

TRAVIO (TOUR AND TRAVEL MANAGEMENT APPLICATION) FOR ANDROID

A PROJECT REPORT

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

Under the Guidance of

SHREYA BANERJEE

Project Carried Out At



Ardent Computech Pvt Ltd (An ISO 9001:2015 Certified)

CF-137, Sector - 1, Salt Lake City, Kolkata - 700 064

Submitted By

**SAPTARSHI ROY
SUNANDAN BHAKAT
DEBARYA GANGULY
SAGNIK DEY SARKAR
SARBAJIT CHATTERJEE**



FUTURE INSTITUTE OF ENGINEERING & MANAGEMENT

FIEM, WEST BENGAL, KOLKATA, INDIA

JUNE – JULY 2018



AR DENT

COMPUTECH PVT. LTD.

High-End Technology Training and Project



Microsoft
Technology Associate



Summer Industrial Training Certificate

Awarded to

SAPTARSHI ROY

of

FUTURE INSTITUTE OF ENGINEERING AND MANAGEMENT

for successfully completing the Summer Industrial Training on

ANDROID APPLICATION DEVELOPMENT

from

15/06/18 to 16/07/18

and implementing the project titled

TOUR AND TRAVEL AGENCY

Director
Technology Services

Director
Operations

Date of Achievement: 21/07/18

Certificate ID: ACPL/SVT 2018/1855

To verify the genuinity of this certificate visit www.ardentcollaborations.com or email to hr@ardentcollaborations.com



TO WHOMSOEVER IT MAY CONCERN

This is to certify that the project entitled **"TOUR AND TRAVEL AGENCY"**, designed on **"ANDROID APPLICATION DEVELOPMENT"** technology is an original work carried out by **"SAPTARSHI ROY"** student of **"FUTURE INSTITUTE OF ENGINEERING AND MANAGEMENT"** in the Summer Internship Program at Ardent Computech Private Limited, from 15/06/18 to 16/07/18.

The matter furnished in the project is an authentic work done by the student has not been submitted to any University/Institute for the fulfilment of any course of study.

Thanks & Regards

Director
Technology Services

Date : 27/07/2018

SDF Building, Module 132, Ground Floor,
Sector V, Salt Lake, Kolkata 700091

033 4007 3507

ardent.kolkata@gmail.com

www.ardentcollaborations.com





GRADE CARD

INDUSTRIAL TRAINING 2018

ARDENT COMPUTECH PVT LTD.

SDF Building, Module No.- 132, Ground Floor, Sector V, Salt Lake - 700091

www.ardentcollaborations.com

Microsoft
Technology Associate

AUTODESK
Certified User

CERTIFIED ASSOCIATE

Microsoft
Office Specialist

MAAD
Skills For Tomorrow

Quality
Partner

Name : **SAPTARSHI ROY**
Course : **ANDROID APPLICATION DEVELOPMENT**
Course Duration : **15-06-2018 to 16-07-2018**

Parameter	Grade
Assignment	E
Assessment	O
Attendance	O
Project Implementation	O

Total Grade : **O**

Date:21/07/18

Grade Explanations :

O- Outstanding 90-100, E- Excellent 80-90, A- Very Good 70-80, B- Good 60-70,
C - Average/fair 50-60, D- Below Average 40-50, F- Fail Below 40, I- Absent

* This is a system generated file. Hence no signature is required.

'Knowledge is power, power leads to growth, growth is life'

In association with



(Note: All entries of the proforma of approval should be filled up with appropriate and complete information. Incomplete proforma of approval in any respect will be summarily rejected.)

1. Title of the Project: **Tour and Travel Management**
2. Project Members: 1. Saptarshi Roy, 2. Sunandan Bhakat 3. Debarya Ganguly
4. Sagnik Dey Sarkar 5. Sarbajit Chatterjee

3. Name and Address of the Guide: **Mrs. Shreya Banerjee**

Ardent Computech Pvt Ltd (An ISO 9001:2015 Certified)

Module 132, SDF Building, Sector 5, Salt Lake, Kolkata - 64

Ph.D.* M.Tech.* B.E*/B.Tech.* MCA* M.Sc.*

4. Educational Qualification of the Guide: ☐ ☒ ☒ ☐ ☐

5. Working / Training experience of the Guide: **8.5 Years**

6. Project Version Control History

Version	Primary Authors	Description of Version	Date Completed
Final	Saptarshi Roy, Sunandan Bhakat, Debarya Ganguly, Sagnik Dey Sarkar, Sarbajit Chatterjee	Project Report	10 th July, 2018

1. *Saptarshi Roy*
2. *Sunandan Bhakat*
3. *Debarya Ganguly*
4. *Sagnik Dey Sarkar*
5. *Sarbajit Chatterjee*

Signatures of Team Members

Shreya Banerjee
Signature of Approval

Date: 10.07.2018

Date: 10.07.2018

For Office Use Only

Approved

Not Approved

Mrs. Shreya Banerjee
Project Proposal Evaluator

PROJECT RESPONSIBILITY FORM

Tour and Travel Management Application

GROUP NO.	SL.NO.	NAME OF MEMBER	RESPONSIBILITY
1	1	Saptarshi Roy	Project Leader, Coding & Designing
	2	Sunandan Bhakat	Coding & Documentation
	3	Debarya Ganguly	System Analysis, Testing
	4	Sagnik Dey Sarkar	Implementation and Maintenance
	5	Sarbajit Chatterjee	PPT, Testing


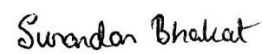
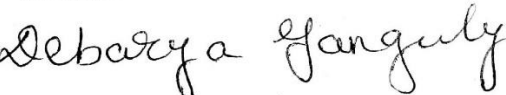
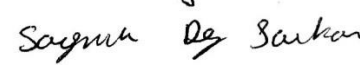

Each group member must participate in project development and developing the ideas for the required elements. Individual group members will be responsible for completing tasks which help to finalize the project and the performance. All group members must be assigned a task.

Date: 10.07.2018

Name of the Students

1. Saptarshi Roy
2. Sunandan Bhakat
3. Debarya Ganguly
4. Sagnik Dey Sarkar
5. Sarbajit Chatterjee

Signatures of the students

- a. 
- b. 
- c. 
- d. 
- e. 

DECLARATION


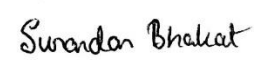
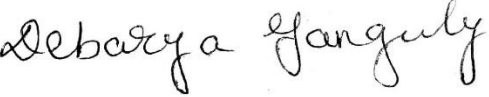


We hereby declare that the project work being presented in the project proposal entitled **“TRAVIO (TOUR AND TRAVEL MANAGEMENT APPLICATION)”** in partial fulfilment of the requirements for the award of the degree of **BACHELOR OF TECHNOLOGY IN ELECTRONICS AND COMMUNICATION ENGINEERING** at **ARDENT COMPUTECH PVT. LTD, SALT LAKE, KOLKATA, WEST BENGAL**, is an authentic work carried out under the guidance of **Mrs. SHREYA BANERJEE**. The matter embodied in this project work has not been submitted elsewhere for the award of any degree of our knowledge and belief.

Date:

Name of the Students

- 1.Saptarshi Roy
- 2.Sunandan Bhakat
- 3.Debarya Ganguly
- 4.Sagik Dey Sarkar
- 5.Sarbajit Chatterjee

Signature of the students

- a. 
- b. 
- c. 
- d. 
- e. 

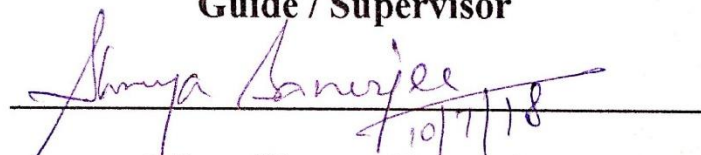


Ardent Computech Pvt Ltd (An ISO 9001:2008 Certified)
CF-137, Sector - 1, Salt Lake City, Kolkata - 700 064

CERTIFICATE

This is to certify that this proposal of minor project entitled “**TRAVIO (TOUR AND TRAVEL MANAGEMENT APPLICATION)**” is a record of bona fide work, carried out by **1. SAPTARSHI ROY 2. SUNANDAN BHAKAT 3. DEBARYA GANGULY 4. SAGNIK DEY SARKAR 5. SARBAJIT CHATTERJEE** and under my guidance at **ARDENT COMPUTECH PVT LTD**. In my opinion, the report in its present form is in partial fulfilment of the requirements for the award of the degree of **BACHELOR OF TECHNOLOGY IN ELECTRONICS AND COMMUNICATION ENGINEERING** and as per regulations of the **ARDENT®**. To the best of my knowledge, the results embodied in this report, are original in nature and worthy of incorporation in the present version of the report.

Guide / Supervisor


Mrs. Shreya Banerjee

Ardent Computech Pvt Ltd (An ISO 9001:2015 Certified)
Module 132, SDF Building, Sector 5, Salt Lake, Kolkata – 64

ACKNOWLEDGEMENT

Success of any project depends largely on the encouragement and guidelines of many others. We take this sincere opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project work.

Our heartfelt thanks to Dr. Dipankar Ghosh, HOD, ECE Department and Prof. Debabrata Pandit, Electronics and Communication Engineering Department, FIEM, for providing us the opportunity to develop the project at Ardent Computech Pvt. Ltd.

We would like to show our greatest appreciation to Mrs. Shreya Banerjee, Project Manager at Ardent, Kolkata. We always feel motivated and encouraged every time by her valuable advice and constant inspiration; without her encouragement and guidance this project would not have materialized.

Words are inadequate in offering our thanks to the other trainees, project assistants and other members at Ardent Computech Pvt. Ltd. for their encouragement and cooperation in carrying out this project work. The guidance and support received from all the members and who are contributing to this project, was vital for the success of this project.

Place: Kolkata
Date: 10.07.2018

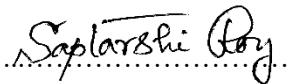

Signature of Student

TABLE OF CONTENTS

S.NO.	Name of the Topic	Page No.
1.	Company Profile	9
2.	Introduction	11-13
2.1	Objective	12
2.2	Scope	13
3.	System Analysis	15-21
3.1	Identification of Need	15
3.2	Feasibility Study	16
3.3	Work flow	17-18
3.4	Functional Requirements	19
3.5	Non-Functional Requirements	20
3.6	Hardware and Software Requirements	21
4.	System Design	22-51
4.1	Data Flow Diagram (DFD)	23-25
4.2	Entity-Relationship Diagram	26-28
4.3	Use-case Diagram	29-31
4.4	Modularization Details	32-33
4.5	Database Design	34-36
4.6	User Interface Design	37-51

5.	Implementation & Testing	52-68
5.1	Introduction	54
5.2	Objectives of Testing	55-56
5.3	Test Cases	57-62
5.4	White Box Testing	63
5.5	Black Box Testing	64
5.6	Output Testing	65
5.7	User Acceptance Testing	66
5.8	Integration Testing	67
6.	Gantt Chart	68
7.	System Security Measures	69-72
7.1	Database Security Measures	70
7.2	System Security Measures	71
7.3	Limitations	72
8.	Conclusion	73
9.	Future Scope & Further Enhancements	74
10.	Bibliography/References	75

ARDENT COMPUTECH PVT. LTD.

Ardent Computech Private Limited is an ISO 9001-2008 certified Software Development Company in India. It has been operating independently since 2003. It was recently merged with ARDENT TECHNOLOGIES.

Ardent Technologies

ARDENT TECHNOLOGIES is a Company successfully providing its services currently in UK, USA, Canada and India. The core line of activity at ARDENT TECHNOLOGIES is to develop customized application software covering the entire responsibility of performing the initial system study, design, development, implementation and training. It also deals with consultancy services and Electronic Security systems. Its primary clientele includes educational institutes, entertainment industries, resorts, theme parks, service industry, telecom operators, media and other business houses working in various capacities.

Ardent Collaborations

ARDENT COLLABORATIONS, the Research Training and Development Department of ARDENT COMPUTECH PVT LTD is a professional training Company offering IT enabled services & industrial trainings for B-Tech, MCA, BCA, MSc and MBA fresher's and experienced developers/programmers in various platforms. Summer Training / Winter Training / Industrial training will be provided for the students of B.TECH., M.TECH., MBA, MCA and BCA only. Deserving candidates may be awarded stipends, scholarships and other benefits, depending on their performance and recommendations of the mentors.

Associations

Ardent is an ISO 9001:2008 company.

It is affiliated to National Council of Vocational Training (NCVT), Directorate General of Employment & Training (DGET), Ministry of Labor & Employment, and Government of India.

TRAVIO

(TOUR AND TRAVEL
MANAGEMENT APPLICATION)

1. INTRODUCTION

What is Tour and Travel Management Application?

With the advent of technology and online systems coming to more and more use, each and every user of the internet would want things to be sorted or completed without going to any place physically. So as in case of travelers, finding a travel agent is very important for proper planning and organizing their dream tours. In general, the earlier method of booking tours was with the help of tour agents or companies where physically going to them for booking assistance was a must. In this way there are many problems pertaining to money, time and reliability. So in order to resolve all these problems and achieve perfect compatibility as much as possible, several tour and travel management applications have come into existence.

2.1 OBJECTIVE

- Our objective is to offer a variety of travel services that are sure to match all your priorities.
- Our objective is to globalism, organize, standardize and goal of journey towards perfectionism
- Our objective is to make strong relationship with customers so that they can enjoy the holiday of their dreams.
- Our objective is to offer variety of travel services at an economic and efficient rate.
- Our objective is just an initiative, it will be made to more further and developed work of art.

2.2 SCOPE

The project is all about providing users the opportunity to book their dream tour without physically going to any travel agent or company. Suppose a person lives in a particular area and the particular travel company is far away from his/her home then he/she doesn't need to go and visit the travel company physically and then ask for tour planning and booking.

This Project provides security for the users with the use of Login-id and Password, so that unauthorized users cannot use anyone else's account. The only authorized will have proper access authority can access the software.

SYSTEM ANALYSIS

3.1 IDENTIFICATION OF NEED

System analysis is a process of gathering and interpreting facts, diagnosing problems and the information to recommend improvements on the system. It is a problem solving activity that requires intensive communication between the system users and system developers. System analysis or study is an important phase of any system development process. The system is studied to the minutest detail and analyzed. The system analyst plays the role of the interrogator and dwells deep into the working of the present system. The System is viewed as a whole and the input to the system are identified. The outputs from the organization are traced to the various processes. System analysis is concerned with becoming aware of the problem, identifying the relevant and Decisional variables, analysis and synthesizing the various factors and determining an optimal or at least a satisfactory solution or program of action.

A detailed study of the process must be made by various techniques like interviews, questionnaires etc. The data collected by these sources must be scrutinized to arrive to a conclusion. The conclusion is an understanding of how the system functions. This system is called the existing system. Now the existing system is subjected to close study and problem area are identified. The designer now functions as a problem solver and tries to sort out the difficulties that the enterprise faces. The solutions are given as proposals. The proposal is then weighed with the existing system analytically and the best one is selected. The proposal is presented to the user for an endorsement by the user. The proposal is reviewed on user request and suitable changes are made. This is a loop that ends as soon as the user is satisfied with the proposal.

3.2 FEASIBILITY STUDY

Feasibility study is made to see if the project on completion will serve the purpose the organization for the amount of work.

Effort and the time that spend on it. Feasibility study lets the developer foresee the future of the project and the usefulness. A feasibility study of a system proposal is according to its workability, which is the impact on the organization, ability to meet their user needs and effective use of resources. Thus when a new application is proposed it normally goes through a feasibility study before it is approved for development.

The document provide the feasibility of the project that is being designed and lists various area that were considered very carefully during the feasibility study of this project such as Technical, Economic and operational feasibilities.

3.3 WORK FLOW

This Document plays a vital role in the development life cycle (SDLC) as it describes the complete requirement of the system. It is meant for use by the developers and will be the basic during testing phase. Any changes made to the requirements in the future will have to go through formal change approval process.

WATER FALL MODEL was being chosen because all requirements were known beforehand and the objective of our software development is the computerization/automation of an already existing manual working system.

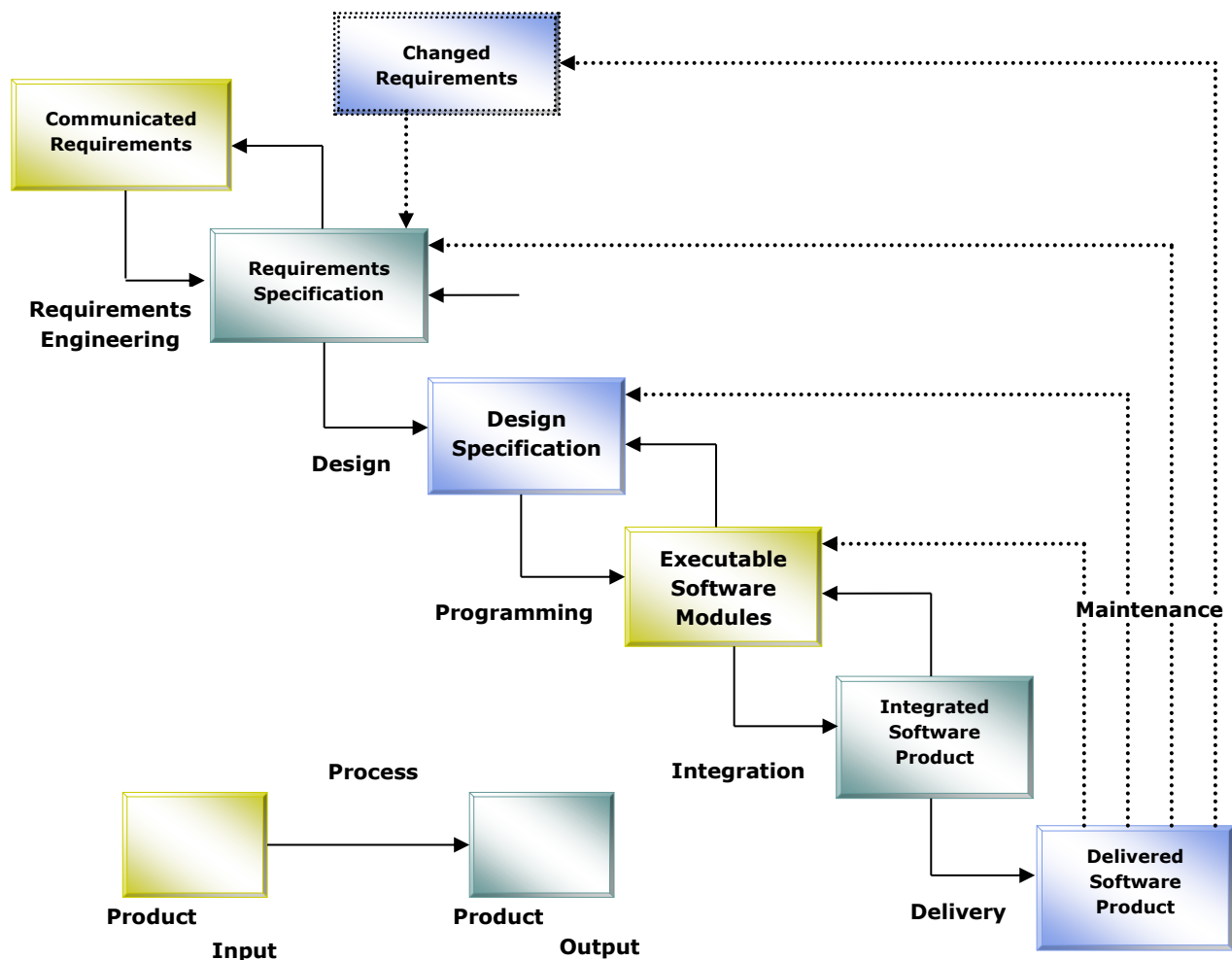


Fig: Water Fall Model

The developer is responsible for:

- Developing the system, which meets the SRS and solving all the requirements of the system?
- Demonstrating the system and installing the system at client's location after the acceptance testing is successful.
- Submitting the required user manual describing the system interfaces to work on it and also the documents of the system.
- Conducting any user training that might be needed for using the system.
- Maintaining the system for a period of one year after installation.

3.4 FUNCTIONAL REQUIREMENTS

Modules:

The modules used in this software are as follows:

- **Home:** This layout contains the tour packages that are offered by the App. User can view the packages without Login/Sign-up
- **Sign In:** This module is for registered users to login. The **ADMIN** has the authority to Add, Delete, Update and View Bookings etc. The **USER** can only view fixtures, book tour packages, take screenshots of their bookings and view other information.
- **Sign Up:** This Module can be accessed by a **NEW USER** to register by using Email id, Password and Name.
- **Admin Login:** In this Module, **The ADMIN** can login and view booking and enquiry details from the user.
- **Update Information:** In this Module, **THE USER** can update details regarding his/her account.
- **Contact Us:** This Module gives the user to contact the **ADMIN/THE COMPANY** for any queries.

3.5 NON-FUNCTIONAL REQUIREMENTS

- **Usability Requirement:** The system shall allow the users to access the system from any browsers, no special training is required. The system user friendly and the system is written in simple English.
- **Availability Requirement:** The system is available 100% for the user and is used by 24 hours a day and 365 days a year. The system shall be operational 24 hours a day and 7 days a week.
- **Accuracy:** The system should accurately provide real time information taking into consideration various issues. The system shall provide 100% access reliability.
- **Performance Requirement:** The information is refreshed at regular intervals depending upon whether some updates have occurred or not. The system shall respond the member in less than 2 seconds.
- **Security Requirement:** System will use a secured database and the system will have different users and each user has different types of constraints. Only admins have the rights to update database information of other users.
- **Reliability Requirement:** The system has to be 100% reliable due to the importance of data and the damages that can be caused by incorrect data. The system will run 7 days a week and 24 hours a day.

3.6 HARDWARE and SOFTWARE REQUIREMENTS

HARDWARE REQUIREMENTS

- CPU TYPE : INTEL Pentium 4
- CLOCK SPEED : 3.0 GHZ
- RAM SIZE : 512MB
- HARD DISK CAPACITY : 8GB
- MOBILE : ANDROID DEVICE

SOFTWARE REQUIREMENTS

OPERATING SYSTEM	:	ANDROID
LANGUAGE	:	ANDROID SDK 2.3 OR ABOVE
CLOUD DATABASE	:	FIREBASE

SYSTEM DESIGN

4.1 DATA FLOW DIAGRAM

A Data Flow Diagram (DFD) is a diagram that describes the flow of data and the processes that change data throughout a system. A structured analysis and design tool that can be used for flowcharting in place of or in association with information. Oriented and process oriented system flowcharts. When analysts prepare the Data Flow Diagram, they specify the user needs at a level of detail that virtually determines the information flow into and out of the system and the required data resources. This network is constructed by using a set of symbols that do not imply physical implementations. The Data Flow Diagram reviews the current physical system, prepares input and output specification, specifies the implementation plan etc.

Four basic symbols are used to construct data flow diagrams. They are symbols that represent data source, data flows, and data transformations and data storage. The points at which data are transformed are represented by enclosed figures, usually circles, which are called nodes.

DATA FLOW DIAGRAM SYMBOLS



Source or Destination of Data



Data Flow



Process



Storage

Steps to Construct Data Flow Diagram

Four Steps are generally used to construct a DFD.

- Process should be named and referred for easy reference. Each name should be representative of the reference.
- The destination of flow is from top to bottom and from left to right.
- When a process is distributed into lower level details they are numbered.
- The names of data stores, sources and destinations are written in capital letters.

Rules for constructing a Data Flow Diagram-

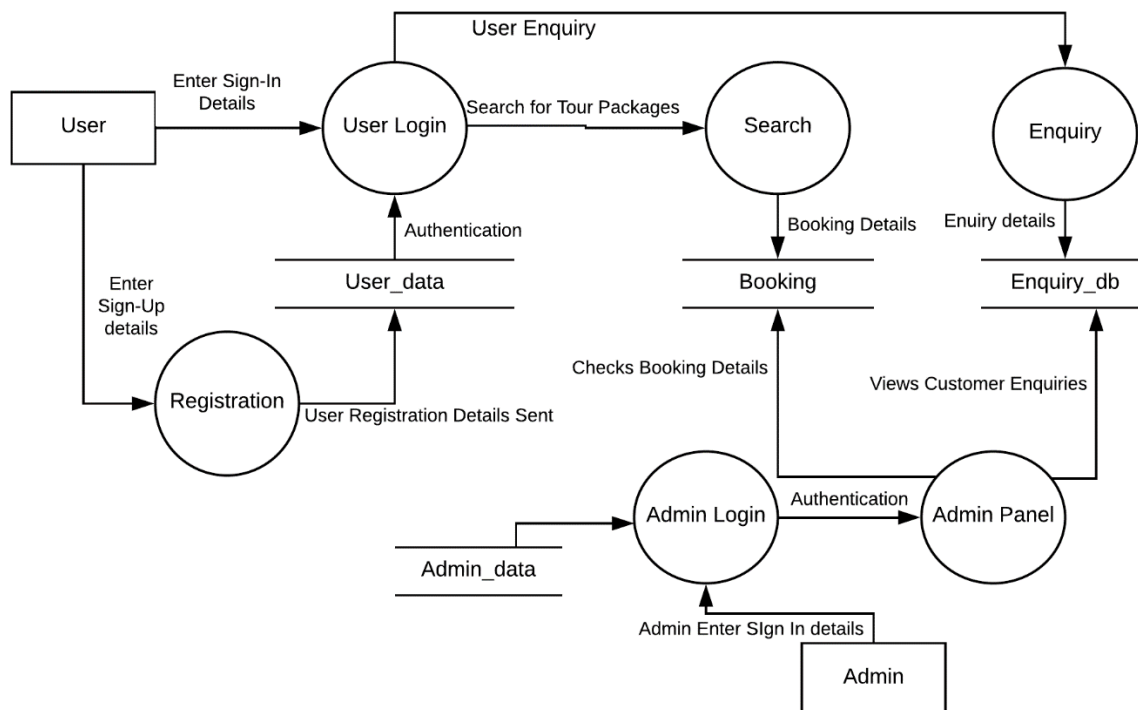
- Arrows should not cross each other.
- Squares, Circles, Files must bear a name.
- Decomposed data flow squares and circles can have same names.
- Draw all data flow around the outside of the diagram

DATA FLOW DIAGRAM

Level 0



Level 1



4.2 ENTITY RELATIONSHIP DIAGRAM

In software engineering, an **entity–relationship model (ER model)** is a data model for describing the data or information aspects of a business domain or its process requirements, in an abstract way that lends itself to ultimately being implemented in a database such as a relational database. The main components of ER models are entities (things) and the relationships that can exist among them.

However, variants of the idea existed previously, and have been devised subsequently such as super type and subtype data entities and commonality relationships.

An entity–relationship model is a systematic way of describing and defining a business process. The process is modeled as components (*entities*) that are linked with each other by *relationships* that express the dependencies and requirements between them, such as: *one building may be divided into zero or more apartments, but one apartment can only be located in one building*. Entities may have various properties (*attributes*) that characterize them. Diagrams created to represent these entities, attributes, and relationships graphically are called entity–relationship diagrams.

An ER model is typically implemented as a database. In the case of a relational database, which stores data in tables, every row of each table represents one instance of an entity. Some data fields in these tables point to indexes in other tables; such pointers represent the relationships.

The three schema approach to software engineering uses three levels of ER models that may be developed.

An entity may be defined as a thing capable of an independent existence that can be uniquely identified. An entity is an abstraction from the complexities of a domain.

When we speak of an entity, we normally speak of some aspect of the real world that can be distinguished from other aspects of the real world.

A relationship captures how entities are related to one another. Relationships can be thought of as verbs, linking two or more nouns.

Cardinality constraints are expressed as follows:

- a double line indicates a *participation constraint*, totality or subjectivity : all entities in the entity set must participate in *at least one* relationship in the relationship set;
- an arrow from entity set to relationship set indicates a key constraint, i.e. injectivity: each entity of the entity set can participate in *at most one* relationship in the relationship set;
- A thick line indicates both, i.e. bijectivity: each entity in the entity set is involved in *exactly one* relationship.
- An underlined name of an attribute indicates that it is a key: two different entities or relationships with this attribute always have different values for this attribute.

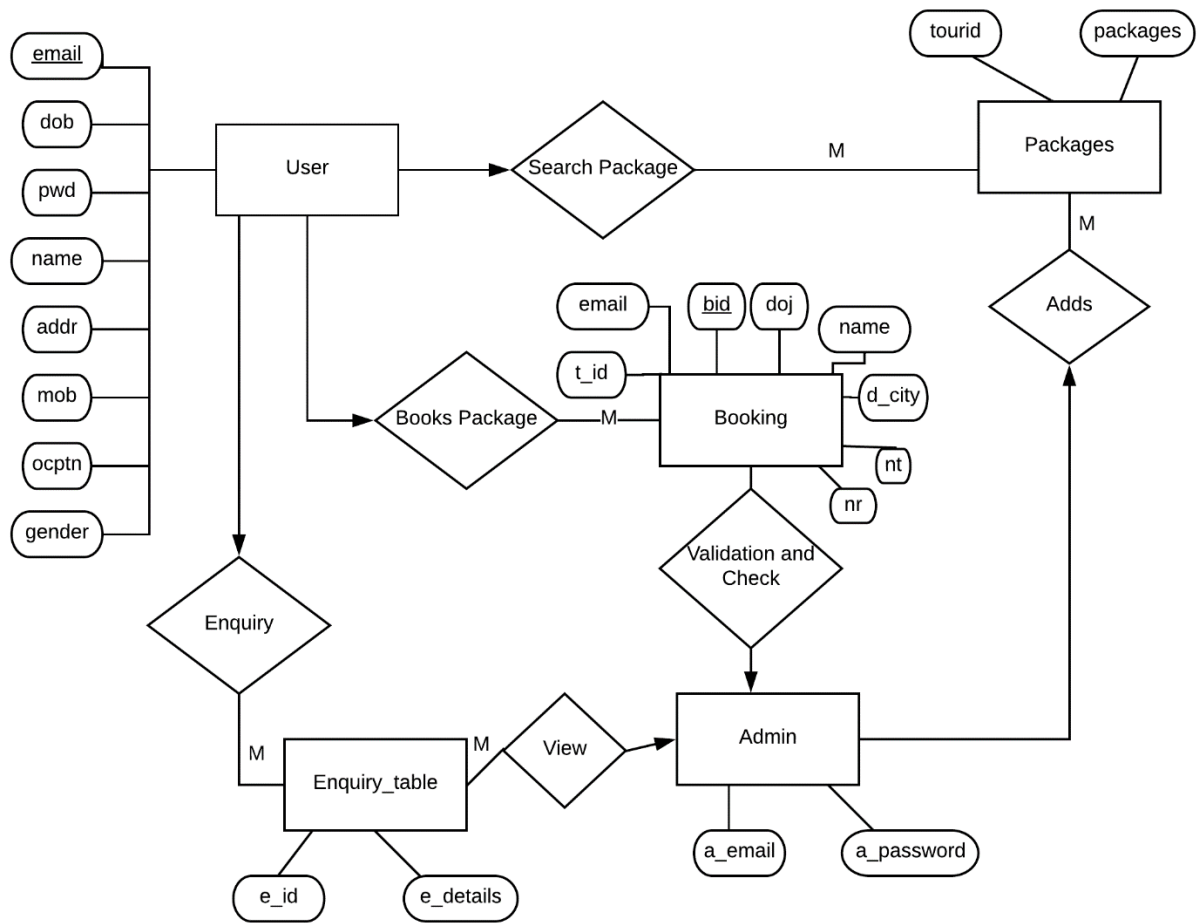


Figure: Entity-Relationship diagram

4.3 USE CASE DIAGRAM

A **use case diagram** at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well.

So only static behavior is not sufficient to model a system rather dynamic behavior is more important than static behavior. In UML there are five diagrams available to model dynamic nature and use case diagram is one of them. Now as we have to discuss that the use case diagram is dynamic in nature there should be some internal or external factors for making the interaction.

These internal and external agents are known as actors. So use case diagrams are consists of actors, use cases and their relationships. The diagram is used to model the system/subsystem of an application. A single use case diagram captures a particular functionality of a system. So to model the entire system numbers of use case diagrams are used. The purpose of use case diagram is to capture the dynamic aspect of a system. But this definition is too generic to describe the purpose. Because other four diagrams (activity, sequence, collaboration and State chart) are also having the same purpose. So we will look into some specific purpose which will distinguish it from other four diagrams.

Use case diagrams are used to gather the requirements of a system including internal and external influences. These requirements are mostly design requirements. So when a system is analyzed to gather its functionalities use cases are prepared and actors are identified.

Now when the initial task is complete use case diagrams are modelled to present the outside view. So in brief, the purposes of use case diagrams can be as follows:

- Used to gather requirements of a system.
- Used to get an outside view of a system.
- Identify external and internal factors influencing the system.
- Show the interacting among the requirements are actors.

How to draw Use Case Diagram?

Use case diagrams are considered for high level requirement analysis of a system. So when the requirements of a system are analyzed the functionalities are captured in use cases. So we can say that uses cases are nothing but the system functionalities written in an organized manner. Now the second things which are relevant to the use cases are the actors. Actors can be defined as something that interacts with the system.

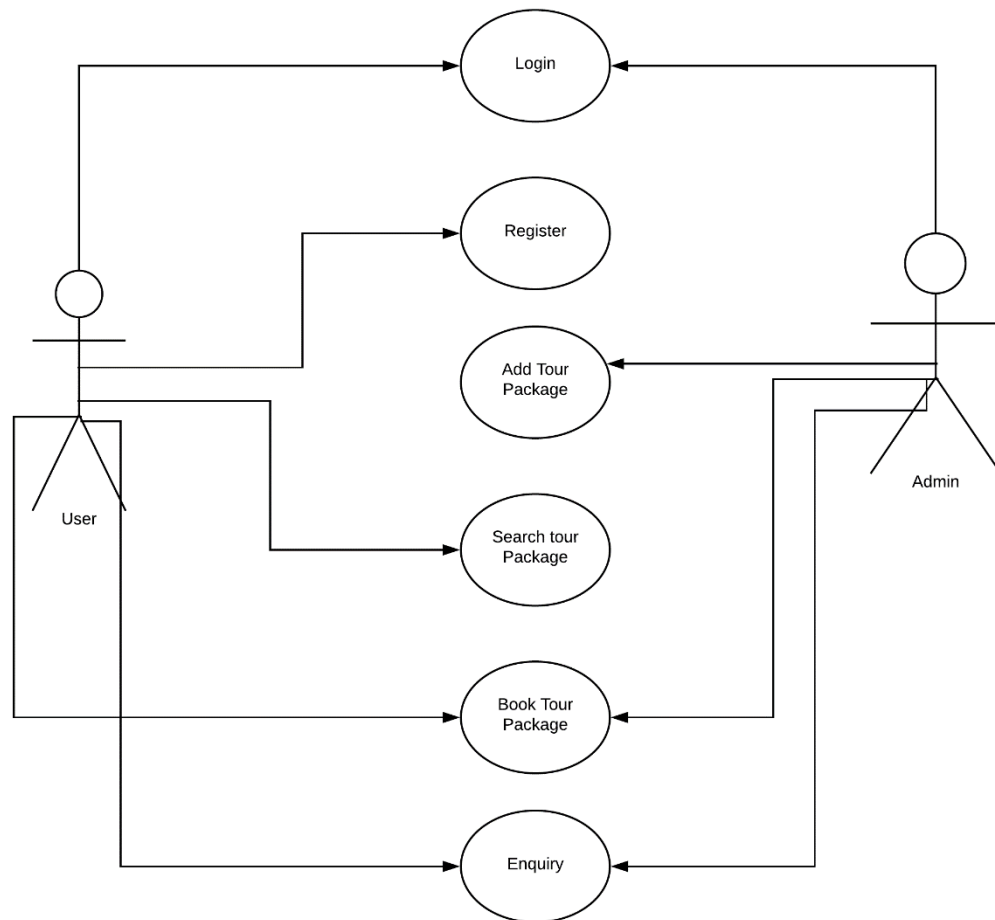
The actors can be human user, some internal applications or may be some external applications. So in a brief when we are planning to draw a use case diagram we should have the following items identified.

- Functionalities to be represented as an use case
- Actors
- Relationships among the use cases and actors.

Use case diagrams are drawn to capture the functional requirements of a system. So after identifying the above items we have to follow the following guidelines to draw an efficient use case diagram.

- The name of a use case is very important. So the name should be chosen in such a way so that it can identify the functionalities performed.
- Give a suitable name for actors.
- Show relationships and dependencies clearly in the diagram.

- Do not try to include all types of relationships. Because the main purpose of the diagram is to identify requirements.
- Use note whenever required to clarify some important points.



4.4 MODULARIZATION DETAILS

As Modularization has gained increasing focus from companies outside its traditional industries of aircraft and automotive, more and more companies turn to it as strategy and product development tool. I intend to explain the importance aspects of modularization and how it should be initiated within a company. After determining the theoretical steps of modularization success described in literature, I intend to conduct a multiple case study of companies who have implemented modularization in order to find how real world modularization was initiated and used to improve the company's competitiveness. By combining theory and practical approach to modularization I will derive at convergence and divergence between theoretical implementation to modularization and real world implementation to modularization. This gives a valuable input for both implantations in companies as well as new aspects to be further.

DATA INTEGRITY AND CONSTRAINTS

Data integrity is normally enforced in a database system by a series of integrity constraints or rules. Three types of integrity constraints are an inherent part of the relational data model: entity integrity, referential integrity and domain integrity:

- *Entity integrity* concerns the concept of a primary key. Entity integrity is an integrity rule which states that every table must have a primary key and that the column or columns chosen to be the primary key should be unique and not null.
- Concerns the concept of a foreign key. The referential integrity rule states that any foreign-key value can only be in one of two states. The usual state of affairs is that the foreign-key value refers to a primary key value of some table in the database. Occasionally, and this will depend on the rules of the data owner, a

foreign-key value can be null. In this case we are explicitly saying that either there is no relationship between the objects represented in the database or that this relationship is unknown.

- *Domain integrity* specifies that all columns in a relational database must be declared upon a defined domain. The primary unit of data in the relational data model is the data item. Such data items are said to be non-decomposable or atomic. A domain is a set of values of the same type.

4.5 DATABASE DESIGN

A database is an organized mechanism that has capability of storing information through which a user can retrieve stored information in an effective and efficient manner. The data is the purpose of any database and must be protected.

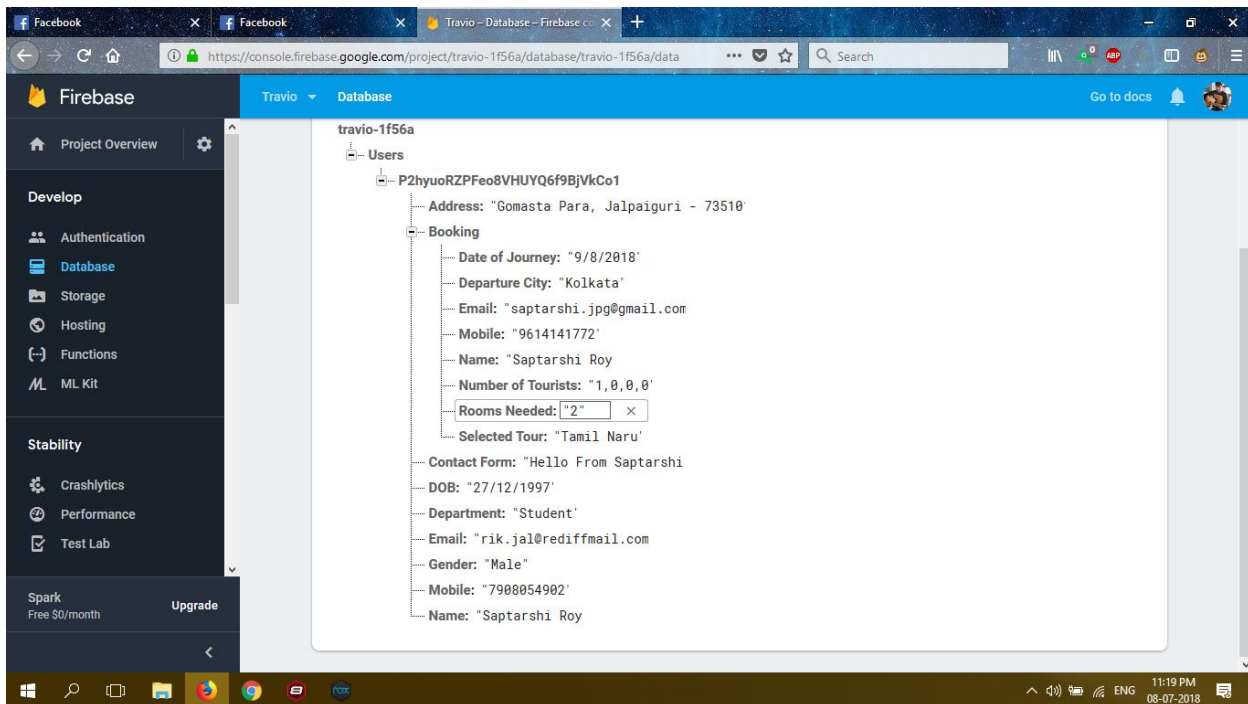
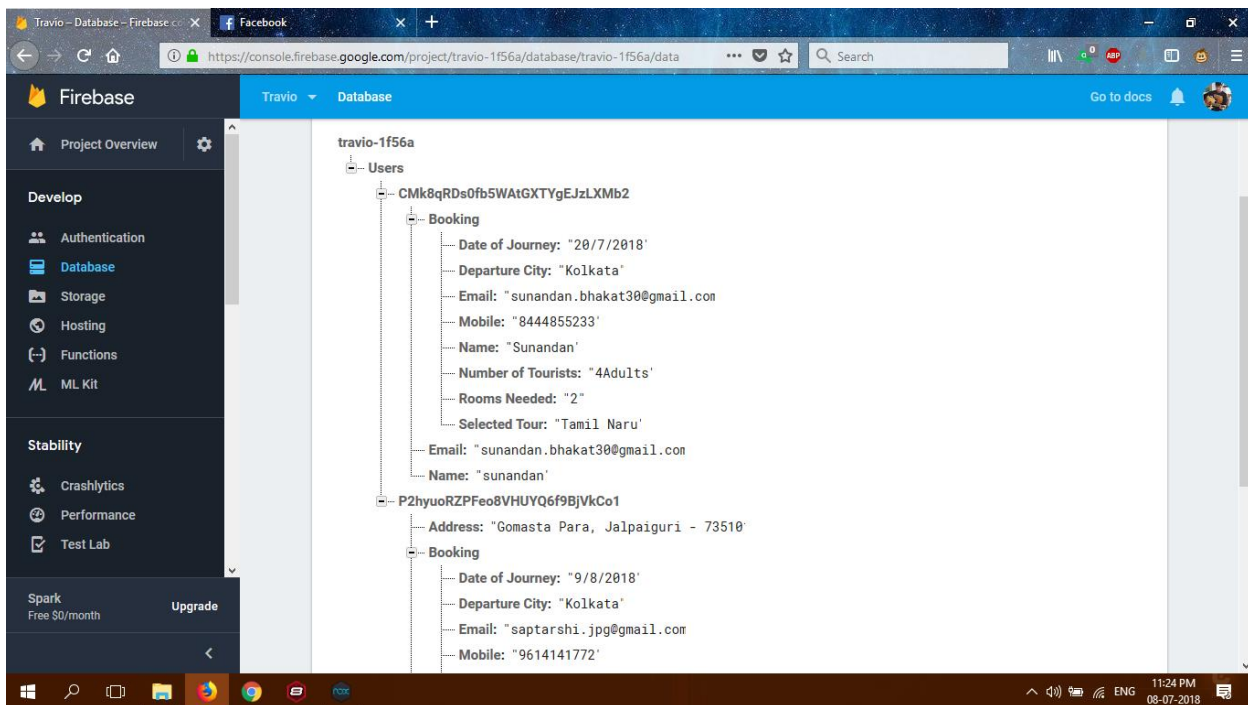
The database design is two level processes. In the first step, user requirements are gathered together and a database is designed which will meet these requirements as clearly as possible. This step is called information Level design and it is taken independent of any individual DBMS.

In the following snapshots we display the way we have used FIREBASE as the backend for our project and the various entities that have been used along with their table definition and table data.

HISTORY AND USE OF FIREBASE

Firebase provides a realtime database and backend as a service. The service provides application developers an API that allows application data to be synchronized across clients and stored on Firebase's cloud.^[19] The company provides client libraries that enable integration with [Android](#), [iOS](#), [JavaScript](#), [Java](#), [Objective-C](#), [swift](#) and [Node.js](#) applications. The database is also accessible through a REST API and bindings for several [JavaScript frameworks](#) such as [AngularJS](#), [React](#), [Ember.js](#) and [Backbone.js](#).^[19] The REST API uses the [Server-Sent Events](#) protocol, which is an API for creating HTTP connections for receiving push notifications from a server. Developers using the realtime database can secure their data by using the company's server-side-enforced security rules.^[20] Cloud Firestore which is Firebase's next generation of the Realtime Database was released for beta use.

DATA DICTIONARY



4.6 USER INTERFACE DESIGN

User interface design (UID) or **user interface engineering** is the design of user interfaces for machines and software, such as computers, home appliances, mobile devices, and other electronic devices, with the focus on maximizing the user experience. The goal of user interface design is to make the user's interaction as simple and efficient as possible, in terms of accomplishing user goals (user-centered design).

Good user interface design facilitates finishing the task at hand without drawing unnecessary attention to it. Graphic design and typography are utilized to support its usability, influencing how the user performs certain interactions and improving the aesthetic appeal of the design; design aesthetics may enhance or detract from the ability of users to use the functions of the interface. The design process must balance technical functionality and visual elements (e.g., mental model) to create a system that is not only operational but also usable and adaptable to changing user needs.

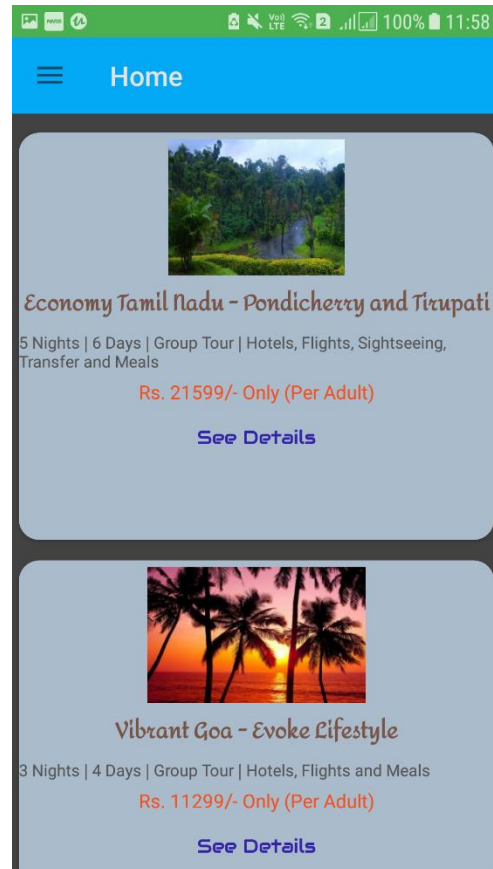
Interface design is involved in a wide range of projects from computer systems, to cars, to commercial planes; all of these projects involve much of the same basic human interactions yet also require some unique skills and knowledge. As a result, designers tend to specialize in certain types of projects and have skills centered on their expertise, whether that be software design, user research, web design, or industrial design.

SNAPSHOTS

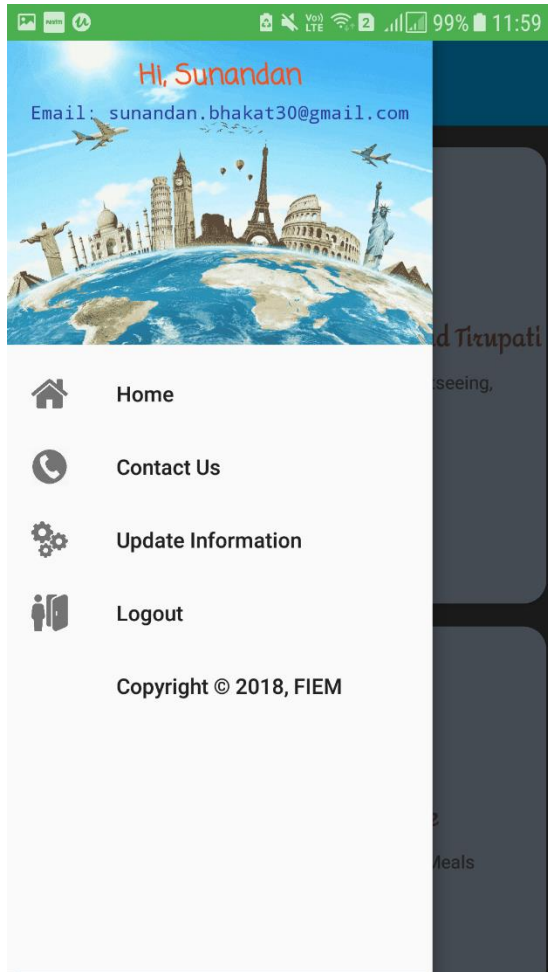
LOADING SCREEN



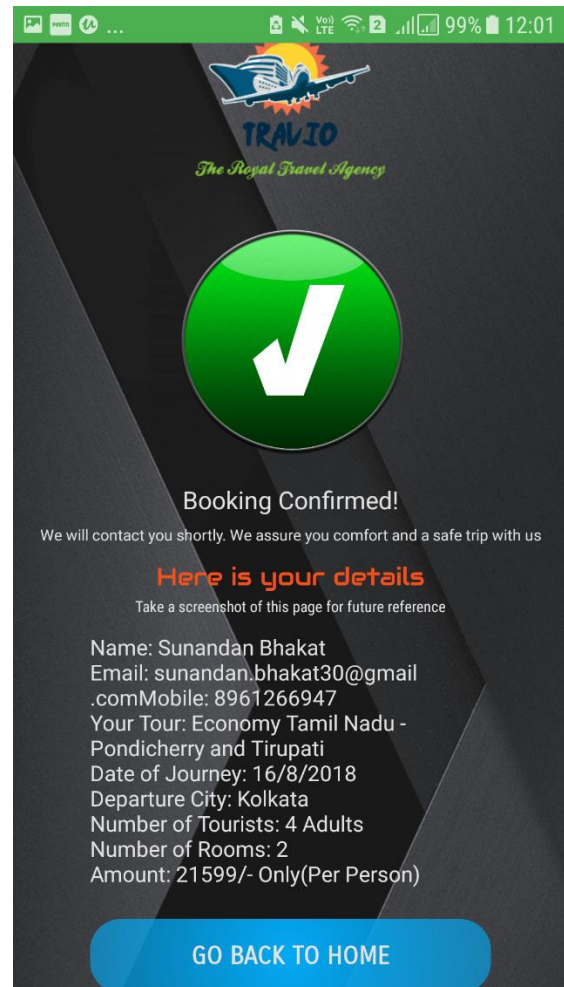
HOME SCREEN



SIGN IN



BOOKING CONFIRMATION



INDIVIDUAL PACKAGE VIEW




Vibrant Goa - Evoke Lifestyle

Tour Details

3 Nights | 4 Days
Goa(3N)

Accommodation:





Goa - Hotel Evoke Lifestyle

Plot No 252/4 Sequirra Vaddo Candolim, Goa.

What tour price includes?

- Welcome drinks on arrival (Non-Alcoholic)
- Mini Bar in room on chargeable basis
- Complimentary use of Tea/Coffee maker in all rooms, replenished once a day. Additional requirement will be at an extra cost
- Complimentary 02 bottles of mineral water, replenished daily
- Wi-Fi connectivity on complimentary basis for browsing only
- All applicable hotel taxes
- Complimentary use of Swimming Pool (with proper swim wear)

What your tour price does not include?

- Any monument entrance fees, camera fees, guide charges
- Any services not mentioned in inclusion list

Book Now!

CODING

```
<!--Android Manifest.xml-->

<?xml version="1.0" encoding="utf-8"?>

<manifest xmlns:android="http://schemas.android.com/apk/res/android"

    package="tk.musicoflife.travio">

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

    <uses-permission android:name="android.permission.READ_FRAME_BUFFER" />

    <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />

    <uses-permission android:name="android.permission.GET_TASKS" />

    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />

    <uses-permission android:name="android.permission.SYSTEM_ALERT_WINDOW" />

    <application

        android:allowBackup="true"

        android:icon="@mipmap/ic_launcher"

        android:label="@string/app_name"

        android:largeHeap="true"

        android:roundIcon="@mipmap/ic_launcher_round"

        android:supportRtl="true"

        android:theme="@style/AppTheme">

        <activity android:name=".MainActivity">
```

```

        <intent-filter>

            <action android:name="android.intent.action.MAIN" />

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

        </intent-filter> </activity>

<activity android:name=".HomeActivity" />

<activity android:name=".SigninActivity" />

<activity android:name=".SignupActivity" />

<activity android:name=".ContactActivity" />

<meta-data

    android:name="preloaded_fonts"

    android:resource="@array/preloaded_fonts" />

<activity android:name=".SettingsActivity" />

<activity android:name=".TamilnaruActivity" />

<activity android:name=".GoaActivity" />

<activity android:name=".KeralaActivity" />

<activity android:name=".BookingActivity" />

<activity android:name=".CompleteActivity" />

<activity android:name=".AdminLoginActivity" />

<activity android:name=".AdminHomeActivity"></activity>

</application> </manifest>

```

BOOKINGACTIVITY.JAVA

```
package tk.musicoflife.travio;

import android.app.DatePickerDialog;

import android.app.Dialog;

import android.app.ProgressDialog;

import android.content.Intent;

import android.support.annotation.NonNull;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.text.TextUtils;

import android.view.View;

import android.view.inputmethod.EditorInfo;

import android.widget.AdapterView;

import android.widget.AdapterView.OnItemClickListener;

import android.widget.ArrayAdapter;

import android.widget.Button;

import android.widget.DatePicker;

import android.widget.EditText;

import android.widget.Spinner;

import android.widget.Toast;


import com.google.android.gms.tasks.OnCompleteListener;

import com.google.android.gms.tasks.Task;

import com.google.firebase.auth.FirebaseAuth;

import com.google.firebase.database.DataSnapshot;

import com.google.firebase.database.DatabaseError;

import com.google.firebase.database.DatabaseReference;

import com.google.firebase.database.FirebaseDatabase;
```

```

import com.google.firebase.database.ValueEventListener;

import java.util.Calendar;

import java.util.HashMap;

public class BookingActivity extends AppCompatActivity {

    Spinner spin, spin2;

    ArrayAdapter<CharSequence> adapter, adapter2;

    private Button datebtn, submitbutton;

    int yearx, dayx, monthx;

    static final int DIALOG_ID =0;

    private EditText rooms, numbers, name, contact, email, dateofjourney;

    private FirebaseAuth mAuth;

    private DatabaseReference userref, uref, urr, ur;

    private ProgressDialog loadingbar;

    String currentID, cityy, tour, namee;

    int amount;

    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_booking);

        mAuth = FirebaseAuth.getInstance();

        currentID = mAuth.getCurrentUser().getUid();

        userref = FirebaseDatabase.getInstance().getReference().child("Users").child(currentID).child("Booking");

        uref = FirebaseDatabase.getInstance().getReference().child("Userinfo");
    }

```

```

urr = FirebaseDatabase.getInstance().getReference().child("Users").child(currentID);

loadingbar = new ProgressDialog(this);

final Calendar cal = Calendar.getInstance();

yearx = cal.get(Calendar.YEAR);

monthx = cal.get(Calendar.MONTH);

dayx = cal.get(Calendar.DAY_OF_MONTH);


rooms = (EditText)findViewById(R.id.booking_room);

numbers = (EditText)findViewById(R.id.booking_numbers);

name = (EditText)findViewById(R.id.booking_name);

contact = (EditText)findViewById(R.id.booking_contactno);

email = (EditText)findViewById(R.id.booking_email);

dateofjourney = (EditText)findViewById(R.id.booking_date);

datebtn = (Button)findViewById(R.id.booking_date_btn);

submitbutton = (Button)findViewById(R.id.booking_btn);

spin = (Spinner)findViewById(R.id.spinnercity);

spin2 = (Spinner)findViewById(R.id.spinnerplace);

adapter          =          ArrayAdapter.createFromResource(this,          R.array.city,
R.layout.support_simple_spinner_dropdown_item);

adapter2          =          ArrayAdapter.createFromResource(this,          R.array.place,
R.layout.support_simple_spinner_dropdown_item);

adapter.setDropDownViewResource(R.layout.support_simple_spinner_dropdown_item);

adapter2.setDropDownViewResource(R.layout.support_simple_spinner_dropdown_item);

spin.setAdapter(adapter);

spin2.setAdapter(adapter2);

showDialogonbtn();

spin.setOnItemClickListener(new AdapterView.OnItemClickListener() {

```



```

@Override

public void onItemSelected(AdapterView<?> parent, View view, int position, long id) {

    cityy = (String) parent.getItemAtPosition(position);

}

@Override

public void onNothingSelected(AdapterView<?> parent) {

}

});

spin2.setOnItemClickListener(new AdapterView.OnItemClickListener() {

@Override

public void onItemSelected(AdapterView<?> parent, View view, int position, long id) {

    tour = (String) parent.getItemAtPosition(position);

    switch (tour){

        case "Economy Tamil Nadu - Pondicherry and Tirupati":

            amount = 21599;

            break;

        case "Kerala - Hills and Backwaters":

            amount = 13599;

            break;

        case "Vibrant Goa - Evoke Lifestyle":

            amount = 11299;

            break;

    }

}

}

```

```

@Override

public void onNothingSelected(AdapterView<?> parent) {

}

});

submitButton.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

    SaveInformation();

}

});

urr.addValueEventListener(new ValueEventListener() {

@Override

public void onDataChange(DataSnapshot dataSnapshot) {

    if(dataSnapshot.exists()){

        if(dataSnapshot.hasChild("Name")){

            nameee = dataSnapshot.child("Name").getValue().toString();

            ur = uref.child(nameee).child("Booking");

        }

        else {

            Toast.makeText(BookingActivity.this, "No name exists!",
Toast.LENGTH_SHORT).show();

        }

    }

    else {

        Toast.makeText(BookingActivity.this, "Error in database!",
Toast.LENGTH_SHORT).show();

```

```

        }
    }

    @Override
    public void onCancelled(DatabaseError databaseError) {
    }
});
}

private void SaveInformation() {
    String roomsneeded = rooms.getText().toString();
    String numbersoftourists = numbers.getText().toString();
    String fullname = name.getText().toString();
    String mobileneno = contact.getText().toString();
    String emailaddress = email.getText().toString();
    String journeydate = dateofjourney.getText().toString();

    if(TextUtils.isEmpty(roomsneeded)){
        Toast.makeText(this, "Please enter the number of rooms needed",
Toast.LENGTH_SHORT).show();
    }
    else if(TextUtils.isEmpty(numbersoftourists)){
        Toast.makeText(this, "Please enter the number of persons", Toast.LENGTH_SHORT).show();
    }
    else if(TextUtils.isEmpty(fullname)){
        Toast.makeText(this, "Please enter your full name", Toast.LENGTH_SHORT).show();
    }
}

```

```

else if(TextUtils.isEmpty(mobileno)){

    Toast.makeText(this, "Please enter your contact number", Toast.LENGTH_SHORT).show();

}

else if(TextUtils.isEmpty(emailaddress)){

    Toast.makeText(this, "Please enter your Email Address", Toast.LENGTH_SHORT).show();

}

else if(TextUtils.isEmpty(journeydate)){

    Toast.makeText(this, "Please enter the date of journey", Toast.LENGTH_SHORT).show();

}

else {

    loadingbar.setTitle("Confirming your booking");

    loadingbar.setMessage("Please wait while we confirm your booking with us...");

    loadingbar.show();

    loadingbar.setCanceledOnTouchOutside(true);


    HashMap usermap = new HashMap();

    usermap.put("Rooms Needed", roomsneeded);

    usermap.put("Number of Tourists", numbersoftourists);

    usermap.put("Name", fullname);

    usermap.put("Mobile", mobileno);

    usermap.put("Email", emailaddress);

    usermap.put("Date of Journey", journeydate);

    usermap.put("Departure City", cityy);

    usermap.put("Selected Tour", tour);

    usermap.put("Amount", amount);

    userref.updateChildren(usermap).addOnCompleteListener(new OnCompleteListener() {

```

```

@Override

public void onComplete(@NonNull Task task) {

    if(task.isSuccessful()){

        SendToComplete();

        loadingbar.dismiss();

        Toast.makeText(BookingActivity.this, "Your booking was successfully done!",
Toast.LENGTH_SHORT).show();

    }

    else {

        String mesg = task.getException().getMessage();

        Toast.makeText(BookingActivity.this, "Error occurred: " + mesg,
Toast.LENGTH_LONG).show();

        loadingbar.dismiss();

    }

}

});

ur.updateChildren(usermap);

}

}

private void SendToComplete() {

    Intent in = new Intent(BookingActivity.this, CompleteActivity.class);

    in.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK | Intent.FLAG_ACTIVITY_CLEAR_TASK);

    startActivity(in);

    finish();

}

public void showDialogonbtn(){

```

```

datebtn.setOnClickListener(new View.OnClickListener() {

    @Override

    public void onClick(View v) {

        showDialog(DIALOG_ID);

    }

});

}

@Override

protected Dialog onCreateDialog(int ID){

    if(ID == DIALOG_ID)

        return new DatePickerDialog(this, dpListener, yearx, monthx, dayx);

    return null;

}

private DatePickerDialog.OnDateSetListener dpListener = new DatePickerDialog.OnDateSetListener()
{

    @Override

    public void onDateSet(DatePicker view, int year, int month, int dayOfMonth) {

        yearx = year;

        monthx = month+1;

        dayx = dayOfMonth;

        String dateofbirth = dayx+"/"+monthx+"/"+yearx;

        dateofjourney.setText(dateofbirth);

    }

};

}

```

IMPLEMENTATION AND TESTING

IMPLEMENTATION AND TESTING

A software system test plan is a document that describes the objectives, scope, approach and focus of software testing effort. The process of preparing a test plan is a usual way to think the efforts needed to validate the acceptability of a software product. The complete document will help people outside the test group understand the "WHY" and "HOW" product validation. It should be thorough enough to be useful but not so thorough that no one outside the test group will read it.

5.1 INTRODUCTION

Testing is the process of running a system with the intention of finding errors. Testing enhances the integrity of a system by detecting deviations in design and errors in the system. Testing aims at detecting error-prone areas. This helps in the prevention of errors in a system. Testing also adds value to the product by conforming to the user requirements.

The main purpose of testing is to detect errors and error-prone areas in a system. Testing must be thorough and well-planned. A partially tested system is as bad as an untested system. And the price of an untested and under-tested system is high.

The implementation is the final and important phase. It involves user-training, system testing in order to ensure successful running of the proposed system. The user tests the system and changes are made according to their needs. The testing involves the testing of the developed system using various kinds of data. While testing, errors are noted and correctness is the mode.

5.2 OBJECTIVES OF TESTING

The objective our test plan is to find and report as many bugs as possible to improve the integrity of our program. Although exhaustive testing is not possible, we will exercise a broad range of tests to achieve our goal. Our user interface to utilize these functions is designed to be user-friendly and provide easy manipulation of the tree. The application will only be used as a demonstration tool, but we would like to ensure that it could be run from a variety of platforms with little impact on performance or usability.

Process Overview

The following represents the overall flow of the testing process:

1. Identify the requirements to be tested. All test cases shall be derived using the current Program Specification.
2. Identify which particular test(s) will be used to test each module.
3. Review the test data and test cases to ensure that the unit has been thoroughly verified and that the test data and test cases are adequate to verify proper operation of the unit.
4. Identify the expected results for each test.

5. Document the test case configuration, test data, and expected results.
6. Perform the test(s).
7. Document the test data, test cases, and test configuration used during the testing process. This information shall be submitted via the Unit/System Test Report (STR).
8. Successful unit testing is required before the unit is eligible for component integration/system testing.
9. Unsuccessful testing requires a Bug Report Form to be generated. This document shall describe the test case, the problem encountered, its possible cause, and the sequence of events that led to the problem. It shall be used as a basis for later technical analysis.
10. Test documents and reports shall be submitted. Any specifications to be reviewed, revised, or updated shall be handled immediately.

5.3 TEST CASES

A test case is a document that describe an input, action, or event and expected response, to determine if a feature of an application is working correctly. A test case should contain particular such as test case identifier, test condition, input data Requirement expected results. The process of developing test cases can help find problems in the requirement or design of an application, since it requires completely thinking through the operation of the application.

TESTING STEPS

UNIT TESTING

Unit testing focuses efforts on the smallest unit of software design. This is known as module testing. The modules are tested separately. The test is carried out during programming stage itself. In this step, each module is found to be working satisfactory as regards to the expected output from the module.

INTEGRATION TESTING

Data can be lost across an interface. One module can have an adverse effect on another, sub functions, when combined, may not be linked in desired manner in major functions. Integration testing is a systematic approach for constructing the program structure, while at the same time conducting test to uncover errors associated within the interface. The objective is to take unit tested modules and builds program structure. All the modules are combined and tested as a whole.

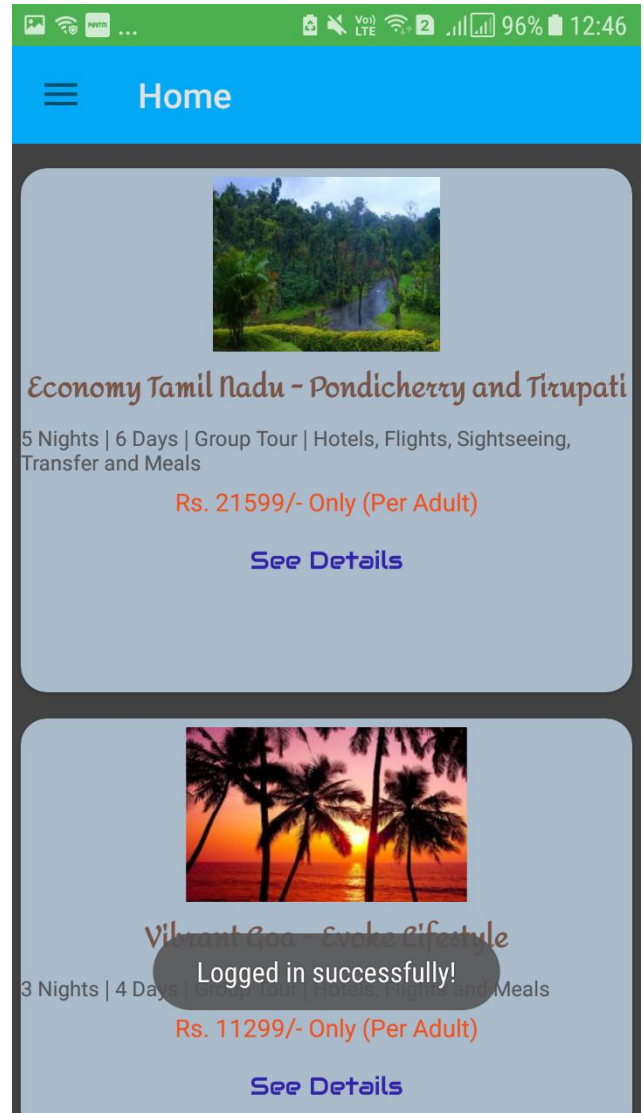
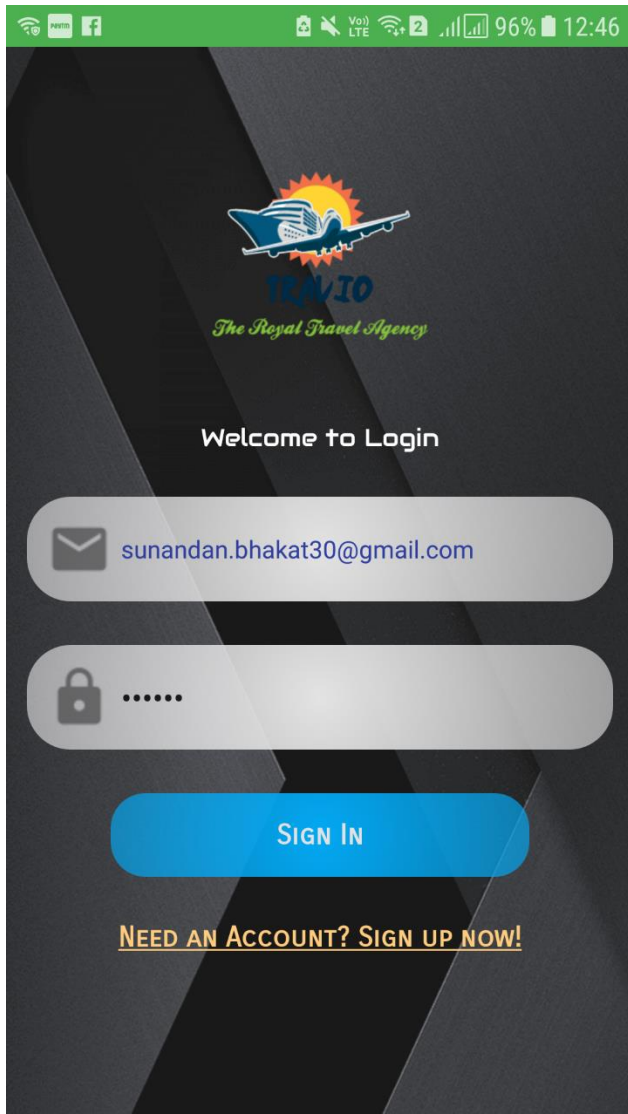
VALIDATION

At the culmination of the integration testing, Software is completely assembled as a package. Interfacing errors have been uncovered and corrected and a final series of software test begin in validation testing. Validation testing can be defined in many ways, but a simple definition is that the validation succeeds when the software functions in a manner that is expected by the customer. After validation test has been conducted, one of the three possible conditions exists.

- a) The function or performance characteristics confirm to specification and are accepted.
- b) A deviation from specification is uncovered and a deficiency lists is created.
- c) Proposed system under consideration has been tested by using validation test and found to be working satisfactory.

Tested By:	Debarya Ganguly
Test Type	Unit Testing
Test Case Number	1
Test Case Name	User Identification
Test Case Description	The user should enter his/ her accurate user id and password so that he/she can able to go for the further options. The test case will check the application for the same since a user can only login with the correct user id and password.
Item(s) to be tested	
1	Verification of the user id and password with the record in the database.
Specifications	
Input	Expected Output/Result
1) Correct User id and password 2) Incorrect Id or Password	1) Successful login 2) Failure Message

SNAPSHOT



Tested By:		Sagnik Dey Sarkar
Test Type		Unit Testing
Test Case Number		2
Test Case Name		Booking
Test Case Description		A user will choose all his/her respective package and fill in details. These details wil be stored in the database.
Item(s) to be tested		
1	Required fields in the form are not empty, validation of proper credentials	
Specifications		
Input		Expected Output/Result
1) Trip, Departure City, no. of rooms required, no. of adults, email, name, date of journey. 2) Empty field, Invalid entry		1) Successful Booking 2) Failure Message

SNAPSHOT

ET ... 95% 12:56

Please fill in your details


Select Trip: Economy Tamil Nadu - Pon..


Select Departure City: Kolkata


1


2

Your contact details:

 Sarbajit Chatterjee


 8017910478


 sarbajitchatterjee@hotmail.com

 26/9/2018 [Select Date](#)

[CONFIRM BOOKING](#)

ET ... 95% 12:56


TRAVIO
The Royal Travel Agency



Booking Confirmed!

We will contact you shortly. We assure you comfort and a safe trip with us

Here is your details

Take a screenshot of this page for future reference

Name: Sarbajit Chatterjee
Email: sarbajitchatterjee@hotmail.com
Mobile: 8017910478
Your Tour: Economy Tamil Nadu - Pondicherry and Tirupati
Date of Journey: 26/9/2018
Departure City: Kolkata
Number of Tourists: 2
Number of Rooms: 1
Amount: 21599/- Only(Per Person)

[GO BACK TO HOME](#)

5.4 WHITE BOX TESTING

In white box testing, the UI is bypassed. Inputs and outputs are tested directly at the code level and the results are compared against specifications. This form of testing ignores the function of the program under test and will focus only on its code and the structure of that code. Test case designers shall generate cases that not only cause each condition to take on all possible values at least once, but that cause each such condition to be executed at least once. To ensure this happens, we will be applying Branch Testing. Because the functionality of the program is relatively simple, this method will be feasible to apply.

Each function of the binary tree repository is executed independently; therefore, a program flow for each function has been derived from the code.

5.5 BLACK BOX TESTING

Black box testing typically involves running through every possible input to verify that it results in the right outputs using the software as an end-user would. We have decided to perform Equivalence Partitioning and Boundary Value Analysis testing on our application.

System Testing

The goals of system testing are to detect faults that can only be exposed by testing the entire integrated system or some major part of it. Generally, system testing is mainly concerned with areas such as performance, security, validation, load/stress, and configuration sensitivity. But in our case we will focus only on function validation and performance. And in both cases, we will use the black-box method of testing.

5.6 OUTPUT TESTING

After performing the validation testing, the next step is output testing of the proposed system, since no system could be useful if it does not produce the required output in a specific format. The output format on the screen is found to be correct. The format was designed in the system design time according to the user needs. For the hardcopy also; the output comes as per the specified requirements by the user. Hence output testing did not result in any correction for the system.

5.7 USER ACCEPTANCE TESTING

User acceptance of a system is the key factor for the success of any system. The system under consideration is tested for the user acceptance by constantly keeping in touch with the prospective system users at the time of developing and making changes whenever required.

This is done in regard to the following point:

- a) Input Screen Design
- b) Output Screen Design
- c) Format of reports and other outputs.

5.8 INTEGRATION TESTING

Software testing is always used in association with verification and validation. In the testing phase of this project our aim is to find the answer to following two questions.

- Whether the software matches with the specification (i.e. process base) to verify the product.
- Whether this software in one client what wants (i.e. product base) to validate the product.
- Unit testing and integration testing has been carried out to find the answer to above questions. In unit testing each individual module was test to find any unexpected behaviour if exists. Later all the module was integrated and flat file was generated.

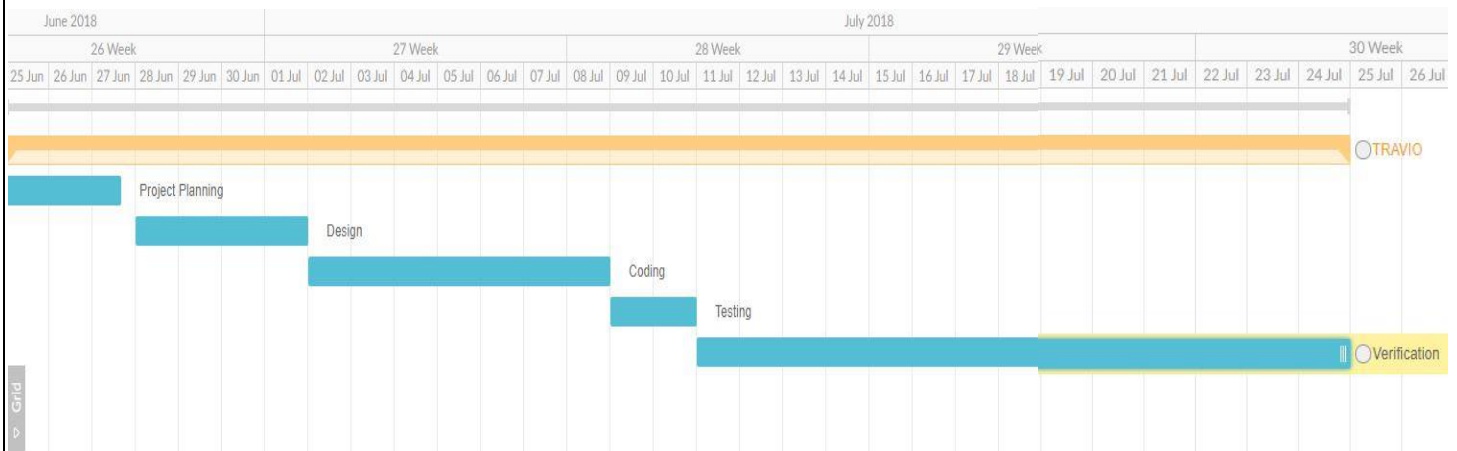
FUNCTIONAL TESTING

These are the points concerned during the stress test:

- Nominal input: character is in putted in the place of digits and the system has to flash the message "Data error"
- Boundary value analysis: exhaustive test cases have designed to create an output report that produces the maximum (and minimum) allowable number of table entries.

6 GANTT CHART

Task name		Start date	End date	Duration day	+
<input type="checkbox"/> Total estimate		25/06/2018	24/07/2018	30	
<input type="checkbox"/> TRAVIO		25/06/2018	24/07/2018	30	
Project Planning		25/06/2018	27/06/2018	2.67	
Design		28/06/2018	01/07/2018	4	
Coding		02/07/2018	08/07/2018	7	
Testing		09/07/2018	10/07/2018	2	
Verification		11/07/2018	24/07/2018	14	



SOFTWARE SYTEM ATTRIBUTES

7.1 DATABASE SECURITY

System security measure is meant to be provided to make your system reliable and secured from unauthorized user may create threats to the system. So you should follow some security measures. We have used security levels in database level at system level.

7.2 SYSTEM SECURITY

If we talk about the system security in our proposed system we have implemented with the help of maintain the session throughout the system's use. Once a user has logged out than he/she will not be able to perform any task before signing back again. A high level of authentic login is given to the system so this is a very tedious task to enter without authorization and authentication.

7.3 LIMITATIONS

- Since it is an online project, customers need internet connection to use it.
- People who are not familiar with android platform can't use this application.
- Customer must have debit card or credit card to book tickets.

8. CONCLUSION

This project has been appreciated by all the users in the organization. It is easy to use, since it uses the GUI provided in the user dialog. User friendly screens are provided. The usage of software increases the efficiency, decreases the effort. It has been efficiently employed as a Site management mechanism. It has been thoroughly tested and implemented.

The project “**TRAVIO**” is the ideal place for every traveler who will get all the information about different Tour Packages. It will provide user the benefit of booking tour packages just with a touch sitting in his/her home without having to make the long walk and standing in queues.

The software collects the credit card information of the user and it provides a secure gateway for all kinds of transaction online. The software provides a reliable platform for keeping all sensitive information. For this kind of online business, the special software must be installed on the server which host the site, or on a secure server which receives all sensitive data.

9 FUTURE SCOPE AND FURTHER ENHANCEMENTS

In future, we would like to keep working on this project and make new additions to provide users with more advanced features and more detailed information. We have set our sights on the following additions in future-

1. Addition of Flight, Train and car rental facility.
2. Apply Machine Learning on previous User Booking and Search data and give appropriate suggestions based on user's choices.
3. Introduce Foreign Tourism
4. Integration of a chat bot with the app to provide better assistance to users for booking and enquiry.

10 REFERENCES

1. <https://www.android.com>
2. <https://www.slideshare.com>
3. <https://www.scribd.com>
4. <https://www.tutorialspoint.com>
5. <https://www.youtube.com>

THANK YOU