Integrated Tourist Guide Application - Increment 1

Group Number: Team 15

Project Title: Integrated Tourist Guide Application

Group Members:

- Muktevi, Venkata Sathya Vamsi Krishna (31)
- Tummala, Vijay Kumar Tummala (55)
- Murakonda, Sravani (34)
- Panja, Kumara Satya Gopal (41)

Project Goal and Objectives

- **Motivation:**
  - We have several tourists/visitors struggling a lot by browsing many different websites for gathering information about the restaurants, transport, entertainment, places, events etc., when they visit a new place/city. They have to make note of all the information they like and sort they out manually to plan a tour schedule. All these takes a lot of time, so avoid this and make users tour a memorable part of their life, we came up with an idea of developing an application. This application is a one stop platform for the users, where we integrate information from different web services APIs and provide them to users in a customized and user friendly interface based on the search criteria. Also users can short list their preferences and make a tour schedule using the application.

- **Significance/ Uniqueness:**
  - There are several web services which address these issues individually. We are here with an idea of integrating all these services and allowing user to make a trip schedule. Here in this application we are addressing two user specific issues.
  - We provide the requested data based on two specific parameters Time, Location.
    Example: If the time is 1:00 PM in the Noon and user is searching restaurants, we will retrieve the data of restaurants which provide lunch to users and nearby user.
  - We provide the user an option of scheduling, where user add his/her interests in to his queue. User can sort the interests and make a plan, add time to get a reminder, and also can remove an Item from the queue if he/she wishes to do so.
  - Based on the user profile we suggest the places, restaurants etc. which user might be interested.

- **Objectives:**
  - Our main objective is to integrate all the useful web services that provide information about restaurants, entertainment, transport, events and places on to a single platform. We integrate the information available in different web services and display them to user based on user search criteria. This application also allows user to prepare a schedule for the tour by combining all the information in the user's wish-list. This app avoids users from browsing different websites and wasting time in gathering information and making a pen-paper schedule. Based on user information and history of visits we make the search pattern more user friendly.

❖ **System Features:**

The system will possess the following features:

- User should open the application on mobile and see the home page. On clicking on home page user will see a login page.
- User needs to login to the application with his Facebook login credentials.
- On successful login we navigate user to menu page.
- In the Menu Page the different menu options (Restaurants, places, Events, Transport and Schedule) are displayed.
- User has to select a menu option and click on it. So this will navigate user to next Search Page.
- In the Search Page user has to enter the specific interested query to search bar (Example: If user is looking for restaurants, then he/she can enter Chinese as a query).
- After processing the query all the results are displayed in list to users based on current location and Time as parameters.
- On each item the image, name and address are displayed. Based on user interest user can select item of his interest.
- On clicking specific item user will be navigated to next page where the details of the venue are displayed.
- The Details page displays venue pictures, contact details, address etc. Also the Map showing the venue location.
- User has an option to add the venue to his wish list.
- In the Menu Page user has an option to open My Plan page.
- In My Plan page user has option to add time, and remove the venue if not interested.

Software Requirement Specification (SRS):

This is an android application developed on android studio IDE. We used xml to build our layouts and scripting is done on java. The Android studio provides good platform to develop android applications.

Code Development Tool Kit:

JDK 7: The Java Development Kit (JDK) is an implementation of either one of the Java SE, Java EE or Java ME platforms released by Oracle Corporation in the form of a binary product aimed at Java developers on Solaris, Linux, Mac OS X or Windows. The JDK includes a private JVM and a few other resources to finish the development of a Java Application. Java 1.7 is used in our project, implementation and scripting is done in java script.

Android SDK: Since the application is an android application we used android SDK to build the application based on SDK tools.

Programming Language:

Java: We use java for our application development. Java is a general-purpose 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 compiled Java code can run on all platforms that support Java without the need for recompilation. Java applications are typically compiled to bytecode that can run on any Java virtual machine (JVM) regardless of computer architecture.

Integrated Development Environment:

Android Studio: At the core of Android Studio is an intelligent code editor capable of advanced code completion, refactoring, and code analysis. The powerful code editor helps you be a more productive Android app developer. New project wizards make it easier than ever to start a new project. Start projects using template code for patterns such as navigation drawer and view pagers, and even import Google code samples from GitHub. Build apps for Android phones, tablets, Android Wear, Android TV, Android Auto and Google Glass. With the new Android Project View and module support in Android Studio, it's easier to manage app projects and resources. Android Studio comes pre-configured with an optimized emulator image. The updated and streamlined Virtual Device Manager provides pre-defined device profiles for common Android devices. Android Studio comes pre-configured with an optimized emulator image. The updated and streamlined Virtual Device Manager provides pre-defined device profiles for common Android devices.

Designing UI screens:

XML: We used XML for designing our layouts. We have developed all screens using this XML Layouts available in android studio.

Versioning:

We have used GitHub cloud tool to store all our code in remote location. This is accessible to all our project mates. We have created a master and a development branch for project purposes. All our source code, documentation, UML diagrams, Wireframes are stored in this repository. Also Zen Hub plug-in is integrated to trace the phases and issues in project, we can assign tasks to our team mates using this tool.

Modeling Tools:

We have used https://www.Creately.com and Microsoft Visio to design UML diagrams in our project.

Web Services:

Foursquare API: We used this web service to get the data related to restaurants, transport, places and events. This API provides the details of all the venues based on search pattern entered by user. http://developers.foursquare.com/ is used for getting data

related various categories. We process the retrieved data and make it more comfortable for end user.

Facebook API: We made use of Facebook plugin to enable user to login using his Facebook credentials. We make use of users Facebook data to make assumptions of users interests and give best possible suggestions. https://developers.facebook.com/ web service is used in our project to enable users to login through there Facebook login credentials.

Google Maps API: We have taken Google Maps API to trace user's current location and time. Based on this details we suggest users the venues nearby him and also preferable restaurants based on time. https://developers.google.com/maps/ is used for fetching

Project Development Plan:

We are implementing **AGILE** methodologies to in development of our project. We have split our entire project into four increments. These increments or mile stones are based on time interval. We have divided our project into four modules.

1. **Venue Search Module**: Retrieving data from foursquare web service and displaying the results based on user current location.
2. **Facebook Login Integration Module:** Implementing Facebook Login in our project and retrieving user details to optimize search
3. **Category based searches Module**: We will display the data to user by splitting data into different categories and we will improve UI interface.
4. **Scheduling Module**: A My Schedule page which has all the user's interests and functionalities like edit remove a venue are added to this page.

Increment 1:

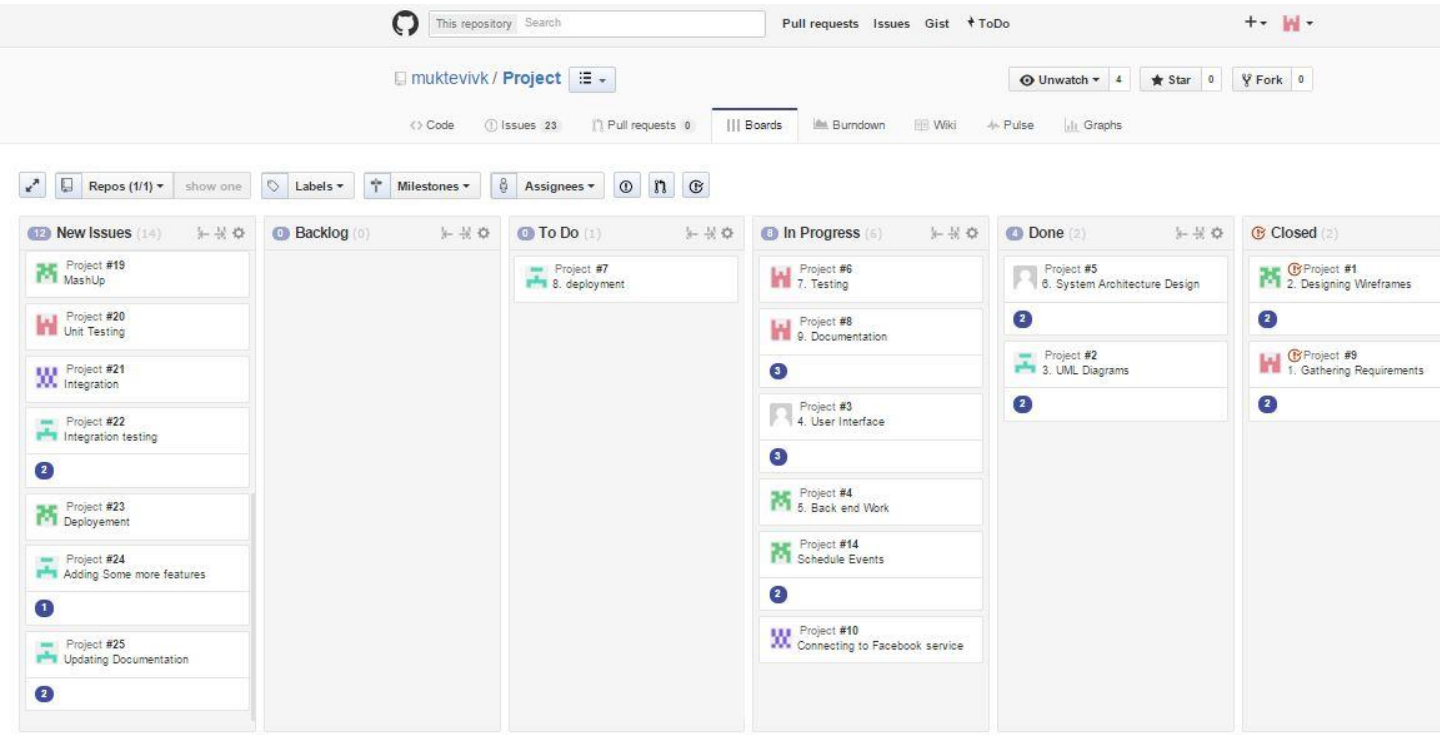
In first increment we worked on Search Module. After finalizing the project idea. We have done a lot of research work for tracing out the best web services available and which full fill all our requirements. Foursquare API Facebook API, Google Maps API are the web services we finalized to implement in our project. We have developed a basic layout and utilized foursquare API to Google Maps API to send Search query and retrieve data based on query and location.

Functionalities implemented in Increment 1:

- A Search page with a search bar and button.
- Based on search Query and Current Location of User we retrieve data from Foursquare API web service.
- We display all the search results to user in a list. Each Item in the list has an icon, Name and Address.
- When user selects an item in the list and clicks on it we navigate user to next page.
- In the Details page all the details like Name, Address, Contact details, Reviews are displayed.

As a part of future development plan we will implement other three modules in coming increments.

Zen Hub Screen Shots:



Burndown Charts:

0	0	9	
Total Story Points	Completed Story Points	Total Issues	Unestim

Issues

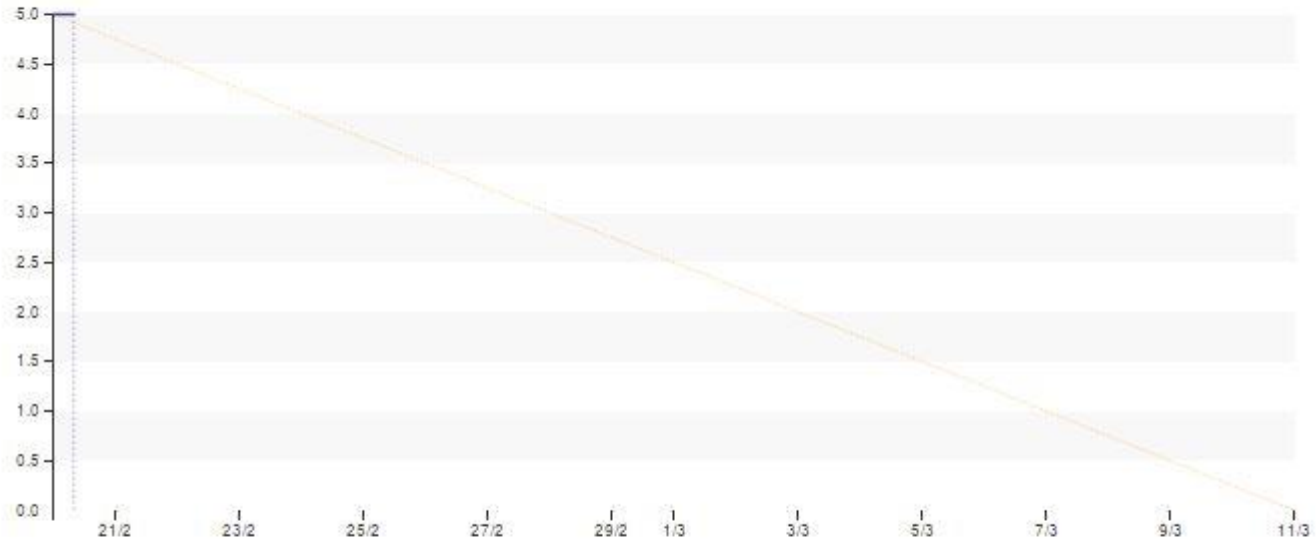
- **#8** 9. Documentation
- **#3** 4. User Interface
- **#5** 6. System Architecture Design
- **#2** 3. UML Diagrams
- **#7** 8. deployment
- **#6** 7. Testing
- **#4** 5. Back end Work
- **#9** 1. Gathering Requirements
- **#1** 2. Designing Wireframes

Increment 2

Edit Milestone | Change Milestone

Start: **Feb 20, 2016** Edit | Due: **Mar 11, 2016** Edit

powered by | ZenHub



5	0	5	2
Total Story Points	Completed Story Points	Total Issues	Unestimated Issues

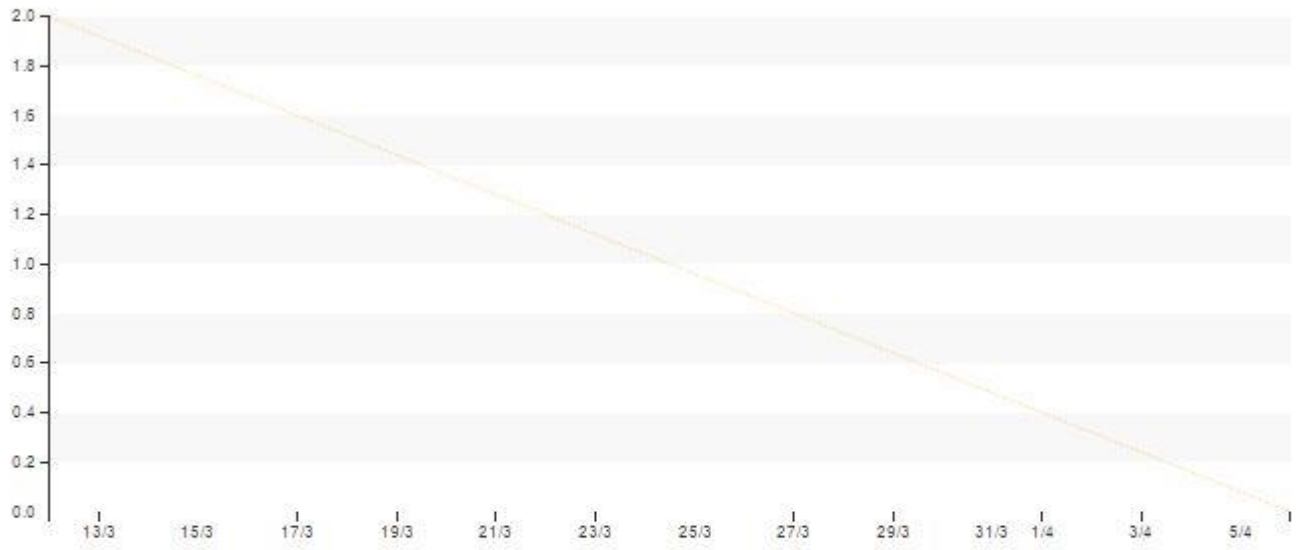
Issues	Story Points
#15 View Schedule	2
#14 Schedule Events	2
#13 Update Report	1
#11 Checking Login Credentials	Not estimated
#10 Connecting to Facebook service	Not estimated

Increment 3

Edit Milestone | Change Milestone

Start: **Mar 12, 2016** Edit | Due: **Apr 6, 2016** Edit

powered by | ZenHub



2	0	5	4
Total Story Points	Completed Story Points	Total Issues	Unestimated Issues

Issues	Story Points
#16 Storing Past history	2
#20 Unit Testing	Not estimated
#19 MashUp	Not estimated
#18 Storing data in database	Not estimated
#17 Managing Database	Not estimated

Increment 4

Edit Milestone | Change Milestone

Start: **Apr 7, 2016** Edit | Due: **Apr 29, 2016** Edit

5	0	5	2
Total Story Points	Completed Story Points	Total Issues	Unestimated Issues

Issues	Story Points
#25 Updating Documentation	2
#22 Integration testing	2
#24 Adding Some more features	1
#23 Deployement	Not estimated
#21 Integration	Not estimated

Timelines:

Increment	**Date**
Plan & Increment 1	**2/19/2016**
Increment 2	3/11/2016
Increment 3	4/6/2016
Increment 4	4/29/2016

Final Submission	5/6/2016

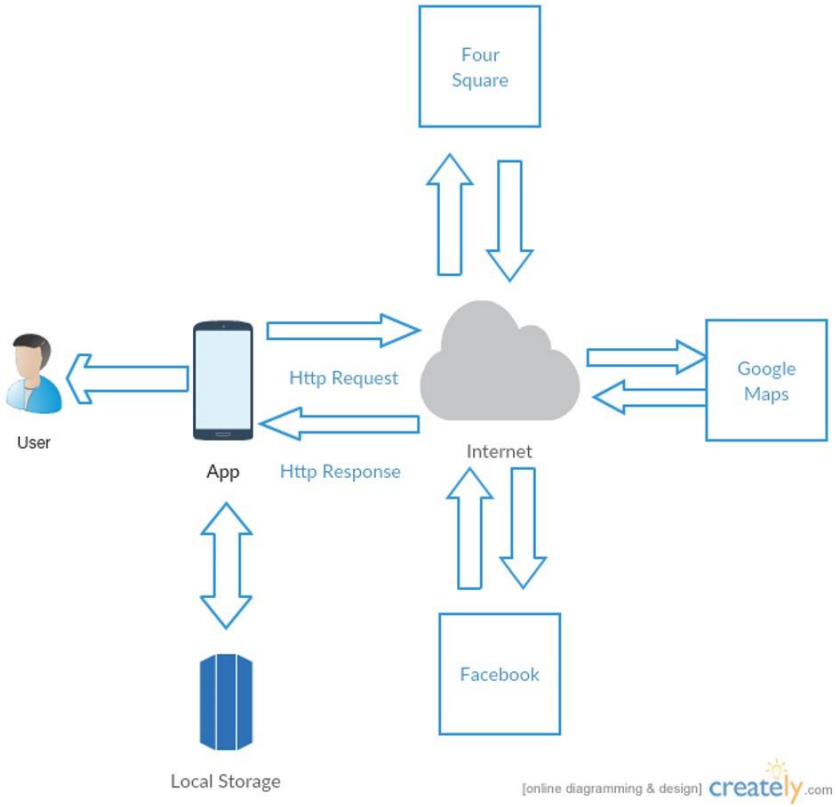
Members:

S.no	Name	Class ID
1	Muktevi, Venkata Sathya Vamsi Krishna	31
2	Tummala, Vijay Kumar Tummala	55
3	Murakonda, Sravani	34
4	Panja, Kumara Satya Gopal	41

Task Responsibility:

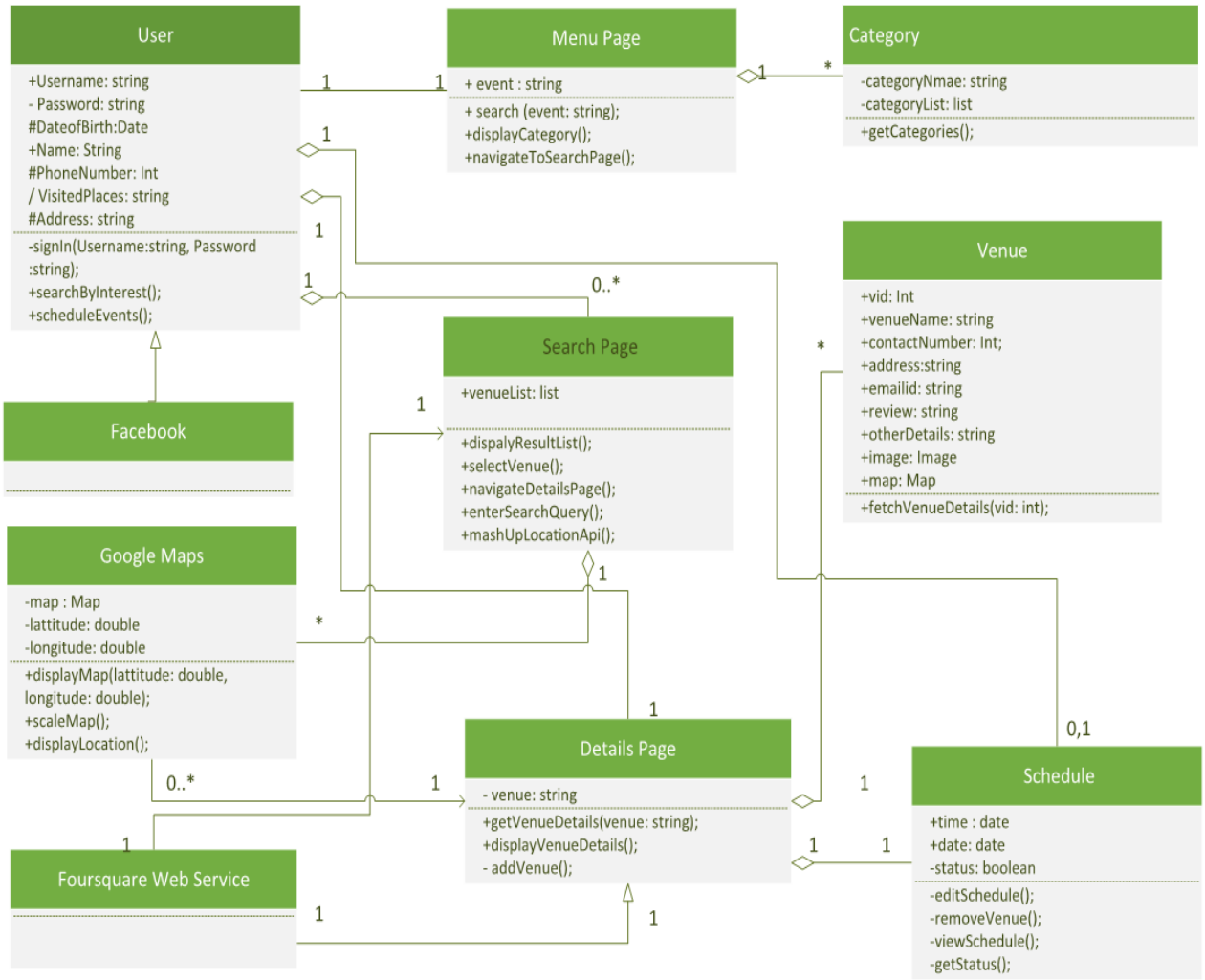
S.no	Name	Task Responsibility
1	Muktevi, Venkata Sathya Vamsi Krishna	Development ,Functional Scheduling , Android Framework Architecture
2	Tummala, Vijay Kumar Tummala	Development, Requirements Gathering, Documentation, Development , Wire frames
3	Murakonda, Sravani	UI Designing, Android Framework Architecture, , Functional Testing, UML Diagrams
4	Panja, Kumara Satya Gopal	Task Scheduling, , Unit Testing, Requirements Gathering, Web service Integration

System Architecture:

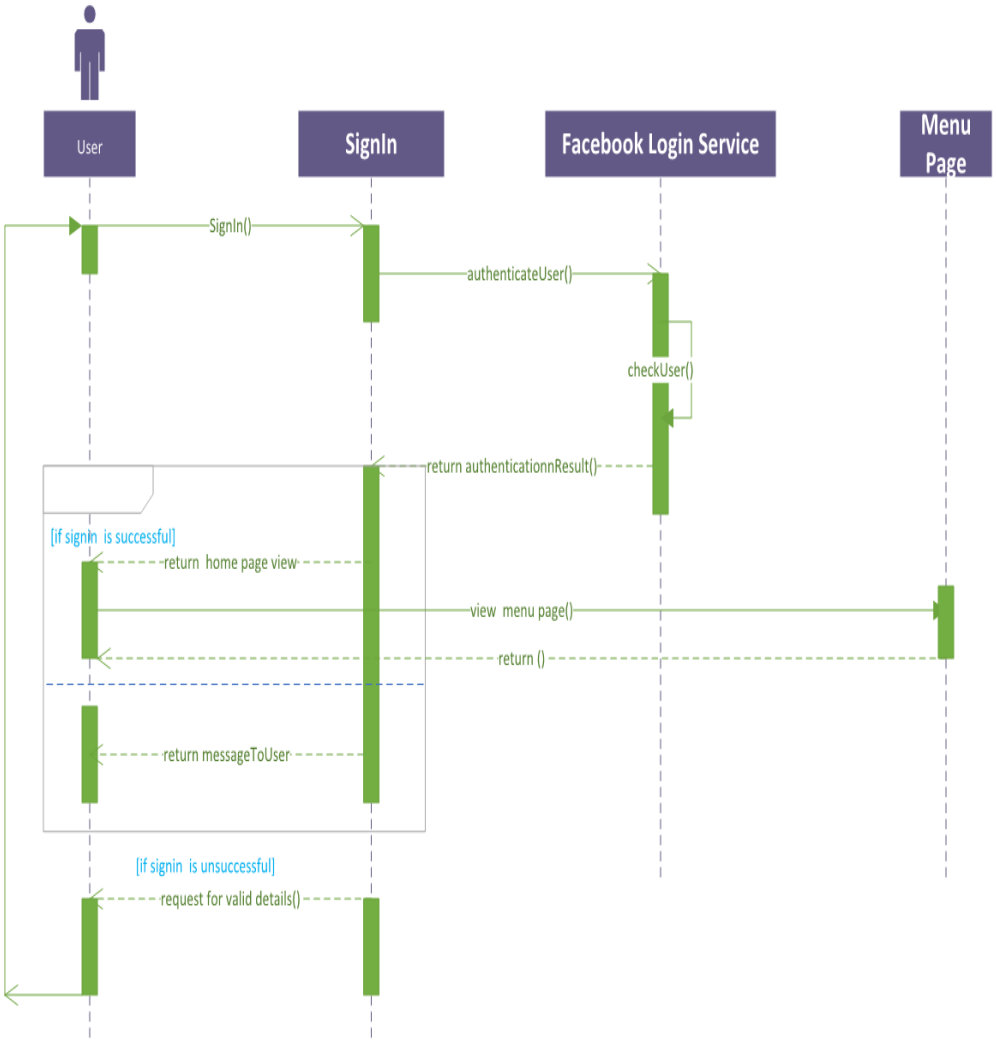


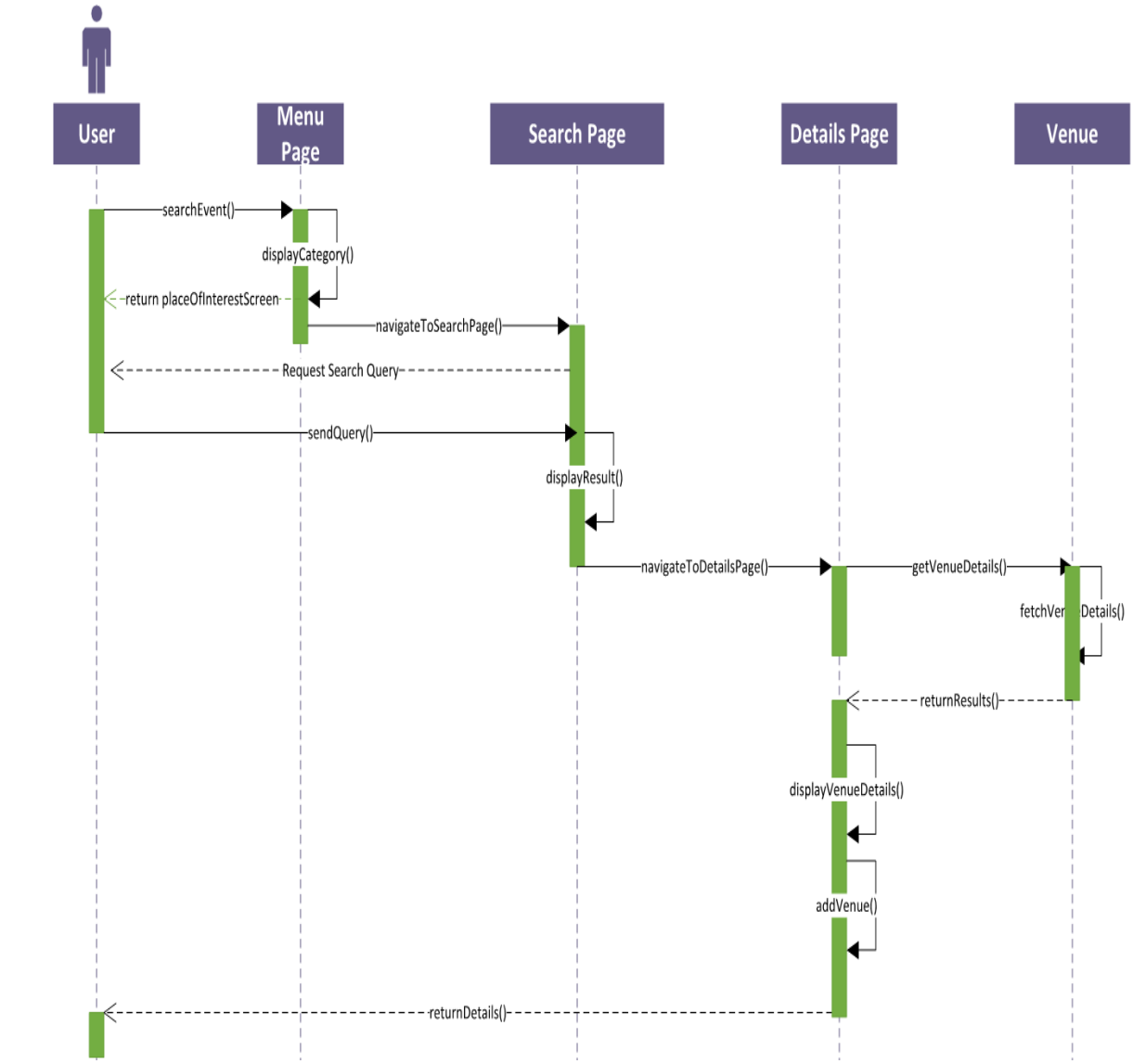
Designing:

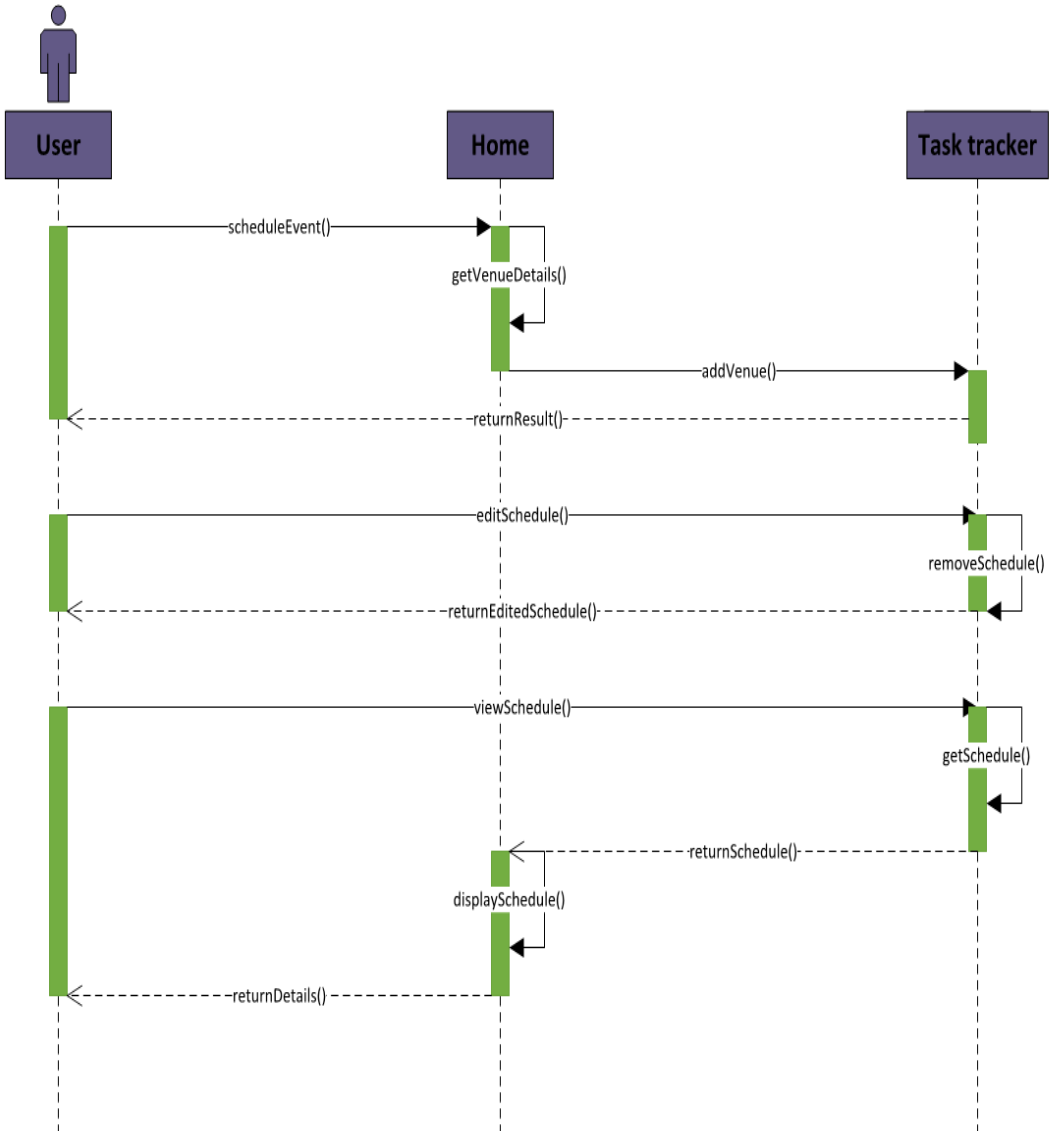
UML Diagrams:


User
+Username: string
- Password: string
#DateofBirth:Date
+Name: String
#PhoneNumber: Int
/ VisitedPlaces: string
#Address: string
-signIn(Username:string, Password :string);
+searchByInterest();
+scheduleEvents();
Menu Page
+ event : string
+ search (event: string);
+displayCategory();
+navigateToSearchPage();
Category
-categoryNmae: string
-categoryList: list
+getCategories();
Venue
+vid: Int
+venueName: string
+contactNumber: Int;
+address:string
+emailid: string
+review: string
+otherDetails: string
+image: Image
+map: Map
+fetchVenueDetails(vid: int);
Search Page
+venueList: list
+dispalyResultList();
+selectVenue();
+navigateDetailsPage();
+enterSearchQuery();
+mashUpLocationApi();
Facebook
Google Maps
-map : Map
-lattitude: double
-longitude: double
+displayMap(lattitude: double, longitude: double);
+scaleMap();
+displayLocation();
Details Page
- venue: string
+getVenueDetails(venue: string);
+displayVenueDetails();
- addVenue();
Schedule
+time : date
+date: date
-status: boolean
-editSchedule();
-removeVenue();
-viewSchedule();
-getStatus();
Foursquare Web Service
1
*
0..*
0,1

Login


User
SignIn
Facebook Login Service
Menu Page
SignIn()
authenticateUser()
checkUser()
return authenticationnResult()
[if signin is successful]
return home page view
view menu page()
return ()
return messageToUser
[if signin is unsuccessful]
request for valid details()


User
Menu Page
Search Page
Details Page
Venue
searchEvent()
displayCategory()
return placeOfInterestScreen
navigateToSearchPage()
Request Search Query
sendQuery()
displayResult()
navigateToDetailsPage()
getVenueDetails()
fetchVer Details()
returnResults()
displayVenueDetails()
addVenue()
returnDetails()


User
Home
Task tracker
scheduleEvent()
getVenueDetails()
addVenue()
returnResult()
editSchedule()
removeSchedule()
returnEditedSchedule()
viewSchedule()
getSchedule()
returnSchedule()
displaySchedule()
returnDetails()

Wire Frames:

Figure 1: This is a home page of the application

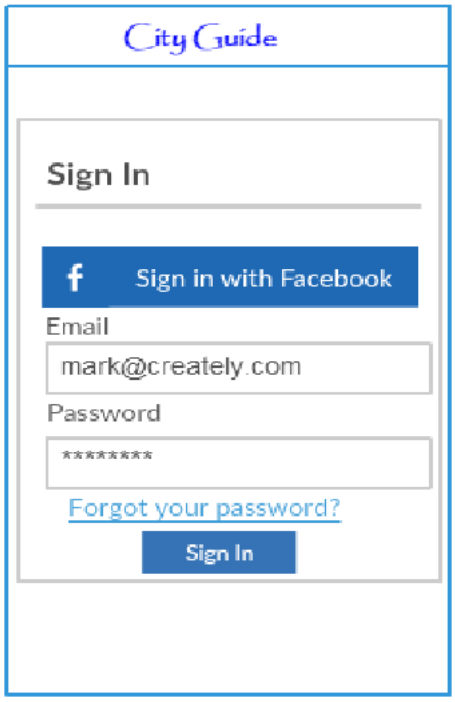


Figure 2: Facebook Login Page

Figure 3: This is Menu page

Figure 4: This is a search page

Figure 5: This is List of Retrieved Values

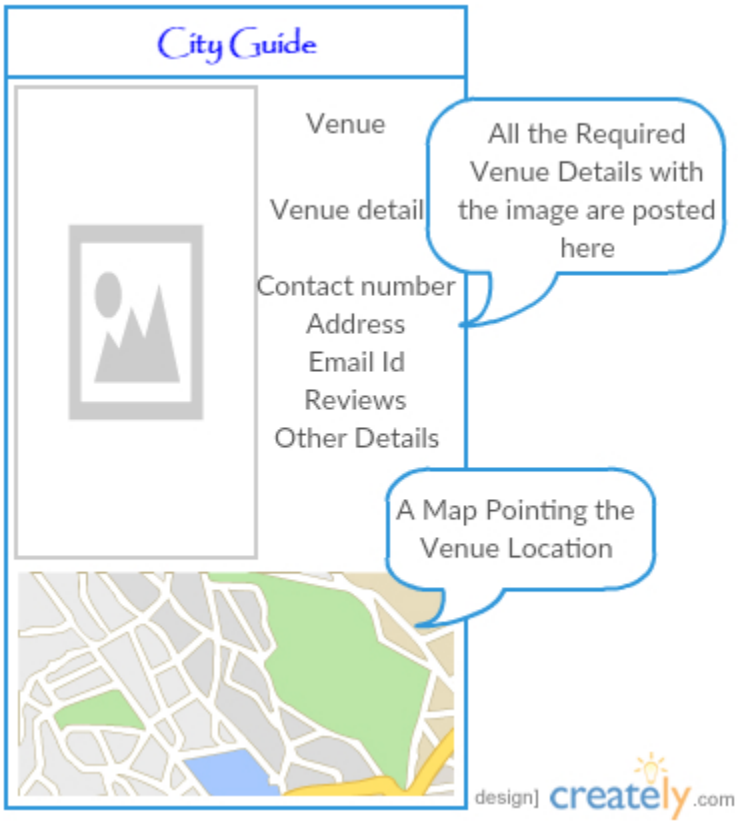


Figure 6: This is a Details page displaying all the Venue details.

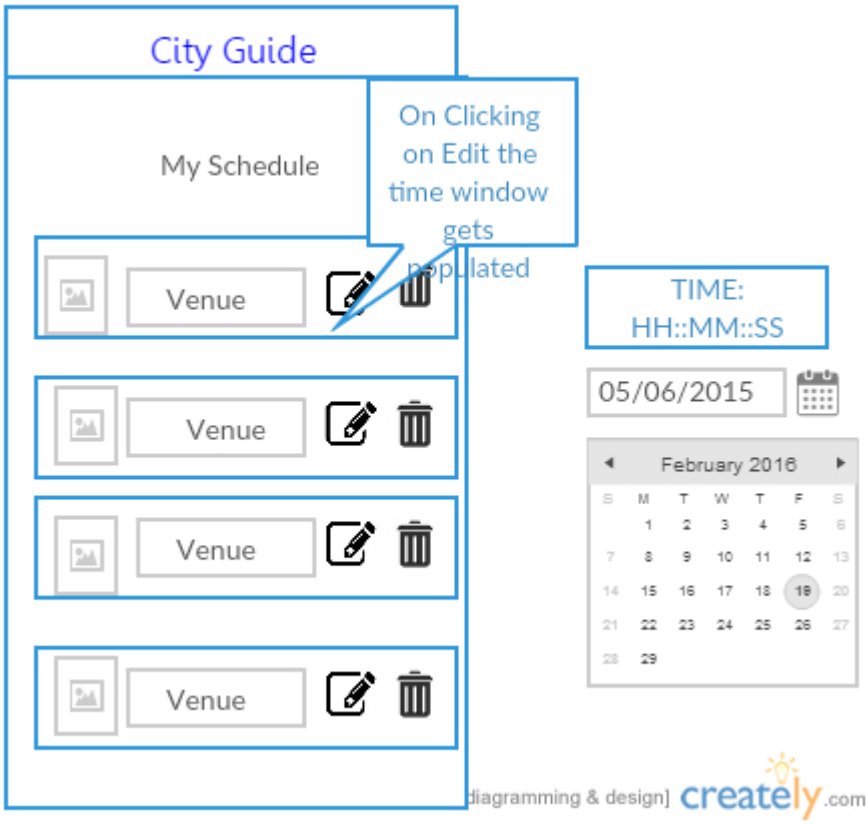
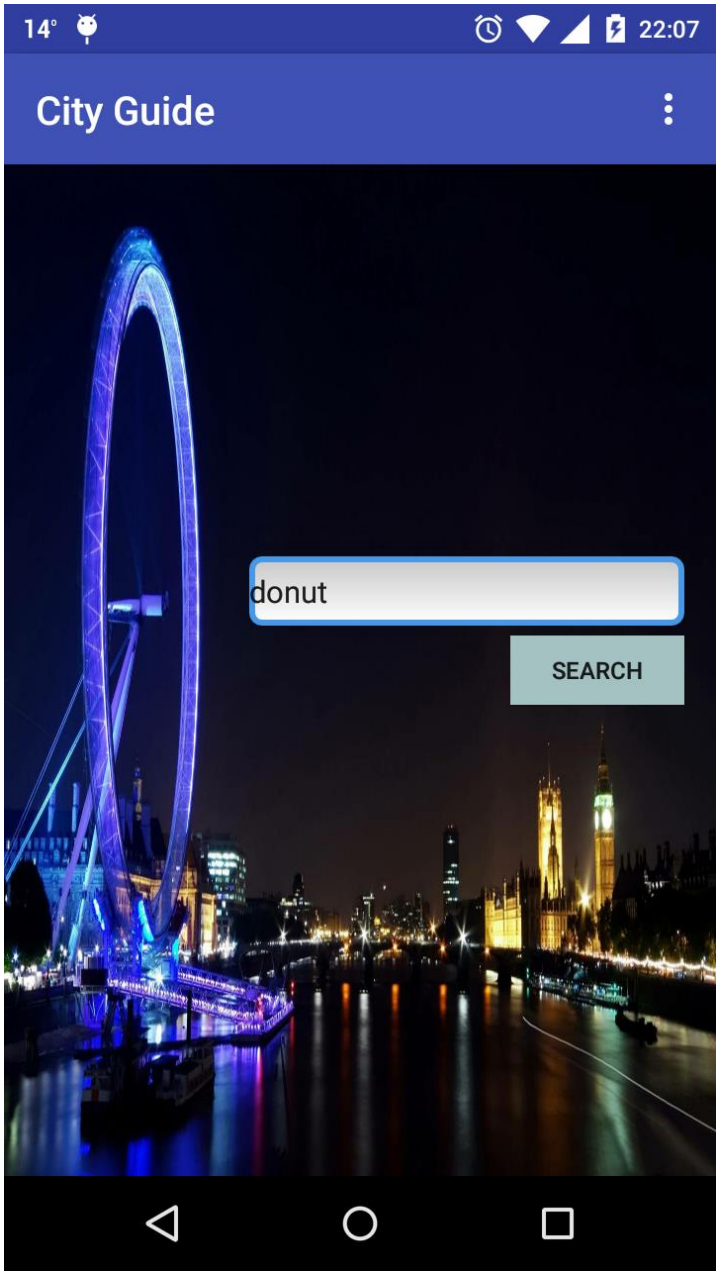
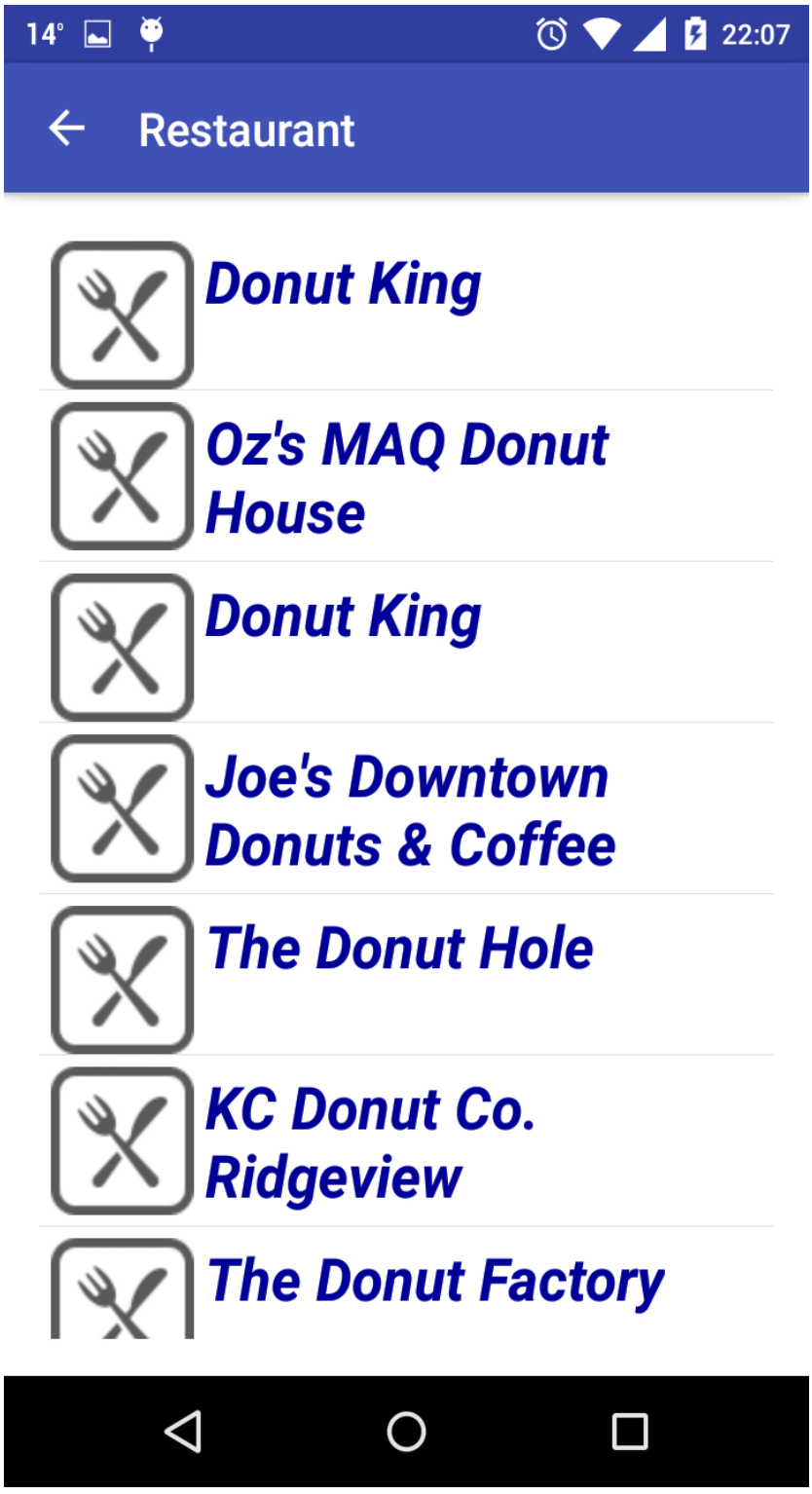


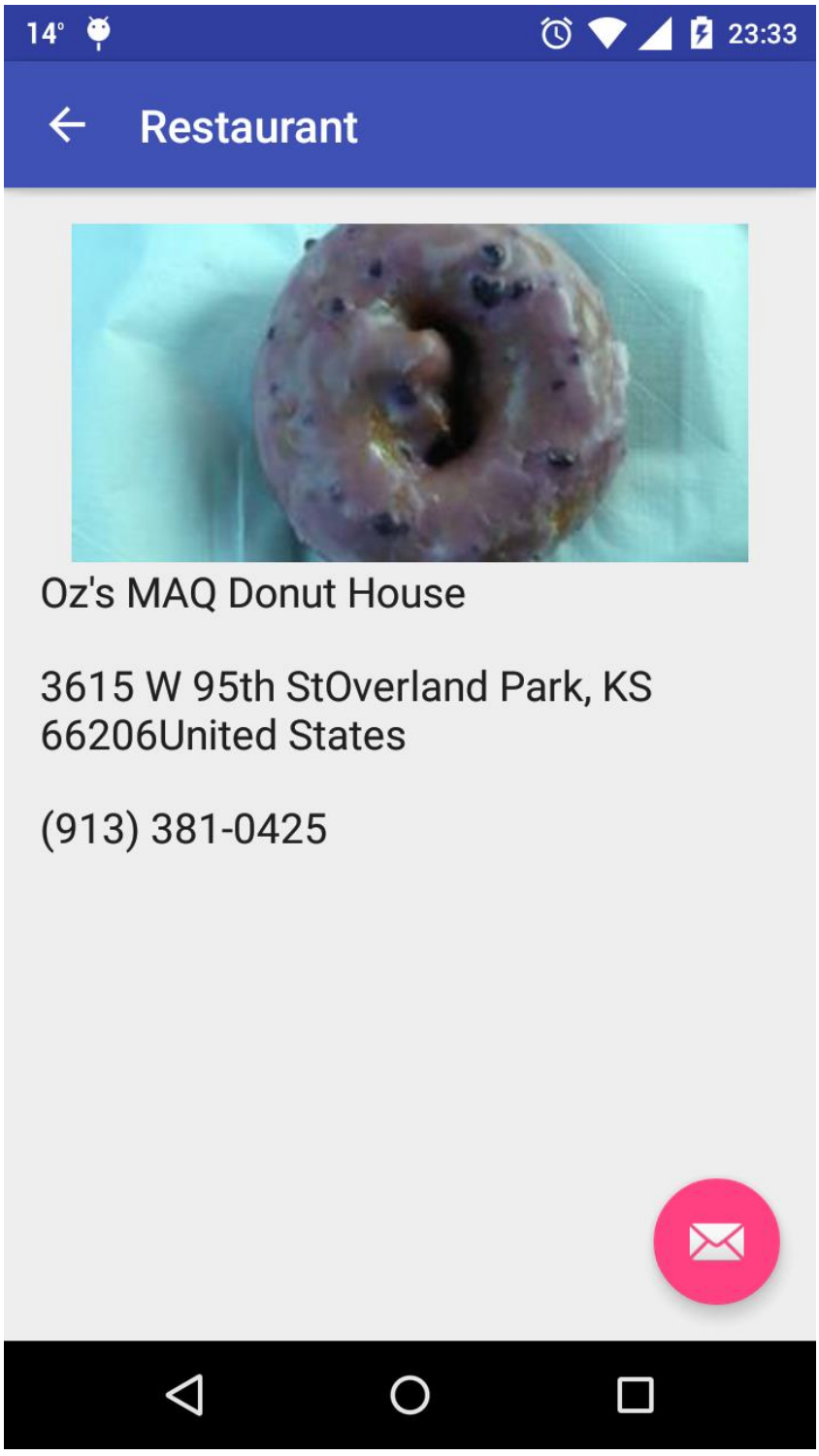
Figure 7: This is Schedule page

Application Screen Shots:

UI:


14°
22:07
City Guide
donut
SEARCH


14°
22:07
Restaurant
Donut King
Oz's MAQ Donut House
Donut King
Joe's Downtown Donuts & Coffee
The Donut Hole
KC Donut Co. Ridgeview
The Donut Factory



Test Cases:

ID	Test_case#1

Title	Search for Restaurants
Pre-Conditions	GPS should be enabled, Internet Service must be Provided, Application Should be opened
Test Steps	Open the Application, Place a query in the search bar, press enter
Expected Results	A list of Restaurants Near and Around the user must be displayed.
Actual Result	A list of Restaurants Near and Around are displayed.

ID	Test_case#2
Title	Search for Museums
Pre-Conditions	GPS should be enabled, Internet Service must be Provided, Application Should be opened
Test Steps	Open the Application, Place a query in the search bar, press enter
Expected Results	A list of Museums Near and Around the user must be displayed.
Actual Result	A list of Museums near and Around are displayed.

ID	Test_case#3
Title	Get Details of a Venue
Pre-Conditions	GPS should be enabled, Internet Service must be Provided, Application Should be opened, Query for a Venue

Test Steps	Query for a key venue, From the results Select a Venue
Expected Results	A New Page Displaying Venue
Actual Result	New Page with venue image, address, Contact details are displayed.

Bibliography:

https://developer.android.com/training/

https://developer.foursquare.com/docs/

https://developers.google.com/maps/documentation/

https://developers.facebook.com/docs/

https://docs.oracle.com/javase/tutorial/

http://www.programmableweb.com/

https://stackoverflow.com/questions/2736389/how-to-pass-an-object-from-one-activity-to-another-on-android