ABSTRACT

Sponsorship is an Android app which sets up an online platform for sponsorship related events. This app allows the users to post their requirements regarding the current ongoing events. The Sponsors looking to provide sponsorship can login into the app and view the posts regarding the current ongoing events and can respond to the post by contacting the respective candidate or organization. This app provides a medium for various kinds of organizations, businesses, schools, students, charities, social events etc., to meet the public by sponsoring events and there by promoting their organizations. Users can login into the application and can upload the documents that are required for the event, such as event logo, student transcripts and brochure which contains the information regarding the event. Sponsors can view the documents and can post their comments for the respective event. Users can list the amount or quantity of sponsorship they are looking for. In addition sponsorship can be of many types such as clothes for social events or charities, food for college events, furniture for local events etc., this app includes all these functionalities. Facebook and Gmail login is integrated into the application there by allowing users and sponsors to directly login to the application without registering for the application.
# TABLE OF CONTENTS

Abstract ................................................................................................................................. ii

Table of Contents ................................................................................................................. iii

List of Figures ...................................................................................................................... vii

List of Tables ..................................................................................................................... xi

1. Background and Rationale .............................................................................................. 1

   1.1 Android ....................................................................................................................... 3

   1.2 Android Studio ........................................................................................................... 4

   1.3 Facebook and Gmail Integration ............................................................................... 4

   1.4 List of Events ............................................................................................................. 6

       1.4.1 Business Event .................................................................................................. 6

       1.4.2 College Event ..................................................................................................... 7

       1.4.3 Orphanage Event ............................................................................................... 7

       1.4.4 Proposal Listing Event ...................................................................................... 8

       1.4.5 Student Scholarship Event ................................................................................ 8

       1.4.6 Social Event ....................................................................................................... 9

   1.5 Existing Applications ................................................................................................. 9

       1.5.1 Web-based Application: thesponsorshipcompany.in .................................. 9

       1.5.2 Social Networking Applications like Facebook, Twitter ......................... 10
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.6</td>
<td>Google Cloud Messaging for Push Notifications</td>
<td>10</td>
</tr>
<tr>
<td>2.</td>
<td>Narrative</td>
<td>12</td>
</tr>
<tr>
<td>2.1</td>
<td>Problem Statement</td>
<td>12</td>
</tr>
<tr>
<td>2.2</td>
<td>Motivation</td>
<td>12</td>
</tr>
<tr>
<td>2.3</td>
<td>Product Scope</td>
<td>13</td>
</tr>
<tr>
<td>2.4</td>
<td>Functionalities of the Product</td>
<td>13</td>
</tr>
<tr>
<td>3.</td>
<td>System Design</td>
<td>15</td>
</tr>
<tr>
<td>3.1</td>
<td>Architecture of Sponsorship Application</td>
<td>15</td>
</tr>
<tr>
<td>3.2</td>
<td>Use Case Diagram</td>
<td>16</td>
</tr>
<tr>
<td>3.3</td>
<td>Class Diagram</td>
<td>18</td>
</tr>
<tr>
<td>3.4</td>
<td>Sequence Diagram</td>
<td>20</td>
</tr>
<tr>
<td>3.5</td>
<td>User Interface</td>
<td>21</td>
</tr>
<tr>
<td>3.5.1</td>
<td>User Login and Registration Screen</td>
<td>22</td>
</tr>
<tr>
<td>3.5.2</td>
<td>User Home and Dashboard Screen</td>
<td>23</td>
</tr>
<tr>
<td>3.5.3</td>
<td>Event Creation Screen</td>
<td>24</td>
</tr>
<tr>
<td>3.5.4</td>
<td>Upload Document Feature</td>
<td>25</td>
</tr>
<tr>
<td>3.5.5</td>
<td>View all Posts Screen</td>
<td>26</td>
</tr>
<tr>
<td>3.5.6</td>
<td>My Posts Screen: Editing</td>
<td>27</td>
</tr>
<tr>
<td>3.5.7</td>
<td>Adding Sponsor Names and Amount or Quantity</td>
<td>29</td>
</tr>
</tbody>
</table>
3.5.8 Sponsor Login Screen and Facebook Authentication Screen .......................... 29
3.5.9 Sponsor Home and Dashboard Screen ..................................................... 31
3.5.10 View all Posts Detailed Screen .............................................................. 32
3.5.11 Willing to Support Feature for Sponsor ................................................ 33
3.5.12 Event Filter ............................................................................................ 34
3.5.13 Send Message Activity Screens .............................................................. 35
3.5.14 Settings .................................................................................................. 36
3.5.15 Notifications .......................................................................................... 37
3.5.16 Share the Events .................................................................................... 38
3.5.17 Sponsorship Type other than, Money Feature .......................................... 40
3.5.18 Guest Screens ........................................................................................ 41

4. Implementation of Application Modules ....................................................... 42

4.1 Integration of Facebook and Gmail Login .................................................... 42
4.2 Setting Permissions in AndroidManifest.xml File ........................................ 42
4.3 Facebook and Gmail Login Integration logic .............................................. 43
4.4 Google Cloud Messaging Integration ......................................................... 45
4.5 Setting Permissions in AndroidManifest.xml file for GCM Integration ....... 46
4.6 Registering the GCM ID ............................................................................. 46
4.7 Sending Message to GCM .......................................................................... 48
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.8</td>
<td>Receiving Message from GCM</td>
<td>48</td>
</tr>
<tr>
<td>5.</td>
<td>Testing and Evaluation</td>
<td>50</td>
</tr>
<tr>
<td>5.1</td>
<td>Launching the Application</td>
<td>50</td>
</tr>
<tr>
<td>5.2</td>
<td>Registration</td>
<td>51</td>
</tr>
<tr>
<td>5.3</td>
<td>Login</td>
<td>53</td>
</tr>
<tr>
<td>5.4</td>
<td>Facebook Login Authentication</td>
<td>54</td>
</tr>
<tr>
<td>5.5</td>
<td>Event Creation</td>
<td>56</td>
</tr>
<tr>
<td>5.6</td>
<td>Adding participating Colleges and attending VIP’s</td>
<td>57</td>
</tr>
<tr>
<td>5.7</td>
<td>Event Editing</td>
<td>58</td>
</tr>
<tr>
<td>5.8</td>
<td>Event Deleting</td>
<td>60</td>
</tr>
<tr>
<td>6.</td>
<td>Conclusion and Future Work</td>
<td>61</td>
</tr>
<tr>
<td>6.1</td>
<td>Future Work</td>
<td>61</td>
</tr>
<tr>
<td>7.</td>
<td>Bibliography and References</td>
<td>63</td>
</tr>
</tbody>
</table>
LIST OF FIGURES

Figure 1.1 Time Spent on Mobile, Desktop, TV and Other Connected Devices Usage from 2010-2015 .................................................................01

Figure 1.2 Most Popular Devices ..................................................................02

Figure 1.3 Current Number of Applications in leading App Stores ............05

Figure 1.4 Application Functionality with Google Cloud Messaging ..........11

Figure 3.1 Architecture of Sponsor Application ......................................15

Figure 3.2 Use Case Diagram for Sponsorship Application ......................17

Figure 3.3 Class Diagram for Sponsorship Application ..........................19

Figure 3.4 Sequence Diagram of User/Sponsor Interaction with the Application 20

Figure 3.5 Home Screen of Sponsorship App ....................................21

Figure 3.6 User Login and Registration Screens ....................................22

Figure 3.7 User Home and Dashboard Screens .....................................23

Figure 3.8 College Event Creation Screen ...........................................24

Figure 3.9 Upload Document Screen .......................................................25

Figure 3.10 View All Posts Screen .............................................................26

Figure 3.11 Detailed View of Event and Download Logo Screen ...............27

Figure 3.12 User Post with Edit and Delete Option ..................................28

Figure 3.13 User Event Editing Screen .....................................................28
Figure 3.14 Adding Sponsors Names and amount or quantity Screen ..........29
Figure 3.15 Sponsor Login Screen and Facebook Authentication Screen........30
Figure 3.16 Sponsor Home Screen and Dashboard Screen ..........................31
Figure 3.17 Sponsor Detailed view of Event Screen ................................32
Figure 3.18 Willing to Support Feature for Sponsor ................................33
Figure 3.19 Event Filter Screen .................................................................34
Figure 3.20 Message Activity Screen ......................................................35
Figure 3.21 User Settings Screen .............................................................36
Figure 3.22 Notifications Screen ..............................................................37
Figure 3.23 Share Button Action in Sponsorship Application .....................38
Figure 3.24 Sharing the Event in Google Plus ..........................................39
Figure 3.25 Event Shared in Google Plus .................................................39
Figure 3.26 Sponsorship Type other than, Money Feature ..........................40
Figure 3.27 Guest Home Screen ..............................................................41
Figure 4.1 Code Snippet for Adding Libraries to build.gradle file ..........42
Figure 4.2 Code Snippet for including Facebook id in AndroidManifest.xml ....42
Figure 4.3 Code Snippet for Facebook Login Authentication ....................43
Figure 4.4 Code Snippet for Gmail Login Authentication ..........................43
Figure 4.5 Code Snippet for Requesting PHP file at AWS .......................44
Figure 4.6 Code snippet to get and Register Facebook Details in Database. .......44

Figure 4.7 Creating Google Cloud Messaging System Application ID ...............45

Figure 4.8 Code Snippet for Setting Permission for GCM in AndroidManifest ...46

Figure 4.9 Code Snippet to Register the Device with GCM ID ..................47

Figure 4.10 Code Snippet to Register the User Id with GCM ID..................47

Figure 4.11 Code Snippet for sending Message to GCM ..........................48

Figure 4.12 Code Snippet for declaring Class file in AndroidManifest.xml ......48

Figure 4.13 Code Snippet to Receive Message from GCM ..........................49

Figure 5.1 Application Home Screen ..........................................................50

Figure 5.2 Registration Page Negative Test Case with Empty Fields ............51

Figure 5.3 Registration Page given with Existing Email id as input, Negative Test Case with Empty Fields ..........................................................52

Figure 5.4 Registration Positive Test Case ....................................................52

Figure 5.5 User Login Negative Test Case ....................................................53

Figure 5.6 User Dashboard Screen ...............................................................54

Figure 5.7 Facebook Login Negative Test Case ............................................55

Figure 5.8 User Dashboard Screen ...............................................................55

Figure 5.9 Event Creation Negative Test Cases ............................................56

Figure 5.10 Event Creation Positive Test Case .............................................57
Figure 5.11 Adding Participating Colleges and VIP’s to an Event .....................58

Figure 5.12 Event Editing Negative Test Case ..............................................59

Figure 5.13 Event Editing Positive Test Case ...............................................59

Figure 5.14: Event Deleting Positive Test Case ...........................................60
LIST OF TABLES

Table 1.1 Android Versions and Releases .................................................................3
1. BACKGROUND AND RATIONALE

In contemporary society, people find difficulty in searching for a sponsor, because of their busy schedules and it’s hard to find the sponsors and request them to sponsoring an event. Marketing and promotion is the key to success in business. Many companies and organizations are searching for events to sponsor which can help promote their name. It’s hard for the companies to schedule a meeting with each and every individual who asks for sponsorships. Both public and organizations needs an easy to use medium to communicate and make the task simple. Apart from companies and organizations there are others like charity societies, students, colleges, schools, and orphanages etc., who search for sponsorships for their events. In addition, sponsorships can be of many forms such as money, food, clothes, and toys for Christmas etc.

In today’s world smart phones are used as an alternative to personal computers as they are always handy, rather than going to a website it is easy to download the application software and install it on a smart phone. Figure 1.1 shows mobile desktop/laptop, Television and other connected devices usage and time spent on respective devices from 2010 to 2015.

![Figure 1.1: Time Spent on Mobile, Desktop, TV and Other Connected Devices Usage from 2010-2015](image)
Smartphones are used for much more than calling, texting, or basic internet browsing. Users are turning to these mobile devices as they navigate a wide range of life events: look up information about a health condition, online banking, real estate listings or other information about a place to live, search job, government services or information, to take an online class or get educational content and entertainment etc.

There are many ways to access internet to get these life events connected such as Laptop, Smartphone, Tablet, Games Console, smart devices like smart TV, watches etc. Figure 1.2 shows the most popular devices used to search the internet.

![Most Popular Devices](https://example.com/image)

**Figure 1.2: Most Popular Devices [1]**

Smartphones are going to dominate the future. Many operating systems are available to these smart phones, out of them the leading ones are Google Android, Apple
iOS, Windows Phone etc. This sponsorship app is developed for phones running on android operating system.

1.1 Android

Android is mobile operating system (OS) based on the Linux kernel and currently developed by Google. Android was primarily built for digital cameras but then realized that there is not much market for the digital cameras [2]. Later the Android operating system was built for smartphones, smart glasses, smart homes, cars, game consoles, watches, computers, tablets etc. Android creates an open market for wide variety of applications such as games, entertainment, education etc. Android provides a world class platform for developing apps and games for Android users everywhere, as well as an open market place for distributing them instantly. Most of the applications are developed using programming languages like Java, PHP and databases such as MySQL etc. This application is developed using android Java. The app interacts with server running PHP and MySQL as the database. The sponsorship application is compatible with minimum SDK version of Android 4.0 (Ice Cream Sandwich) to recent update Android 6.0 (Marshmallow).

The various versions of the Android operating systems are listed in Table 1.1:

<table>
<thead>
<tr>
<th>S. No</th>
<th>Name</th>
<th>Version</th>
<th>Release Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Android</td>
<td>1.0</td>
<td>September 2008</td>
</tr>
<tr>
<td>2</td>
<td>Android</td>
<td>1.1</td>
<td>February 2009</td>
</tr>
<tr>
<td>3</td>
<td>Cupcake</td>
<td>1.5</td>
<td>April 2009</td>
</tr>
<tr>
<td>4</td>
<td>Donut</td>
<td>1.6</td>
<td>September 2009</td>
</tr>
<tr>
<td>5</td>
<td>Éclair</td>
<td>2.0–2.1</td>
<td>October 2009</td>
</tr>
<tr>
<td>6</td>
<td>Froyo</td>
<td>2.2–2.2.3</td>
<td>May 2010</td>
</tr>
<tr>
<td>7</td>
<td>Gingerbread</td>
<td>2.3–2.3.7</td>
<td>December 2010</td>
</tr>
<tr>
<td>8</td>
<td>Honeycomb</td>
<td>3.0–3.2.6</td>
<td>February 2011</td>
</tr>
<tr>
<td>9</td>
<td>Ice Cream Sandwich</td>
<td>4.0–4.0.4</td>
<td>October 2011</td>
</tr>
<tr>
<td>10</td>
<td>Jelly Bean</td>
<td>4.1–4.3.1</td>
<td>July 2012</td>
</tr>
</tbody>
</table>
Most android applications are developed using Android software development kit SDK and the java programming language that has access to the Android APIs. The android studio IDE is used to develop the application.

1.2 Android Studio

Android studio is an integrated development environment for developing android applications. It is an open source software available under the Apache License 2.0. Android Studio is the official IDE for Android application development, based on IntelliJ Idea. On top of the capabilities offered by IntelliJ, Android Studio offers [4]:

- Flexible Gradle-based build system.
  - Gradle is an open source build automation system. Gradle can automate the building, testing, publishing and deployment of an application.
- Build variants and multiple apk file generation.
- Code templates such as application templates, activity templates and object templates to assist with building common app features.
- Rich layout editor with support for drag and drop theme editing.
- Lint tools to catch performance, usability, version compatibility, and other problems.
- Pro Guard and app-signing capabilities.
- Built-in support for Google Cloud Platform, making it easy to integrate Google Cloud Messaging and App Engine.

1.3 Facebook and Gmail Login Integration

Most users do not show interest in registering or creating the accounts whenever a new application comes in to the market. A recent survey conducted by statista.com, Figure 1.3 shows us current number of applications in leading app stores as of July 2015
in Apple App store, Google Play, Amazon App store, Windows Phone store and BlackBerry World. It becomes very difficult for the users to create an account whenever a new application comes to the market and also hard to remember all applications login credentials.

![Figure 1.3: Current Number of Applications in Leading App Stores](image)

**Figure 1.3: Current Number of Applications in Leading App Stores [5]**

Market leader Facebook has 1 billion user base, with 900,000,000 - estimated unique monthly visitors and Gmail has 120,000,000 - estimated unique monthly visitors [6], this application supports login to the application using both the Facebook and Google credentials. Sponsorship application makes login to the application simple by integrating Facebook login to the application, where users can easily login in to the application using their Facebook credentials without registering a new account. Similarly they can login using their Google credentials. After users login to the application for first time they are asked to enter few other details like email, phone for contact information purposes etc.
1.4 List of Events

Sponsorship is an Android app which sets up an online platform for those looking for sponsorship for an event and also those look to sponsor an event. This app allows the users to post their sponsorship requirements regarding the events. The Sponsors who are looking to provide some kind of sponsorship can login into the app and view the posts regarding the events and can respond to the post by contacting the respective candidate or organization. This app provides a medium for various kinds of organizations, businesses, schools, students, charities, social events etc., to meet the public and there by promoting their organizations. Sponsorship application supports various kinds of events, which allows the user to create and post an event in the respective category. The list of events that are currently integrated in Sponsorship application are

- Business Event
- College Event
- Orphanage Event
- Proposal Listing
- Student Scholarship
- Social Event

The user can create a new event by selecting the appropriate event category. The user can then enter the information like title of the event, location of the event etc. Each and every event is discussed in detail in 1.4.1-1.4.6 sections.

1.4.1 Business Event

In today’s world there are many companies or organizations looking forward to collaborate with leading companies in their respective field. They need a platform where they can find a source to generate and create the relationship with other companies or organizations. Either these companies want to join with the other competitors or they want to create an event where they can invite others to join. These companies need a platform to post their event information online, where this event can be searched by others. Sponsorship application allows the users to create an event in the business event category, Sponsors can directly lookup this event in the business event category.
Business event contains a registration form to register the event with fields existing data such as person name, event title, organization name, event location, event information, participating organizations, VIP’s attending, sponsorship type, event start date, event end date, and event documents. Here sponsorship type refers to money, food, and infrastructure etc., Participating organizations refers to the list of participating organizations, VIP’s attending refers to the list of VIP’s attending the event.

1.4.2 College Event

As a general scenario, many of the clubs and departments at school organize events. Whenever a club is organizing an event they normally look for sponsors to sponsor the event. These clubs or departments need a platform to post their event online, where this event can be searched by sponsors. Clubs at school looks for various types of sponsorship such as food, money, infrastructure, clothes like T-shirts related to the event etc. where the clubs wants to enter the number of participants for the event and not the amount of funds in some cases. In addition companies or organization look forward to provide sponsorship which can help promote their name or product by creating a booth at the event etc.

Sponsorship application allows the students to create the event in College Event category and sponsors can directly search in college event category. College event contains a registration form to register the event with fields such as person name, event title, college name, event location, event information, participating colleges, VIP’s attending, sponsorship type, event start date, event end date, and event documents. Here sponsorship type refers to money, food, and infrastructure etc., participating colleges refers to the list of participating colleges in the event.

1.4.3 Orphanage Event

An Orphanage is a residential institution devoted to the care of orphans, where they typically are in need of various things. Sponsorship application allows the user or head of the particular orphanage to create a post in orphanage event category where the application allows to register the post by filling a registration form related to the event. The type of sponsorship can be foe food, clothes or funds.
Sponsorship application allows the users to create the event in orphanage event category and sponsors can directly search in orphanage event category. Orphanage event contains a registration form to register the event with fields such as person name, event title, orphanage name, orphanage location, event information, sponsorship type, orphanage capacity, other sponsors, event start date, event end date, and event documents.

1.4.4 Proposal Listing Event

Proposal listing event are events inspired by a person or group of individuals for a good cause or for social awareness or for any other reasons. Here the user or the person is looking for two things: searching for other co-organizers to organize the event as he alone cannot organize the event and also looking for sponsorship for the event.

Sponsorship application allows the users to create the event in proposal listing event category and sponsors or other users can directly search in the proposal listing event category either to join the event or to provide sponsorship. Proposal listing event contains a registration form to register the event with fields such as person name, contact information, event type, event title, event location, event information, no of coordinators, coordinators type, sponsorship type, event start date, event end date, and event documents. Here sponsorship type refers to funds, food, and infrastructure etc. event type refers to the type of event such as musical, social, sports or other. Coordinators type refers to the type of coordinator such as student, organization other etc. where the person is looking to organize an event.

1.4.5 Student Scholarship Event

Nowadays we observe the huge increase in school fees, where the student finds difficulty in paying the fees. Scholarships, on campus jobs like Teaching Assistants, Graduate Assistants are some means to help them, but there are only limited scholarships and the institution cannot support all students. So student needs a platform to post his request for external scholarship, where someone can look the post and can provide scholarship if interested. In addition student can upload his documents like transcripts, essay etc. which provides more information about the student.
Student scholarship event contains a registration form to register the event with fields such as person name, person email, person phone, student qualification, post title, post information, total fee, sponsorship amount, and student documents.

1.4.6 Social Events

Social Events are events organized by a group of people related to a community, religion, and small societies. These people organize events like festivals related to their religion, musical events or some kind of sports events. Sponsorship application allows the users to post the event in social event category where they can select the type of event they are going to organize, which makes it easy for sponsors to follow the event and to provide the required sponsorship.

Social Event contains a registration form to register the event with fields such as person name, person email, person phone, event type, event title, event information, participating sponsors, event start date, event end date, sponsorship type, event location, and event documents.

1.5 Existing Applications

Some of the existing applications in current market are discussed in following sections.

1.5.1 Web-based Application: thesponsorshipcompany.in [7].

- Doesn’t have an option to view current ongoing events without logging in to the website.
- There is no option to specify the category of event while posting the event or searching.
- No push notifications for users and sponsors.
- There is no option for students to post about scholarship they are looking for.
- No Facebook or Gmail login integration, users need to create an account.
1.5.2 Social Networking Applications like Facebook, Twitter

Social networking applications such as Twitter and Facebook can be used to post the events but it’s hard to find them by the sponsors because of irrelevant data such as too many posts related to entertainment, related to news etc. It’s really difficult to search for a particular category of the event even searching using location or name.

1.6 Google Cloud Messaging for Push Notifications

Google Cloud Messaging (GCM) is an administration that empowers designers to send information from servers to both Android applications or Chrome applications and expansions. Push notifications enables users to be notified of incoming messages. If sponsor responds to an event the user will be notified. In addition, if user commented on the post then whoever involved with the event; that means all sponsors and users will get notified about the respective comment. In order to avail these services Sponsorship application is integrated with Google Cloud Messaging services. The web server used for this Sponsorship application is Amazon Web Server in which an instance is created which supports PHP and MySQL database. Figure 1.4 shows the functionality of application along with Google Cloud Messaging System.
Figure 1.4: Application Functionality with Google Cloud Messaging [8].
2. NARRATIVE

2.1 Problem Statement

In the current Android market, there are few social networking applications such as Twitter and Facebook where the users can post their event requirements but it’s hard for sponsors to find them because of a large number of irrelevant posts by other users on social networking sites. There are few websites that connects the both sponsors and the event organizers such as thesponsorshipcompany.in etc., which has few disadvantages as discussed in existing applications section 1.5.1. There exists no dedicated application developed for the event organizers and for the sponsors with different category of events such as social, orphanage, business, student, college, proposal listing. Many companies and organizations looking forward to provide sponsorships on different category of events. It’s hard for the sponsors to keep track of all the events in the society. Even if the sponsor knows about the event, it’s hard to track the status of the event such as amount funded, organizations that are sponsored, VIP’s attending the event, and other related information in the posted event. In addition, there is no application available for students who can post their requests regarding the scholarships. Sponsorship application is developed to provide solutions to the above problems for both users and sponsors.

2.2 Motivation

Firstly, there is no dedicated platform for people who are looking for sponsors and also for sponsors looking forward to provide sponsorship. Secondly, in today’s competitive world small organizations are in search for marketing their products in order to compete with large organizations. Sponsorship is one of the sources to promote their products. Even if a sponsor got an event to provide the sponsorship there isn’t a way to discuss the requirements of the event without contacting the person. Moreover it’s hard to get the latest updates about the event. These are some of the difficulties that motivated the development of an application that includes all functionalities in to a single application and thereby providing the users best experience.
2.3 Product Scope

Technology has advanced at a rapid pace in the last two decades, and what once was futuristic and difficult to imagine is now a part of our everyday lives. Especially smartphones play a crucial role in one’s daily life.

The popular mobile operating systems for smartphones as per 2015 market are
1. Apple’s iOS
2. Google’s Android
3. Microsoft’s Window’s
4. RIM Blackberry.

Out of all these operating systems, Android operating system is most popular and widely used operating system with over billion users according to a 2015 survey[9]. Developing this project in Android operating system finds a large user base in today’s world. Users should connect to internet in order to avail the service of this application because communication between event managers and sponsors is done through a cloud server. This application is compatible with minimum SDK version of Android 4.0 (Ice Cream Sandwich) to recent update Android 6.0 (Marshmallow).

2.4 Functionalities of the Project

This project presents a Sponsorship application in android platform. The following functionalities are provided by the app.

1. A platform which supports users to post their events and for sponsor’s to look for events and get in touch with the event organizers.
2. Application for students, colleges, business organizations, orphanages etc. to post their events and related information.
3. Ability to create universities social events.
4. Users and Sponsors can login into the application by registering their account or they can login using the registered accounts of Facebook and Gmail.
5. Users and Sponsors can filter the type of events while searching or posting the event such as college events, student scholarships, business events, Orphanage events, proposal listing and social events.
6. Sponsors and user get push notifications about the event they are involved with.
7. Sponsors can provide their comments on events like “I’m interested in this event” or “I want to sponsor this event” etc.
8. Sponsors can look for the amount of sponsorship funded and amount needs to be funded for a particular event.
9. Sponsorship is not all about money, so it can be in any form like clothes, food, infrastructure etc. which are supported by this application.
10. User can upload documents such as brochure, flyer and logo’s etc. and Sponsor can download the documents that provides more information on the event.
3. SYSTEM DESIGN

3.1 Architecture of Sponsorship Application

Figure 3.1 depicts the architecture of Sponsorship application.

![Figure 3.1 Architecture of Sponsorship Application](image)

In order to avail the services of Sponsorship application, the application should be downloaded and installed on both users and sponsors android mobiles. Users and sponsors initially have to login into the application. Account verification and validation, posting the events, upload documents, commenting on posts, event editing or deleting etc. requests are sent to the Amazon web server for processing and the required data is stored in the database. Accounts information and the posts related to users are stored in tables at the database. Users or Sponsors requested data is processed and fetched through Amazon web server from the database. Here Amazon web server acts as intermediate for the interactions between both the user and sponsor. In order to get push notifications services for both users and sponsors Sponsorship application is integrated with Google Cloud.
Messaging service. Both users and sponsors should be connected to internet in order to avail the services of Sponsorship application.

3.2 Use Case Diagram

Figure 3.2 depicts the use case diagram for the Sponsorship application. In the use case diagram all ovals represents the activities involved between both user and sponsor. Registration use case deals with the registration functionality for the sponsorship application. Account Login use case provides the user to login to the application with credentials given at Registration phase, the Facebook and Gmail Login use case activity deals with login functionality of the application for both sponsor and user. Once User login to the application Event Creation use case activity deals with the creation of events like College, Business, Student Scholarship, Social, Orphanage and Proposal Listing. User can also edit or delete the events created earlier with the help of Edit or Delete Events functionality. Both the user and sponsor can view all events and contact information related to user and whoever post comments to the event with the help of View Events and View Contact Information use cases.

Both user and sponsor can provide their comments to different category of events with the functionality of Message/Comments use case. Whenever a new comment is posted for particular event then the user respective to that event gets notified about the comment with the help of Push Notifications use case. Guest users can directly view the events and contact information of users who created the event.
Figure 3.2: Use Case Diagram for Sponsorship Application
3.3 Class Diagram

The Class Diagram for the Sponsorship application consists of interfaces, methods, variables and relationship between them. Figure 3.3 shows class diagram for the user and sponsor with few classes which contains different functionalities such as registration, authentication, Facebook authentication, Gmail authentication, user and sponsor dashboards, creating events, inserting comments, loading comments, push notifications. Both Facebook and Gmail logins are identical to the user and sponsor but the interface and functionalities vary based on their roles after login. Dashboard screen is common to both user, sponsor and guest, but content displayed in the dashboard varies with their roles after login.
Figure 3.3: Class Diagram of Sponsorship Application
3.4. Sequence Diagram

Figure 3.4 depicts the sequence diagram that shows the request and response messages between user/sponsor, Sponsorship application, Amazon Web Server and the database. Initially request is sent from either user or sponsor with the help of Sponsorship application, then the requests are authenticated at AWS for example requests such as login requests and requested data is sent back from the database in the form of responses.

![Sequence Diagram of User/Sponsor Interaction with the Application](image.png)

Figure 3.4: Sequence Diagram of User/Sponsor Interaction with the Application
3.5. User Interface

The main activities in Sponsor application are *Home Screen, User Login Screen* and *Sponsor Login screen*. Once the user or sponsor login to the application, the landing screens for user are *User Home screen, User Dashboard, Event creation, Message Typing Screen etc.* and for sponsor are *Sponsor Home screen, Sponsor Dashboard Screen, Filtering Events Screen and Message Typing Screen etc.* Figure 3.5 depicts the *Home Screen* for the Sponsorship application. From this screen the user, sponsor and guest can navigate to other screens.

![Sponsorship Application](image)

**Figure 3.5: Home Screen of Sponsorship App**
3.5.1. User Login and Registration Screens

Figure 3.6(a) depicts the User Login screen, where the user has various options to get logged into the application such as through the registered account, Facebook and Gmail.

Figure 3.6(b) depicts the User Registration screen which allows the user to sign up for a new account by entering the details such as Gender, First Name, Last Name, Address, city, state, country, zip code, Occupation, Phone, Email, Username, Password, Confirm Password and then the user is registered to the application.

Figure 3.6: User Login and Registration Screens
3.5.2. User Home and Dashboard Screen

When a user successfully logs in to the application, then user will be navigated to the User Home Screen as shown in Figure 3.7(a). Here the user is presented with functionality for creating new events in six different categories.

Figure 3.7(b) shows the User dashboard that consists of tabs *Events, My Posts, All Posts, Settings, Logout and About Us.*

![User Home and Dashboard Screens](image-url)
3.5.3. Event Creation Screen

When user clicks on college event shown in Figure 3.7(a) then user is navigated to college event creation screen shown in Figure 3.8. The user can then enter the information about the event such as *event title*, *event information*, *VIP’s attending*, *participating colleges*, *amount of sponsorship required*, *location of the event* etc. and upload documents related to event such as event logo or brochure. On clicking the create button the event is created and posted it to the database.

![College Event Creation Screen](image)

**Figure 3.8: College Event Creation Screen**
3.5.4. Upload Document Feature

Sponsorship application allows the user to upload the documents related to the event such as event logo or brochure. The user can upload an image, pdf and word file at the time of event creation by clicking on the attach file symbol on the event creation screen. Figure 3.9 shows the upload document feature of the application.

![Upload Document Screen](image)

*Figure 3.9: Upload Document Screen*
3.5.5. View All Posts Screen

Figure 3.10 shows the screen that contains all posted events with brief description i.e. all categories of events created by users. The filter option on the page can be used to filter the events based on a category.

![View All Posts Screen](image)

**Figure 3.10: View All Posts Screen**

When the user clicks on any event shown in Figure 3.10 a new screen that contains the detailed description for the event is displayed as shown in figure 3.11(a). When user clicks on any document that is related to particular event, they will be navigated to a screen from where the documents can be downloaded by clicking on the download button as shown in Figure 3.11(b).
3.5.6. My Posts Screen: Editing, Deleting

When the user clicks on any event shown in Figure 3.10 a new screen that contains the detailed description for the event is displayed as shown in figure 3.12, which contains the edit, delete and share buttons.

Figure 3.13 shows the screen for editing the created event, where the user can edit previously created event data by clicking the edit button. Similarly user can delete the created event by clicking on the delete button.
Figure 3.12: User Post with Edit and Delete Option

Figure 3.13: User Event Editing Screen
3.5.7. Adding Sponsors Names and Amount or Quantity

Whenever a user gets partial or complete sponsorship from the sponsor, the user can update the sponsor name and the sponsorship received by clicking on Add Sponsor Details button on the event. Once user clicks the button it will show a dialog box that contains Amount or Quantity and Sponsor name field. Quantity refers to the number of participants. Once the user enters name and amount or quantity, then event will be displayed with the name and sponsorship provided. Figure 3.14 shows this feature of application.

![Figure 3.14: Adding Sponsors Names and amount or quantity Screen](image)
3.5.8. Sponsor Login Screen and Facebook Authentication Screen

Figure 3.15(a) depicts the Sponsor Login screen, where the sponsor is provided with Facebook and Gmail login options.

Figure 3.15(b) depicts the Sponsor Facebook authentication screen where the user can enter the Facebook login credentials.

Figure 3.15: Sponsor Login and Facebook Authentication Screen
3.5.9. Sponsor Home and Dashboard Screen

When a sponsor successfully logs in to the application, the sponsor will be navigated to the sponsor home screen as shown in Figure 3.16(a) containing the list of events created by users.

Figure 3.16(b) shows the sponsor dashboard that consists of tabs *All Posts, Settings, Logout and About Us.*

![Sponsor Home Screen](image1)

![Sponsor Dashboard Screen](image2)

**Figure 3.16: Sponsor Home Screen and Dashboard Screen**
3.5.10. View All Posts Detailed Screen

When a sponsor clicks on any event in Figure 3.16(a) he will be navigated to screen which contains detail description of the event as shown in Figure 3.17. When sponsor clicks on any document that was related to particular event for example event logo or event flyer, he will be navigated to the screen where user can download the document using download button as shown in Figure 3.11(b).

![Event Information](image)

**Figure 3.17: Sponsor Detailed View of Event Screen**
3.5.11. Willing to Support Feature for Sponsor

Sponsorship application allows the sponsors to enter sponsorship they are looking forward to provide for a particular event by clicking on *willing to support* option on the right side top corner on the event as shown in Figure 3.17. When user clicks the *willing to support* option a dialog box is displayed for the sponsor to enter the sponsorship details as shown in Figure 3.18.

![Figure 3.18: Willing to Support Feature for Sponsor](image)
3.5.12. Event Filter

Both users and sponsors can filter the list of events based on particular category of event with the help of filter option provided on the top right corner of the screen in event display screen as shown in Figure 3.16(a). Whenever user clicks the filter icon it will display a dialog box containing all event categories Figure 3.19 shows the event filter screen.

![Event Filter Screen](image)

Figure 3.19: Event Filter Screen
3.5.13. Send Message Activity Screens

When a sponsor clicks on any event in Figure 3.16(a) he will be navigated to screen which contains detail description of the event along Type your message option at bottom of the screen as shown in Figure 3.17. Sponsorship application supports the user and sponsor to provide their comments for events using Send Message option. Figure 3.20 (a) and (b) shows the messaging activity screens.

Figure 3.20: Message Activity Screens
3.5.14. Settings

Figure 3.16(b) shows the sponsor dashboard which consists of tabs *All Posts, Settings, Logout* and *About Us*. When a user clicks on settings tab, he can filter the events either by creation time or event time as shown in Figure 3.21. This option is similar to both user and sponsor.

![Figure 3.21: Users Settings Screen](image)
3.5.15. Notifications

The application allows the user and sponsor to send messages or comments using the events created. Whenever a comment is posted by the sponsor then the user posting the event should be notified about the comment and similarly sponsor should be notified if the user provides a comment. Figure 3.22 (a) and (b) shows the push notifications coming in.

![Figure 3.22: Notifications Screen](image-url)
3.5.16. Share the Events

Sponsorship application allows the user to share the event on social networking applications such as Google plus, WhatsApp, Gmail etc. Both sponsor and user can share the event on social networking sites using the share button in the Sponsorship application. Figure 3.23 shows the share button action and Figure 3.24 and Figure 3.25 shows the event sharing in Google plus.

![Share button action in Sponsorship Application](image)

Figure 3.23: Share button action in Sponsorship Application
Figure 3.24: Sharing the Event in Google Plus

Figure 3.25: Event Shared in Google Plus
3.5.17 Sponsorship Type other than, Money Feature

Figure 3.26(a) shows the feature of selecting money, clothes, infrastructure etc. from drop down menu when creating an event. For example, Figure 3.26(b) shows the event which requests for Clothes/T-shirts.

Figure 3.26: Sponsorship Type other than, Money Feature
3.5.18. Guest Screens

Figure 3.27 depicts the Guest screen of the Sponsorship application. Guest users do not have an option such as *willing to support* or *send message*. They can simply view the events and contact details related to the event along with share button to share the event on Google plus, WhatsApp and Gmail etc.

![Guest Home Screen](image)

Figure 3.27: Guest Home Screen
4. IMPLEMENTATION OF APPLICATION MODULES

4.1 Integration of Facebook and Gmail Login

Sponsorship application is integrated with three different kinds of logins, in order to avail the full services of the application. User account login where the user is first registered with the application and then can login to the application using the login credentials provided at registration. In addition to account login by registration user and sponsor can also login to the application using registered accounts such as Facebook and Gmail.

In order to integrate the Facebook and Gmail login functionality, the libraries shown in Figure 4.1 should be included in build.gradle file. Client side code is written in Android Studio using Java, and http request calls are made to PHP using apache library on Amazon Web Server (AWS). In order make http requests to Amazon Web Server an instance is created which in turn provides a URL to make http requests as shown in Figure 4.1. In build.gradle file: the statement shown in Figure 4.1 are added.

```groovy
compile 'com.google.android.gms:play-services:8.1.0'
compile 'com.facebook.android:facebook-android-sdk:4.7.0'
compile 'org.apache.httpcomponents:httpclient:4.2.3'
```

URL required to make Http Requests to Amazon Web Server is

```java
```

Figure 4.1: Code Snippet for Adding Libraries to build.gradle file

4.2 Setting Permissions in AndroidManifest.xml file

All permission are set in AndroidManifest.xml file, in order to support Facebook login functionality in the application Facebook developer accounts should be created which in turn provides a Facebook id. This Facebook id should be added to AndroidManifest.xml as shown in Figure 4.2 which authenticates the application by Facebook.

```xml
<meta-data
    android:name="com.facebook.sdk.ApplicationId"
    android:value="@string/facebook_application_id" />
```

Figure 4.2: Code Snippet for including Facebook id to AndroidManifest.xml

42
4.3 Facebook and Gmail Login Integration Logic

When user clicks the Facebook or Gmail login buttons the code shown in Figure 4.3 is executed where `facebookRegisterCallback()` method is called which on successful login returns `email`, `Facebook id and name` as shown in Figure 4.3 and Figure 4.4 represents the code for Gmail authentication.

```java
public LoginButton facebookRegisterCallback(LoginButton login_button, CallbackManager callbackManager, final BaseActivity activity) {
    login_button.setReadPermissions(Arrays.asList("public_profile, email"));

    login_button.registerCallback(callbackManager, new FacebookCallback<LoginResult>() {
        public void onSuccess(LoginResult loginResult) {
            GraphRequest.newMeRequest(
                loginResult.getAccessToken(), new
                GraphRequest.GraphJSONObjectCallback() {
                    public void onCompleted(JSONObject jsonObject, GraphResponse response) {
                        String email = jsonObject.optString("email");
                        String id = jsonObject.optString("id");
                        String name = jsonObject.optString("name");

                        new FBLoginTaskRunner(id, email, name, activity).execute(null, null, null);
                    }
                }).executeAsync();
        }
    });
}
```

**Figure 4.3:** Code Snippet for Facebook Login Authentication

```java
public void onConnected(Bundle arg0) {
    System.out.println("called on connected");
    try {
        if (Plus.PeopleApi.getCurrentPerson(mGoogleApiClient) != null) {
            Person currentPerson = Plus.PeopleApi.getCurrentPerson(mGoogleApiClient);
            String name = currentPerson.getDisplayName();
            String email = Plus.AccountApi.getAccountName(mGoogleApiClient);
            new GmailLoginTaskRunner(email, name, this).execute(null, null, null);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

**Figure 4.4:** Code Snippet for Gmail Login Authentication
After getting the values such as email, Facebook id and name from Facebook, the values are sent to AWS using Apache Http Request call which contains the data, respective PHP file and URL, as shown in Figure 4.5

```java
private class FBLoginTaskRunner extends AsyncTask<Void, Void, Void> {
    protected Void doInBackground(Void... params) {
        UserLoginJSON login = new UserLoginJSON();
        jsonArray = login.loginUserFB(facebookid, email, userType, name);
        return null;
    }

    public JSONArray loginUserFB(String facebookid, String email, String userType, String name) {
        final ArrayList<NameValuePair> postdata = new ArrayList<NameValuePair>();
        postdata.add(new BasicNameValuePair("facebookid", facebookid));
        postdata.add(new BasicNameValuePair("name", name));
        postdata.add(new BasicNameValuePair("usertype", userType));
        postdata.add(new BasicNameValuePair("email", email));
        GetJsonFromUrl getjson = new GetJsonFromUrl();
        JSONArray jsonarray = getjson.makeHttpRequest(postdata, "facebook_registration.php");
        return jsonarray;
    }
}
```

**Figure 4.5: Code Snippet for requesting PHP file at AWS**

The facebook_registration.php implements the functionality to receive the request. It checks the Facebook registration table in the database for Facebook id that is sent by Http Request. If it returns false then it registers the user in the database and also adds the Facebook id to Facebook registration table, as shown in Figure 4.6. Similarly on successful Gmail login returns email id, name is be added to Gmail registration table.

```php
$result = $loginDao->getUserProfileByFBId($facebookid);
if($result==false){
    $affected_rows = $registrationsDao->addUser('', $name, '', '', '', '', '', '', '', '', '', $email, '', '', '', '', $usertype);
    if ($affected_rows == 1) {
        $row_array["success"] = "1";
        $row_array["userid"] = mysql_insert_id();
        $registrationsDao->addfbId($facebookid, $row_array["userid"]);
        $row_array["email"] = '';
        $row_array["role"] = $usertype;
        $row_array["phone"] = '';
        $row_array["name"] = $name;
    } else {
        $row_array["success"] = "0";
        $row_array["failurereason"] = "Registration failed. Try again";}
}
```

**Figure 4.6: Code snippet to get and Register Facebook Details in Database**
4.4 Google Cloud Messaging Integration

Sponsorship application is integrated with Google Cloud Messaging to provide push notification functionality to users and sponsors. Whenever user or sponsor comments on an event, the user who created that respective event is notified about the comment. In addition whenever user replies to any comment, sponsors are notified. Google Cloud Messaging (GCM) is service that empowers designers to send information from servers to both Android applications or Chrome applications and extensions. In order to avail the services of Google Cloud Messaging system, we need to create an application id as shown in Figure 4.7 which registers the application with GCM using Google Developer Console.

This application id has to be integrated in the Sponsorship application,
Public static final String SENDER_ID = "938157410150";

Figure 4.7: Creating Google Cloud Messaging System Application ID
4.5 Setting Permissions in AndroidManifest.xml file for GCM Integration

Figure 4.8 shows all the permissions that has to be set in AndroidManifest.xml file for integrating Google Cloud Messaging

```xml
<permission
    android:name="com.app.sponsorship.permission.C2D_MESSAGE"
    android:protectionLevel="signature" />

<uses-permission
    android:name="com.app.sponsorship.permission.C2D_MESSAGE" />
<uses-permission
    android:name="com.google.android.c2dm.permission.RECEIVE" />
<android:uses-permission
    android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<android:uses-permission
    android:name="android.permission.READ_PHONE_STATE" />
<android:uses-permission
    android:name="android.permission.READ_EXTERNAL_STORAGE" />
<receiver
    android:name="com.google.android.gms.gcm.GcmReceiver"
    android:exported="true"
    android:permission="com.google.android.c2dm.permission.SEND">
    <intent-filter>
        <action android:name="com.google.android.c2dm.intent.RECEIVE" />
        <category android:name="com.example.gcm" />
    </intent-filter>
</receiver>
```

**Figure 4.8: Code Snippet for Setting Permission for GCM in Android Manifest file**

4.6 Registering the GCM ID

In order to send the push notification to either user android device or sponsor android device, first the application needs to register the device with a unique id called GCMID. Figure 4.9 shows the code snippet to registers the device with GCMID
private void registerInBackground() { new AsyncTask<Void, Void, Void>()
    {
        protected Void doInBackground(Void... params) {
            String msg = "";
            try {
                if (gcm == null) {
                    gcm = GoogleCloudMessaging.getInstance(context);
                }
                regId = gcm.register(Constants.SENDER_ID);
                Log.d("RegisterActivity", "registerInBackground -
                regId: " + regId);
                msg = "Device registered, registration ID=" + regId;
                storeRegistrationId(context, regId);
            }
            catch (IOException ex) {
                msg = "Error :" + ex.getMessage();
                Log.d("RegisterActivity", "Error: " + msg);
            }
            Log.d("RegisterActivity", "AsyncTask completed: " + msg);
            return null;
        }
    }

    Figure 4.9: Code Snippet to register the device with GCM ID

    Once the application is registered with GCMID, the user or sponsor need to be
    registered along with that GCMID. The code snippet shown in Figure 4.10 will register
    the userid with GCMID.

    private class GCMregistrationTaskRunner extends AsyncTask<Void, Void, Void> {
        private String gcmid;
        private String userid;

        public GCMregistrationTaskRunner(String gcmid, String
        userid, Activity activity) {
            this.gcmid = gcmid;
            this.userid = userid;
        }
        protected Void doInBackground(Void... params) {
            GcmJSON gcm = new GcmJSON();
            gcm.registergcm(gcmid, userid);
            return null;
        }
    }

    Figure 4.10: Code Snippet to Register the User Id with GCM ID
4.7 Sending Message to GCM

Figure 4.11 shows the code snippet for sending the message to the Google Cloud Messaging system. Whenever a new comment is inserted, this code will be executed which will send message, userid and GCMID to the Google Cloud Messaging system using the URL.

```php
$url = 'https://android.googleapis.com/gcm/send';
$fields = array(  
    'registration_ids' => $registration_ids,  
    'data' => $message  
);
$headers = array(  
    'Authorization: key=' . GOOGLE_API_KEY,  
    'Content-Type: application/json'
);

$ch = curl_init();  
// Set the url, number of POST vars, POST data
curl_setopt($ch, CURLOPT_URL, $url);
curl_setopt($ch, CURLOPT_POST, true);
curl_setopt($ch, CURLOPT_HTTPHEADER, $headers);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);  
// Disabling SSL Certificate support temporarily
curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
curl_setopt($ch, CURLOPT_POSTFIELDS, json_encode($fields));

$result = curl_exec($ch);
if ($result === FALSE) {
    die('Curl failed: ' . curl_error($ch));
}
```

Figure 4.11: Code Snippet for sending Message to GCM

4.8 Receiving Message from GCM

Once the message is sent to the Google Cloud Messaging system it sends the notification to the respective registered GCMID. In order to receive the notification sent by GCM, the respective intent service should be invoked to receive the notification. This service is declared in AndroidManifest.xml file, Figure 4.12 shows the service that is declared in the manifest file.

```xml
<service android:name=".GCMIntentService" >
    <intent-filter>
        <action android:name="com.google.android.c2dm.intent.RECEIVE" />
    </intent-filter>
</service>
```

Figure 4.12: Code snippet for declaring class file in AndroidManifest.xml

48
The `GCMIntentService` class shown in Figure 4.13 is invoked whenever a new notification arrives.

```java
protected void onHandleIntent(Intent intent) {
    String notificationText = intent.getExtras().getString("text");
    Notification notification = new NotificationCompat.Builder(getApplicationContext())
        .setContentTitle("Sponsorship")
        .setContentText(notificationText)
        .setTicker("Notification!")
        .setWhen(System.currentTimeMillis())
        .setDefaults(Notification.DEFAULT_SOUND)
        .setAutoCancel(true)
        .setSmallIcon(R.drawable.ic_launcher)
        .build();

    NotificationManager mNotificationManager;
    mNotificationManager = (NotificationManager)
        getSystemService(Context.NOTIFICATION_SERVICE);
    Intent notificationIntent = new Intent(this, DetailedEventDisplayActivity.class);
    notificationIntent.putExtra("eventid", intent.getExtras().getString("eventid");
    notificationIntent.setFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP | Intent.FLAG_ACTIVITY_SINGLE_TOP);
    PendingIntent intent1 = PendingIntent.getActivity(this, 0, notificationIntent, 0);
    notification.flags |= Notification.FLAG_AUTO_CANCEL;
    notification.contentIntent = intent1;
    mNotificationManager.notify(MY_NOTIFICATION_ID, notification);
```

**Figure 4.13: Code Snippet to Receive Message from GCM**
5. TESTING AND EVALUATION

This chapter deals with the functional evaluation of the Sponsorship Application. This application is tested using Nexus 7 tablet running Android Version 4.4.4. Sponsorship application is compatible with minimum SDK version of Android 4.0 (Ice Cream Sandwich) to recent update Android 6.0 (Marshmallow). In order to avail the services of the application the device should be connected to the Internet. To install android application package on Nexus 7 the option “install from other locations” should be enabled in device settings.

Each module of the application is tested with Positive and Negative test cases, where positive test case uses valid input and negative test case uses invalid input.

5.1. Launching the Application

Figure 5.1 depicts the launching screen or Home screen of the application, whenever user clicks on the Sponsorship application icon on the android device.

![Figure 5.1: Application Home Screen](image)
5.2. Registration

a. **Negative Test Case:** When a user clicks on *Register* button without entering any information in the given registration fields, the application displays toast message “Please fix errors” along with indicating the mandatory fields with error symbol shown in Figure 5.2. If the user enters existing email or username then the application displays a dialog box with Email already used or username already used. Figure 5.3 shows the test case of entering an existing email id.

![Registration Page Negative Test Case with Empty Fields](image_url)

Figure 5.2: Registration Page Negative Test Case with Empty Fields
Figure 5.3. Registration Page given with existing email id, Negative Test Cases

b. Positive Test Case: Figure 5.4 shows a dialog displaying User Registered Successfully on successful registration.

Figure 5.4. Registration Positive Test Case
5.3. Login

a. Negative Test Case: When a user clicks on Login button without entering the values in the given registration fields then the application shows the error symbol on username and password fields. If the user enters invalid inputs then the application shows a dialog box with “Please enter valid credentials”. Figure 5.5 shows the test case with invalid user login credentials.

![User Login Screen](image)

**Figure 5.5: User Login Negative Test Case**

b. Positive Test Case: Figure 5.6 shows the User dashboard screen on successful login with valid login credentials.
5.4. Facebook Login Authentication

a. **Negative Test Case:** When a user clicks on *Login* button with invalid inputs then the application shows an error with “Please enter valid credentials”. Figure 5.7 shows the test case with invalid login credentials.
b. **Positive Test Case:** Figure 5.8 shows the User dashboard screen on successful login with valid Facebook login credentials.
5.5. Event Creation

a. Negative Test Case: When a user clicks on CREATE button without entering the values in the given event details fields the application displays the error symbols at respective fields. The application also displays a dialog box with an error message for the first mandatory field in error. Figure 5.9 shows the test case with invalid input.

Figure 5.9: Event Creation Negative Test Cases

b. Positive Test Case: Figure 5.10 shows the User dashboard screen on successful event creation with valid inputs. A toast message “Event created” is displayed on user dashboard screen.
5.6. Adding Participating Colleges and Attending VIP’s.

Most of the events have features and extra fields like adding VIP’s. The user can enter the VIP’s names list if some VIP is attending the event. Most of the companies looks for these VIP’s where the event becomes a platform to meet them. For example, college event has attending VIP’s and participating colleges. User can enter the participating college’s names in the panel and can also delete the added names. Figure 5.11 (a) and (b) shows these features.
5.7. Event Editing

a. Negative Test Case: When a user clicks on Update button with invalid inputs the application shows an error with “Field cannot be Empty”. Figure 5.12 shows the test case with invalid input.
b. **Positive Test Case:** Figure 5.13 shows the toast message “Event updated successfully” on successful updating with valid inputs.
5.8. Event Deleting

a) **Positive Test Case:** Figure 5.14 shows the toast message “Deleting Event”, when user clicks delete button.

![Figure 5.14: Event Deleting Positive Test Case](image-url)
6. CONCLUSION AND FUTURE WORK

Sponsorship application creates a platform for both users and sponsors, where they can post or view the events. It also saves the time of both users and sponsors searching for each other. In addition it can also be useful for orphanages, students, colleges and other entities to post information regarding what is required for them. The functionalities supported by this application:

- An android application with full functionalities that make users and sponsors easy to find themselves and interact with each other.
- A platform which supports users to post their events and for sponsor’s to look for events and get in touch with the event organizers.
- Users and Sponsors can login into the application by registering their account or they can login using the registered accounts of Facebook and Gmail.
- Application for students, colleges, business organizations, orphanages etc. to post their events and related information.
- An application where users and sponsors get notified frequently about the event they are involved with.
- Users and Sponsors can filter the type of events while searching or posting the event such as college events, student scholarships, business events, Orphanage events, proposal listing and social events.
- Sponsors can look for the amount of sponsorship funded and amount needs to be funded for a particular event.

6.1 Future Work

The application can be improved by adding the following functionalities such as:

- A Follow button to make it easy for sponsors to follow or track the event.
- Deleting fraud sponsors by user reporting method, where the user reports about the sponsor. Once admin receives a significant number of reports against particular sponsor the admin can take action against the respective sponsor.
- A payment method integration where the sponsor can directly provide sponsorship amount.
• Adding more events other than included ones such as Sports etc.
• Integration of this application with existing University apps (or) organization apps.
• A search bar for searching the events based on location, time etc.
• Providing security for the application from hackers, fraud users and against SQL injection attacks etc.
• Testing the application with third party testing applications like Nessus, Blue box etc.
BIBLIOGRAPHY and REFERENCES


