LAB REPORT

Submitted by

PREETAM KONAR [RA2011003010041]

Under the Guidance of

Dr. Jayakumar L

Assistant Professor , Computing Technologies

In partial satisfaction of the requirements for the degree of

BACHELOR OF TECHNOLOGY
in
COMPUTER SCIENCE ENGINEERING

SCHOOL OF COMPUTING

COLLEGE OF ENGINEERING AND TECHNOLOGY

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

KATTANKULATHUR - 603203

JUNE 2022

SRM INSTITUTION OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR-603203

BONAFIDE CERTIFICATE

Certified that this lab report titled **“ ”** is the bonafide work done by Preetam Konar (RA2011003010041) who carried out the lab exercises under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other work.

SIGNATURE
Dr.
SEPM – Course Faculty
Assistant Professor
Department of

ABSTRACT

Now a day mobile phone is a necessary part of the people’s life. There is continuously rising in a number of mobile computing applications, centered on the people’s daily life. In such applications, location dependent systems have been detected as an important application. Such application which presents the architecture and implementation of such a location is commonly known as Smart Travel Guide. We propose architecture of mobile tourist guide system for Android Mobile Phones that is able to provide tourism information to the mobile users conveniently. Our system takes advantage of light-weighted mashup technology that can combine more than one data sources to create value-added services, while overcomes the limitations of mobile devices.

LIST OF FIGURES

LIST OF ABBREVIATIONS

ADK : ANDROID DEVELOPMENT KIT

SDK : SOFTWARE DEVELOPMENT KIT

ADB : ANDROID DEBUG BRIDGE

AOSP : ANDROIOD OPEN SYSTEM PROJECT

API – APPLICATION PROGRAMMING INTERFACE

APK – ANDROID APPLICATION PACKAGE

CSR – CERTIFICATE SIGNING REQUEST.

CSV – COMMON SEPARATED VALUES

FTP – FILE TRANSFER PROTOCOL

GUI – GRAPHIC USER INTERFACE

IDE – INTEGRATED DEVELOPMENT ENVIRONMENT.

OS – OPERATING SYSTEM

UDID – UNIQUE DEVICE IDENTIFIER.

UI – USER INTERFACE.

UX – USER EXPERIENCE

JRON JAVASCRIPT RDF OBJECT NOTATION

JSON JAVASCRIPT OBJECT NOTATION

JSP JAVA SERVER PAGES

JSR JAVA SPECIFICATION REQUEST

JQL JAVA QUERY LANGUAGE

KIF KNOWLEDGE INTERCHANGE FORMAT

AI ARTIFICIAL INTELLIGENCE

AUI ABSTRACT USER INTERFACE

BFO BASIC FORMAL ONTOLOGY

CSS CASCADING STYLE SHEET

TABLE OF CONTENTS

SRM
INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)

Department of Networking and Communications

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	1
Title of Experiment	To identify the Software Project, Create Business Case, Arrive at a Problem Statement
Name of the candidate	Preetam Konar
Team Members	Rishu Shukla, G Abiishek
Register Number	RA2011003010041
Date of Experiment	

Mark Split Up

S.No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
	Total	**10**	

Staff Signature with date

Aim:

To Frame a project team, analyze and identify a Software project. To create a business case and Arrive at a Problem Statement for the <title of the project>

Team Members:

S. No	Register No	Name	Role
1	RA2011003010041	Preetam Konar	Lead/Rep
2	RA2011003010047	Rishu Shukla	Member
3	RA2011003010063	G Abiishek	Member

Project Title: *LocalBuddy*

Project Description:

India, a country with natural beauty and historical places, is visited by many foreigners from different countries. Foreigners here also refers to the Indians who travel out of their hometown and have language barriers. The purpose of foreigners‟ visit to this country is diverse. The most dominant purpose is tourism. However, being foreigners in India, they face some challenges during residing in the country such as lack of getting language and transport information. For meeting up some of the requirements by the travelers, we have come up with an online solution by developing android application.

The purpose of our project is to provide the basic idea on some common conversation in the different places that the travelers need to go while travelling in India. Besides, the project provides the travelers concept of transportation cost of different transport medium in different places of India.

Although people can get some general information regarding traveling over the internet, it is sometimes problematic for the newcomers in a place to get familiar with the new environment. Basically, they face difficulties in communicating and finding proper routes and associated costs for distinct routes.

The application will use the map navigation and a database to store information of various transport mediums and accommodations. The app allows the user to enter the place around which you want to spend your holiday. The in-built algorithm will show various options for accommodation and transport according to the price range and number of people. The user will have an option to choose and make booking through the app only.

The app has lots of further potential for enhancement. The addition of more cities within the app can help broaden the scope of travelers to visiting other cities in India. Ridesharing app integration can also be implanted within the proposed Travel app to further enhance the app user's satisfaction.

Result

Thus, the project team formed, the project is described, the business case was prepared and the problem statement was arrived.

Department of Networking and Communications

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	2
Title of Experiment	Identification of Process Methodology and Stakeholder Description
Name of the candidate	Preetam Konar
Team Members	Rishu Shukla, G Abiishek
Register Number	RA2011003010041
Date of Experiment	

Mark Split Up

S.No	**Description**	**Maximum Mark**	**Mark Obtained**
1	Exercise	5	
2	Viva	5	
	Total	**10**	

Staff Signature with date

Aim

To identify the appropriate Process Model for the project and prepare Stakeholder and User Description.

Team Members:

Sl No	Register No	Name	Role
1	RA2011003010041	Preetam Konar	Rep/Member
2	RA2011003010047	Rishu Shukla	Member
3	RA2011003010063	G Abiishek	Member

Project Title:

Selection of Methodology

Methodology: Agile Methodology

- An agile methodology is an iterative approach to software development.
- Each iteration of agile methodology takes a short time interval of 1 to 4 weeks.
- The agile development process is aligned to deliver the changing business requirement.
- It distributes the software with faster and fe wer changes.
- The single-phase software development takes 6 to 18 months.
- In single-phase development, all the requirement gathering and risks management factors are predicted initially.
- The agile software development process frequently takes the feedback of workable product.

Incorporate information to below table regarding stakeholders of the project [Make use of below examples]

Stakeholder Name	Activity/ Area /Phase	Interest	Influence	Priority (High/ Medium/ Low)
Developers	Development of application	High	High	1
Government	Tourism policy, rules, requirements and practices, Ticket booking of State and Centrally	Med	High	2

	Owned Tourism Sites			
Travel Agencies	Providing transport facilities	High	Med	3
Hotels and Guesthouses	Providing accommodation facilities	High	Med	3
Education Sector	Trips to sites with historical significance	Low	Med	4
Suppliers/Local Enterprises	Providing local delights and delicacies exclusive to the respective place of visit	High	Low	6
Tourists/Users	Practical use	High	High	3

Result

Thus the Project Methodology was identified and the stakeholders were described.

SRM
INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)

Department Of Networking and Communications

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	3
Title of Experiment	System, Functional and Non-Functional Requirements of the Project
Name of the candidate	Preetam Konar
Team Members	Rishu Shukla, G Abiishek
Register Number	RA2011003010041
Date of Experiment	

Mark Split Up

S.No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
	Total	**10**	

Staff Signature with date

Aim

To identify the system, functional and non-functional requirements for the project.

Team Members:

S No	Register No	Name	Role
1	RA2011003010041	Preetam Konar	Rep/Member
2	RA2011003010047	Rishu Shukla	Member
3	RA2011003010063	G Abiishek	Member

Project Title: Local Buddy

System Requirements

OS: Android

Minimum Specifications: Android 8.0 (Oreo)

Online Database

JSON/GSON support

RAM minimum: 2GB

Functional Requirements

- User should be able to give location access to the application.
- User should be able to use maps.
- User should receive a confirmation email upon booking.
- User should be able to make payments through online banking system.
- General Purpose Notification support
- Active Internet
- Map service
- Context providers

Non-Functional Requirements

- Loading speed
- Time taken to deliver server response
- User response time
- Data consumption limits
- Robust
- Scalable
- Diverse
- Display popular destinations
- Load main screen within 5 seconds of opening the app

Result

Thus the requirements were identified and accordingly described.

DEPT. Of Computer Science Engineering

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	4
Title of Experiment	Prepare Project Plan based on scope, Find Job roles and responsibilities, Calculate Project effort based on resources
Name of the candidate	Preetam Konar
Team Members	Rishu Shukla, G Abiishek
Register Number	RA2011003010041
Date of Experiment	

Mark Split Up

S.No	Description	Maximum Mark	Mark Obtained
1	Presentation	5	
2	Requirements document	5	
	Total	**10**	

Staff Signature with date

Aim

To Prepare Project Plan based on scope, Find Job roles and responsibilities, Calculate Project effort based on resources

Team Members:

Sl No	Register No	Name	Role
1	RA2011003010041	Preetam Konar	**Lead**
2	RA2011003010047	Rishu Shukla	**Member**
3	RA2011003010063	G Abiishek	**Member**

Key Issues Driving the Project

Focus Area	Details
Integration Management	● **Project Team Structure-** 2 UI/UX developers and front end developers, 1 fullstack developers ● **Governance Framework-** Based on features of newsletters and updates
Resource Management	● **Managing needs-** People management- Skillful people are required to develop and manage the project Finance management- Minimum budget is required to produce the final product
Stakeholders	● **APIs, Database, Research -** Source of information ● **Developers-** Platform development ● **Hotels, Local Enterprises, Travel Agencies -** Services

	● **Government of India –** Guidelines, Advisory, Taxes ● **Tourists, Local People-** End users
Communication Management	● Communication between tourists (end users) and different agencies ● Real-time schedule (regular synced updates)

Cost and Effort Estimation:

Activity Description	**Subtask Description**	**Effort (in hours/per day or week)**	**Cost (in INR/per month)**
Application Development	To create Android application using Kotlin, Java, XML and MySQL	4 - 6	Full Stack-40K Frontend-25K Backend-45K
Research and Development	Evaluation of software technology trends and incorporating them with updates and patches	4	For Research - 45K For Rolling out patches/updates – 30K
Data Analytics	Administrator of Data, Database Management, Evaluating trends in data	10 With shifts	Data Analyst-90K Database Manager-60K
Customer Care	Customer care	24 With shifts	Customer Care-16K to 18K
Website Maintenance	Weekly website maintenance check	5 hours a week	Web developer-10K

Licensing/ Trademarks/ Copyrighting	Database IDE Server Terms Trademarking Copyrighting	All time	1.2L (one time)
Marketing	Advertisements, Public Relations	8-10 On requirement	80K

Infrastructure/ Resource Cost:

Infrastructure Requirement	Qty	Cost per Qty	Cost per Item
Google Play Store Licence	1	1.5k - 2k	1.5k - 2K
Server firm and other hardware	1	15K	15K
Legal and other government documents	1	1L	1L

Maintenance and Support Cost:

Category	Details	Qty	Cost per qty per annum	Cost per item
People	Network, DB Admin, Customer Support, Developer	4	10 L	40 L
License	Database, server, Middleware, Government License	3	10,000	30,000
Infrastructures	Server Storage and Network	2	20,000	40,000

Project Team Formation:

Name	**Role**	**Responsibilities**
Preetam Konar	Full Stack Developer	UI Development, Coding and Debugging, UI Testing to Optimize Performance
Rishu Shukla	UI/UX Designer and Front End Developer, Quality Assurance	UI Development, and come up with questionnaires, surveys, random quality checks and UI Development
G Abiishek	Database Management, Research, intermediate between team and stakeholders	Managing database, Research for more options and come up with new features, intermediate for communication between stakeholders and development team

Result: Thus, the Project Plan was documented successfully.

Department of Networking and Communications

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	5
Title of Experiment	Prepare Work breakdown structure, Timeline chart, Risk identification table
Name of the candidate	Preetam Konar
Team Members	Rishu Shukla, G Abiishek
Register Number	RA2011003010041
Date of Experiment	

Mark Split Up

S.No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
	Total	**10**	

Staff Signature with date

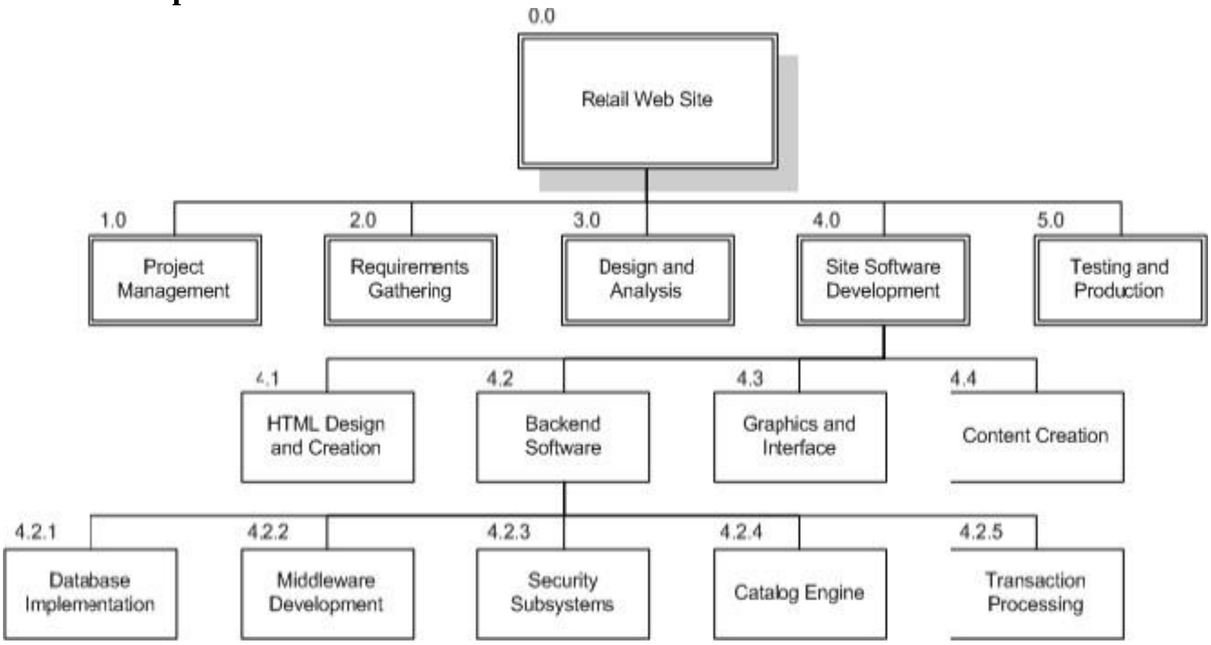
Aim

To Prepare Work breakdown structure, Timeline chart and Risk identification table

Team Members:

Sl No	Register No	Name	Role
1	**RA2011003010041**	**Preetam Konar**	**Rep**
2	**RA2011003010047**	**Rishu Shukla**	**Member**
3	**RA2011003010063**	**G Abiishek**	**Member**

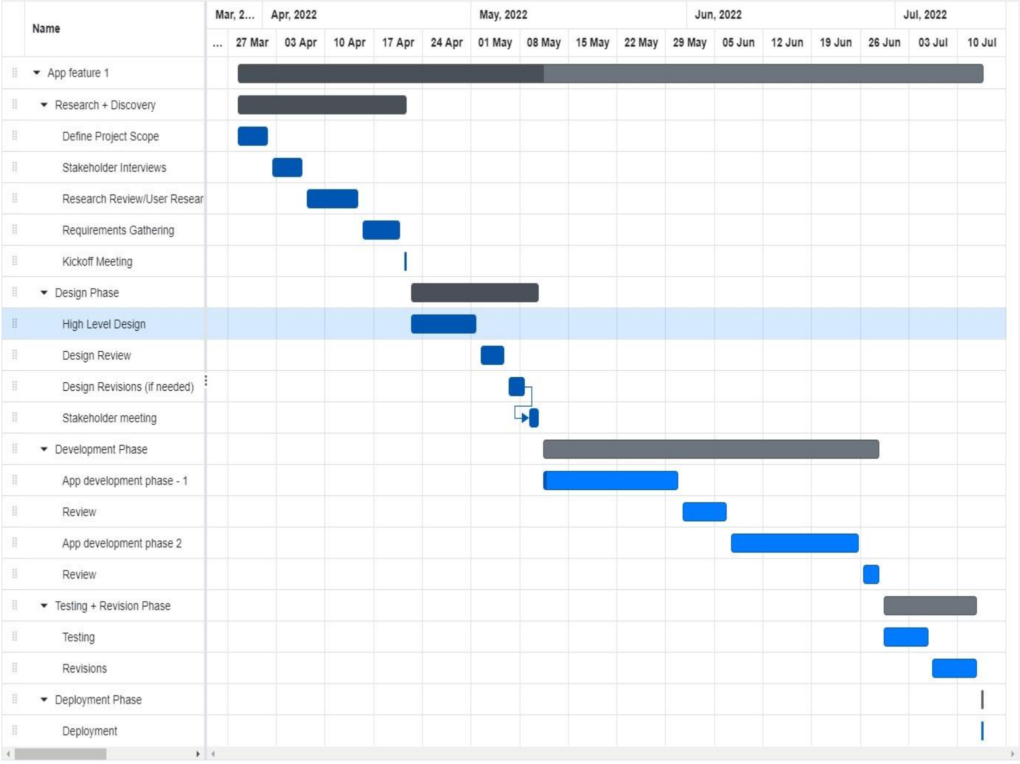
WBS – Examples



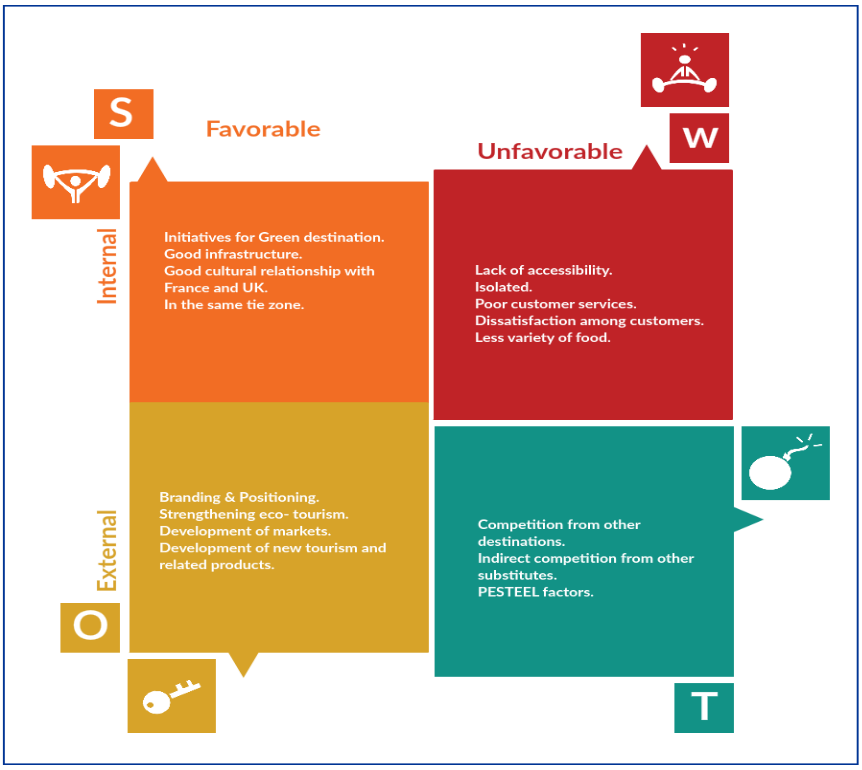
- 0.0 Retail Web Site
- 1.0 Project Management
- 2.0 Requirements Gathering
- 3.0 Analysis & Design
- 4.0 Site Software Development
    - 4.1 HTML Design and Creation
    - 4.2 Backend Software
        - 4.2.1 Database Implementation
        - 4.2.2 Middleware Development
        - 4.2.3 Security Subsystems
        - 4.2.4 Catalog Engine
        - 4.2.5 Transaction Processing
    - 4.3 Graphics and Interface
    - 4.4 Content Creation

- 5.0 Testing and Production

TIMELINE – GANTT CHART



RISK ANALYSIS – SWOT & RMMM



Initiatives for Green destination.
Good infrastructure.
Good cultural relationship with France and UK.
In the same tie zone.

Lack of accessibility.
Isolated.
Poor customer services.
Dissatisfaction among customers.
Less variety of food.

Branding & Positioning.
Strengthening eco- tourism.
Development of markets.
Development of new tourism and related products.

Competition from other destinations.
Indirect competition from other substitutes.
PESTEEL factors.

S. No	Risk Summary	Risk Category	Probability	Impact	RMMM Plan
1.	Requirements not defined properly	Management risk	Medium	Confusion in scope of project causing delay	Gather proper instructions and scope from client and follow communication plan accordingly
2.	Network Security	Technical risk	High	Loss of private data, corruption	Update all security programs periodically, data backup and keep network operation recovery team on sight
3.	Deficiency of sufficient human resources due to unavailability	Budget risk	Medium	Slowing down of development process	More allocation of fund towards human resource and temporary hiring of freelancers
4.	Hardware failure	Technical risk	Medium	Potential of loss of user data, network communication	Redundant communication paths, data mirroring and compulsory vendor agreement, backup equipment

5.	User Interface	Management risk	High	UI is not liked by the users, leading to change in design causing delay	Application should be programmed in such a way that it is compatible with all devices and regular tests and feedback should be taken
6.	Poor testing	Technical risk	Medium	When user use the application, they cannot complete transaction due to this error	Automated testing and manual testing should be conducted and testing should be documented to keep proper record

Result:

Thus, the work breakdown structure with timeline chart and risk table were formulated successfully.

SRM
INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)

Department of Networking and Communications

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	6
Title of Experiment	Design a System Architecture, Use Case and Class Diagram
Name of the candidate	Preetam Konar
Team Members	Rishu Shukla, G Abiishek
Register Number	RA2011003010041
Date of Experiment	

Mark Split Up

S.No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
	Total	**10**	

Staff Signature with date

Aim

To Design a System Architecture, Use case and Class Diagram

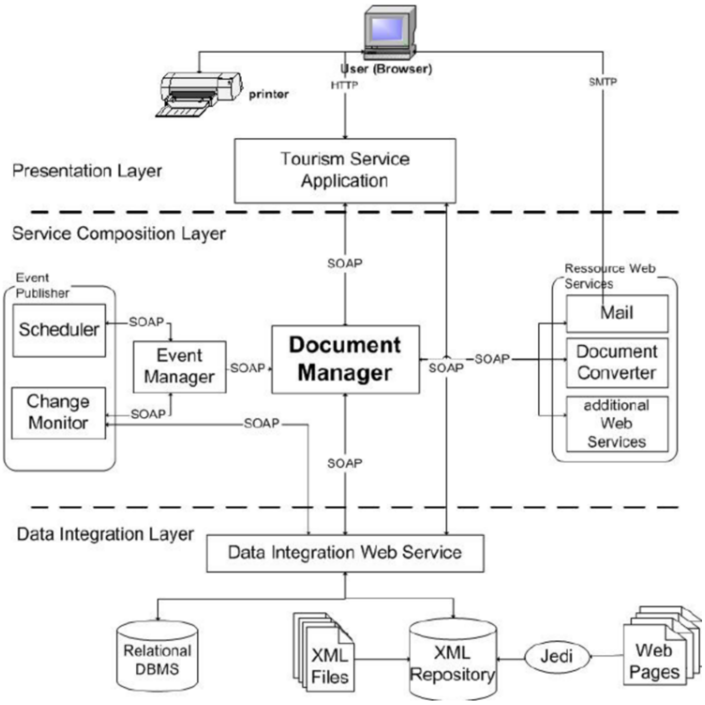
Team Members:

Sl No	**Register No**	**Name**	**Role**
1	**RA2011003010041**	**Preetam Konar**	**Rep**
2	**RA2011003010047**	**Rishu Shukla**	**Member**
3	**RA2011003010063**	**G Abiishek**	**Member**

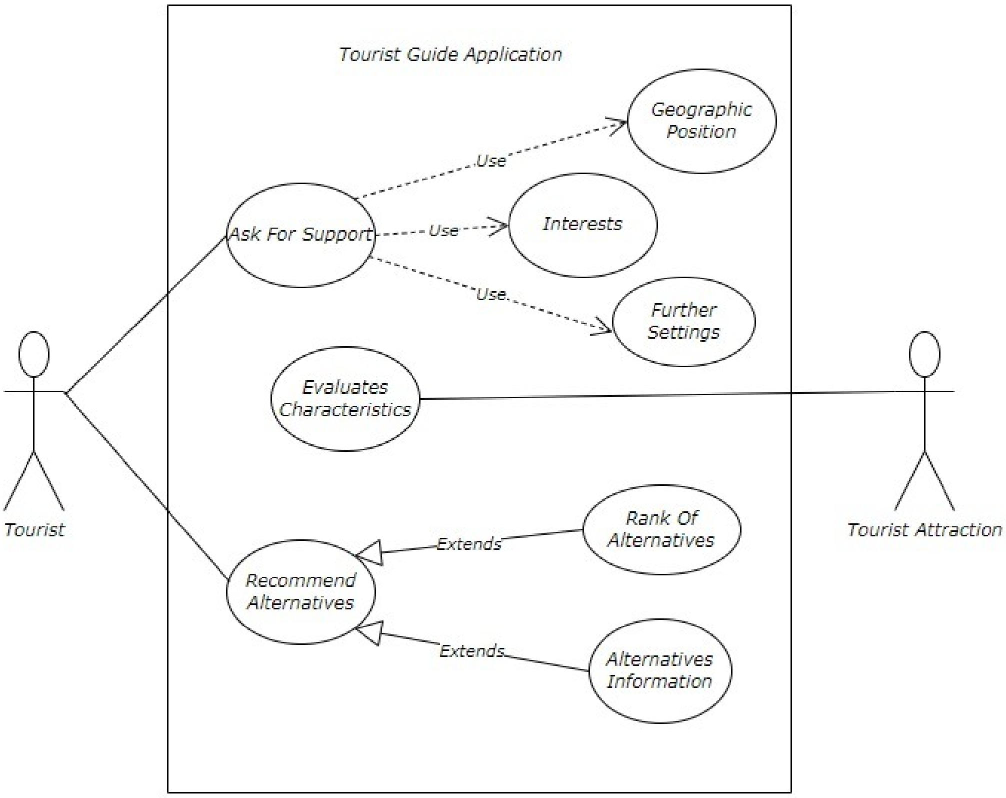
System Architecture

Feature driven development is an **agile methodology** for mobile app development process that is model driven. At the beginning of the project, a model is designed that specifies several features and feature iterations. It consists of different processes. These are:

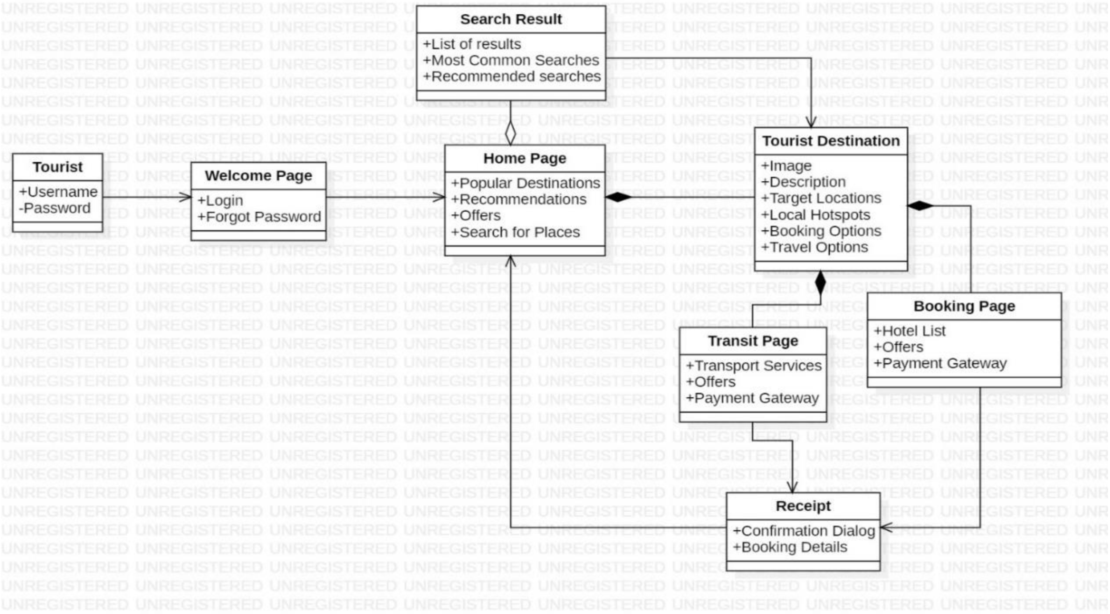
- Develop an overall model
- Build feature list
- Plan by feature
- Design by feature
- Build be feature



Use Case Diagram



Class Diagram



Result: Thus, the system architecture, use case and class diagram created successfully.

SRM
INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)

School of Computing

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	7
Title of Experiment	Design a Entity relationship diagram
Name of the candidate	Preetam Konar
Team Members	Rishu Shukla, G Abiishek
Register Number	RA2011003010041
Date of Experiment	

Mark Split Up

S. No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
	Total	**10**	

Staff Signature with date

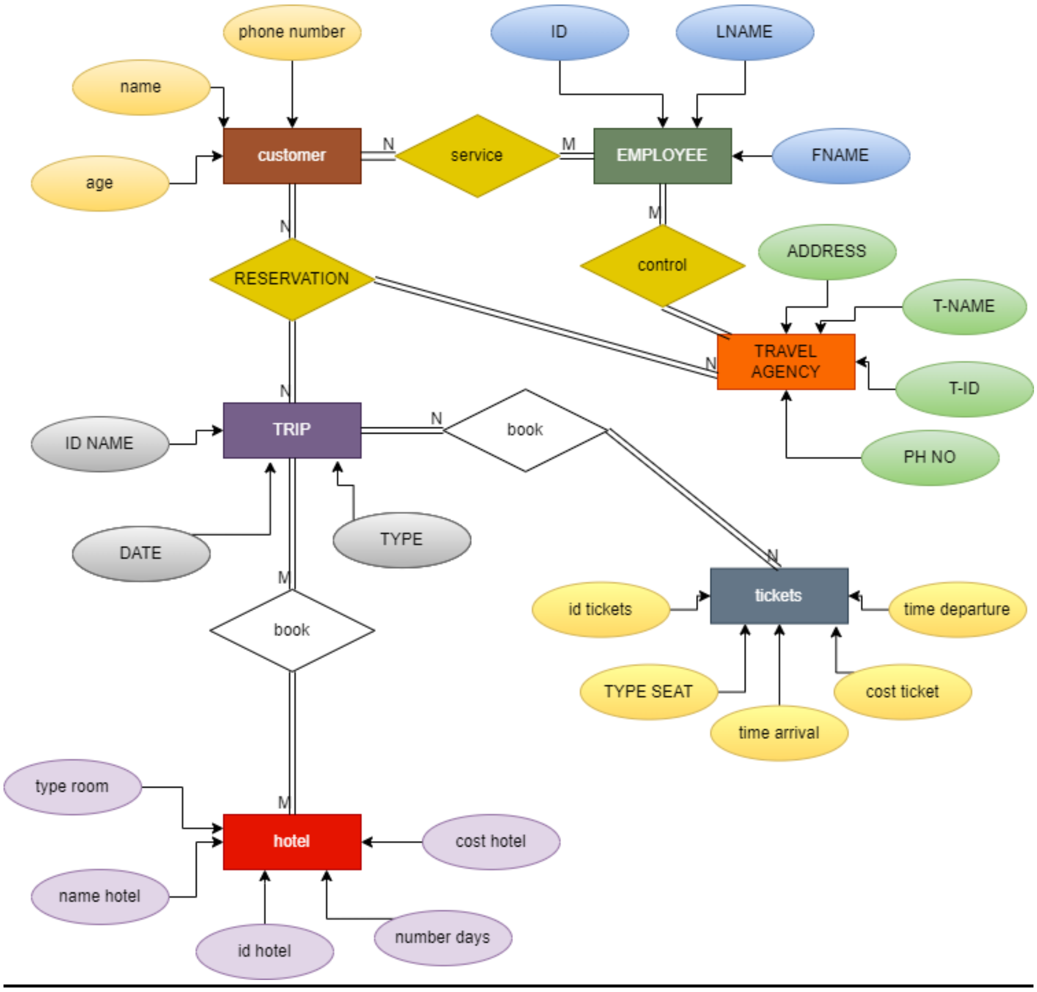
Aim

To create the Entity Relationship Diagram

Team Members:

S No	Register No	Name	Role
1	RA2011003010041	Preetam Konar	Rep
2	RA2011003010047	Rishu Shukla	Member
3	RA2011003010063	G Abiishek	Member

ER Diagram



Result:

Thus, the entity relationship diagram was created successfully.

SRM
INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)

School of Computing

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	8
Title of Experiment	Develop a Data Flow Diagram (Process-Up to Level 1)
Name of the candidate	Preetam Konar
Team Members	Rishu Shukla, G Abiishek
Register Number	RA2011003010041
Date of Experiment	

Mark Split Up

S. No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
	Total	**10**	

Staff Signature with date

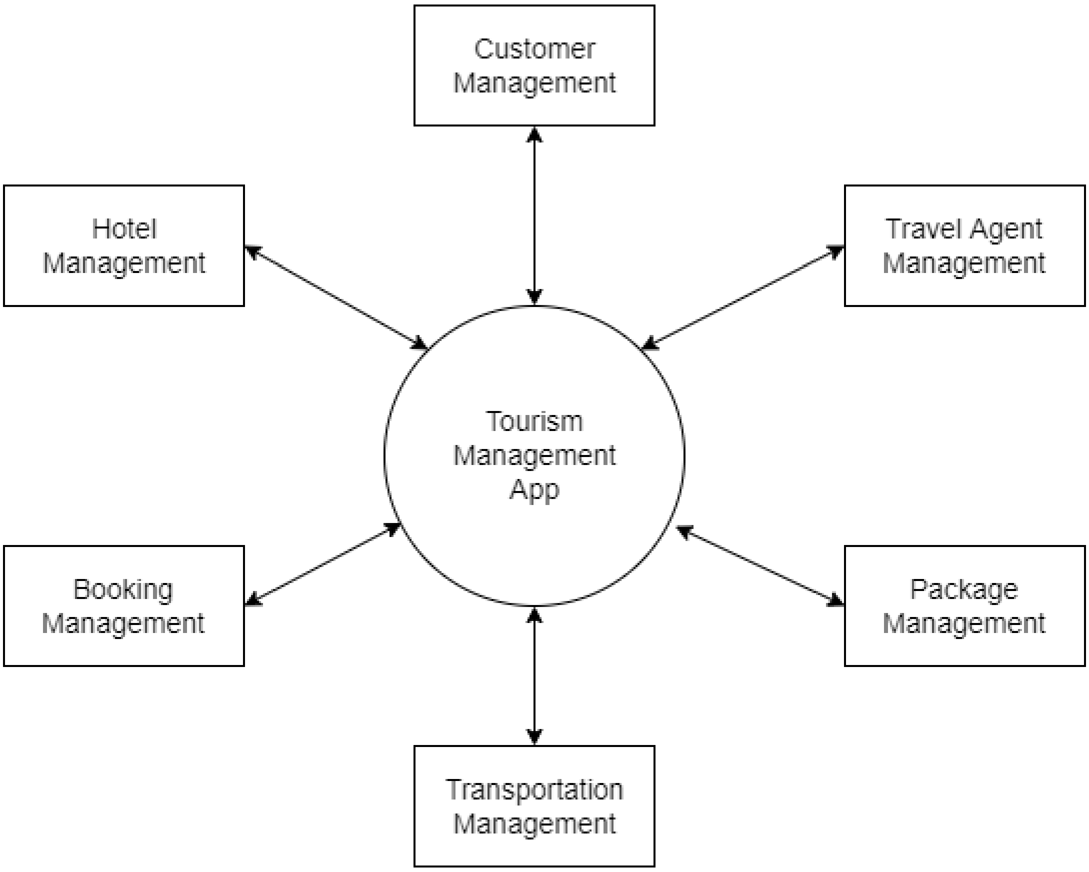
Aim

To develop the data flow diagram up to level 1 for the <project name>

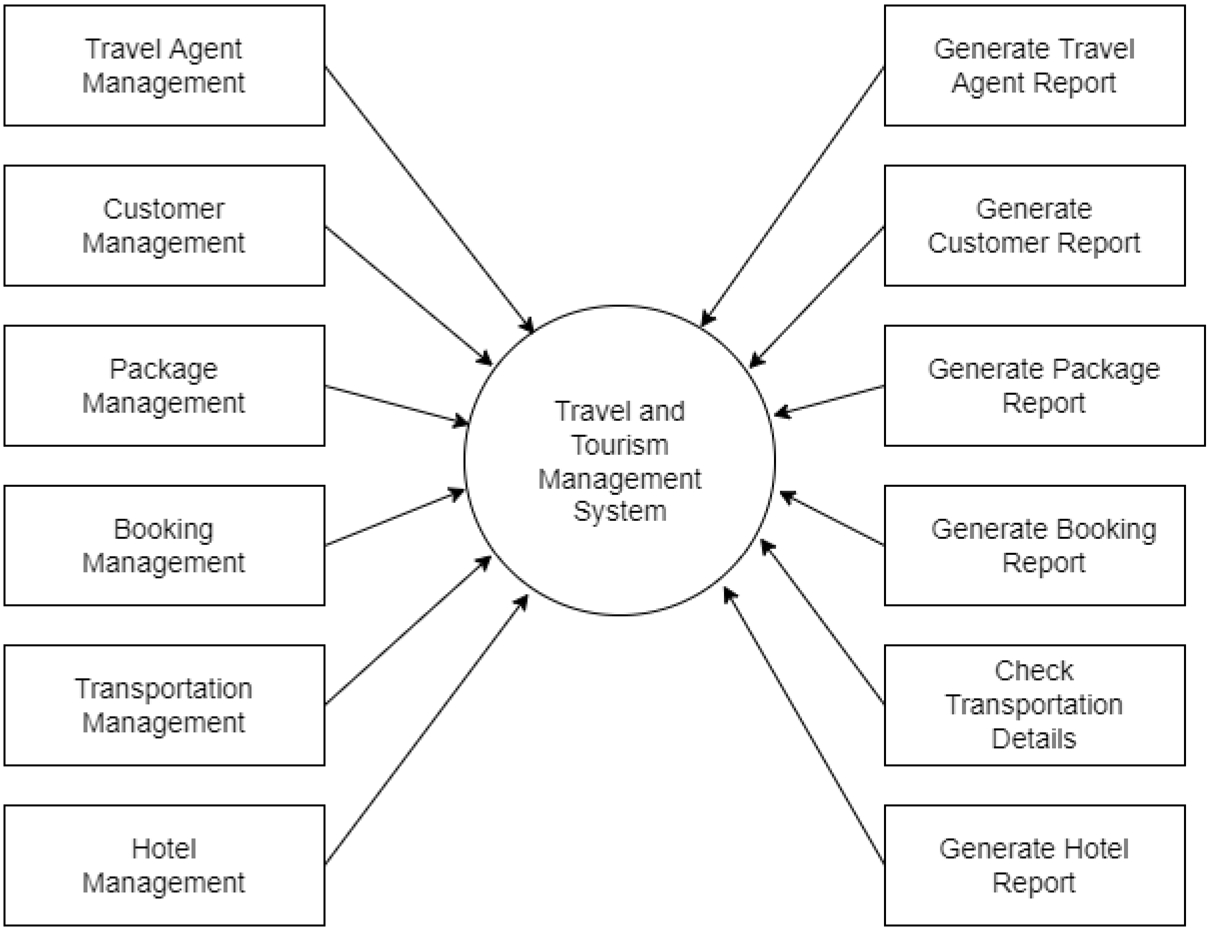
Team Members:

S No	Register No	Name	Role
1	**RA2011003010041**	**Preetam Konar**	**Rep**
2	**RA2011003010047**	**Rishu Shukla**	**Member**
3	**RA2011003010063**	**G Abiishek**	**Member**

Data Flow Diagram



Zero Level DFD - Tourism Management System



First Level DFD - Travel and Tourism Management System

Result:

Thus, the data flow diagrams have been created for the Local Buddy, The Tourist Guide App

SRM
INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)

School of Computing

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	9
Title of Experiment	Design a Sequence and Collaboration Diagram
Name of the candidate	Preetam Konar
Team Members	Rishu Shukla, G Abiishek
Register Number	RA2011003010041
Date of Experiment	

Mark Split Up

S. No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
	Total	**10**	

Staff Signature with date

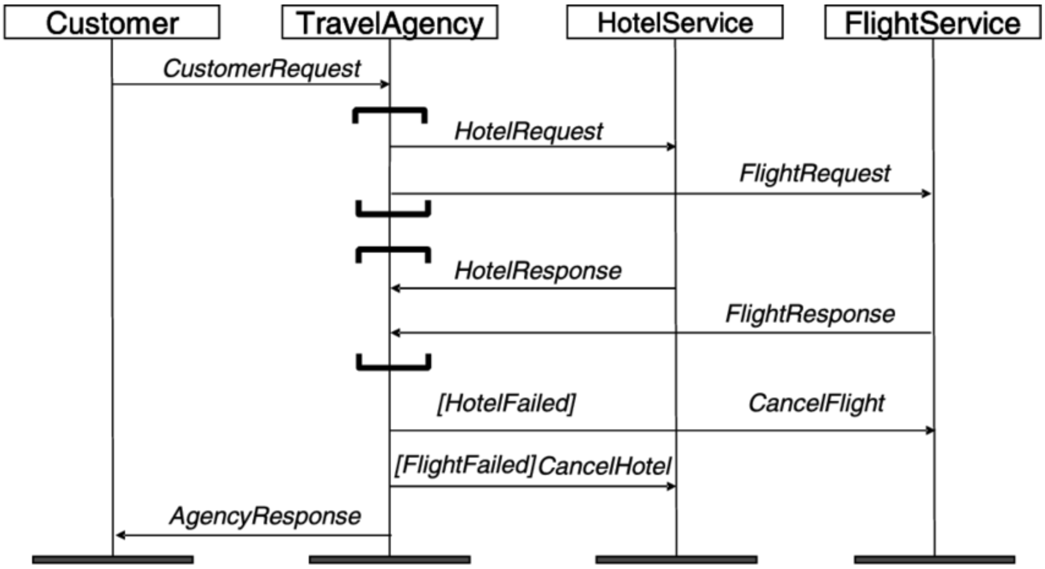
Aim

To create the sequence and collaboration diagram for the <project name>

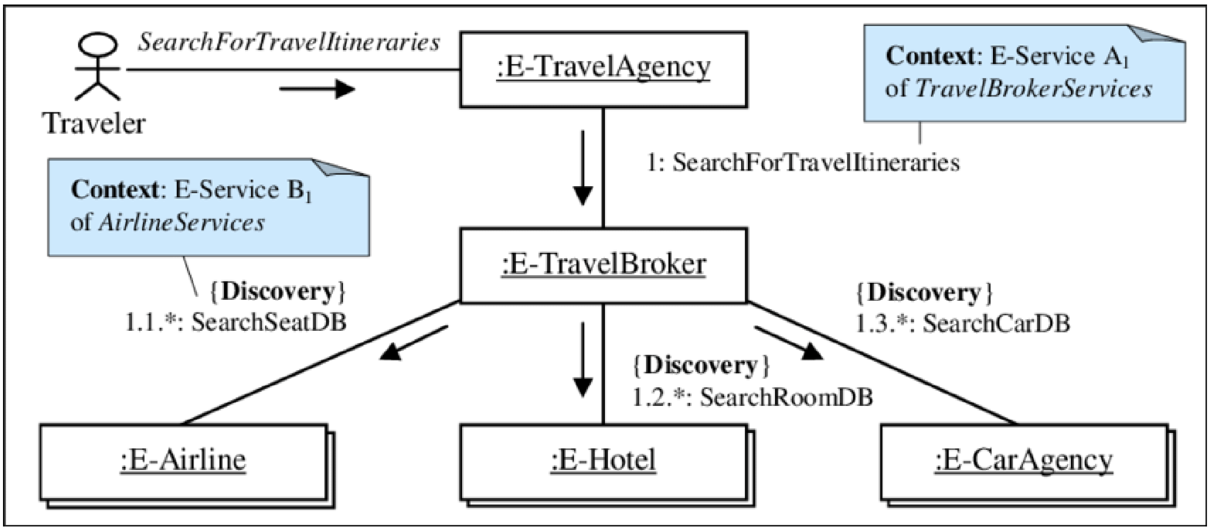
Team Members:

S No	**Register No**	**Name**	**Role**
1	**RA2011003010041**	**Preetam Konar**	**Lead/Rep**
2	**RA2011003010047**	**Rishu Shukla**	**Member**
3	**RA2011003010063**	**G Abiishek**	**Member**

Sequence Diagram



Collaborations Diagram :


SearchForTravelItineraries
Traveler
:E-TravelAgency
Context: E-Service A1 of TravelBrokerServices
1: SearchForTravelItineraries
Context: E-Service B1 of AirlineServices
:E-TravelBroker
{Discovery} 1.1.*: SearchSeatDB
{Discovery} 1.3.*: SearchCarDB
{Discovery} 1.2.*: SearchRoomDB
:E-Airline
:E-Hotel
:E-CarAgency

Result:

Thus, the sequence and collaboration diagrams were created for the Local Buddy.

SRM INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)

School of Computing

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	10
Title of Experiment	Develop a Testing Framework/User Interface
Name of the candidate	Preetam Konar
Team Members	Rishu Shukla, G Abiishek
Register Number	RA2011003010041
Date of Experiment	

Mark Split Up

S. No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
	Total	**10**	

Staff Signature with date

Aim

To develop the testing framework and/or user interface framework for the LocalBuddy App

Team Members:

S No	Register No	Name	Role
1	RA2011003010041	Preetam Konar	Rep/Member
2	RA2011003010047	Rishu Shukla	Member
3	RA2011003010063	Abiishek G Kumar	Member

Executive Summary:

- The Scope of testing of our travel assistance application includes Testing Application Functionalities with consistent data with Computed-Generated tests ,making test cases for different modules to check if the code can withstand boundary cases that can arise if an exception arises. Eg: Test for payment portal or for selecting places
- The objective of this testing includes testing of all modules and to check if any exception exists in any of the modules.
- Regression (Re-running the test cases after change) testing would be important part of software practice that would ensure our application still functions as expected after any code changes , updates, or improvements.
- Lastly critical path testing would be aimed at exploring the functionality used by typical daily activities.

Test Plan:

We as team have decided that the testing will follow top-down approach, as it will go well with Agile method of software development. With this method the testing can begin at the start of the project with continuous integration between development and testing. Agile Testing methodology will be continuous and helps us finish our projects before deadline. We will 1st go through the components, then test the archetype and then other minute details. After completing the functional testing we will move on to non-functional testing.

Scope of Testing

The scope of testing for application includes testing Event API with consistent data along with Computed-Generated test cases, making test-cases for different modules to check if the code canwithstand boundary cases can arise if an exception arises.

Functional Testing

Testing of our application Local Buddy is done with four stages:

Unit Testing	Unit testing is the first level of testing and will be performed by the developers themselves. It is a process of ensuring individual components of piece of software at the code level are functional and work as they were designed to.
System testing	System testing is a black box testing method used to evaluate completed and integrated system, as a whole , to ensure if it meets specific requirements.
Integration testing	After each unit is thoroughly tested, itis integrated with other units to create modules or components that are designed to perform specific tasks or activities. The integration of the various modules are tested in the phase.
Acceptance testing	Acceptance testing is the last phase of functional testing and it is used to access whether or not the final piece of software is ready for delivery. If not, user feedback is taken and following changes are made.

Modules and the aspects they will be testing

User interface testing	The testing of this particular module code will consist of checking if the application displace all the required buttons and check if the settings panel is in with the main screen. The layout and search bar should be placed in a user friendly manner .
API Integartion	The testing of the code of this module deal with the connectivity and processing of different applications. We will check that how much traffic that the app can handle, in short the number of applications running simultaneously to provide data such as scraping tool.
Payment	The need of this module is to ensure the security reliability and performance of payment gateway by encrypting and securing the payment details between user and merchant while providing a smooth payment experience
Backup	In this module, we will be checking if the backup is being made and is being retrieved as and when the user wants to.

Non-Functional

Performance testing	It is an non functional testing technique used to determine how an app will behave under various conditions
Security testing	With the presence of cloud base testing platforms and cyber attacks, there is a growing concern and need for security of data being used and stored in software. It is an non functional and software testing technique used to determine if the information and data in a system is protected.
Data latency	It is an key metric that helps to determine the effectiveness of the application this makes the faster delivery of the data that is much more important.
Availability	It is an user friendly app. Loging in is effective.

Types of testing, methodology, tools

Category	Methodology	Tools required
Functional requirements	Manual	1. Acceptance testing 2. White box testing 3. Black box testing 4. Unit testing 5. System testing 6. Integration testing
Non functional requirements	Manual User or crowd validation	1. Authentication 2. Security 3. Data latency 4. Performance 5. Availability

Result:

Thus, the testing framework/user interface framework has been created for the Local Buddy App.

School of Computing

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	11
Title of Experiment	Test Cases
Name of the candidate	Preetam Konar
Team Members	Rishu Shukla, G Abiishek
Register Number	RA2011003010041
Date of Experiment	

Mark Split Up

S. No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
	Total	**10**	

Staff Signature with date

Aim

To develop the test cases manual for the LOCAL BUDDY.

Team Members:

S No	Register No	Name	Role
1	RA2011003010041	Preetam Konar	Rep/Member
2	RA2011003010047	Rishu Shukla	Member
3	RA2011003010063	G Abiishek	Member

Test Case

Functional Test Cases

Test ID (#)	Test Scenario	Test Case	Execution Steps	Expected Outcome	Actual Outcome	Status	Remarks
TS_AG_REGISTRATION_001	New user registration	Accept valid user information	1. User clicks on the new user registration button. 2. Enters user details like phone number and email. 3. Verifies email and phone number through OTP. 4. User registered	User should be taken to the next page to enter more personal information and information about themself.	-	-	In progress

TS_AG_LOGIN_002	Verify Login functionality of application login page	Enter valid user id and password.	Successful login	User should be taken to the next page for entering more user details	-	-	In progress
TS_AG_ACCESS_RECENT	Verify the functionality recent places recycler view	Click on recent places recycler view button	Leads to recent places details activity	User should be taken to the details activity where the details of recent page will be shown	-	-	In progress
TS_AG_ACCESS_TOP	Verify the functionality top places recycler view	Click on top places recycler view button	Leads to top places details activity	User should be taken to the details activity where the details of top page will be shown	-	-	In progress

Non-Functional Test Cases

Test ID (#)	Test Scenario	Test Case	Execution Steps	Expected Outcome	Actual Outcome	Status	Remarks
TS_AG_NF_001	Security	Check application security	Bugs, errors and virus checks are conducted on the application.	Provides secure user login and registration	-	-	In progress
TS_AG_NF_002	Low Latency of data	Check speed of data updation.	Changes are made in the back end and the updation time for each code is recorded. Code having least data latency chosen.	Data is updated on the interface without delay	-	-	In progress
TS_AG_NF_003	Performance	Check response time.	Changes are made in the back end and the response time for each code is recorded. Code having least response time is chosen.	The site response time should be minimum.	-	-	In progress
TS_AG_NF_004	Availability	Check service availability.	-	The service should be available all the time.	-	-	In progress

Result: Thus, the test case manual has been created for project Local Buddy

SRM INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)

School of Computing

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	12
Title of Experiment	Manual Test Case Reporting
Name of the candidate	Preetam Konar
Team Members	Rishu Shukla, G Abiishek
Register Number	RA2011003010041
Date of Experiment	

Mark Split Up

S. No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
	Total	**10**	

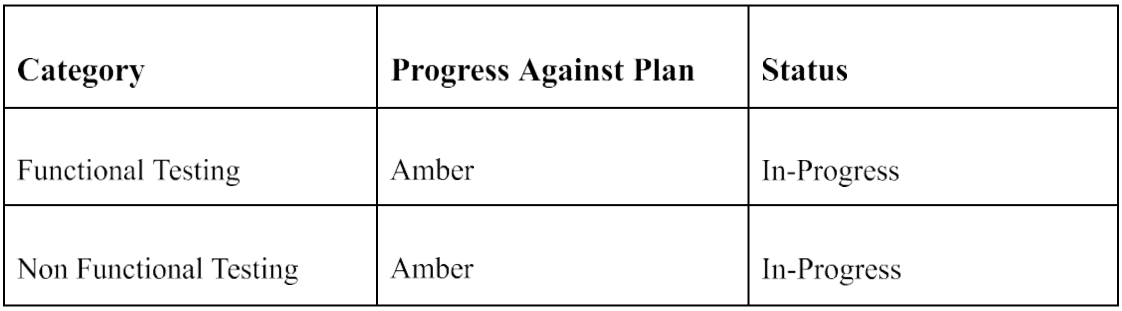
Staff Signature with date

Aim

To prepare the manual test case report for the Eden Springs.

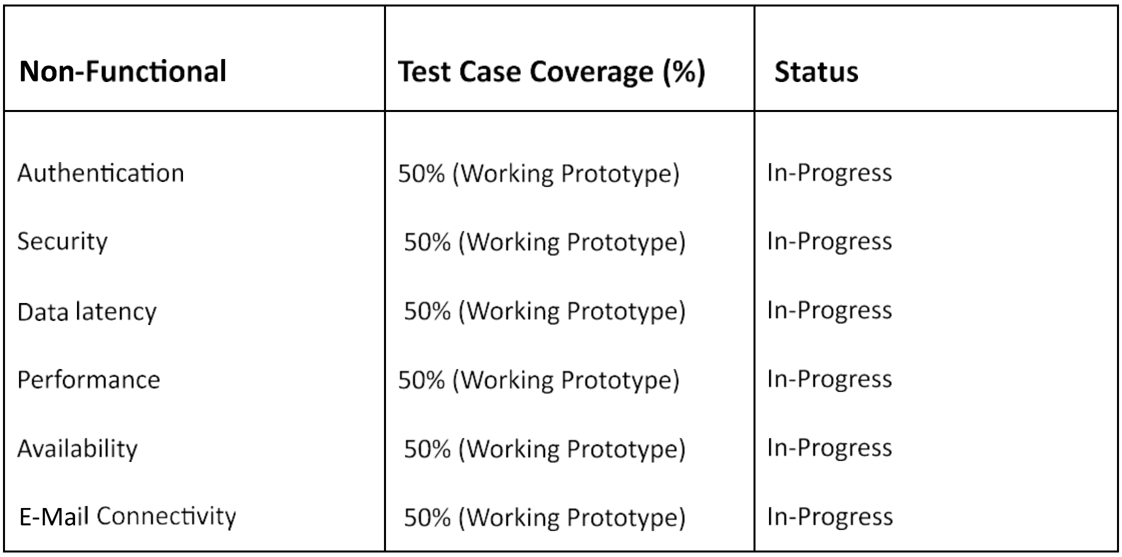
Team Members:

S No	Register No	Name	Role
1	**RA2011003010041**	**Preetam Konar**	**Rep/Member**
2	**RA2011003010047**	**Rishu Shukla**	**Member**
3	**RA2011003010063**	**G Abiishek**	**Member**



Category	Progress Against Plan	Status
Functional Testing	Amber	In-Progress
Non Functional Testing	Amber	In-Progress

Functional	Test Case Coverage (%)	Status
User Registration	75%	In-Progress
Login id	80%	In-Progress
Services provided	70%	In-Progress
Budget estimation	70%	In-Progress
Errors when screen sizes vary	40%	In-Progress



Non-Functional	Test Case Coverage (%)	Status
Authentication	50% (Working Prototype)	In-Progress
Security	50% (Working Prototype)	In-Progress
Data latency	50% (Working Prototype)	In-Progress
Performance	50% (Working Prototype)	In-Progress
Availability	50% (Working Prototype)	In-Progress
E-Mail Connectivity	50% (Working Prototype)	In-Progress

Result:

Thus, the test case report has been created for the Local Buddy App .

SRM
INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)

School of Computing

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	13
Title of Experiment	Provide the details of Architecture Design/Framework/Implementation
Name of the candidate	Preetam Konar
Team Members	Rishu Shukla, G Abiishek
Register Number	RA2011003010041
Date of Experiment	

Mark Split Up

S. No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
	Total	**10**	

Staff Signature with date

Aim

To provide the details of architectural design/framework/implementation

Team Members:

S No	Register No	Name	Role
1	RA2011003010041	Preetam Konar	Rep/Member
2	RA2011003010047	Rishu Shukla	Member
3	RA2011003010063	G Abiishek	Member

Pure Native and Kotlin Framework is Used in this Project With Other functionalities .

MODULE 1

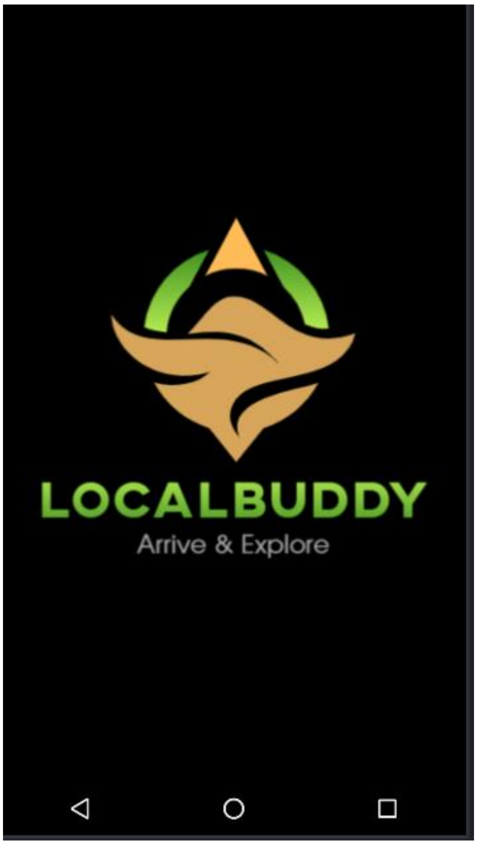
Logo Section

```
package com.konar.localbuddy

import android.content.Intent
import android.os.Bundle
import android.os.CountDownTimer
import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity() {
    private var timer: CountDownTimer? = null

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        timer = object : CountDownTimer(2000, 1000) {
            override fun onTick(millisUntilFinished: Long) {

            }

            override fun onFinish() {
                val intent = Intent(this@MainActivity,
LoginActivity::class.java)
                startActivity(intent)
                finish()
            }

        }.start()

    }
    override fun onDestroy() {
        super.onDestroy()
        timer?.cancel()
        timer = null
    }
}
```

XML

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/black"
    tools:context=".MainActivity">

    <ImageView
        android:id="@+id/iv_logo"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:scaleType="center"
        android:contentDescription="Logo"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:srcCompat="@drawable/ic_logo" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

Sample O/P



MODULE 2

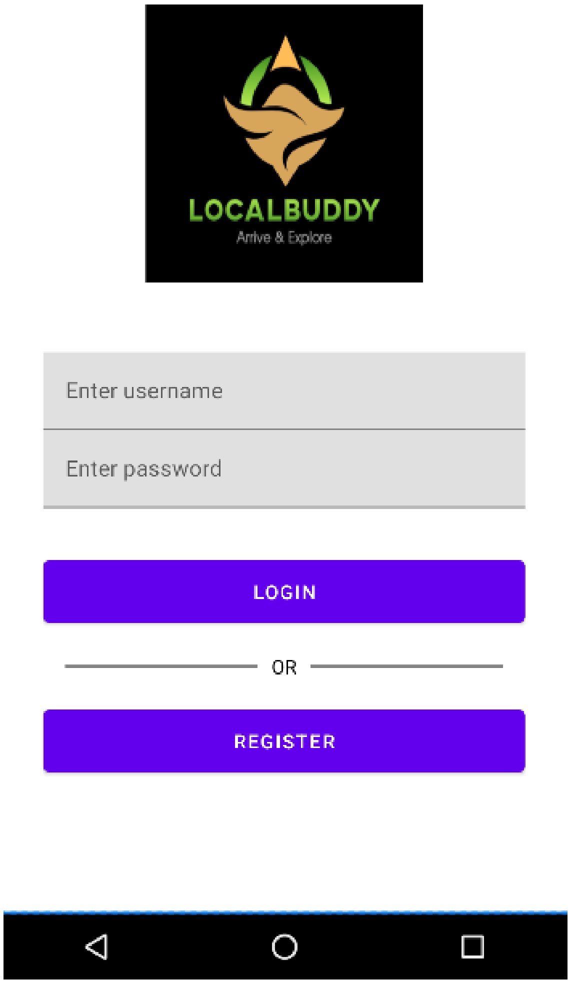
Login Section

```kotlin
package com.konar.localbuddy

import android.content.Intent
import android.os.Bundle
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity
import androidx.lifecycle.lifecycleScope
import com.konar.localbuddy.databinding.ActivityLoginBinding
import com.konar.localbuddy.db.UserDao
import com.konar.localbuddy.db.UserEntity
import kotlinx.coroutines.launch

class LoginActivity : AppCompatActivity() {

    private lateinit var binding: ActivityLoginBinding

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        binding = ActivityLoginBinding.inflate(layoutInflater)
        setContentView(binding.root)

        val userDao = (application as UserApp).db.userDao()

        binding.btnRegister.setOnClickListener {
            registerUser(userDao)
        }

        binding.btnLogin.setOnClickListener {
            loginUser(userDao)
        }
    }

    private fun registerUser(userDao: UserDao) {
        var flag = 0
        val username = binding.etUsername.text.toString()
        val password = binding.etPassword.text.toString()
        if (username.isNotBlank() && password.isNotBlank()) {
            lifecycleScope.launch {
                userDao.fetchAllUsernames().collect {
                    if (it.contains(username)) {
                        Toast.makeText(
                            this@LoginActivity,
                            "Username already exists!!",
                            Toast.LENGTH_SHORT
                        ).show()
                    } else {
                        lifecycleScope.launch {
                            userDao.insert(
                                UserEntity(
                                    userName = username,
                                    password = password
                                )
                            )
                            Toast.makeText(
                                this@LoginActivity,
                                "Successfully registered!!",
                                Toast.LENGTH_SHORT
                            ).show()
                        }
                    }
                }
            }
        } else {
            Toast.makeText(this, "Username or password can't be blank!!",
Toast.LENGTH_SHORT).show()
        }
    }

    private fun loginUser(userDao: UserDao) {
        val username = binding.etUsername.text.toString()
        val password = binding.etPassword.text.toString()

        if (username.isNotBlank() && password.isNotBlank()) {
            lifecycleScope.launch {
                userDao.fetchUserByUsername(username).collect {
                    if (password == it) {
                        val intent = Intent(this@LoginActivity,
BaseActivity::class.java)
                        intent.putExtra("Username", username)
                        startActivity(intent)
                        finish()
                    } else {
                        Toast.makeText(
                            this@LoginActivity,
                            "Wrong username or password entered!!",
                            Toast.LENGTH_SHORT
                        ).show()
                    }
                }
            }
        } else {
            Toast.makeText(this, "Username or password can't be blank!!",
Toast.LENGTH_SHORT).show()
        }
    }
}
```

XML

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".LoginActivity">

    <ImageView
        android:id="@+id/iv_logo"
        android:layout_width="200dp"
        android:layout_height="200dp"
        app:layout_constraintTop_toTopOf="parent"
        android:scaleType="fitXY"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        android:src="@drawable/ic_logo"
        android:layout_margin="30dp"/>

    <LinearLayout

        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/iv_logo"
        android:layout_marginLeft="32dp"
        android:layout_marginRight="32dp"
        android:layout_marginTop="50dp"
        android:orientation="vertical">

        <com.google.android.material.textfield.TextInputLayout
            android:layout_width="match_parent"
            android:layout_height="56dp">

            <EditText
                android:id="@+id/et_username"
                android:layout_width="match_parent"
                android:layout_height="match_parent"
                android:hint="Enter username"/>

        </com.google.android.material.textfield.TextInputLayout>

        <com.google.android.material.textfield.TextInputLayout
            android:layout_width="match_parent"
            android:layout_height="56dp">

            <EditText
                android:id="@+id/et_password"
                android:layout_width="match_parent"
                android:layout_height="match_parent"
                android:hint="Enter password"/>

        </com.google.android.material.textfield.TextInputLayout>

        <Button
            android:id="@+id/btn_login"
            android:layout_width="match_parent"
            android:layout_height="56dp"
            android:layout_marginTop="32dp"
            android:text="Login"/>


        <RelativeLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_centerVertical="true"
            android:layout_marginTop="16dp">

            <TextView
                android:id="@+id/tvText"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_centerInParent="true"
                android:layout_marginLeft="10dp"
                android:layout_marginRight="10dp"
                android:text="OR"
                android:textColor="#000000"/>

            <View
                android:layout_width="match_parent"
                android:layout_height="1dp"
                android:layout_centerVertical="true"
                android:layout_marginLeft="16dp"
                android:layout_toLeftOf="@id/tvText"
                android:background="#000000" />

            <View
                android:layout_width="match_parent"
                android:layout_height="1dp"
                android:layout_centerVertical="true"
                android:layout_marginRight="16dp"
                android:layout_toRightOf="@id/tvText"
                android:background="#000000" />

        </RelativeLayout>

        <Button
            android:id="@+id/btn_register"
            android:layout_width="match_parent"
            android:layout_height="56dp"
            android:layout_marginTop="16dp"
            android:text="Register" />

    </LinearLayout>

</androidx.constraintlayout.widget.ConstraintLayout>
```

Sample O/P



MODULE 3

Home Screen

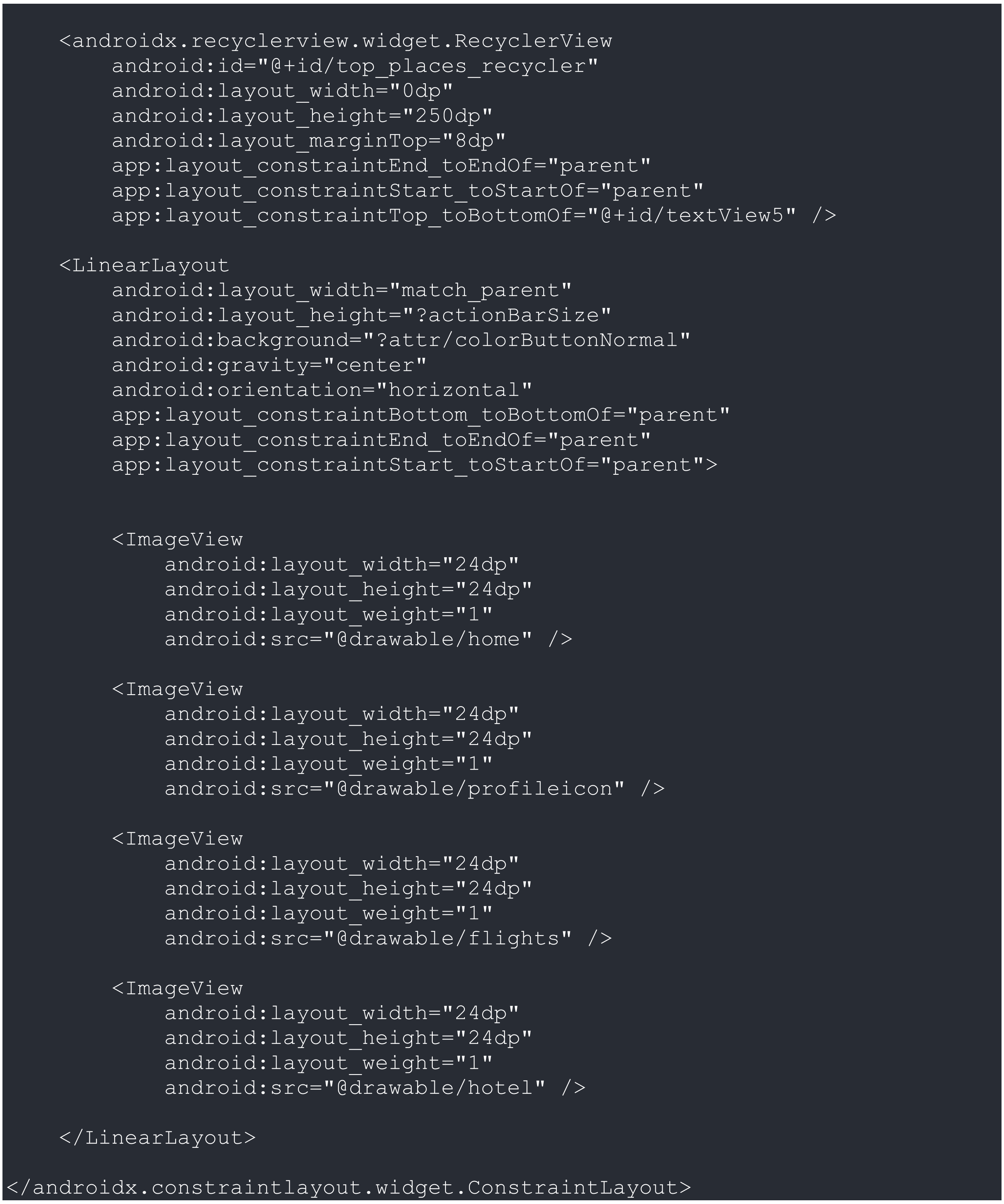
```
package com.konar.localbuddy

import android.content.Intent
import android.os.Bundle
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity
import androidx.recyclerview.widget.LinearLayoutManager
import com.konar.localbuddy.adapter.RecentPlacesAdapter
import com.konar.localbuddy.adapter.TopPlacesAdapter
import com.konar.localbuddy.databinding.ActivityBaseBinding
import com.konar.localbuddy.model.RecentPlacesConstants
import com.konar.localbuddy.model.RecentPlacesModel
import com.konar.localbuddy.model.TopPlacesConstant

class BaseActivity : AppCompatActivity() {

    private lateinit var binding: ActivityBaseBinding
    private lateinit var topPlacesAdapter: TopPlacesAdapter
    private lateinit var recentPlacesAdapter: RecentPlacesAdapter
    private var backPressedTime: Long = 0
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        binding = ActivityBaseBinding.inflate(layoutInflater)
        setContentView(binding.root)

        val username = intent.getStringExtra("Username")
        binding.tvUsername.text = username

        initRecentPlacesRecyclerView()
        initTopPlacesRecyclerView()


    }

    private fun initTopPlacesRecyclerView() {
        topPlacesAdapter = TopPlacesAdapter()
        binding.topPlacesRecycler.layoutManager =
            LinearLayoutManager(this, LinearLayoutManager.VERTICAL, false)
        topPlacesAdapter.setList(TopPlacesConstant.getTopPlacesList())
        binding.topPlacesRecycler.adapter = topPlacesAdapter
    }

    private fun initRecentPlacesRecyclerView() {
        recentPlacesAdapter = RecentPlacesAdapter({ place ->
onClickRecentPlace(place) })
        binding.recentRecycler.layoutManager =
            LinearLayoutManager(this, LinearLayoutManager.HORIZONTAL,
false)

recentPlacesAdapter.setList(RecentPlacesConstants.getRecentPlacesList())
        binding.recentRecycler.adapter = recentPlacesAdapter
    }

    private fun onClickRecentPlace(place: RecentPlacesModel) {
        val intent = Intent(this, DetailsActivity::class.java)
        intent.putExtra("RecentPlaceName", place.placeName)
        intent.putExtra("RecentPlaceImage", place.imageURL)
        intent.putExtra("RecentPlaceCountry", place.countryName)
        intent.putExtra("RecentPlacePrice", place.price)
        startActivity(intent)
    }

    override fun onBackPressed() {
        if (backPressedTime + 3000 > System.currentTimeMillis()) {
            super.onBackPressed()
        } else {
            Toast.makeText(this, "Press back again to exit the app!",
Toast.LENGTH_SHORT).show()
        }
        backPressedTime = System.currentTimeMillis()
    }
}
```

XML

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
```

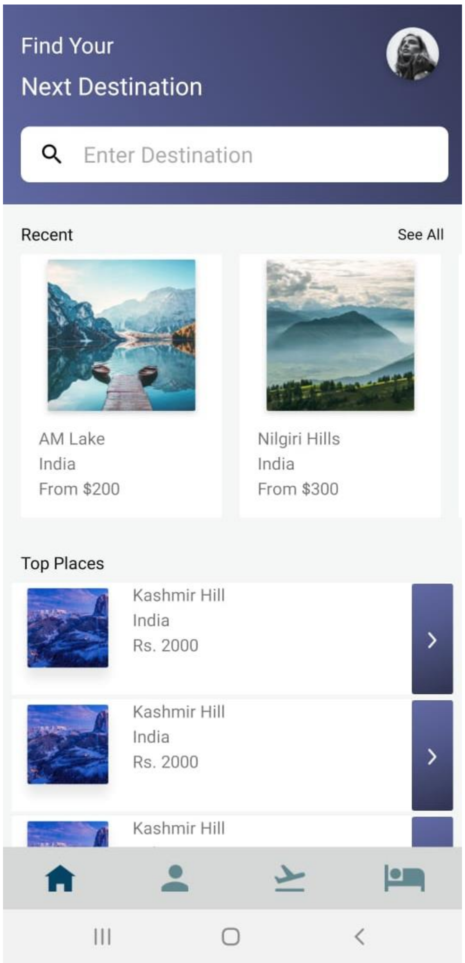
```
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#F5F7F7"
    tools:context=".MainActivity">

    <androidx.constraintlayout.widget.ConstraintLayout
        android:id="@+id/constraintLayout"
        android:layout_width="0dp"
        android:layout_height="180dp"
        android:background="@drawable/home_bg"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent">


        <TextView
            android:id="@+id/textView"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="16dp"
            android:layout_marginTop="24dp"
            android:text="Find Your"
            android:textColor="#FFFFFF"
            android:textSize="18sp"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent" />

        <TextView
            android:id="@+id/textView2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="8dp"
            android:text="Next Destination"
            android:textColor="#FFFFFF"
            android:textSize="20sp"
            app:layout_constraintStart_toStartOf="@+id/textView"
            app:layout_constraintTop_toBottomOf="@+id/textView" />

        <ImageView
            android:id="@+id/iv_profile_photo"
            android:layout_width="64dp"
            android:layout_height="64dp"
            android:layout_marginTop="16dp"
            android:layout_marginEnd="12dp"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintTop_toTopOf="parent"
            app:srcCompat="@drawable/profile" />

        <EditText
            android:id="@+id/editText"
            android:layout_width="0dp"
            android:layout_height="50dp"
            android:layout_marginTop="22dp"
            android:background="@drawable/search_bg"
            android:drawableLeft="@drawable/ic_search_black_24dp"
            android:drawablePadding="16dp"
            android:ems="10"
            android:inputType="textPersonName"
```

```xml
            android:paddingStart="16dp"
            android:hint="Enter Destination"
            android:textColorHint="@android:color/darker_gray"
            android:textColor="#62878F"
            app:layout_constraintEnd_toEndOf="@+id/iv_profile_photo"
            app:layout_constraintStart_toStartOf="@+id/textView2"
            app:layout_constraintTop_toBottomOf="@+id/textView2" />

        <TextView
            android:id="@+id/tv_username"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Username"
            android:textColor="@color/white"
            app:layout_constraintEnd_toEndOf="@+id/iv_profile_photo"
            app:layout_constraintStart_toStartOf="@id/iv_profile_photo"
            app:layout_constraintTop_toBottomOf="@id/iv_profile_photo" />
    </androidx.constraintlayout.widget.ConstraintLayout>

    <TextView
        android:id="@+id/textView3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:layout_marginTop="16dp"
        android:text="Recent"
        android:textColor="@android:color/background_dark"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/constraintLayout" />

    <TextView
        android:id="@+id/textView4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginEnd="16dp"
        android:text="See All"
        android:textColor="@android:color/background_dark"
        android:textSize="12sp"
        app:layout_constraintBottom_toBottomOf="@+id/textView3"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toTopOf="@+id/textView3" />

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/recent_recycler"
        android:layout_width="0dp"
        android:layout_height="250dp"
        android:layout_marginTop="8dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textView3" />

    <TextView
        android:id="@+id/textView5"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:layout_marginTop="16dp"
        android:text="Top Places"
        android:textColor="@android:color/background_dark"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/recent_recycler" />
```



```
    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/top_places_recycler"
        android:layout_width="0dp"
        android:layout_height="250dp"
        android:layout_marginTop="8dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textView5" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="?actionBarSize"
        android:background="?attr/colorButtonNormal"
        android:gravity="center"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent">


        <ImageView
            android:layout_width="24dp"
            android:layout_height="24dp"
            android:layout_weight="1"
            android:src="@drawable/home" />

        <ImageView
            android:layout_width="24dp"
            android:layout_height="24dp"
            android:layout_weight="1"
            android:src="@drawable/profileicon" />

        <ImageView
            android:layout_width="24dp"
            android:layout_height="24dp"
            android:layout_weight="1"
            android:src="@drawable/flights" />

        <ImageView
            android:layout_width="24dp"
            android:layout_height="24dp"
            android:layout_weight="1"
            android:src="@drawable/hotel" />

    </LinearLayout>

</androidx.constraintlayout.widget.ConstraintLayout>
```

Sample O/P



Find Your
Next Destination

Enter Destination

Recent

See All

AM Lake
India
From $200

Nilgiri Hills
India
From $300

Top Places

Kashmir Hill
India
Rs. 2000

Kashmir Hill
India
Rs. 2000

Kashmir Hill

Result:

Thus, the details of architecture used with source code along with the screenshots were provided.

CONCLUSION

In this paper, we presented the design and implementation of a mobile application called Smart Travel Guide, with which mobile users can get tourism guidance information they need anytime and anywhere. By Smart Travel Guide, users can get an attraction’s detailed information, including text, picture and video. In particular, Smart Travel Guide can provide users with location-based information, which can be browsed or queried through a map. User can search the nearby attractions after he or she configures the distance between the current location and the view spots. When the user moves out of the current location, the mobile phone will automatically send its new position to the server side, and the corresponding attraction list will be received by the user.

REFERENCES

[1] "What are some good tourism planning apps," Quora, [Online]. Available: www.quora.com/what-aresome-

[2] S. Li, Food Phone Application . cs.sjsu.edu, 2010, pp. 1-40.

[3] DELOUCHE and P. TSNOBILADZE, Dynamic Music Creation on a Smartphone. sonoscaphes.com, 2013.

[4] Y. Dzezhyts, Android application development. Haaga Helia, 2015, pp. 1-40.

[5] Iversen and M. Eierman, Learning Mobile App Development. Pearson Education, Inc, 2013, pp. 1-350.

[6] Nilanchala , Javatechig | Resources for Developers, 'Android ScrollView Example | JavaTechig', 2015. [Online]. Available: http://javatechig.com/android/android-scrollviewexample. [Accessed: 11- May- 2015].

[7] S. Montoro, Mobile application for obtain information from our geolocation TRAVEL GUIDE, 1st ed. Barcelona: University of Politechnica De Catalunya, 2014, pp. 5-70.

[8] A. Singhal, Location - Based Mobile App for Android Platform, 1st ed. Austin: University of Texus, 2010, pp. 5-95.

[9] D. Jinendra et al. Smart Travel Guide: Application for Android Mobile, 1st ed. ijecscse.org, 2012, pp. 1-6.

[10] H. Shu, City Guide over Android, 1st ed. Norway: Norweigian University of Science and Technology, 2010, pp. 5-95.

APPENDIX (CODE)

MODULE 1

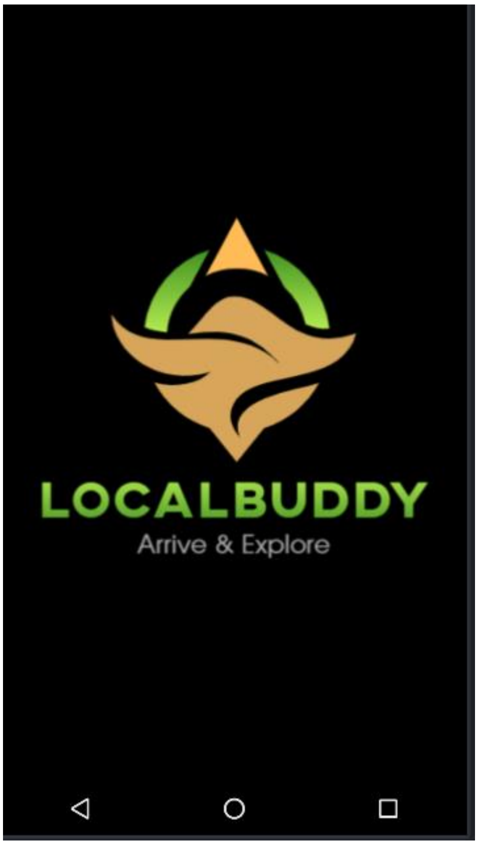
Logo Section

```
package com.konar.localbuddy

import android.content.Intent
import android.os.Bundle
import android.os.CountDownTimer
import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity() {
    private var timer: CountDownTimer? = null

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        timer = object : CountDownTimer(2000, 1000) {
            override fun onTick(millisUntilFinished: Long) {

            }

            override fun onFinish() {
                val intent = Intent(this@MainActivity,
LoginActivity::class.java)
                startActivity(intent)
                finish()
            }

        }.start()

    }

    override fun onDestroy() {
        super.onDestroy()
        timer?.cancel()
        timer = null
    }
}
```

XML

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/black"
    tools:context=".MainActivity">

    <ImageView
        android:id="@+id/iv_logo"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:scaleType="center"
        android:contentDescription="Logo"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:srcCompat="@drawable/ic_logo" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

Sample O/P



MODULE 2

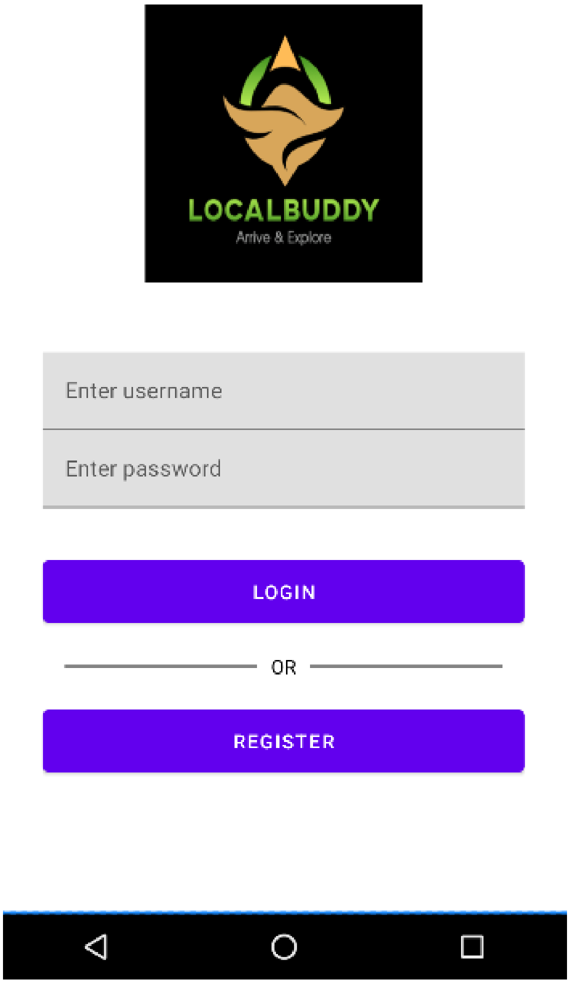
Login Section

```
package com.konar.localbuddy

import android.content.Intent
import android.os.Bundle
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity
import androidx.lifecycle.lifecycleScope
import com.konar.localbuddy.databinding.ActivityLoginBinding
import com.konar.localbuddy.db.UserDao
import com.konar.localbuddy.db.UserEntity
import kotlinx.coroutines.launch

class LoginActivity : AppCompatActivity() {

    private lateinit var binding: ActivityLoginBinding
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        binding = ActivityLoginBinding.inflate(layoutInflater)
        setContentView(binding.root)

        val userDao = (application as UserApp).db.userDao()

        binding.btnRegister.setOnClickListener {
            registerUser(userDao)
        }

        binding.btnLogin.setOnClickListener {
            loginUser(userDao)
        }
    }

    private fun registerUser(userDao: UserDao) {
        var flag = 0
        val username = binding.etUsername.text.toString()
        val password = binding.etPassword.text.toString()
        if (username.isNotBlank() && password.isNotBlank()) {
            lifecycleScope.launch {
                userDao.fetchAllUsernames().collect {
                    if (it.contains(username)) {
                        Toast.makeText(
                            this@LoginActivity,
                            "Username already exists!!",
                            Toast.LENGTH_SHORT
                        ).show()
                    } else {
                        lifecycleScope.launch {
                            userDao.insert(
                                UserEntity(
                                    userName = username,
                                    password = password
                                )
                            )
                            Toast.makeText(
                                this@LoginActivity,
                                "Successfully registered!!",
                                Toast.LENGTH_SHORT
                            ).show()
                        }
                    }
                }
            }
        } else {
            Toast.makeText(this, "Username or password can't be blank!!",
Toast.LENGTH_SHORT).show()
        }
    }

    private fun loginUser(userDao: UserDao) {
        val username = binding.etUsername.text.toString()
        val password = binding.etPassword.text.toString()

        if (username.isNotBlank() && password.isNotBlank()) {
            lifecycleScope.launch {
                userDao.fetchUserByUsername(username).collect {
                    if (password == it) {
                        val intent = Intent(this@LoginActivity,
BaseActivity::class.java)
                        intent.putExtra("Username", username)
                        startActivity(intent)
                        finish()
                    } else {
                        Toast.makeText(
                            this@LoginActivity,
                            "Wrong username or password entered!!",
                            Toast.LENGTH_SHORT
                        ).show()
                    }
                }
            }
        } else {
            Toast.makeText(this, "Username or password can't be blank!!",
Toast.LENGTH_SHORT).show()
        }
    }
}
```

XML

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".LoginActivity">

    <ImageView
        android:id="@+id/iv_logo"
        android:layout_width="200dp"
        android:layout_height="200dp"
        app:layout_constraintTop_toTopOf="parent"
        android:scaleType="fitXY"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        android:src="@drawable/ic_logo"
        android:layout_margin="30dp"/>

    <LinearLayout

        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/iv_logo"
        android:layout_marginLeft="32dp"
        android:layout_marginRight="32dp"
        android:layout_marginTop="50dp"
        android:orientation="vertical">

        <com.google.android.material.textfield.TextInputLayout
            android:layout_width="match_parent"
            android:layout_height="56dp">

            <EditText
                android:id="@+id/et_username"
                android:layout_width="match_parent"
                android:layout_height="match_parent"
                android:hint="Enter username"/>

        </com.google.android.material.textfield.TextInputLayout>

        <com.google.android.material.textfield.TextInputLayout
            android:layout_width="match_parent"
            android:layout_height="56dp">

            <EditText
                android:id="@+id/et_password"
                android:layout_width="match_parent"
                android:layout_height="match_parent"
                android:hint="Enter password"/>

        </com.google.android.material.textfield.TextInputLayout>

        <Button
            android:id="@+id/btn_login"
            android:layout_width="match_parent"
            android:layout_height="56dp"
            android:layout_marginTop="32dp"
            android:text="Login"/>


        <RelativeLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_centerVertical="true"
            android:layout_marginTop="16dp">

            <TextView
                android:id="@+id/tvText"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_centerInParent="true"
                android:layout_marginLeft="10dp"
                android:layout_marginRight="10dp"
                android:text="OR"
                android:textColor="#000000"/>

            <View
                android:layout_width="match_parent"
                android:layout_height="1dp"
                android:layout_centerVertical="true"
                android:layout_marginLeft="16dp"
                android:layout_toLeftOf="@id/tvText"
                android:background="#000000" />

            <View
                android:layout_width="match_parent"
                android:layout_height="1dp"
                android:layout_centerVertical="true"
                android:layout_marginRight="16dp"
                android:layout_toRightOf="@id/tvText"
                android:background="#000000" />

        </RelativeLayout>

        <Button
            android:id="@+id/btn_register"
            android:layout_width="match_parent"
            android:layout_height="56dp"
            android:layout_marginTop="16dp"
            android:text="Register" />

    </LinearLayout>

</androidx.constraintlayout.widget.ConstraintLayout>
```

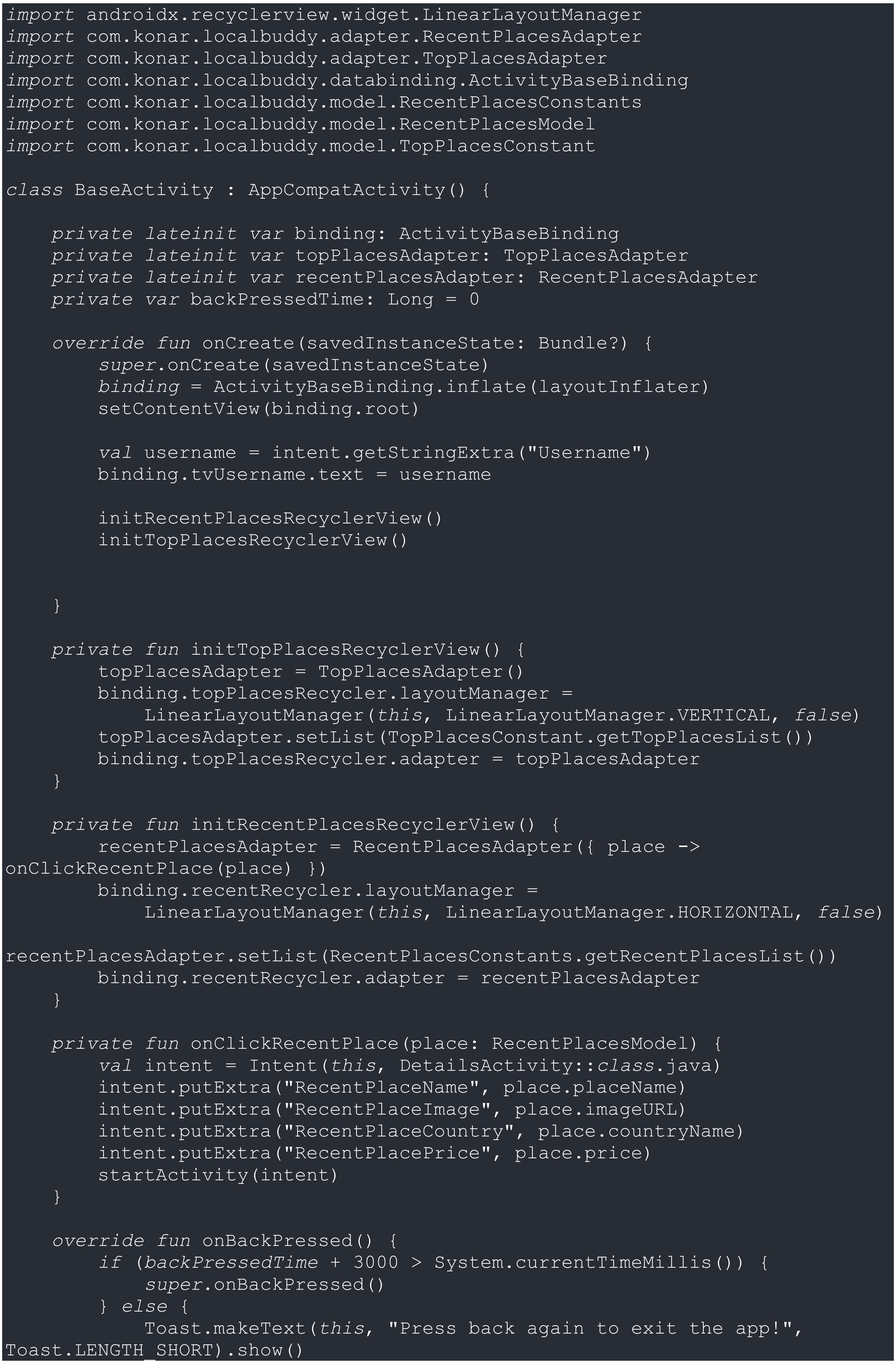
Sample O/P



MODULE 3

Home Screen

```
package com.konar.localbuddy

import android.content.Intent
import android.os.Bundle
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity
```



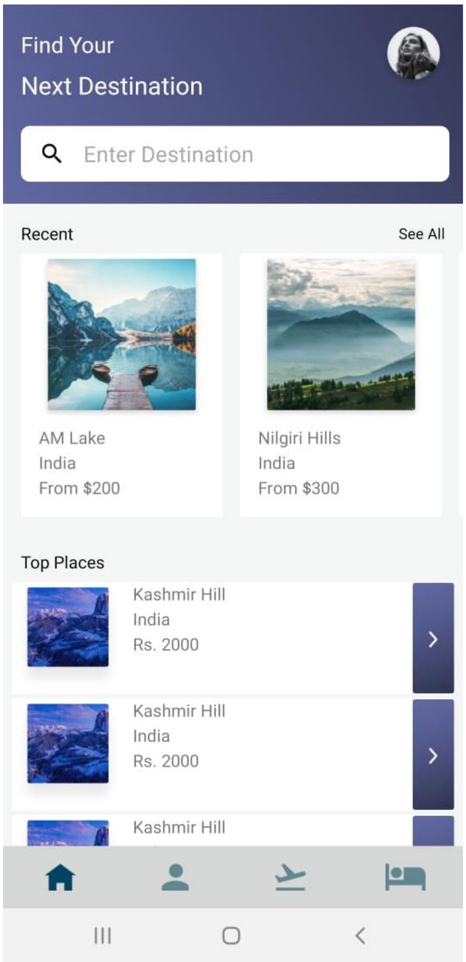
```
import androidx.recyclerview.widget.LinearLayoutManager
import com.konar.localbuddy.adapter.RecentPlacesAdapter
import com.konar.localbuddy.adapter.TopPlacesAdapter
import com.konar.localbuddy.databinding.ActivityBaseBinding
import com.konar.localbuddy.model.RecentPlacesConstants
import com.konar.localbuddy.model.RecentPlacesModel
import com.konar.localbuddy.model.TopPlacesConstant

class BaseActivity : AppCompatActivity() {

    private lateinit var binding: ActivityBaseBinding
    private lateinit var topPlacesAdapter: TopPlacesAdapter
    private lateinit var recentPlacesAdapter: RecentPlacesAdapter
    private var backPressedTime: Long = 0

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        binding = ActivityBaseBinding.inflate(layoutInflater)
        setContentView(binding.root)

        val username = intent.getStringExtra("Username")
        binding.tvUsername.text = username

        initRecentPlacesRecyclerView()
        initTopPlacesRecyclerView()


    }

    private fun initTopPlacesRecyclerView() {
        topPlacesAdapter = TopPlacesAdapter()
        binding.topPlacesRecycler.layoutManager =
            LinearLayoutManager(this, LinearLayoutManager.VERTICAL, false)
        topPlacesAdapter.setList(TopPlacesConstant.getTopPlacesList())
        binding.topPlacesRecycler.adapter = topPlacesAdapter
    }

    private fun initRecentPlacesRecyclerView() {
        recentPlacesAdapter = RecentPlacesAdapter({ place ->
onClickRecentPlace(place) })
        binding.recentRecycler.layoutManager =
            LinearLayoutManager(this, LinearLayoutManager.HORIZONTAL, false)

recentPlacesAdapter.setList(RecentPlacesConstants.getRecentPlacesList())
        binding.recentRecycler.adapter = recentPlacesAdapter
    }

    private fun onClickRecentPlace(place: RecentPlacesModel) {
        val intent = Intent(this, DetailsActivity::class.java)
        intent.putExtra("RecentPlaceName", place.placeName)
        intent.putExtra("RecentPlaceImage", place.imageURL)
        intent.putExtra("RecentPlaceCountry", place.countryName)
        intent.putExtra("RecentPlacePrice", place.price)
        startActivity(intent)
    }

    override fun onBackPressed() {
        if (backPressedTime + 3000 > System.currentTimeMillis()) {
            super.onBackPressed()
        } else {
            Toast.makeText(this, "Press back again to exit the app!",
Toast.LENGTH_SHORT).show()
```

```
        }
        backPressedTime = System.currentTimeMillis()
    }
}
```

XML

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#F5F7F7"
    tools:context=".MainActivity">

    <androidx.constraintlayout.widget.ConstraintLayout
        android:id="@+id/constraintLayout"
        android:layout_width="0dp"
        android:layout_height="180dp"
        android:background="@drawable/home_bg"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent">


        <TextView
            android:id="@+id/textView"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="16dp"
            android:layout_marginTop="24dp"
            android:text="Find Your"
            android:textColor="#FFFFFF"
            android:textSize="18sp"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent" />

        <TextView
            android:id="@+id/textView2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="8dp"
            android:text="Next Destination"
            android:textColor="#FFFFFF"
            android:textSize="20sp"
            app:layout_constraintStart_toStartOf="@+id/textView"
            app:layout_constraintTop_toBottomOf="@+id/textView" />

        <ImageView
            android:id="@+id/iv_profile_photo"
            android:layout_width="64dp"
            android:layout_height="64dp"
            android:layout_marginTop="16dp"
            android:layout_marginEnd="12dp"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintTop_toTopOf="parent"
            app:srcCompat="@drawable/profile" />
        <EditText
            android:id="@+id/editText"
            android:layout_width="0dp"
            android:layout_height="50dp"
            android:layout_marginTop="22dp"
            android:background="@drawable/search_bg"
            android:drawableLeft="@drawable/ic_search_black_24dp"
            android:drawablePadding="16dp"
            android:ems="10"
            android:inputType="textPersonName"
            android:paddingStart="16dp"
            android:hint="Enter Destination"
            android:textColorHint="@android:color/darker_gray"
            android:textColor="#62878F"
            app:layout_constraintEnd_toEndOf="@+id/iv_profile_photo"
            app:layout_constraintStart_toStartOf="@+id/textView2"
            app:layout_constraintTop_toBottomOf="@+id/textView2" />

        <TextView
            android:id="@+id/tv_username"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Username"
            android:textColor="@color/white"
            app:layout_constraintEnd_toEndOf="@+id/iv_profile_photo"
            app:layout_constraintStart_toStartOf="@id/iv_profile_photo"
            app:layout_constraintTop_toBottomOf="@id/iv_profile_photo" />
    </androidx.constraintlayout.widget.ConstraintLayout>

    <TextView
        android:id="@+id/textView3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:layout_marginTop="16dp"
        android:text="Recent"
        android:textColor="@android:color/background_dark"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/constraintLayout" />

    <TextView
        android:id="@+id/textView4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginEnd="16dp"
        android:text="See All"
        android:textColor="@android:color/background_dark"
        android:textSize="12sp"
        app:layout_constraintBottom_toBottomOf="@+id/textView3"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toTopOf="@+id/textView3" />

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/recent_recycler"
        android:layout_width="0dp"
        android:layout_height="250dp"
        android:layout_marginTop="8dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textView3" />
    <TextView
        android:id="@+id/textView5"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:layout_marginTop="16dp"
        android:text="Top Places"
        android:textColor="@android:color/background_dark"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/recent_recycler" />

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/top_places_recycler"
        android:layout_width="0dp"
        android:layout_height="250dp"
        android:layout_marginTop="8dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textView5" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="?actionBarSize"
        android:background="?attr/colorButtonNormal"
        android:gravity="center"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent">


        <ImageView
            android:layout_width="24dp"
            android:layout_height="24dp"
            android:layout_weight="1"
            android:src="@drawable/home" />

        <ImageView
            android:layout_width="24dp"
            android:layout_height="24dp"
            android:layout_weight="1"
            android:src="@drawable/profileicon" />

        <ImageView
            android:layout_width="24dp"
            android:layout_height="24dp"
            android:layout_weight="1"
            android:src="@drawable/flights" />

        <ImageView
            android:layout_width="24dp"
            android:layout_height="24dp"
            android:layout_weight="1"
            android:src="@drawable/hotel" />

    </LinearLayout>

</androidx.constraintlayout.widget.ConstraintLayout>
```

Sample O/P



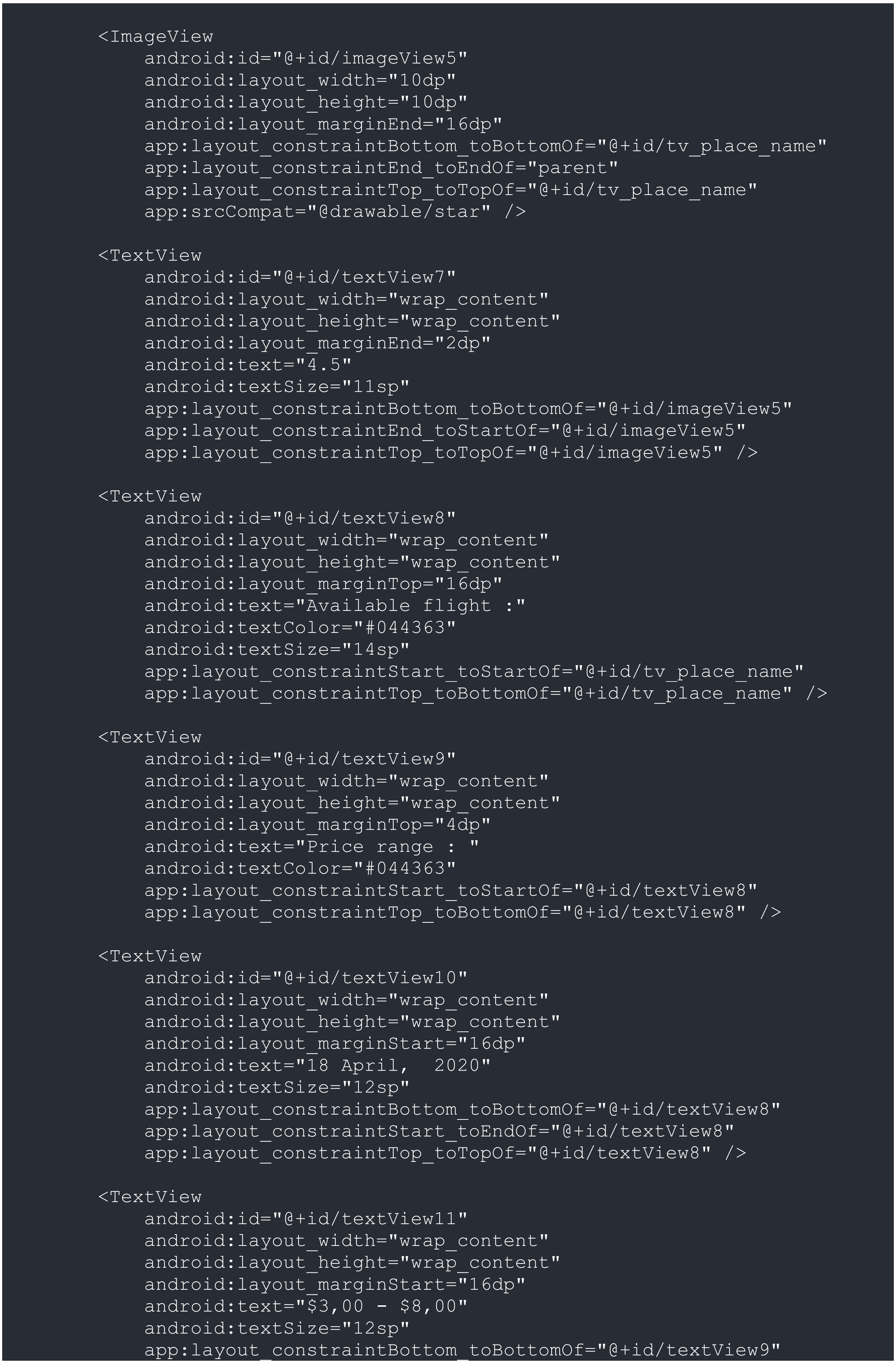
Module 4

Details Screen

```
package com.konar.localbuddy

import android.os.Bundle
import androidx.appcompat.app.AppCompatActivity
import com.konar.localbuddy.databinding.ActivityDetailsBinding

class DetailsActivity : AppCompatActivity() {

    private lateinit var binding: ActivityDetailsBinding

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        binding = ActivityDetailsBinding.inflate(layoutInflater)
        setContentView(binding.root)
        binding.btnBack.setOnClickListener {
            onBackPressed()
        }
    }

    private fun initDetailsActivity() {
        binding.tvPlaceName.text = intent.getStringExtra("RecentPlaceName")

binding.ivMainImage.setImageResource(intent.getIntExtra("RecentPlaceImage",0
))
    }
}
```

XML

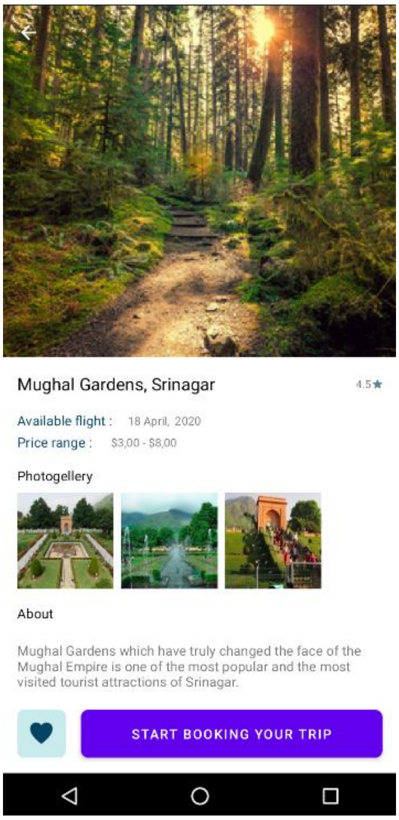
```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_height="match_parent"
    android:fillViewport="true"
    android:layout_width="match_parent">

    <androidx.constraintlayout.widget.ConstraintLayout
xmlns:app="http://schemas.android.com/apk/res-auto"
        xmlns:tools="http://schemas.android.com/tools"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        tools:context=".DetailsActivity">

        <ImageView
            android:id="@+id/iv_main_image"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:adjustViewBounds="true"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent"
            app:srcCompat="@drawable/bg" />

        <ImageView
            android:id="@+id/btn_back"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:clickable="true"
            android:layout_marginStart="16dp"
            android:layout_marginTop="16dp"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent"
            android:src="@drawable/ic_arrow_back_black_24dp" />

        <TextView
            android:id="@+id/tv_place_name"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="16dp"
            android:layout_marginTop="16dp"
            android:text="Mughal Gardens, Srinagar"
            android:textColor="@android:color/background_dark"
            android:textSize="18sp"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toBottomOf="@+id/iv_main_image" />
```



```
        <ImageView
            android:id="@+id/imageView5"
            android:layout_width="10dp"
            android:layout_height="10dp"
            android:layout_marginEnd="16dp"
            app:layout_constraintBottom_toBottomOf="@+id/tv_place_name"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintTop_toTopOf="@+id/tv_place_name"
            app:srcCompat="@drawable/star" />

        <TextView
            android:id="@+id/textView7"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginEnd="2dp"
            android:text="4.5"
            android:textSize="11sp"
            app:layout_constraintBottom_toBottomOf="@+id/imageView5"
            app:layout_constraintEnd_toStartOf="@+id/imageView5"
            app:layout_constraintTop_toTopOf="@+id/imageView5" />

        <TextView
            android:id="@+id/textView8"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="16dp"
            android:text="Available flight :"
            android:textColor="#044363"
            android:textSize="14sp"
            app:layout_constraintStart_toStartOf="@+id/tv_place_name"
            app:layout_constraintTop_toBottomOf="@+id/tv_place_name" />

        <TextView
            android:id="@+id/textView9"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="4dp"
            android:text="Price range : "
            android:textColor="#044363"
            app:layout_constraintStart_toStartOf="@+id/textView8"
            app:layout_constraintTop_toBottomOf="@+id/textView8" />

        <TextView
            android:id="@+id/textView10"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="16dp"
            android:text="18 April,  2020"
            android:textSize="12sp"
            app:layout_constraintBottom_toBottomOf="@+id/textView8"
            app:layout_constraintStart_toEndOf="@+id/textView8"
            app:layout_constraintTop_toTopOf="@+id/textView8" />

        <TextView
            android:id="@+id/textView11"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="16dp"
            android:text="$3,00 - $8,00"
            android:textSize="12sp"
            app:layout_constraintBottom_toBottomOf="@+id/textView9"
            app:layout_constraintStart_toEndOf="@+id/textView9"
            app:layout_constraintTop_toTopOf="@+id/textView9" />

        <TextView
            android:id="@+id/textView12"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="16dp"
            android:text="Photogellery"
            android:textColor="@android:color/background_dark"
            app:layout_constraintStart_toStartOf="@+id/textView9"
            app:layout_constraintTop_toBottomOf="@+id/textView9" />

        <ImageView
            android:id="@+id/imageView8"
            android:layout_width="100dp"
            android:layout_height="100dp"
            android:layout_marginStart="16dp"
            android:layout_marginTop="8dp"
            android:scaleType="fitXY"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toBottomOf="@+id/textView12"
            app:srcCompat="@drawable/img1" />

        <ImageView
            android:id="@+id/imageView9"
            android:layout_width="100dp"
            android:layout_height="100dp"
            android:layout_marginStart="8dp"
            android:scaleType="fitXY"
            app:layout_constraintBottom_toBottomOf="@+id/imageView8"
            app:layout_constraintStart_toEndOf="@+id/imageView8"
            app:layout_constraintTop_toTopOf="@+id/imageView8"
            app:srcCompat="@drawable/img2" />

        <ImageView
            android:id="@+id/imageView10"
            android:layout_width="100dp"
            android:layout_height="100dp"
            android:layout_marginStart="8dp"
            android:scaleType="fitXY"
            app:layout_constraintBottom_toBottomOf="@+id/imageView9"
            app:layout_constraintStart_toEndOf="@+id/imageView9"
            app:layout_constraintTop_toTopOf="@+id/imageView9"
            app:srcCompat="@drawable/img3" />

        <ImageView
            android:id="@+id/imageView11"
            android:layout_width="50dp"
            android:layout_height="50dp"
            android:layout_marginStart="16dp"
            android:layout_marginBottom="16dp"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:srcCompat="@drawable/like" />

        <Button
            android:id="@+id/button"
            android:layout_width="0dp"
            android:layout_height="50dp"
            android:layout_marginStart="16dp"
            android:layout_marginEnd="16dp"
```

```
            android:background="@drawable/book_button_bg"
            android:text="Start Booking Your Trip"
            android:textColor="#FFFFFF"
            app:layout_constraintBottom_toBottomOf="@+id/imageView11"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintStart_toEndOf="@+id/imageView11"
            app:layout_constraintTop_toTopOf="@+id/imageView11" />

        <TextView
            android:id="@+id/textView13"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="16dp"
            android:layout_marginTop="16dp"
            android:text="About "
            android:textColor="@android:color/background_dark"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toBottomOf="@+id/imageView8" />

        <TextView
            android:id="@+id/textView14"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_marginTop="8dp"
            android:layout_marginEnd="16dp"
            android:layout_marginBottom="8dp"
            android:text="Mughal Gardens which have truly changed the face
of the Mughal Empire is one of the most popular and the most visited tourist
attractions of Srinagar."
            app:layout_constraintBottom_toTopOf="@+id/button"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintStart_toStartOf="@+id/textView13"
            app:layout_constraintTop_toBottomOf="@+id/textView13" />

    </androidx.constraintlayout.widget.ConstraintLayout>
</ScrollView>
```

Sample o/p:



Mughal Gardens, Srinagar 4.5★

Available flight : 18 April, 2020

Price range : $3,00 - $8,00

Photogellery

About

Mughal Gardens which have truly changed the face of the Mughal Empire is one of the most popular and the most visited tourist attractions of Srinagar.

Login Database Module:

UserDao

```
package com.konar.localbuddy.db

import androidx.room.*
import kotlinx.coroutines.flow.Flow

@Dao
interface UserDao {
    @Insert
    suspend fun insert(userEntity: UserEntity)

    @Update
    suspend fun update(userEntity: UserEntity)

    @Query("UPDATE `user-table` SET password=:newPassword where 
username=:username")
    suspend fun updatePassword(username: String, newPassword: String)

    @Delete
    suspend fun delete(userEntity: UserEntity)

    @Query("SELECT * FROM 'user-table'")
    fun fetchAllUsers(): Flow<List<UserEntity>>

    @Query("SELECT username FROM `user-table`")
    fun fetchAllUsernames(): Flow<List<String>>

    @Query("SELECT password FROM `user-table` where username=:username")
    fun fetchUserByUsername(username: String): Flow<String>

}
```

UserEntity

```
package com.konar.localbuddy.db

import androidx.room.ColumnInfo
import androidx.room.Entity
import androidx.room.PrimaryKey

@Entity(tableName = "user-table")
data class UserEntity(
    @PrimaryKey
    @ColumnInfo(name = "username")
    var userName: String,
    @ColumnInfo(name = "password")
    var password: String
)
```

UserDatabase

```
package com.konar.localbuddy.db

import android.content.Context
import androidx.room.Database
import androidx.room.Room
import androidx.room.RoomDatabase

@Database(entities = [UserEntity::class], version = 1)
abstract class UserDatabase : RoomDatabase() {

    abstract fun userDao(): UserDao

    companion object {

        @Volatile
        private var INSTANCE: UserDatabase? = null

        fun getInstance(context: Context): UserDatabase {
            synchronized(this) {
                var instance = INSTANCE

                if (instance == null) {
                    instance = Room.databaseBuilder(
                        context.applicationContext,
                        UserDatabase::class.java,
                        "user_database"
                    ).fallbackToDestructiveMigration().build()
                    INSTANCE = instance
                }
                return instance
            }

        }
    }

}
```

RecyclerView Adapters Module

RecentRecyclerView

```
package com.konar.localbuddy.adapter

import android.view.LayoutInflater
import android.view.ViewGroup
import androidx.recyclerview.widget.RecyclerView
import com.konar.localbuddy.databinding.RecentRowsItemBinding
import com.konar.localbuddy.model.RecentPlacesModel

class RecentPlacesAdapter(private val clickListener: (place:
RecentPlacesModel) -> Unit) :
    RecyclerView.Adapter<RecentPlacesViewHolder>() {

    val placeList = ArrayList<RecentPlacesModel>()

    fun setList(places: ArrayList<RecentPlacesModel>) {
        placeList.clear()
        placeList.addAll(places)
    }

    override fun onCreateViewHolder(parent: ViewGroup, viewType: Int):
RecentPlacesViewHolder {
        val layoutInflater = LayoutInflater.from(parent.context)
        val binding = RecentRowsItemBinding.inflate(layoutInflater, parent,
false)
        return RecentPlacesViewHolder(binding)
    }

    override fun onBindViewHolder(holder: RecentPlacesViewHolder, position:
Int) {
        holder.bind(placeList[position])

        holder.itemView.setOnClickListener {
            clickListener.invoke(placeList[position])
        }
    }

    override fun getItemCount(): Int {
        return placeList.size
    }
}

class RecentPlacesViewHolder(val binding: RecentRowsItemBinding) :
    RecyclerView.ViewHolder(binding.root) {
    fun bind(places: RecentPlacesModel) {
        binding.placeName.text = places.placeName
        binding.countryName.text = places.countryName
        binding.price.text = places.price
        binding.placeImage.setImageResource(places.imageURL)
    }
}
```

TopPlacesRecyclerViewAdapter

```
package com.konar.localbuddy.adapter

import android.view.LayoutInflater
import android.view.ViewGroup
import androidx.recyclerview.widget.RecyclerView
import com.konar.localbuddy.databinding.TopPlacesRowItemBinding
import com.konar.localbuddy.model.TopPlacesModel
```

```
class TopPlacesAdapter : RecyclerView.Adapter<TopPlacesViewHolder>() {

    val placeList = ArrayList<TopPlacesModel>()

    fun setList(places: ArrayList<TopPlacesModel>) {
        placeList.clear()
        placeList.addAll(places)
    }

    override fun onCreateViewHolder(parent: ViewGroup, viewType: Int):
TopPlacesViewHolder {
        val layoutInflater = LayoutInflater.from(parent.context)
        val binding = TopPlacesRowItemBinding.inflate(layoutInflater,
parent, false)
        return TopPlacesViewHolder(binding)
    }

    override fun onBindViewHolder(holder: TopPlacesViewHolder, position:
Int) {
        holder.bind(placeList[position])
    }

    override fun getItemCount(): Int {
        return placeList.size
    }
}

class TopPlacesViewHolder(val binding: TopPlacesRowItemBinding) :
    RecyclerView.ViewHolder(binding.root) {
    fun bind(places: TopPlacesModel) {
        binding.placeName.text = places.placeName
        binding.countryName.text = places.countryName
        binding.price.text = places.price
        binding.placeImage.setImageResource(places.imageURL)
    }
}
```