `password`  
FROM `login`  
GROUP BY `password`  
ORDER BY `password`  
LIMIT 0 , 30

Login>>RFID

SELECT COUNT( * ) AS `Rows` , `rfid`  
FROM `login`  
GROUP BY `rfid`  
ORDER BY `rfid`  
LIMIT 0 , 30

Question>>Question

SELECT COUNT( * ) AS `Rows` , `question`  
FROM `question`
GROUP BY `question`
ORDER BY `question`
LIMIT 0 , 30

Question>>Answer1

SELECT COUNT( * ) AS `Rows` , `answer1`
FROM 'question'
GROUP BY `answer1`
ORDER BY `answer1`
LIMIT 0 , 30

Question>>Answer2

SELECT COUNT( * ) AS `Rows` , `answer2`
FROM 'question'
GROUP BY `answer2`
ORDER BY `answer2`
LIMIT 0 , 30

Question>>Answer3

SELECT COUNT( * ) AS `Rows` , `answer3`
FROM 'question'
GROUP BY `answer3`
ORDER BY `answer3`
LIMIT 0 , 30

Question>>Answer4

SELECT COUNT( * ) AS `Rows` , `answer4`
FROM 'question'
GROUP BY `answer4`
ORDER BY `answer4`
LIMIT 0 , 30

Question1>>username
SELECT COUNT( * ) AS `Rows`, `username`
FROM `question1`
GROUP BY `username`
ORDER BY `username`
LIMIT 0, 30

Question1>>answer

SELECT COUNT( * ) AS `Rows`, `answer`
FROM `question1`
GROUP BY `answer`
ORDER BY `answer`
LIMIT 0, 30

Question1>>latitude

SELECT COUNT( * ) AS `Rows`, `latitude`
FROM `question1`
GROUP BY `latitude`
ORDER BY `latitude`
LIMIT 0, 30

Question1>>longitude

SELECT COUNT( * ) AS `Rows`, `longitude`
FROM `question1`
GROUP BY `longitude`
ORDER BY `longitude`
LIMIT 0, 30

Question1>>time

SELECT COUNT( * ) AS `Rows`, `time`
FROM `question1`
GROUP BY `time`
ORDER BY `time`
LIMIT 0, 30

10 LIST OF ACRONYMS

RFID Radio Frequency Identification
NFC       Near Field Communication
RADAR     Radio Detection and Ranging
RF        Radio Frequency
USB       Universal Serial Bus
SPI       Serial Peripheral Interface
I2C       Inter Integrated Circuit
GPS       Global Positioning System
TX        Transmission
RX        Receiving
SQL       Structured Query Language
OS        Operating System
APK       Application Package File
HTML      Hyper Text Markup Language
PHP       Hypertext Preprocessor
URL       Uniform Resource Locator
GUI       Graphical User Interface
HTTP      Hyper Text Transfer Protocol
AOSP      Android Open Source Project
OHA       Open Handset Alliance
CSS       Cascading style sheet
CDMA      Code Division Multiplexing Access
GSM       Global System Mobile
HDMI      High Definition Multimedia Interface
4G        Fourth Generation
GPRS      General Packet for Radio Service
CPU  Central Processing Unit
GPU  Graphics Processing Unit
OS   Operating System
Java SE Java Standard Edition
Java ME Java Mobile Edition
SDK  Software Development Kit
IDE  Integrated Development Environment
A GPS Assisted Global Positioning System
S GPS Simultaneous Global Positioning System
Lat Long  Latitude Longitude
AOA  Android Open Accessory
ADK  Android Development Kit
JJIL  Jon’s Java Imaging Library
KHZ  Kilo Hertz
GHZ  Giga Hertz
SSH  Secure Shell

11 REFERENCES


http://gigaom.com/2012/02/02/google-tv-honeycomb-32/


https://exploreb2b.com/articles/top-10-tools-for-mysql-development-and-administration-of-sandbox