

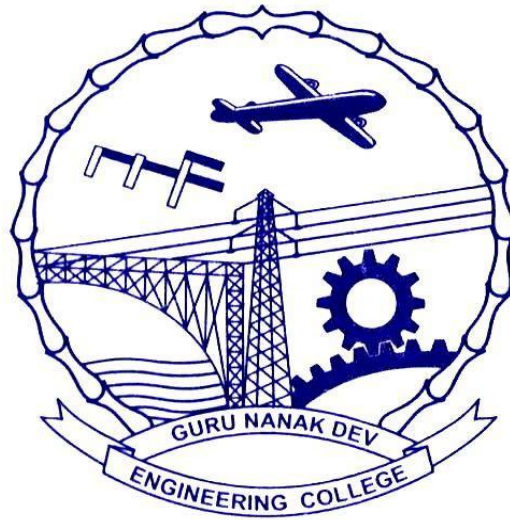
EVENTO-GNDEC Events Guide

Submitted for the partial fulfilment of the Degree

of

Bachelor of Technology

(Computer Science Engineering)



Submitted By:

Aashita Dutta

1243588

Submitted to:

Sukhjit Singh Sehra

Training Co-ordinator

CSE Department

Department of Computer Science & Engineering

Guru Nanak Dev Engineering College

Ludhiana 141006

Company Certificate



Abstract

Evento Application helps you access various events held in our college on your phone. This app brings about the information of events held by various societies of our college where event administrator can create events and students can join events, eliminating the need of manual registration on desk.

The events are an incredible part of students' co-curricular and extra co-curricular activities. Handling day-to-day events can be easier through this App where administrator can easily post event and students are notified about scheduling of various events through email service. Also students can suggest events to administrators which provide help in all way, reducing overall burden on Event Handlers of various societies.

In this way, Evento Application serves as communication link between participants and event creators, which helps in encouraging more participation. This eliminates the need to keep a separate list of participants which is hard to maintain due to the rapid movement of students

Acknowledgements

The satisfaction that accompanies the successful completion of any task would be incomplete without mentioning the people whose ceaseless cooperation made it possible, whose constant guidance and encouragement crowned all efforts with success.

I am highly grateful to Dr. M.S.Saini (Director, Guru Nanak Dev Engineering College, Ludhiana), for providing this opportunity to carry out the six weeks Industrial Training.

The constant guidance and encouragement received from Dr. K.S.Mann Dean Training & Placement Cell, GNDEC Ludhiana has been of great help in carrying out the project work and is acknowledged with reverential thanks.

I would like to express deep debt to Mr. Shwet Kumar (Project Guide, Aptech, Ludhiana), for his vital suggestions, meticulous guidance and constant motivation which went a long way in the successful completion of this project.

I would like to express gratitude to other faculty members of Aptech and of CSE department of GNDEC for their intellectual support throughout the course of this work.

And finally, my deepest gratitude to my team members, Nihar Kanwar (D4 CSE) for her unfailing support and innovative attitude towards the project.

Special thanks to other teachers and seniors for their mentoring.

Table of Contents

S. no.	Contents	Page no.
1.	Definitions, Acronyms and Abbreviations	Vi
2.	List of figures	Vii
3.	List of tables	Viii
4.	[Chapter-1] Introduction to company	1
5.	[Chapter-2] Introduction to project [2.1] Overview [2.2] Existing System [2.3] User Requirement Analysis [2.4] Feasibility Study [2.5] Objectives of Project	2-9
6.	[Chapter-3] Project design [3.1] Product Perspective [3.2] Product Functions [3.3] User Characteristics [3.4] Constraints [3.5] Use Case Model [3.6] Database Design [3.7] Table Structure [3.8] ER Diagrams [3.9] Assumptions and Dependencies [3.10] Specific Requirements	10-28
7.	[Chapter-4] Development and implementation [4.1] Introduction to Languages (Front End and Back End) [4.2] Any other supporting languages	29-51

	[4.3] Implementation with Screen Shots/ Figures [4.4] Testing	
8.	[Chapter-5] Conclusion and future scope [5.1] Conclusion [5.2] Future Scope	52-53
9.	References	54
10.	Appendix	55

Definitions, Acronyms and Abbreviations

1. App: Application
2. Admin: Administrator
3. IDE: Integrated development environment
4. UI: User interface
5. BLOB:(Binary Large Object) It is used to store binary data of any kind.
6. XML: Extensible Markup language.
7. SQL: Structured Query Language

List of Figures

1. Aptech logo.....	1
2. Registration Page layout.....	10
3. Login page layout.....	11
4. Main dashboard.....	11
5. Create event page layout.....	12
6. Use case model for User.....	18
7. Use case model for Admin.....	19
8. ER Diagram.....	23
9. Landing page.....	36
10. Home page.....	37
11. Sign up page.....	38
12. Login page.....	38
13. Dashboard page.....	39
14. Create event page.....	40
15. Event description page for admin.....	40
16. Event description page for user.....	41
17. Join event page.....	41
18. Send email page.....	42
19. Show users page.....	42
20. Navigation bar for users.....	43
21. Navigation bar for admin.....	43

22. User profile page.....	51
23. About app page.....	52
24. Event suggestion page.....	53
25. Society list page.....	54
26. Email validation.....	55
27. Mobile number validation.....	56
28. Reset password page.....	57

List of Tables

1. Registration
2. Timeline
3. User table
4. Image table
5. Admin

[Chapter-1] Introduction to company

Aptech Limited is a global education and training company headquartered in Mumbai, India. In 1993, Aptech became Asia's first IT education and training organization to receive the ISO 9001:2000 quality certification for Education Support Services. The company has its stock listed on the BSE and the NSE. In 2014, Aptech was selected as a finalist for Microsoft Learning Partner of the Year award.

Aptech has over 2600 "centre of learning" in more than 40 emerging countries through its two main streams of business – individual training and enterprise business.



[Chapter-2]Introduction to Project

[2.1]Overview

Evento Application helps you access events' notification on your phone. This app brings about the information of different events held by various societies of our college where admin can create events and students can join events, eliminating the need of manual registration on desk.

The events are an incredible part of students' co-curricular and extra co-curricular activities. Handling day-to-day events can be easier with this electronic notice and announcement system where admin can easily post event and notification alerts are sent out notifying students about scheduling of various events through email service. Also students can suggest events to admin which provide help in all way, reducing overall burden on Event Handlers of various societies.

In this way, Evento Application serves as communication link between participants and event creators, which helps in encouraging more participation. This eliminates the need to keep a separate list of participants which is hard to maintain due to the rapid movement of students.

These are the features that an Evento Application should have:

- An electronic dashboard board for disseminating information out to students about creation and scheduling of events.
- Events can be joined, with notice obtained instantly.
- Students can be notified of new events via email service.
- Event administrator may push important event notices in to selected student's email.
- Event administrator may create/modify/delete any event category with prior notification.

The interface of this application is straightforward and takes you roughly a minute to get started. Adding events to board is easy, just click on the create event button and enter the text. Users can view the posted event on the spot by having an email and notification alert in android phone. Here registration is must for all the users having this application in order they want to have notification and staying tuned.

This application can also acts as a suggestion service and lets you suggest events to administrator Thus, innovation is the main objective of this application. This will help in time utilization for students

and administrators as well. Students can join events on the spot if they wish to participate in that event and registration procedure can be efficient through this. Also administrator is notified when user joins particular event and can view the list of joined users.

This application also facilitates the administrators to post images of various events, so that makes event notice board more interactive though event image that can be the center of attraction. Also students can keep the track of events organized by various societies that have been already held in the college campus.

[2.2]Existing System

Currently our college has manual system of event notification via announcements and putting pamphlets, posters etc. on notice board. It's outdated now, as nobody has a time to stand in rush in order to read various events' posters on noticeboard. Also those students who are doing registration before starting of event seems confused, tensed while maintaining list on paper.

Limitations of Existing System:

1. Complexity: Automated system is less complex than manual system of handling notices, which can make it easier for untrained people to access and manipulate data. Anyone having the basic knowledge of mobiles can work on the automated system.
2. Inconsistency of data: There will be unavailability for future use, since events might get misplaced during manual event management. So events won't be preserved properly for future use.
3. Damage: Manual events stack are vulnerable to damage, destruction and theft in ways that digital databases are not. A company may back up its digital data both on site and at offsite locations, ensuring its security if the office building suffered a fire or similar disaster. A manual database, however, may only exist in one place without any copies. As a result, a manual database would be very vulnerable to a fire or other natural disaster. In addition, while access time in a manual database system, information must be found by hand rather than electronically. While a digital database will typically allow users to search the entire database for specific information in seconds, someone looking for information in a manual system may have to spend hours searching for a

particular piece of data.

4. **Editing and Communication:** Manual event management system do not allow users to easily edit data or information. Manually events often cannot be edited directly, forcing users to make new copies. For pamphlets or charts users must require students. Evento app allow users to edit information fields directly, and because data is stored digitally, it is already in a form that can be easily transmitted.
5. **Order of Events:** Events can get out of order in traditional event management system. If someone accidentally schedulesome event in the wrongly, it can lead to lost data. Automated notice management systems allow users to quickly check whether information already exists somewhere in the system, which helps avoid problems like redundant data.
6. **Inflexibility:** Manual Event Management System causes inflexibility for students doing registration on desk before any event commences. This causes burden on students and creates confusion to maintain list of participating students and it becomes headache when number of students come on desk creating chaos.

[2.3]User Requirement Analysis

A Software Requirement Analysis for a software system is a complete description of the behavior of a system to be developed keeping users' needs in mind. It includes a set of use cases that describe all the interactions the users will have with the software. In addition to use cases, the SRS also contains functional and non-functional requirements. Non-functional requirements are those which impose constraints on the design of implementation.

Purpose:Evento Application is used for managing events and the main purpose of this project is to:

1. Utilize time through faster dissemination of notices regarding education, technical events, cultural events and fun events.
2. Help you to be updated which events going to be held in our College.
3. Maintain list of participating students.

4. Eliminate the need of class announcements.
5. Bring about close interaction between users and administrators.
6. Suggest events to various societies' administrators.
7. Make event handling efficient and interactive.
8. Create/delete/modify events by administrators.
9. Provide facility to users who can join particular event associated with society prior to event notification.
10. Reduce the dependencies and increasing the transparency.

General Description:Evento Application is basically designed to create events and notifying the users about scheduling of events electronically. Also users can join events and administrators are notified about users joining particular event. Also this application is common to all societies. So to make these tasks easy with all functions performed quickly, this electronic way to manage event notifications will be quite helpful.

Administrator will be the super user of the application who will configure system information such as creating new events, and their information dissemination via email service, or editing or deleting the old events, managing users who joined event.

This Evento Application will be a mobile application, open source and user friendly. This application is designed on the basis of android application architecture. In this application, SQLite database will be used to store data related to events, users, users joining events, event images etc. This application is meant for both administrators and users. It is developed using Android, XML.

Users Of The System

1. Administrator: Administrator can add, update or delete events' details and can manage users who joined events and manage images related to events. He can notify registered users about new events or any updates in existing events through email service. He can also delete old events no longer in use, and keeps track of old events.
2. Client: Clients are the end users that benefits from this android application. A client can get all information of all the existed events and the future events as well and thus they can apply for the same. They can also view their profile as well as can check the track of their events that they have joined through email inbox. They can also suggest events to administrators.

Functional Requirements

Specific Requirements: This phase covers the whole requirements for the system. After understanding the system we need the input data to the system then we watch the output and determine whether the output from the system is according to our requirements or not. So this phase explains what we have to input and then what we'll get as output. It also describes the software and non-function requirements of the system.

Input Requirements of the System:

1. Client Details
2. Admin Login Credentials
3. Event Details
4. Joined User Details

Output Requirements of the System:

1. Interface for administrator to configure the system.
2. Interface for clients.
3. An electronic dashboard board for disseminating information out to clients.
4. Notification about events to clients via email service.
5. Notification to administrators about joining of users.
6. Users' profile.
7. Automatic email generation notifying clients about events' upgrades and their registration into application.

Specific User Requirements:

1. Automatic Email Generation to all clients who register into application and administrator can send emails to all as well as to concerned person about events.

Software Requirements:

1. Programming Language: Java
2. Web Languages: XML
3. Database: SQLite
4. IDE: Android Studio
5. Operating System: Android

Non-Functional Requirements:

1. Scalability: System should be able to handle a number of users. For example, handling around thousand users at the same time.
2. Usability: Simple User Interface that a layman can understand.
3. Speed: Speed of the system should be responsive i.e. Response to a particular action should be available in short period of time. For example, Automatic email service, Notification to all clients about events.

[2.4]Feasibility Study

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

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

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

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

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

1. Technical Feasibility: The technical issue usually raised during the feasibility stage of the investigation includes the following :

- Does the necessary technology exist to do what is suggested?
- Do the proposed equipments have the technical capacity to hold the data required to use the new system?
- Will the proposed system provide adequate response to inquiries, regardless of the number or location of users?
- Can the system be upgraded if developed?
- Are there technical guarantees of accuracy, reliability, ease of access and data security?

The current system developed is technically feasible. It is a web based user interface. Thus it provides an easy access to the users. The databases purpose is to create, establish and maintain a work- flow among various entities in order to facilitate all concerned users in their various capacities or roles. Permission to the users would be granted based on the roles specified. Therefore, it provides the technical guarantee of accuracy, reliability and security. The software and hardware requirements for the development of this project are not many and are already available as free as open source. The work for the project is done with the current equipment and existing software technology.

2. Operational Feasibility: Under this category of service we conduct a study to analysis and determine whether your need can be fulfilled by using a proposed solution. The result of our operational feasibility Study will clearly outline that the solution proposed for your business is operationally workable and conveniently solves your problems under consideration after the proposal is implemented. We would precisely describe how the system will interact with the systems and persons around.

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

[2.5]Objectives of Project

The proposed system's objectives are to overcome all the limitations and drawbacks of the existing system. The online Evento Application is user-friendly android application. The main objective of the application is its simplicity of design and ease of implementation that shows and helps to collect most of the information about events going on in college premises and allowing users to join events eliminating the need of manually registration. The interface will be very user-friendly.

The main objectives of the proposed system can be enumerated as follows:

- Faster dissemination of notices regarding education, technical events, cultural events and fun events.
- Easy way to broadcast your message.
- Helps you to be updated which events going to be held in our College.
- Good way to maintain list of participating students.
- Eliminating the need of class announcements.
- Close interaction between users and administrators.
- Suggesting events to various societies' administrators.

[Chapter-3]Product Design

[3.1]Product Perspective

The software will be a new independent product, that it, it is not a component of another program. It is intended for the administration of the event management and other concerned users. This application is mobile application.

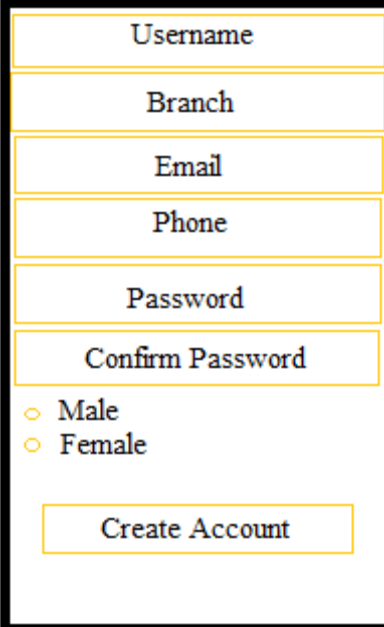
System Interface:

Any Android phone.

User Interface:

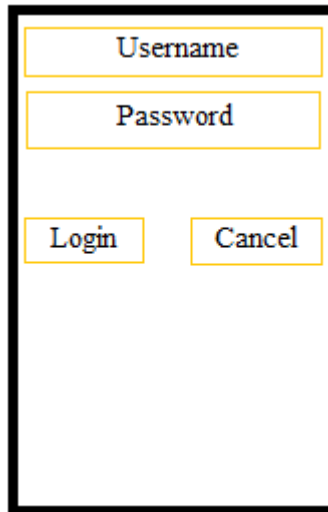
User Interface Design means the design of application with which the user interacts. So it should be kept in mind that UI should be very simple and easy to use. It should be simple enough in look and feel also.

Following screens will be provided:



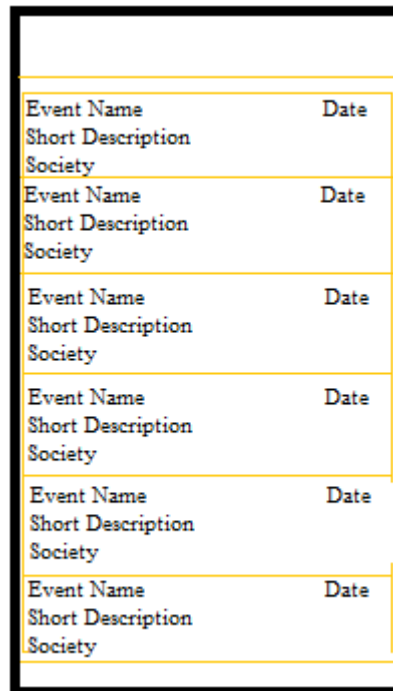
Username
Branch
Email
Phone
Password
Confirm Password
<input type="radio"/> Male <input type="radio"/> Female
Create Account

1. This is the registration page where the user gets itself registered with this application.



A login form with a black border. It contains two input fields: 'Username' and 'Password'. Below these fields are two buttons: 'Login' and 'Cancel'.

2. This is the login page where user enters his username and password in order to access the notices.



Event Name	Date
Short Description Society	
Event Name	Date
Short Description Society	
Event Name	Date
Short Description Society	
Event Name	Date
Short Description Society	
Event Name	Date
Short Description Society	

3. This is the main dashboard, which receives all the notices from the admin. The user can click on any notice, to see its details. In addition to it, admin can search the events.

Event Name
Description
Short Description
Society
Date
Venue
Prize
Organizers
Contacts
Upload Image
Create Event

4. This page is only for the admin. to create events.

Hardware Interface:

Evento is intended as a mobile application for the Android platform and hence is solely supported on Android-powered devices or it can run on AVDs i.e Emulator running on computer. Notification, updates, and data exchanged between Android devices. This produces the illusion of peer-to-peer interactivity between administrators and students. Evento is being developed on Android 4.3 (JellyBean) and can be operated on all versions released after it. Email will be sent using SMTP protocol. The Android platform provides abstractions for all network communication interfaces and thus the hardware as well.

Software Interface:

Operating System: Windows 7

Tools: Android Studio

Platform: Android SDK Framework

Android Emulator: Minimum SDK version = “8” and target SDK version = “18”

Database: SQLite

Memory Constraints:

APK file of Evento Application would require about 5 MB and on Emulator it requires around 12 MB.

[3.2]Product functions

The functions of this application are as follow:

1. Registering a User:

The first step in this application is to get the users registered with the application. For this, user will provide all the necessary details and press the Create Account button. After that, all the information along with registration id is stored in the SQLite Database and the user gets registered.

2. User Login:

After registering, the user is allowed to log in for authentication. After authentication, user gets logged in.

3. Viewing the Events:

After user is logged in, he/she is taken to dashboard of events where all the existed and future events are displayed with its name, date, short description and the society in listview. By clicking on particular event, user is directed to detail page of that event and there he/she has option to join event.

4. Searching a Notice:

The user is able to search the notice in listview depending on the title of the notice. It helps user to get the desired notice instantly.

5. Deleting a Notice:

If the administrator does not want some notices, he/she can delete it from their phone and thus deleted from database too.

6. Posting an Event:

If a user is an admin, he is able to post the event. In order to post the events, he has two options. One option is that, he can post a simple text event. Another option allows him to send some attachment image with the event. In this, he can pick the image from the gallery. After that, press the Create Event button to post the event.

7. Reset Password:

This application also has the facility to reset the password. If one user has forgot his password, he/she can reset the password by giving his email address. The user will be given a page in which he can set his new password. Forgotten password will be updated with the new one in the database.

8. Email Service:

As soon as the user creates an account in this application, user receives an automatic email consisting of username and password by administrator notifying about his/her successful registration. In case he/she forgets password, it can be recovered by email they received. Also automatically email is generated and sent to users notifying about creating and upgrading of events by administrator, and when user updates his profile and successfully joins an event. User has to send email to admin if they want to join an event, so that admin can keep track of the list of events through email service.

Additional features of the system:

9. User Logout:

Before closing the app, whether user is admin or student, he/she has to logout from the application due to security reasons.

10. User Profile:

User can view his/her profile and can update the profile.

11. About App:

User can view about this application and the details how to run this application safely. All the

guidelines are given in About App widget in Navigation Drawer.

12. About Societies:

User can view all the societies that are there in our college and the history about societies, its history, and current status and progress so far as well.

13. Show Users:

If user is admin, then he can view the list of users who has joined the events of any society.

14. Event suggestion:

User can suggest events to administrator by filling in details of events and sending email to admin, so that they can review it and thus it helps student to be in proximity with administrator.

[3.3]User characteristics

We have identified five potential classifications of users of our system:

- Software Designers: These are the most obvious users. They will use the system as a means of laying out the design of an as yet unimplemented system.
- Software Developers: These are the people that take the model generated by the designers and implement it in code. They may also use the system to identify the design of an existing system in order to maintain it.
- Quality Engineers: These users are usually responsible for ensuring that a design is feasible and/or reliable. They will therefore also need to be looking at the output from our system/
- System Administrator: Due to the client/server/concurrent nature of the system, somebody needs to be responsible for security and maintenance of the system. This is the System Administrators role. The Administrator of the system, project or model need not be a member of the any of the other roles identified here.
- The Client: More often than not, software is designed for a client. The client may wish to see the design as laid out by the system and be able to see what exactly they are buying.

[3.3.1]The General User

All users can be assumed to have the following characteristics:

- Ability to read and understand English.
- Familiarity with the operation of the basic android devices and its User Interface.

[3.3.2] The Software Designer

The Software Designer users can be assumed to have the following characteristics:

- A thorough knowledge of object oriented design notation.
- Familiarity with common design environments. For example, the concept of layouts and views in android applications. Software Developers User Characteristics.

[3.3.3] The Quality Engineer

The Quality Engineer users can be assumed to have the following characteristics:

- A thorough knowledge of object oriented design notation.

[3.3.4] The Administrator

The Administrator users must know how to create event, delete event, update event and send emails to registered students on this Evento Application. Also administrators can manage the users who have joined any event. And any number of administrators can manage these events irrespective of their societies as this application is common to all societies.

[3.4] Constraints

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

User Password Constraint:

The application should check the user and password fields before logging in. It should check whether the fields are filled or not. If fields are not filled up, user should be instructed to fill up the fields before moving further.

Constraints During Registration:

There are a lot of constraints that needs to be implemented in the application. They are as follow:

1. Username:

The username can contain only alphabets, digits, underscore and hyphen. It should be at least 3 characters long and maximum of 15 characters. Also first letter of first should be in uppercase. Also a toast notification is displayed if username already exists in database.

2. Password:

The password must contains one digit from 0-9, one lowercase character, one uppercase character, one special symbols in the list ”#\$%” and length of password must be at least 6 characters and maximum of 20. Also a toast notification is displayed if email already exists in database.

3. Email:

The application must validate and email address entered by the user for receiving email automatically.

4. Mobile Number:

The mobile number should be of only ten digits. No more, no less than that.

4. Constraints During Creating Event:

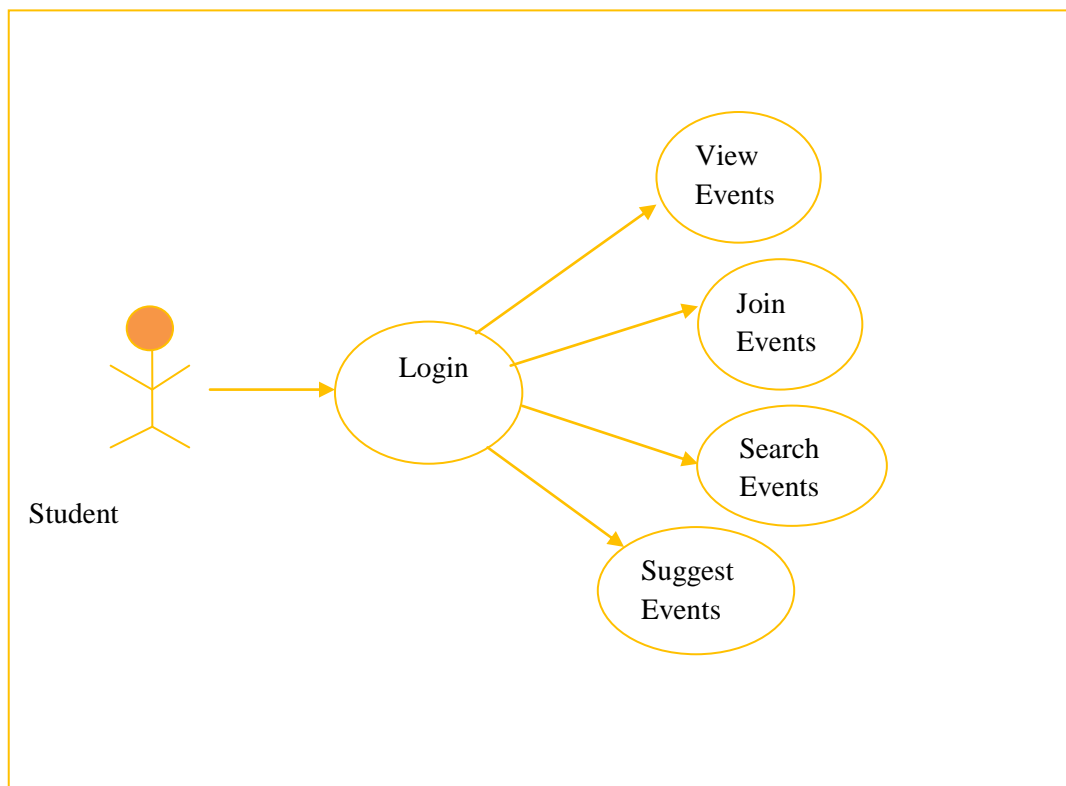
The application should validate the event creation fields before creating any event. All the fields has first letter in capitals It should check whether event name, description, short description, society, date, venue, organizers, contacts and prize fields are filled or not. If not, toast is displayed to fill up the required fields while creating the event.

5. Reset Password Validation:

The application should check that user has entered the correct email in the given filed before pressing the reset password button.

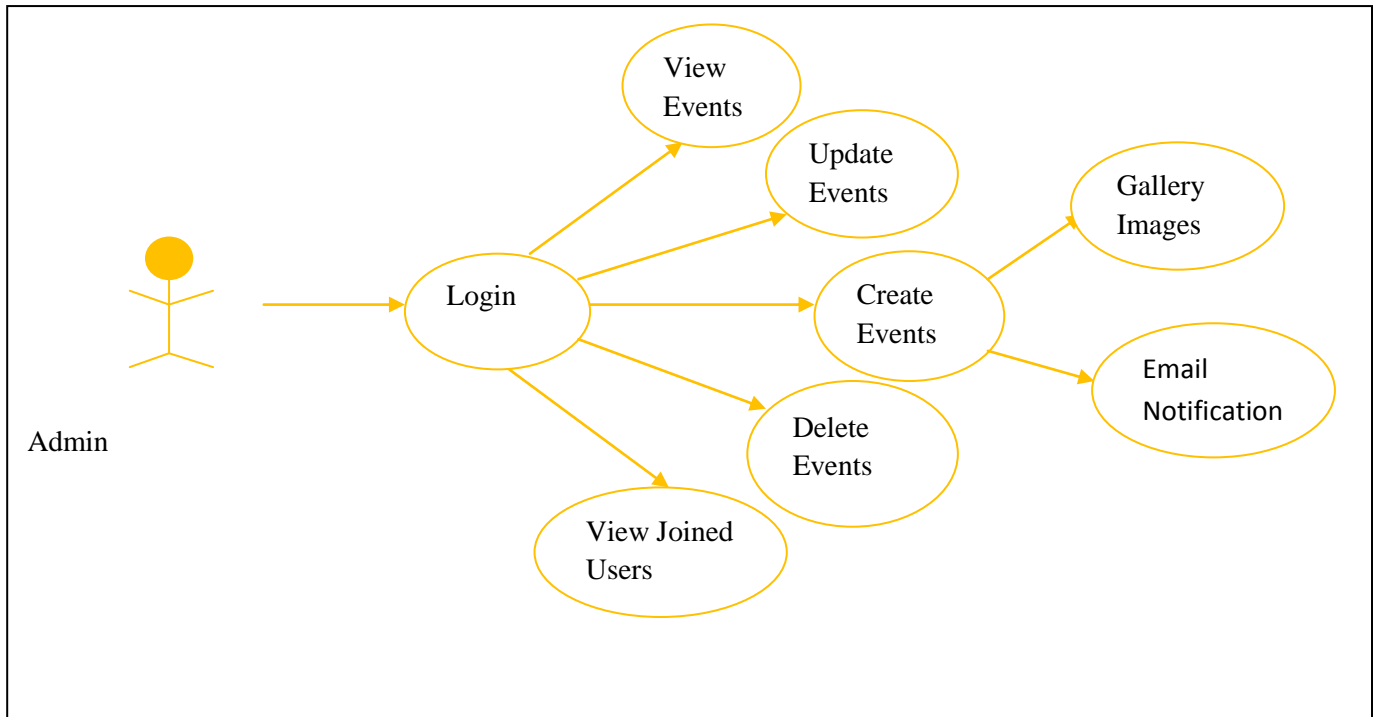
[3.5]Use case model

A Use Case diagram at its simplest is a representation of a user's interaction with the system and depicting the specifications of a use case. A use case diagram can portray the different types of users of a system and the various ways that they interact with the system. This type of diagram is typically used in conjunction with the textual use case and will often be accompanied by other types of diagrams as well. There are two types of user in this application, user and admin. Following depicts their use case diagram:



Use Case Diagram For User

This diagram is showing what a normal user can do with this application. The user can login, after that he can view the events, join the events, search for particular events and can suggest any events.



Use Case Diagram For Admin

This diagram shows the privileges of admin. An admin can create the events in addition to viewing, deleting and updating it. Also he can view the users who have joined any events. He can also post images as an attachment to the events which can be chosen from the gallery. Also he can send notification to students via email service.

[3.6]Database Design

The general theme behind a database is to handle information as an integrated whole. A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and effectively. After designing input and output, the analyst must concentrate on database design or how data should be organized around user requirements. The general objective is to make information access, easy quick, inexpensive and flexible for other users. During database design the following objectives are concerned:-

- Controlled Redundancy
- Data independence
- Accurate and integrating
- More information at low cost
- Recovery from failure
- Privacy and security
- Performance
- Ease of learning and use

[3.7]Table Structure

[3.7.1] Registration

FIELD	TYPE	NULL	KEY
Reg_id		No	Primary
Username		Yes	-
User branch		Yes	-
Usermail		Yes	-
Userphone		Yes	-
Password		Yes	-
Gender		Yes	-
Password		Yes	-

[3.7.2] Timeline

FIELD	TYPE	NULL	KEY
Id	Integer	No	Primary
Event_name	Text	Yes	-
Description	Text	Yes	-
Short description	Text	Yes	-
Society	Text	Yes	-
Date	Text	Yes	-
Venue	Text	Yes	-
Prize	Text	Yes	-
Organiser	Text	Yes	-
Contacts	Text	Yes	-

[3.7.3] User_table

FIELD	TYPE	NULL	KEY
Id	Integer	No	Primary
User_name	Text	Yes	-
User_event_name	Text	Yes	-
User_join_date	Datetime	Yes	-
User_phone	Text	Yes	-
User_society	Text	Yes	-

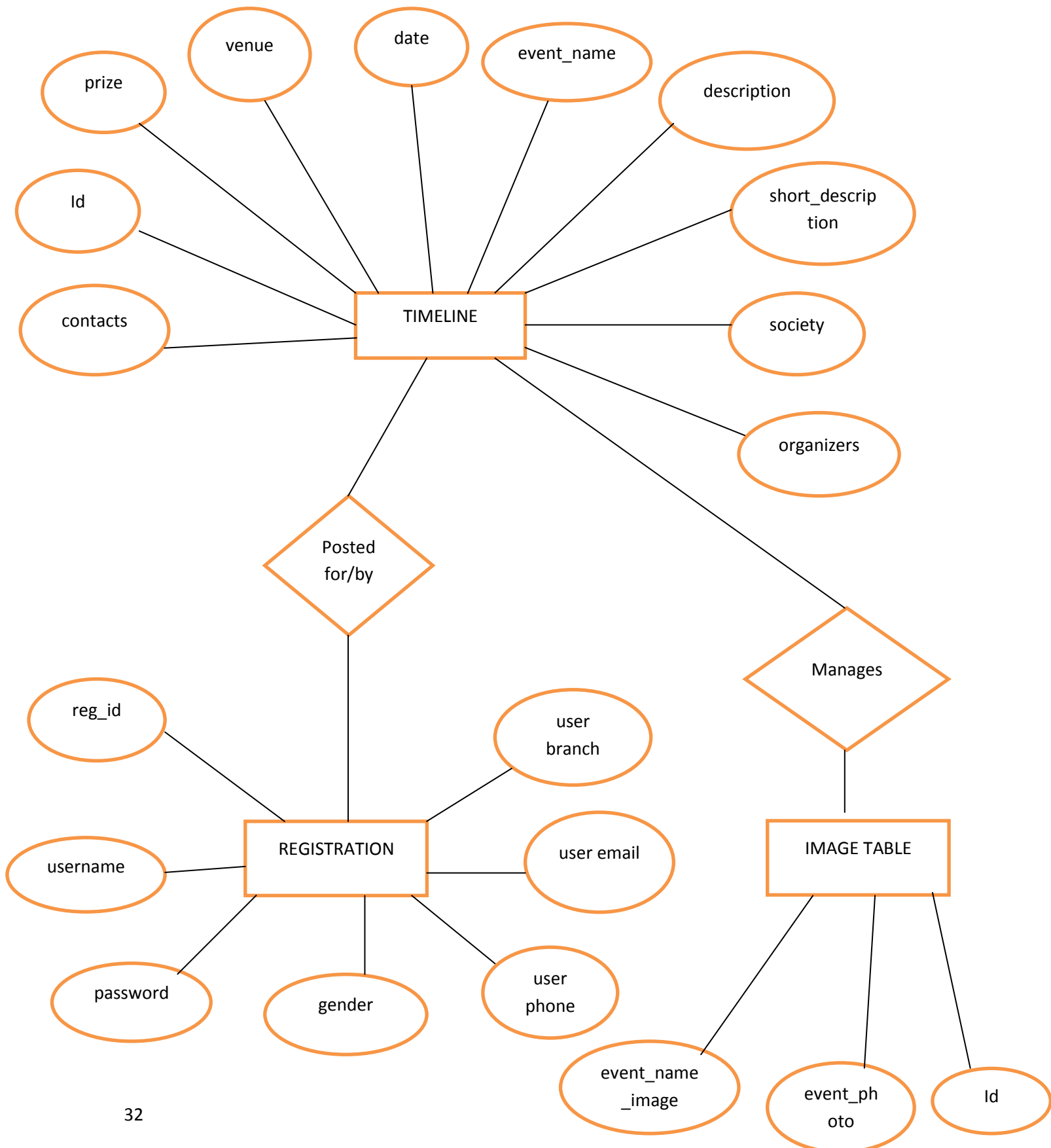
[3.7.4] Image table

FIELD	TYPE	NULL	KEY
Id	Integer	No	Primary
event_name_image	Text	Yes	-
Event_photo	BLOB	Yes	-

[3.7.5] Admin

FIELD	TYPE	NULL	KEY
Id	Integer	No	Primary
Adminuser	Text	Yes	-
Adminpass	Text	Yes	-

[3.8]ER Diagram



[3.9] Assumptions and Dependencies

Assumptions:

- The coding should be error free
- The system should be user-friendly so that it is easy to use for the users
- The information of all users, events must be stored in a database that is accessible by the user and admin
- The system should have more storage capacity and provide fast access to the database

Dependencies:

- Android Studio
- Android Virtual Device (Emulator)
- SQLite Database

[3.10] Specific Requirements

[3.10.1] Data Requirements

Data requirement is meant to be the data that will be used in our application. Data required in this project is all about events, needed to be conveyed to the user. This application also requires the username, passwords and email of persons in order to register them and sending email about updates.

So two main requirements are:

Event Details

User Details

[3.10.2] Functional Requirements

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

Download mobile application:

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

User registration:

Given that a user has downloaded the mobile application, then the user should be able to register through the mobile application. The user must provide user-name, password and e-mail address. The user can choose to provide a regularly used phone number.

User Login:

Given that a user has registered, then the user should be able to log in to the mobile application. The log-in information will be stored on the phone and in the future the user should be logged in automatically.

Reset Password:

Given that a user has registered, then the user should be able to retrieve his/her password by e-mail.

Dashboard:

Given that a user is logged in to the mobile application, then the first page that is shown should be the dashboard page. The user should be able to see all the college events.

Search Events:

The user should be able to search for an event by its title. For example, if a user types Technical, all the events having technical in their content gets displayed.

Selecting an Event:

A user should be able to select any Event from list view. The click on particular event will take him to event details of that particular event.

Joining an Event:

A user should be able to join event by clicking on Join Event button on Event details page, from there he is directed to Join Event page and sending email to administrator giving information about which event he/she is joining.

Navigating back to Event List:

The user should be able to navigate back to events list from the event details section. This is required to give a good user experience.

Deleting Events:

The administrator should have the option to delete the unnecessary events from his phone, by confirming it. This way, admin can save this phone memory from non-required events.

Posting Events:

The admin of this application should be able to create the events. He should be able to add a picture within events. That picture can be taken from gallery of the mobile phone.

Notification Alert:

All the registered users should be able to have a ping or email notification on their mobile phone whenever a new event is posted.

Show Users:

Admin should be able the list of users who have joined particular event.

[3.10.3] Performance Requirements

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

Prominent search feature:

The search feature should be prominent and easy to find for the user.

Usage of the Event Information:

The event link should be prominent and it should be evident that it is a usable link. Selecting the event link should only take one click.

Response Time:

The response time should not be more than 5 seconds if users have a proper internet connection.

Fault Tolerance:

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

[3.10.4] System Dependability

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

Internet Permission:

Application developed, require full internet permissions of mobile so that it can fetch event notices via email. At the same time, it should be able to receive buzz or email notification whenever new event is created by admin.

External SD Card Writable Permissions:

This application would be requiring read write access to SD card. It is required in order to download the events attachment and save in SD card of mobile phone.

System Tools:

This application requires various system tools to be used. For example, it requires Camera of mobile in order to click the image and post in into event. It also require system tool that prevents it from sleeping.

Hardware Control:

It uses vibrator of mobile phone whenever any email notification arrives.

[3.10.5] Maintainability Requirements

Following are the maintainability requirement of Event Management mobile application:

Application extendibility:

The application should be easy to extend. The code should be written in a way that it favors implementation of new functions. It is requires in order for future functions to be implemented easily to the application.

Application testability:

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

[3.10.6] Security Requirements

Admin Login Account Security:

If an admin tries to log in to the web portal with a non-existing account then the admin should not be logged in. The admin should be notified about log-in failure.

Admin Account Security:

There should be security of admin accounts. An admin and IP address should not be able to log-in to the web portal for a certain time period after three times of failed log-in attempts.

User Create Account Security:

The security of creating account for users of the system should be maintained. If a user wants to create an account and the desired user name is occupied, the user should be asked to choose a different user name.

[3.10.7] Look and Feel Requirements

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

Simple and Light:

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

Easy to Use

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

Soft Sound Notification:

The sound for email notification should be very soft. It should not disturb the ears with a loud note.

Everything should be sober in this application.

[Chapter-4] Development and implementation

[4.1] Introduction to Languages (front end & back end)

❖ Java(Front End)

Java is a computer programming language that is concurrent, class-based, object-oriented, and specifically designed to have as few implementation dependencies as possible. It is intended to let application developers "write once, run anywhere" (WORA), meaning that code that runs on one platform does not need to be recompiled to run on another. Java applications are typically compiled to bytecode that can run on any Java virtual machine (JVM) regardless of computer architecture. Java is, as of 2014, one of the most popular programming languages in use, particularly for client-server web applications, with a reported 9 million developers. Java was originally developed by James Gosling at Sun Microsystems (which has since merged into Oracle Corporation) and released in 1995 as a core component of Sun Microsystems' Java platform. The language derives much of its syntax from C and C++, but it has fewer low-level facilities than either of them.

The original and reference implementation Java compilers, virtual machines, and class libraries were originally released by Sun under proprietary licenses. As of May 2007, in compliance with the specifications of the Java Community Process, Sun relicensed most of its Java technologies under the GNU General Public License. Others have also developed alternative implementations of these Sun technologies, such as the GNU Compiler for Java (bytecode compiler), GNU Classpath (standard libraries), and IcedTea-Web (browser plug-in for applets).

Java is:

- **Object Oriented:** In Java, everything is an Object. Java can be easily extended since it is based on the Object model.
- **Platform independent:** Unlike many other programming languages including C and C++, when Java is compiled, it is not compiled into platform specific machine, rather into platform

independent byte code. This byte code is distributed over the web and interpreted by virtual Machine (JVM) on whichever platform it is being run.

- Simple:Java is designed to be easy to learn. If you understand the basic concept of OOP Java would be easy to master.
- Secure: With Java's secure feature it enables to develop virus-free, tamper-free systems. Authentication techniques are based on public-key encryption.
- Architectural: neutral:Java compiler generates an architecture-neutral object file format which makes the compiled code to be executable on many processors, with the presence of Java runtime system.
- Portable:Being architectural-neutral and having no implementation dependent aspects of the specification makes Java portable. Compiler in Java is written in ANSI C with a clean portability boundary which is a POSIX subset.
- Robust:Java makes an effort to eliminate error prone situations by emphasizing mainly on compile time error checking and runtime checking.
- Multithreaded: With Java's multithreaded feature it is possible to write programs that can do many tasks simultaneously. This design feature allows developers to construct smoothly running interactive applications.
- Interpreted:Java byte code is translated on the fly to native machine instructions and is not stored anywhere. The development process is more rapid and analytical since the linking is an incremental and light weight process.
- High Performance: With the use of Just-In-Time compilers, Java enables high performance.
- Distributed:Java is designed for the distributed environment of the internet.
- Dynamic: Java is considered to be more dynamic than C or C++ since it is designed to adapt to an evolving environment. Java programs can carry extensive amount of run-time information that can be used to verify and resolve accesses to objects on run-time.

A Java distribution typically comes in two flavors, the Java Runtime Environment (JRE) and the Java Development Kit (JDK).

- The Java runtime environment (JRE) consists of the JVM and the Java class libraries. Those contain the necessary functionality to start Java programs.
- The JDK additionally contains the development tools necessary to create Java programs. The JDK therefore consists of a Java compiler, the Java virtual machine and the Java class libraries

❖ Sqlite(Back end)

SQLite is an in-process library that implements a self-contained, serverless, zero-configuration, transactional SQL database engine. It is the one database, which is zero-configured, that means like other database you do not need to configure it in your system.

SQLite engine is not a standalone process like other databases, you can link it statically or dynamically as per your requirement with your application. The SQLite accesses its storage files directly.

Features:

- SQLite does not require a separate server process or system to operate.(serverless).
- SQLite comes with zero-configuration, which means no setup or administration needed.
- A complete SQLite database is stored in a single cross-platform disk file.
- SQLite is very small and light weight, less than 400KiB fully configured or less than 250KiB with optional features omitted.
- SQLite is self-contained, which means no external dependencies.
- SQLite transactions are fully ACID-compliant, allowing safe access from multiple processes or threads.

- SQLite supports most of the query language features found in the SQL92 (SQL2) standard.
- SQLite is written in ANSI-C and provides simple and easy-to-use API.
- SQLite is available on UNIX (Linux, Mac OS-X, Android, iOS) and Windows (Win32, WinCE, WinRT).

❖ Android Development Tools

- Android SDK:
The Android Software Development Kit (Android SDK) contains the necessary tools to create, compile and package Android applications. Most of these tools are command line based. The primary way to develop Android applications is based on the Java programming language.
- Android debug bridge (adb):
The Android SDK contains the Android debug bridge (adb), which is a tool that allows you to connect to a virtual or real Android device, for the purpose of managing the device or debugging your application.

Android Developer Tools and Android Studio:

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

Both IDEs contain all required functionality to create, compile, debug and deploy Android applications. They also allow the developer to create and start virtual Android devices for testing. Both tools provide specialized editors for Android specific files. Most of Android's configuration files are based on XML. In this case these editors allow you to switch between the XML representation of the file and a structured user interface for entering the

data. Dalvik Virtual Machine The Android system uses a special virtual machine, i.e., the Dalvik Virtual Machine (Dalvik) to run Java based applications. Dalvik uses a custom bytecode format which is different from Java bytecode.

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

[4.2]Any other supporting languages

❖ XML

XML is independent of the platform, meaning that every program built to use XML can read and manage XML data, no matter the hardware or the operating system. For example, with the right XML tags, you will have the possibility to use a program to open and to operate data from a mainframe computer. Moreover, it won't matter if the XML data are made by somebody else, you can manage them using the programs from MS Office. As it is completely portable, XML has become one of the most popular technologies to exchange data between databases and the desktops of the users.

Features:

XML is widely used in the era of web development. It is also used to simplify data storage and data sharing.

The main features or advantages of XML are given below.

1) XML separates data from HTML:

If you need to display dynamic data in your HTML document, it will take a lot of work to edit the HTML each time the data changes.

With XML, data can be stored in separate XML files. This way you can focus on using HTML/CSS for display and layout, and be sure that changes in the underlying data will not require any changes to the HTML.

With a few lines of JavaScript code, you can read an external XML file and update the data content of your web page.

2) XML simplifies data sharing:

In the real world, computer systems and databases contain data in incompatible formats.

XML data is stored in plain text format. This provides a software- and hardware-independent way of storing data.

This makes it much easier to create data that can be shared by different applications.

3) XML simplifies data transport:

One of the most time-consuming challenges for developers is to exchange data between incompatible systems over the Internet.

Exchanging data as XML greatly reduces this complexity, since the data can be read by different incompatible applications.

4) XML simplifies Platform change:

Upgrading to new systems (hardware or software platforms), is always time consuming. Large amounts of data must be converted and incompatible data is often lost.

XML data is stored in text format. This makes it easier to expand or upgrade to new operating systems, new applications, or new browsers, without losing data.

5) XML increases data availability:

Different applications can access your data, not only in HTML pages, but also from XML data sources.

With XML, your data can be available to all kinds of "reading machines" (Handheld computers, voice machines, news feeds, etc), and make it more available for blind people, or people with other disabilities.

6) XML can be used to create new internet languages:

A lot of new Internet languages are created with XML.

Here are some examples:

XHTML

WSDL for describing available web services

WAP and WML as markup languages for handheld devices

RSS languages for news feeds

RDF and OWL for describing resources and ontology

SMIL for describing multimedia for the web

[4.3] Implementation with Screen Shots

[4.3.1] Landing page



Figure 4.3.1 Landing Page

This is the first page the user is presented with at the first time, when he opens the application.

[4.3.2] Home page



Figure 4.3.2 Home Page

This is the home page where user and admin can sign up and sign in

[4.3.3]Sign Up Page

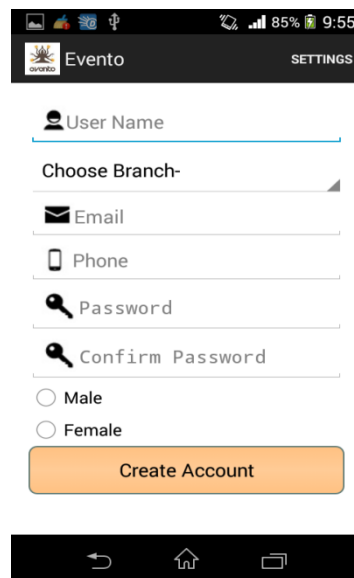


Figure 4.3.3 Sign Up Page

This is the page where user registers himself to the application.

[4.3.4] Login Dialog Box

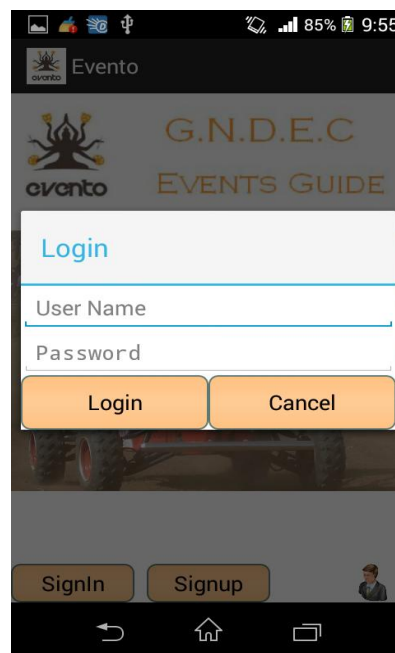


Figure 4.3.4 Login Dialog Box

This page is used to login to the application.

[4.3.5]Event Dashboard Page



Figure 4.3.5 Event Dashboard Page

In this page, all the events are displayed. This is the main panel that is required by the normal user of application. The user will tap on a particular event in order to view details of events.

[4.3.6]Create Event Page

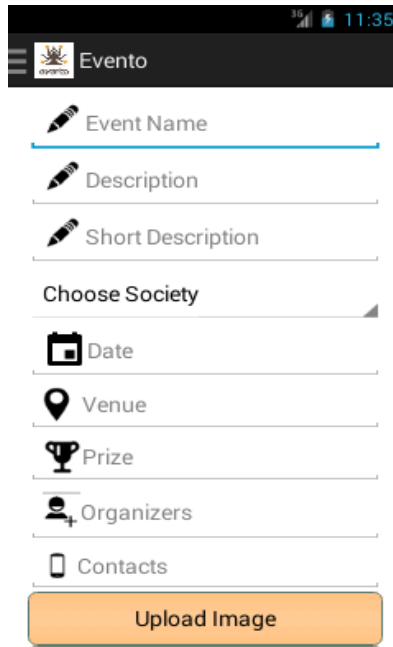


Figure 4.3.6 Create Event Page

This is an Admin Panel Page where admin is allowed to create the events. Admin has an option of attaching image to events from gallery and post with the event.

[4.3.7] Event Description Page for Admin

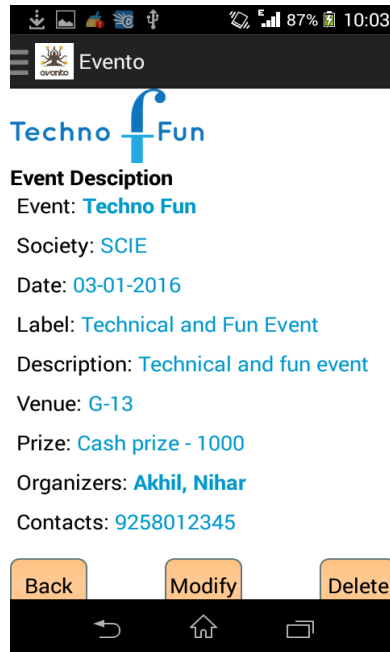


Figure 4.3.7 Event Description Page

This is an Admin Panel Page where this Page gives the details regarding events along with its image.

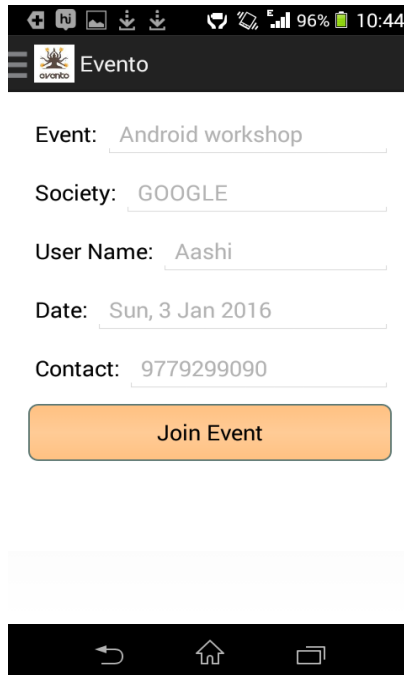
[4.3.8] Event Description Page for User



Figure 4.3.8 Event Description Page

This is User Event Description Page where this page gives the details regarding events along with its image and user can join the events.

[4.3.9] Join Event Page



Evento

Event:

Society:

User Name:

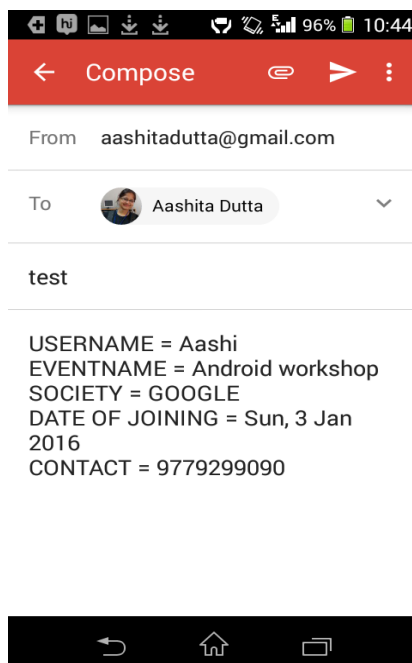
Date:

Contact:

Figure 4.3.9 Join Event Page


This Page is User Join Event Page where user can join an Event

[4.3.10] Send Email Notification



Compose

From: aashitadutta@gmail.com

To:  Aashita Dutta

test

USERNAME = Aashi
 EVENTNAME = Android workshop
 SOCIETY = GOOGLE
 DATE OF JOINING = Sun, 3 Jan 2016
 CONTACT = 9779299090

Figure 4.3.10 Send Email Page

This is a Send Email Page for sending Email Notification to Registered Students to notify about Event

[4.3.11] Show Users Page

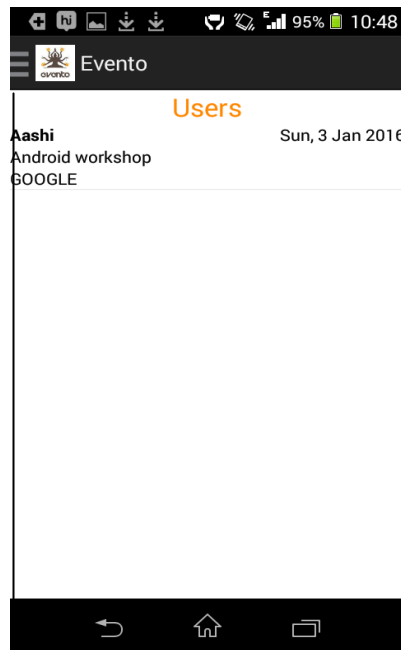


Figure 4.3.11 Show Users Page

This is an Admin Panel Page where this Page shows the list of users who have joined the event

[4.3.12] Navigation Bar for Users

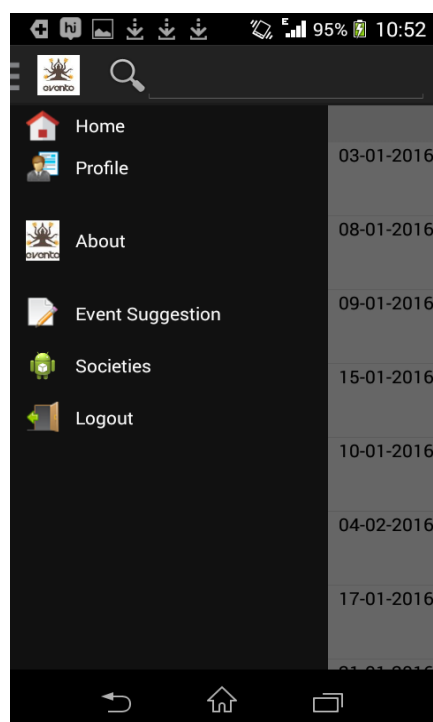


Figure 4.3.12 Navigation Bar for Users

This Picture Describes the Navigation Bar for Users to navigate among different page

[4.3.13] Navigation Bar for Admin

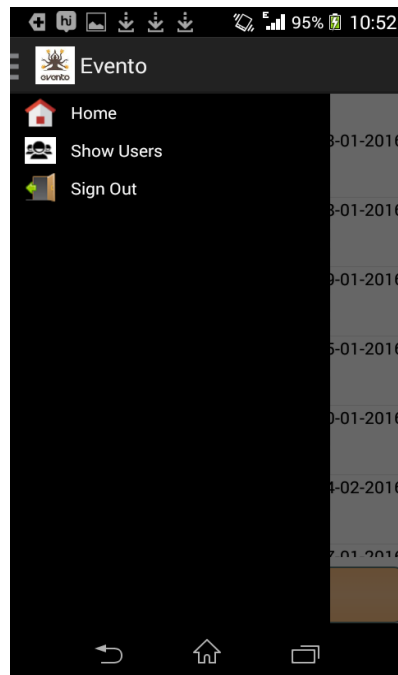


Figure 4.3.13 Navigation Bar for Admin

This Picture Describes the Navigation Bar for Admin to navigate among different pages.

[4.3.14] User Profile Page

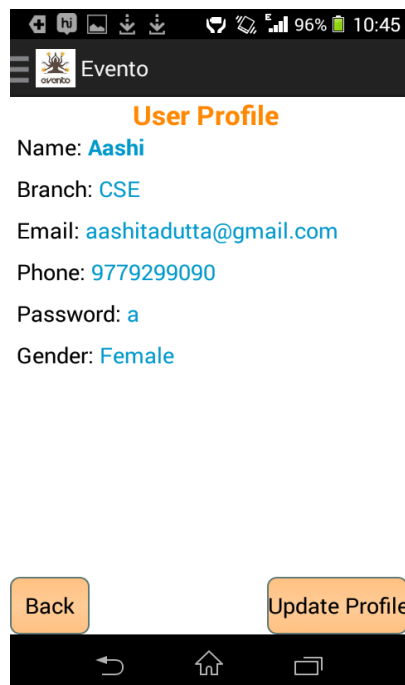


Figure 4.3.14 User Profile Page

This Page describes the user's profile details and can edit its profile by navigating to Update Profile page.

[4.3.15] About App Page



Figure 4.3.15 About App Page

This Page tells us about the app and what it serves.

[4.3.16] Event Suggestion Page for Users

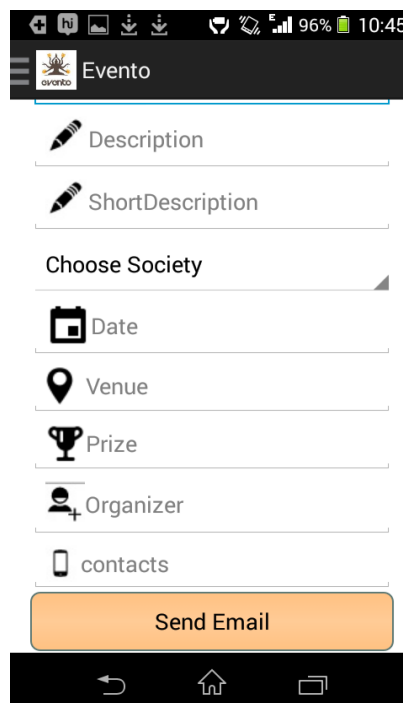


Figure 4.3.16Event Suggestion Page

This page is for Users to suggest events to admin and send email to admin

[4.3.17] Society List Page

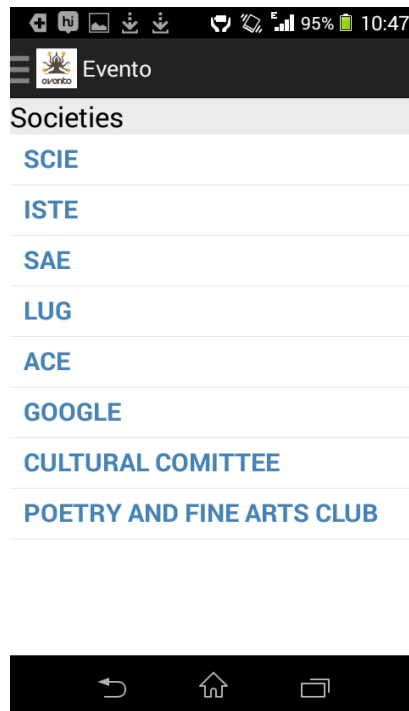


Figure 4.3.17 Society List Page

This Page is for users who can view existing societies in our college and details about each society

[4.3.18] Email Validation

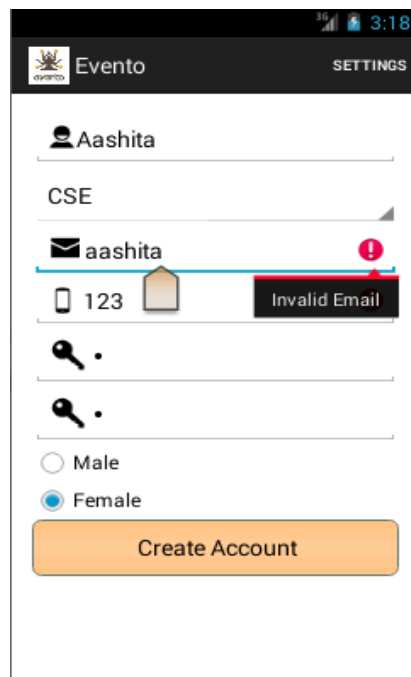


Figure 4.3.18 Email Validation

This figure shows the email validations implemented inside the application

[4.3.19] Mobile Number Validation

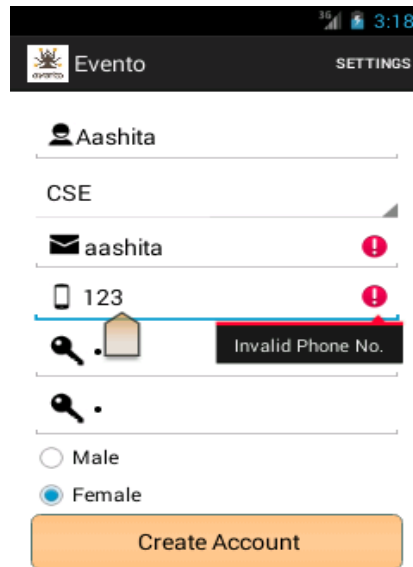


Figure 4.3.19 Mobile Number Validation

This figure shows the mobile number validation implemented in the application.

[4.3.20] Reset Password Page

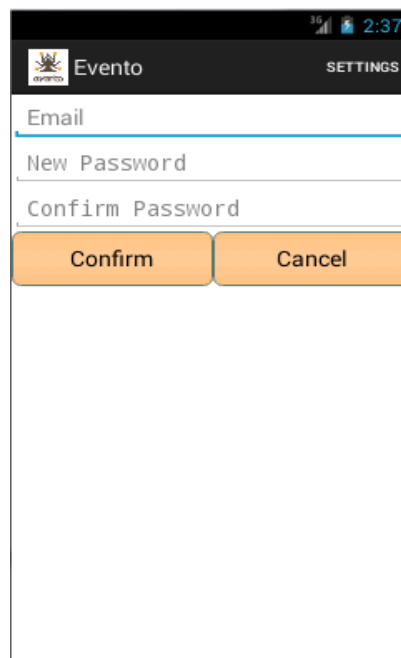


Figure 4.3.20 Reset Password Page.

This page is presented to the user when user forgot his password and wants to reset the password.

[4.4] Testing

After completion of coding, testing phase will get started. This is an important phase as it tells about how our project is working and does it works correctly. It tells about errors found in the project and make it correct.

[4.4.1] Types of Mobile Testing

There are broadly two kinds of testing that take place on mobile devices:

- **Hardware testing:**

The device including the internal processors, internal hardware, screen sizes, resolution, space or memory, camera, radio, Bluetooth, WIFI etc. This is sometimes referred to as, simple “Mobile Testing”.

- **Software or Application testing:**

The applications that work on mobile devices and their functionality is tested. It is called the “Mobile Application Testing” to differentiate it from the earlier method. Even in the mobile applications, there are few basic differences that are important to understand:

a) Native apps: A native application is created for use on a platform like mobile and tablets.

b) Mobile web apps are server-side apps to access website/s on mobile using different browsers like chrome, Firefox by connecting to a mobile network or wireless network like WIFI.

c) Hybrid apps are combinations of native app and web app. They run on devices or offline and are written using web technologies like HTML5 and CSS.

[4.4.2] Significance of Mobile Application Testing

Testing applications on mobile devices is more challenging than testing web apps on desktop due to

- Different range of mobile devices with different screen sizes and hardware configurations like hard keypad, virtual keypad (touch screen) and trackball etc.
- Wide varieties of mobile devices like HTC, Samsung, Apple and Nokia.
- Different mobile operating systems like Android, Symbian, Windows, Blackberry and IOS.
- Different versions of operation system like iOS 5.x, iOS 6.x, BB5.x, BB6.x etc.

- Different mobile network operators like GSM and CDMA.
- Frequent updates – (like android- 4.2, 4.3, 4.4, iOS-5.x, 6.x) – with each update a new testing cycle is recommended to make sure no application functionality is impacted.

As with any application, Mobile application testing is also very important, as clientele is usually in millions for a certain product – and a product with bugs is never appreciated. It often results in monetary losses, legal issue and irreparable brand image damage.

[4.4.3] Types of Mobile App Testing:

To address all the above technical aspects, the following types of testing are performed on Evento Application.

- Functional testing
- Performance testing
- Security testing
- Usability testing
- Compatibility testing

❖ Functional testing:

The most fundamental test scenarios in the functional testing can be considered as :

1. To validate whether all the required mandatory fields are working as required.
2. To validate that the mandatory fields are displayed in the screen in a distinctive way than the non-mandatory fields.
3. To validate whether the application works as per as requirement whenever the application starts/stops.
4. To validate whether the phone is able to store, process and receive SMS whenever the app is running. In order to validate the same we need to use a second phone to send sms to the device which is being tested and where the application under test is currently running.
5. To validate that the device is able to perform required multitasking requirements whenever it is necessary to do so.

6. To validate that the application allows necessary social network options such as sharing, posting and navigation etc.
7. To validate that the page scrolling scenarios are being enabled in the application as necessary.
8. To validate that the navigation between relevant modules in the application are as per the requirement.
9. To validate whether the application performs according to the requirement in all versions of Mobile that is 2g, 3g and 4g.

❖ **Performance testing:**

This type of testing's fundamental objective is to ensure that the application performs acceptably under certain performance requirements such as access by a huge number of users or the removal of a key infrastructure part like a database server.

The general test scenarios for performance testing in a Mobile application are:

1. To determine whether the application performs as per the requirement under different load conditions.
2. To validate whether the response time of the application is as per as the requirements.
3. To evaluate product and/or hardware to determine if it can handle projected load volumes.
4. To validate application performance when network is changed to WIFI from 2G/3G or vice versa.
5. To validate that the battery consumption, memory leaks, resources like GPS, Camera performance is well within required guidelines.
6. To validate the application longevity whenever the user load is rigorous.
7. To validate the network performance while moving around with the device.

❖ **Security testing:**

The fundamental objective of security testing is to ensure that the application's data and networking security requirements are met as per guidelines.

The following are the most crucial areas for checking the security of Mobile applications.

1. To validate whether an application is not permitting an attacker to access sensitive content or functionality without proper authentication.
2. To validate that the application has a strong password protection system and it does not permit an attacker to obtain, change or recover another user's password.
3. To analyze the data storage and data validation requirements.

❖ Usability testing:

The usability testing process of the Mobile application is performed to have a quick and easy step application with less functionality than a slow and difficult application with many features. The main objective is to ensure that we end up having an easy-to-use, intuitive and similar to industry-accepted interfaces which are widely used.

1. To ensure that the buttons are placed in the same section of the screen to avoid confusion to the end users.
2. To ensure that the icons are natural and consistent with the application.
3. To ensure that the buttons, which have the same function should also have the same color.
4. To ensure that the application provides a method for going back or undoing an action, on touching the wrong item, within an acceptable duration.
5. To ensure that the text is kept simple and clear to be visible to the users.
6. To ensure that the short sentences and paragraphs are readable to the end users.
7. To ensure that the font size is big enough to be readable and not too big or too small.
8. To validate that the closing of the application is performed from different states and verify if it re-opens in the same state.
9. To ensure that the application items are always synchronized according to the user actions.
10. To ensure that the end user is provided with a user manual which helps the end user to understand and operate the application who may be not familiar with the application's proceedings.

11. Methods which deal with exiting the application.

12. Notifications settings.

Usability testing is normally performed by manual users since only human beings can understand the sensibility and comfort ability of the other users.

❖ **Compatibility testing:**

Compatibility testing on mobile devices is performed to ensure that since mobile devices have different size, resolution, screen, version and hardware so the application should be tested across all the devices to ensure that the application works as desired.

The following are the most prominent areas for compatibility testing.

1. To validate that the user Interface of the application is as per the screen size of the device, no text/control is partially invisible or inaccessible.
2. To ensure that the text is readable for all users for the application.

[Chapter-5]Conclusion and Future scope

[5.1]Conclusion

Evento application overcomes all the limitations and drawbacks of the existing system. The online Evento Application is user-friendly android application. The user interface is easy to use. There is no longer any need to maintain physical listing of students for events. Also the societies' administrator can easily post events through Evento. It bridges the gap between User and Organiser.

[5.2]Future Scope

The future scope of the project is that it can be used as any news giving application or it can be used to advertise your any new launch, telling the customers about new event schemes for example marriage or parties. This application of Event Management can be further extended to include the following features:

1. Categorization of Events:

Events can be categorized in different categories like technical, techno-fun, fun, or cultural etc. so that its possible for user to easily manage the events. Categorization can also be done by making groups. Defining the event to be circulated in a particular society can make it more secure.

2. Documents and PDF files:

The attachments can be further improved to include PDF files or Doc files. Then there will not be much need to send images with the events. A single file would serve all the purposes.

3. Feedback:

Feedback on the notices can also be taken. It can increase communication among connected members and any issue can be easily sorted out on the spot.

4. Interactivity:

Further implementation of chat facility can be done in this app so as to bring close interactivity between admin and students so that they can discuss any type of event and can exchange files or share image, videos too.

References

1.javatpoint.com – <http://www.javatpoint.com>

2.stackoverflow.com - <http://stackoverflow.com/>

3.Androidhive.com - <http://www.androidhive.info/>

3.developer.android.com - <https://developer.android.com/training/index.html>

Appendix

Source Code URL: <https://github.com/aashitadutta/Event-Management>

Blog : <https://aashitadutta2013.wordpress.com/>