

“SMART APP FOR AIRPORT FEEDBACK”

A

Project Report

submitted

in partial fulfillment

for the award of the Degree of

Bachelor of Technology

in Information Technology



Project Mentor:

Ms. Dolly Mittal
Associate Professor

Submitted By :

Shashank Varshney 19ESKIT086
Shivam Thakkar 19ESKIT087
Sujal Soman 19ESKIT092
Yuvraj Gakkhar 19ESKIT100

Department of Information Technology
Swami Keshvanand Institute of Technology, Management & Gramothan,
Jaipur
Rajasthan Technical University, Kota
Session 2022-2023

**Swami Keshvanand Institute of Technology,
Management & Gramothan, Jaipur
Department of Information Technology**

CERTIFICATE

This is to certify that **Shashank Varshney (19ESKIT086)**, a student of B.Tech(Information Technology) **VIII** semester has submitted his Project Report entitled "**Smart App For Airport Feedback**" under my guidance.

Mentor

Ms. Dolly Mittal

Associate Professor

Signature

Coordinator

Ms. Sanju Choudhary

Associate Professor

Signature

**Swami Keshvanand Institute of Technology,
Management & Gramothan, Jaipur**

Department of Information Technology

CERTIFICATE

This is to certify that **Shivam Thakkar (19ESKIT087)**, a student of
B.Tech(Information Technology) **VIII** semester has submitted his Project Report
entitled **Smart App For Airport Feedback** under my guidance.

Mentor

Ms. Dolly Mittal

Associate Professor

Signature

Coordinator

Ms. Sanju Choudhary

Associate Professor

Signature

**Swami Keshvanand Institute of Technology,
Management & Gramothan, Jaipur**

Department of Information Technology

CERTIFICATE

This is to certify that **Sujal Somani (19ESKIT092)**, a student of
B.Tech(Information Technology) **VIII** semester has submitted his Project Report
entitled **Smart App For Airport Feedback** under my guidance.

Mentor

Ms. Dolly Mittal

Associate Professor

Signature

Coordinator

Ms. Sanju Choudhary

Associate Professor

Signature

**Swami Keshvanand Institute of Technology,
Management & Gramothan, Jaipur**

Department of Information Technology

CERTIFICATE

This is to certify that **Yuvraj Gakkhar (19ESKIT100)**, a student of
B.Tech(Information Technology) **VIII** semester has submitted his Project Report
entitled **Smart App For Airport Feedback** under my guidance.

Mentor

Ms. Dolly Mittal

Associate Professor

Signature

Coordinator

Ms. Sanju Choudhary

Associate Professor

Signature

DECLARATION

We hereby declare that the report of the project entitled " Smart App for Airport Feedback " is a record of an original work done by us at Swami Keshvanand Institute of Technology, Management and Gramothan, Jaipur under the mentorship of "Ms.Dolly Mittal "(Dept. of Information Technology) and coordination of "Ms.Sanju Choudhary" (Dept.of Information Technology). This project report has been submitted as the proof of original work for the partial fulfillment of the requirement for the award of the degree of Bachelor of Technology (B.Tech) in the Information Technology. It has not been submitted anywhere else, under any other program to the best of our knowledge and belief.

Team Members

Shashank Varshney 19ESKIT086
Shivam Thakkar 19ESKIT087
Sujal Somani 19ESKIT092
Yuvraj Gakkhar 19ESKIT100

Signature

Acknowledgement

A project of such a vast coverage cannot be realized without help from numerous sources and people in the organization. We take this opportunity to express our gratitude to all those who have been helping us in making this project successful.

We are highly indebted to our faculty mentor Ms. Dolly Mittal. She has been a guide, motivator source of inspiration for us to carry out the necessary proceedings for the project to be completed successfully. We also thank our project coordinator Ms. Sanju Choudhary for her co-operation, encouragement, valuable suggestions and critical remarks that galvanized our efforts in the right direction.

We would also like to convey our sincere thanks to Prof. Dr. Anil Chaudhary, HOD, Department of Information Technology, for facilitating, motivating and supporting us during each phase of development of the project. Also, we pay our sincere gratitude to all the Faculty Members of Swami Keshvanand Institute of Technology, Management and Gramothan, Jaipur and all our Colleagues for their co-operation and support.

Last but not least we would like to thank all those who have directly or indirectly helped and cooperated in accomplishing this project.

Team Members:

Shashank Varshney 19ESKIT086

Shivam Thakkar 19ESKIT087

Sujal Somanı 19ESKIT092

Yuvraj Gakkhar 19ESKIT100

Table of Content

1	Introduction	2
1.1	Problem Statement & Objective	2
1.2	Investigation and Analysis	3
1.3	Introduction to Project	4
1.4	Proposed Solution	5
1.5	Scope of the Project	6
2	Software Requirement Specification	7
2.1	Overall Description	7
2.1.1	Product Perspective	7
2.1.1.1	System Interfaces	7
2.1.1.2	User Interfaces	7
2.1.1.3	Hardware Interfaces	8
2.1.1.4	Software Interfaces	8
2.1.1.5	Communications Interfaces	9
2.1.1.6	Memory Constraints	9
2.1.1.7	Operations	9
2.1.1.8	Site Adaption Requirements	10
2.1.2	Project Functions	10
2.1.3	User Characteristics	10
2.1.4	Constraints	11
2.1.5	Assumption and Dependencies	11
2.2	Specific Requirements	11
2.2.1	User Interfaces Requirements	11
2.2.2	System Product Features	12
2.2.2.1	Security	12
2.2.2.2	Maintainability	12

2.2.3	Portability	13
3	System Design Specification	14
3.1	System Architecture	14
3.2	Module Decomposition Description	14
3.3	High Level Design Diagrams	16
3.3.1	Use Case Diagram	16
3.3.2	Activity Diagram	17
3.3.3	Data-Flow Diagram	20
3.3.4	Sequence Diagram	21
3.3.5	Class Diagram	24
3.3.6	ER Diagram	25
4	Methodology and Team	26
4.1	Introduction to Waterfall Framework	26
4.2	Team Members, Roles & Responsibilities	29
5	Centering System Testing	30
5.1	Functionality Testing	30
5.2	Performance Testing	31
5.3	Usability Testing	32
6	Test Execution Summary	33
7	Project Screen Shots	34
8	Project Summary and Conclusions	40
8.1	Conclusion	40
9	Future Scope	41
References		42

List of Figures

1.1	Architecture Diagram of the Proposed System Methodology	6
3.1	Use Case diagram	16
3.2	Activity Diagram Representing Passenger Registration	17
3.3	Activity Diagram Representing Passenger Feedback	18
3.4	Activity Diagram Representing Admin Login	19
3.5	Activity Diagram Representing Admin View Feedback	19
3.6	Data-Flow Diagram	20
3.7	Sequence Diagram Representing Passenger	21
3.8	Sequence Diagram Representing Admin View Feedback	22
3.9	Sequence Diagram Representing Admin Login	23
3.10	Class Diagram	24
3.11	ER Diagram	25
4.1	WaterFall Model	27
7.1	Passenger Login Screen	34
7.2	Passenger Home Displaying Flight Details	34
7.3	Feedback Page (Common For both Passenger and Admin)	35
7.4	Passenger Rating Form	35
7.5	Admin Login Screen	36
7.6	FoodCourt View Feedback Page	36
7.7	ChechIn FeedBack Page	37
7.8	About Page	37
7.9	Admin Collection	38
7.10	Airline Feedback Collection	38
7.11	CheckIn Feedback Collection	39
7.12	User Collection	39

List of Tables

2.1	Minimum Client Side Hardware Interface	8
2.2	Minimum Server Side Hardware Interface	8
2.3	Minimum Software Interface	9
4.1	Roles and Responsibilities	29
6.1	Test Case Summary	33

Chapter 1

Introduction

1.1 Problem Statement & Objective

Problem: Improper system for feedback regarding airport facilities.

Reasons: The reason for building the android app is to get the feedback regarding the airport facilities which are provided at the airport and to manage the database from which passengers can access for their belongings.

Solutions: This app would help the management and co-ordination between the passengers and airport staff. The purpose of the Inventory management system for taking feedback at the airport is to create a convenient and easy-to-use application for passengers. The system is based on a relational database. Above all, we hope to provide a comfortable user experience with this managerial application

This system can be used to accomplish following tasks, keeping in mind that this software will have an adequate life:

- This gives the facility to provide co-ordination between airport staff and passengers.
- Also allows the passengers to give their views about facilities

This is an advance step towards an automated system that can be useful to passengers to provide feedback and to access their belongings.

Objective: The purpose of this document is to build an online system to take feedback regarding airport facilities which are provided at the airport. The project is based on an web app using which the airport authority can manage the database and passengers can access for their belonging.

1.2 Investigation and Analysis

All users of this module can do the following functions using this web application.

Passengers

- Can submit their feedback according their preference.
- They can provide their experience based feedback after using airport provided services
- Receive Email Confirmation After Successfully submition of feedback
- Can see their flight details

Administrative

- Check the feedbacks provided by the passengers

System Analysis

System analysis is the process of gathering and interpreting facts, diagnosing problems and using the information to recommend improvements on the system. System analysis 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 an interrogator and dwells deep into the working of the present system. The system is viewed as a whole and the inputs to the system are identified. The outputs from the organization are traced through the various processing that the inputs phase through in the organization.

A detailed study of these processes 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. The designer now functions as a problem solver and tries to sort out the difficulties that the enterprise faces. The solutions are given as a proposal. 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 loop ends as soon as the user is satisfied with the proposal.

Analysis gathers the requirements for the system. This stage includes a detailed study of the business needs of the organization

Design focuses on high level design like, what programs are needed and how are they going to interact, low-level design (how the individual programs are going to work), interface design (what are the interfaces going to look like) and data design (what data will be required).

During these phases, the software's overall structure is defined. Analysis and Design are very crucial in the whole development cycle. Any glitch in the design phase could be very expensive to solve in the later stage of the software development. The logical system of the product is developed in this phase.

1.3 Introduction to Project

The purpose of this document is to build an online system to take feedback regarding airport facilities which are provided at the airport. The project is based on an android app using which the airport authority can manage the database and passengers can access for their belonging

This project is a prototype for the Inventory management system for taking feedback at the airport and it is restricted within the college premises. This has been implemented under the guidance of college professors. This project is useful for the airport authority and as well as to the passengers.

The purpose of the Inventory management system for taking feedback at the airport is to create a convenient and easy-to-use application for passengers. The system is based on a relational database. Above all, we hope to provide a comfortable user experience.

This project is a prototype for the Inventory management system for taking feedback at the airport and it is restricted within the college premises. This has been implemented under the guidance of college professors. This project is useful for the airport authority and as well as to the passengers.

1.4 Proposed Solution

Proposed System

1. The proposed system is intranet based system so passengers can also participate in viewing their belongings.
2. The proposed provides detail general information about the passengers along with their personal/basic information and their arrival and destination places.
3. It enhances the HR management in adding, viewing and updating passenger's details and generates various reports regarding passenger's belongings.

At a conceptual level, there may be many different kinds of objects within a given site that are accessible to users. At the physical level, these objects may be represented by one or more Web pages. Conceptually, each of these entities represents a different type of semantic object. During a visit to this site, a user may access several of these objects together during a session. In contrast to content features, ontological representation of domain knowledge contained in the site makes it possible to have a uniform architecture to model such objects, their properties, and their relationships. In this section we will present a general framework for fully utilizing domain ontologies in Web usage mining and personalization. Figure 1.1 lays out a general process for such an integrated approach. As before, it is composed of 3 main phases: preprocessing, pattern discovery and online recommendation. Each of these phases must now take into account the object properties and their relationships.

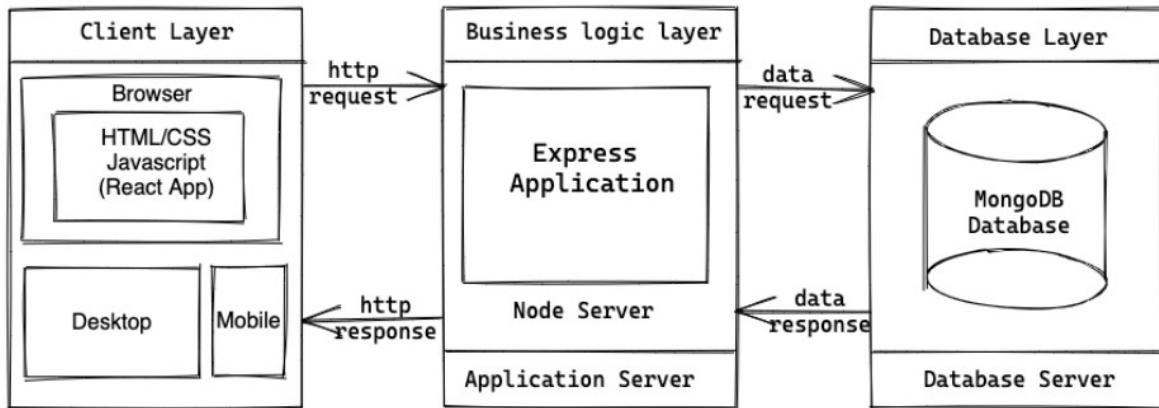


Figure 1.1: Architecture Diagram of the Proposed System Methodology

1.5 Scope of the Project

The purpose of the Inventory management system for taking feedback at the airport is to create a convenient and easy-to-use application for passengers. The system is based on a relational database. Above all, we hope to provide a comfortable user experience.

Chapter 2

Software Requirement Specification

2.1 Overall Description

This section and its subsections contain the description of the project components such as interfaces, performance requirements, design constraints, assumptions and dependencies etc.

2.1.1 Product Perspective

The application will be a web application.

2.1.1.1 System Interfaces

List each system interface and identify the functionality of the system (hardware and software both) to accomplish the system requirement and interface description to match the system.

2.1.1.2 User Interfaces

The application will have a user friendly and menu based interface. Following screens will be provided:

- An inventory management database system stores the following information.
- **User details:** The database consists of the username, password, contact, gender, email id.
- **Admin details:** The DB consists of the username, password, contact, gender, email id, address. These details provide valid login to an admin of a particular airport after getting registered.

2.1.1.3 Hardware Interfaces

- Screen resolution of at least 800 x 600 pixels is required for proper and complete viewing of screens. Higher resolutions in wide-screen mode will be better for a better view
- Support for printer is required. This implies that appropriate drivers should be installed and printer device should be connected for printing of reports and mark sheets.
- A network connection (internet / intranet) is required to make the web service accessible on other systems connected over the network
- Other hardware interface specifications are as follows

	Processor	RAM	Disk Space
Google Chrome	Intel Pentium III or AMD - 800 MHz	256 MB	2 GB

Table 2.1: Minimum Client Side Hardware Interface

	Processor	RAM	Disk Space
NodeJS	Intel Pentium III or AMD - 800 MHz	1 GB	3.5 GB
MongoDB	Intel Pentium III or AMD - 800 MHz	256 MB	512 MB

Table 2.2: Minimum Server Side Hardware Interface

2.1.1.4 Software Interfaces

- Any Microsoft Windows 7 and higher (Windows 7 / 8 / 8.1 / 10) or equivalent Linux based operating system with minimum kernel support 3.X.
- Crystal Reports 8 for generation and viewing of reports
- We have chosen web application for its best support and user-friendliness.

- To implement the project we have chosen VS Code for its more interactive support.

Software Tool	Version	Purpose of Use
Operating System	Windows 7 and higher or Linux with kernel 3.x and higher	Installation and operational platform
Web Browser	Google Chrome, Brave and other higher compatible	Access to the web application
Web Server	Node Server	Running the web application over intranet
Database	MongoDB	Running and linking the database over internet/intranet to the online web application

Table 2.3: Minimum Software Interface

2.1.1.5 Communications Interfaces

- Passenger on Internet will be using HTTP/HTTPS protocol.
- Client (Administrator) on Internet will be using HTTP/HTTPS protocol.

2.1.1.6 Memory Constraints

- At least 256 MB of RAM and 2 GB of space on hard disk will be required for running the application on client end.
- Similarly, a minimum of 4 GB of RAM and 4 GB of space on hard disk will be required for running the application on server end.

2.1.1.7 Operations

- The DBA at the client side will be assumed responsible for manually deleting or archiving obsolete or non-required data from the database as per clients requirements.
- This will include database backup and recovery options also.
- The Node webserver will be hosted and maintained on a remote server addressed by a URL based address.

- The URL address may be intranet or internet based as per clients requirements.
- The 'SYSTEM RESET' function is provided that after confirmation from the administrator, will delete all the selective or complete data from the system.

2.1.1.8 Site Adaption Requirements

The computing terminals connected to network (internet / intranet) at the client end will be required to support the hardware and software interfaces specified in above sections.

2.1.2 Project Functions

The system will allow access only to authorized users with specific roles (Passengers, Administrator etc.). Depending upon the user's role, he / she will be able to access only specific modules of the system.

A summary of the major functions that the software will perform:

1. A Login facility for enabling only authorized access to the system.
2. Passenger can view their flight details and can give feedbacks.
3. Admin can view feedbacks summary and feedback messages.

2.1.3 User Characteristics

- Educational Level: User should be at least graduate and comfortable with English.
- Experience: User should be well versed / informed about the structure of the program.
- Technical Expertise: User should be comfortable using general purpose applications on a computer.

2.1.4 Constraints

- Since the DBMS being used in this project is MongoDB, and the web server is Node Server, that are free open source tools, the server technologies are out of any guarantees, unless specifically purchased for enterprise environment.
- Due to limited features of DBMS being used, performance tuning features will not be applied to the queries and thus the system may become slow with the increase of data records being stored.
- Due to limited features of DBMS, database auditing will also not be provided.
- As the application runs over a network environment, well documented security policy is required to prevent and safeguard data as well as services over the network.

2.1.5 Assumption and Dependencies

- The number of domains being selected by the user does not change.
- The project Code will not change.
- The number of modules assigned to employee cannot be changed.

2.2 Specific Requirements

This section presents the software requirements to a level of detail sufficiency to enable designers to system.

A Software requirements specification (SRS) is a document that captures complete description about how the system is expected to perform. It is usually signed off at the end of requirements engineering phase.

2.2.1 User Interfaces Requirements

Following screens will be provided by the system:

Login Screen: This will be the first screen to be displayed. The first login screen will be for the passenger and it will contain the link for the admin login screen. The passenger will have to enter their name, email, mobile number and PNR (which has to be correct) and admin will have to enter his email and password

Passenger Screen: This screen will be displayed after the passenger enter the correct details and this screen will display his flight details such as aircraft name, airline name, time and date of departure and arrival, etc

Feedback Page: This page displays the eight departments of the airport for which passenger can provide feedback. This page is same for passenger and admin. When passenger clicks on it, it will display the feedback form but when admin will click on it, it will display feedback summary report

Feedback Form: This screen shows the feedback form of the respective form for which passenger can provide rating between 0.5 and 5 based on different parameters.

View Feedback Page: This page show the feedback summary of the respective department in the form of circular progress bar, it also shows the feedback messages given by the passengers.

2.2.2 System Product Features

2.2.2.1 Security

The application will be password protected. Users will have to enter correct user-name, password and role in order to access the application modules allowed to their privilege.

2.2.2.2 Maintainability

The application will be designed in a manner to make it easy to incorporate new requirements in individual modules such as employee info, manager info, task assignment, task completion, report generation and user accounts activation deactivation.

Inevitably the system will need maintenance. Software will definitely undergo change once it is delivered to the customer. There are many reasons for the change. Change could happen because of some unexpected input values into the system.

In addition, the changes in the system could directly affect the software operations. The software should be developed to accommodate changes that could happen during the post implementation period.

2.2.3 Portability

The application will be easily portable among any windows or Linux based systems that have Nodejs and MongoDB database installed.

Chapter 3

System Design Specification

3.1 System Architecture

System architecture presents the schematic view of the complete system along with its major components and their connectivity. The overall architecture of the proposed system will be as follows.

3.2 Module Decomposition Description

The proposed system can be decomposed into following major modules:

1. Passenger
2. Feedback Forms
3. Admin
4. Feedback Management

Module - 1:

Passenger: This is the other user page, i.e. the passenger page after login.

This is the passenger dashboard, passenger can use it to navigate to different section of the website and they can also view their flight details like airline name, Aircraft name, flight departure date and time, flight arrival date time, etc. on this page.

Module - 2:

Feedback Forms: This page shows 8 different departments of the airport for which a passenger can give their feedback.

The 8 different departments are:

- Food Court

- Baggage
- Check In
- Help Desk
- Airlines
- Lounges
- Stores
- Washroom

Feedback Form Page:

After selecting the department in feedback page the passenger is redirect to the respective feedback form of that department and the passenger here can give their feedbacks based on different services or parameters between 0.5 and 5. They can also give the feedback message for the respective department.

Module - 3:

Admin:

This page is the first page after login a manager will see.

Feedback Page:

This page is the same as the one in passenger module. It also has the same 8 departments but when the admin clicks on it, he will be able to view the results to feedbacks given by the passengers.

Module - 4:

Feedback Management:

This displays the feedbacks summary of the respective department in the form of circular progress bar. This also displays the entire feedback message given by the passengers. Mail is sent to user after successfull feedback submittion

The overall objective of Feedback Management is the development of database technology has been to treat data as an organizational resource and as an integrated

whole. DBMS allow data to be protected and organized separately from other resources.

The organization of data in database aims to achieve three major objectives:

- Data integration.
- Data integrity.
- Data independence.

3.3 High Level Design Diagrams

3.3.1 Use Case Diagram

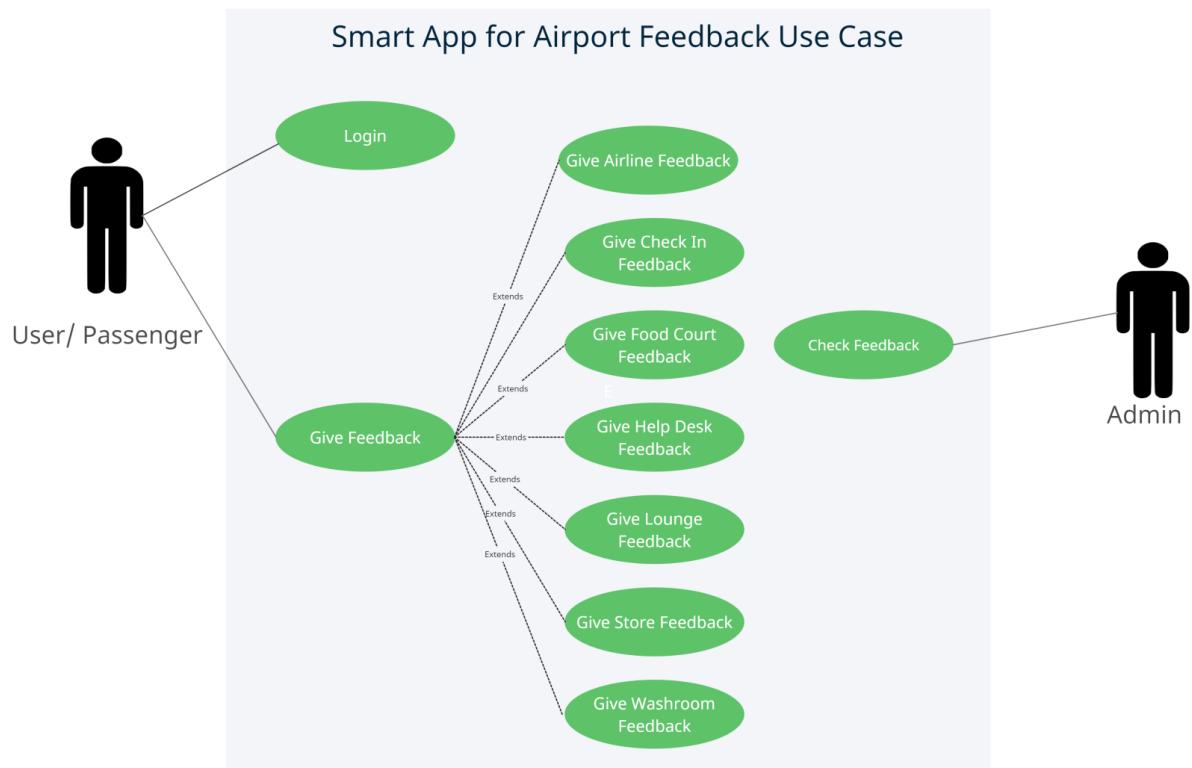


Figure 3.1: Use Case diagram

3.3.2 Activity Diagram

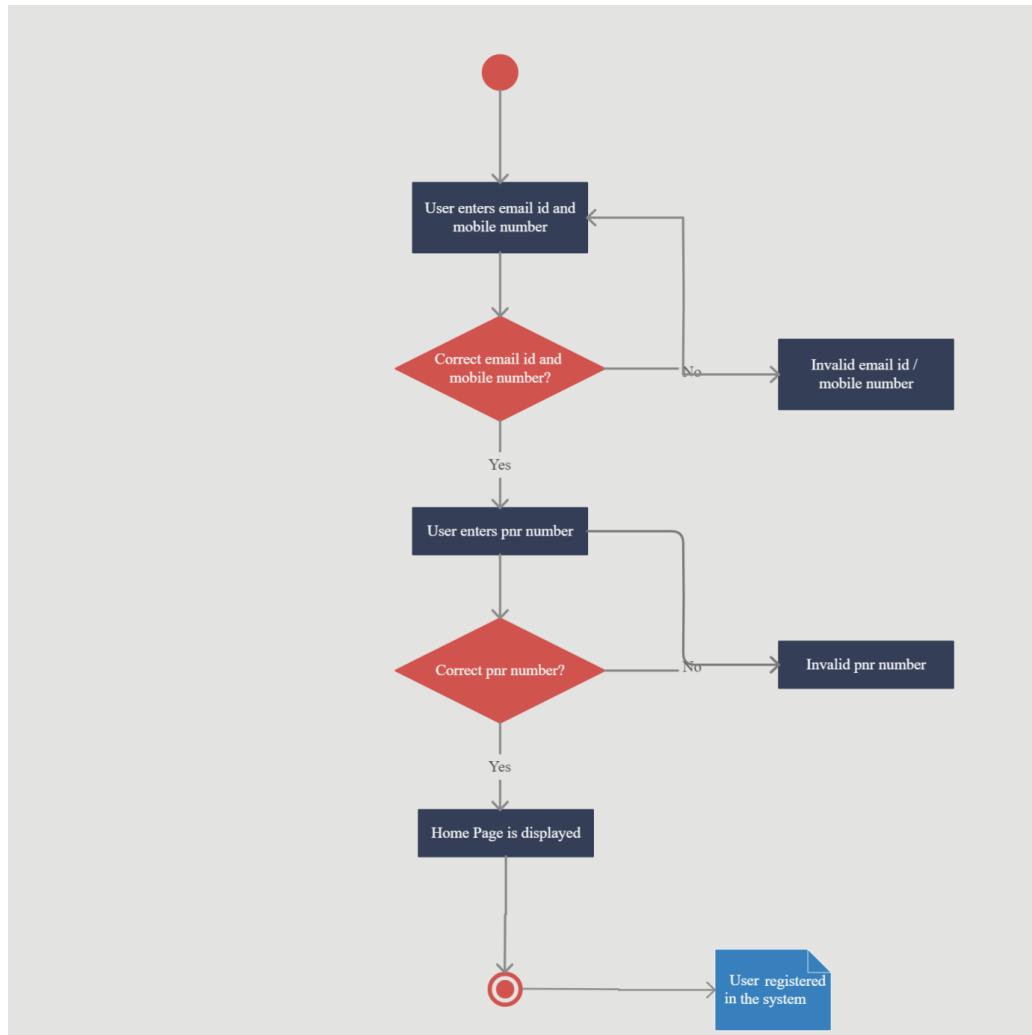


Figure 3.2: Activity Diagram Representing Passenger Registration

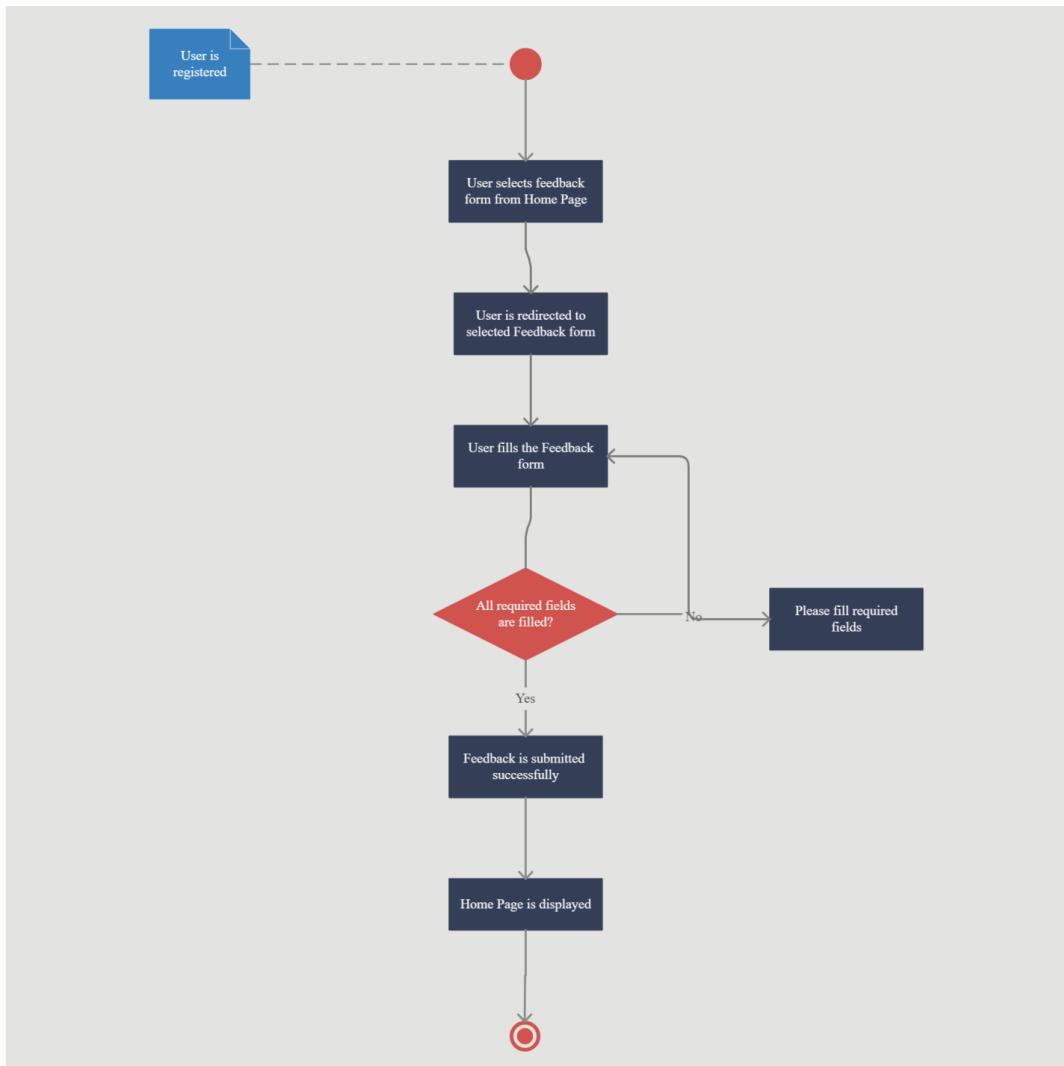


Figure 3.3: Activity Diagram Representing Passenger Feedback

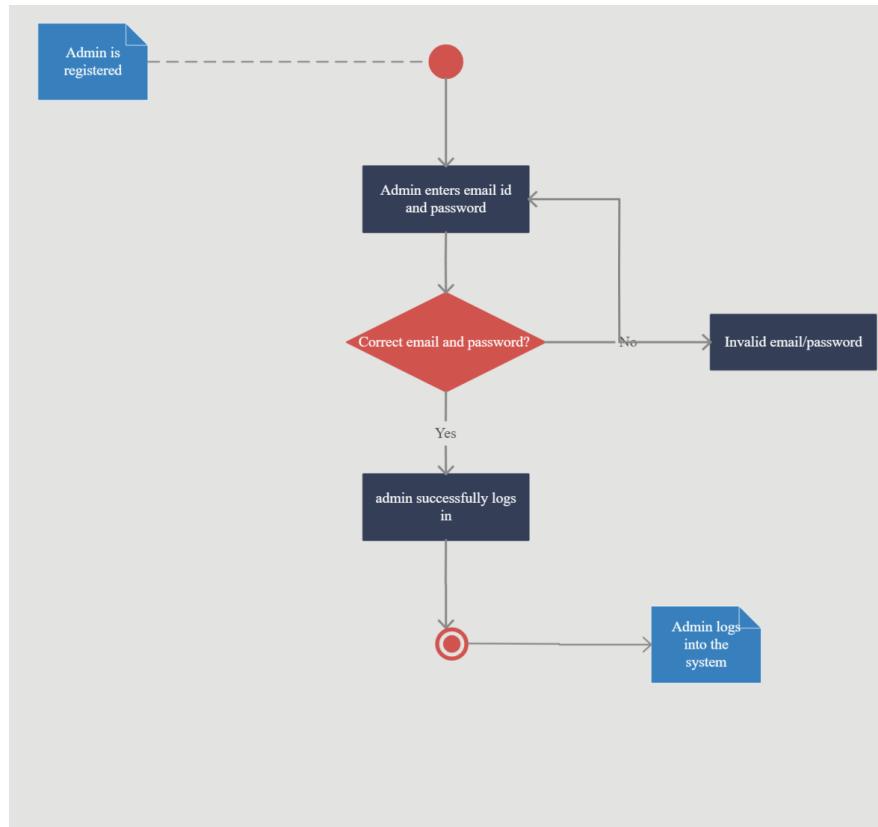


Figure 3.4: Activity Diagram Representing Admin Login

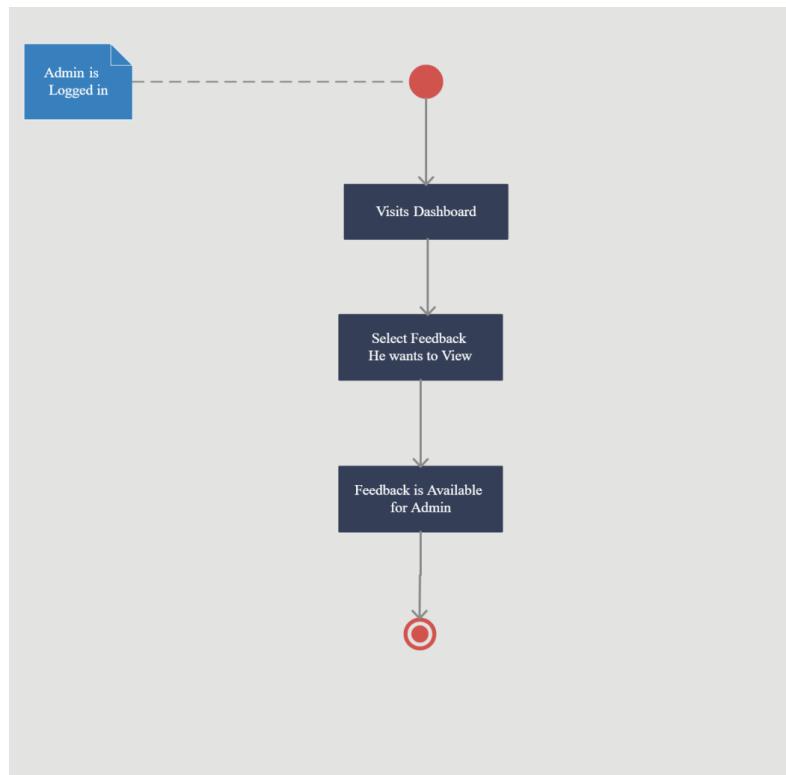


Figure 3.5: Activity Diagram Representing Admin View Feedback

3.3.3 Data-Flow Diagram

LEVEL 0 DATA FLOW DIAGRAM

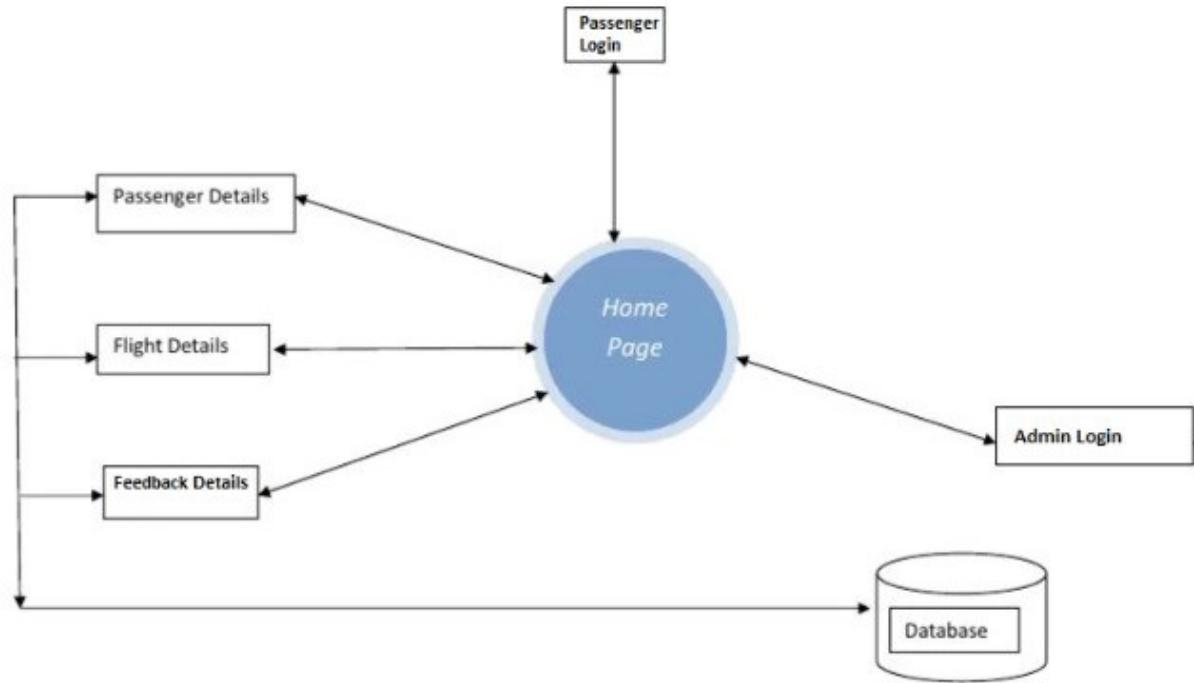


Figure 3.6: Data-Flow Diagram

3.3.4 Sequence Diagram

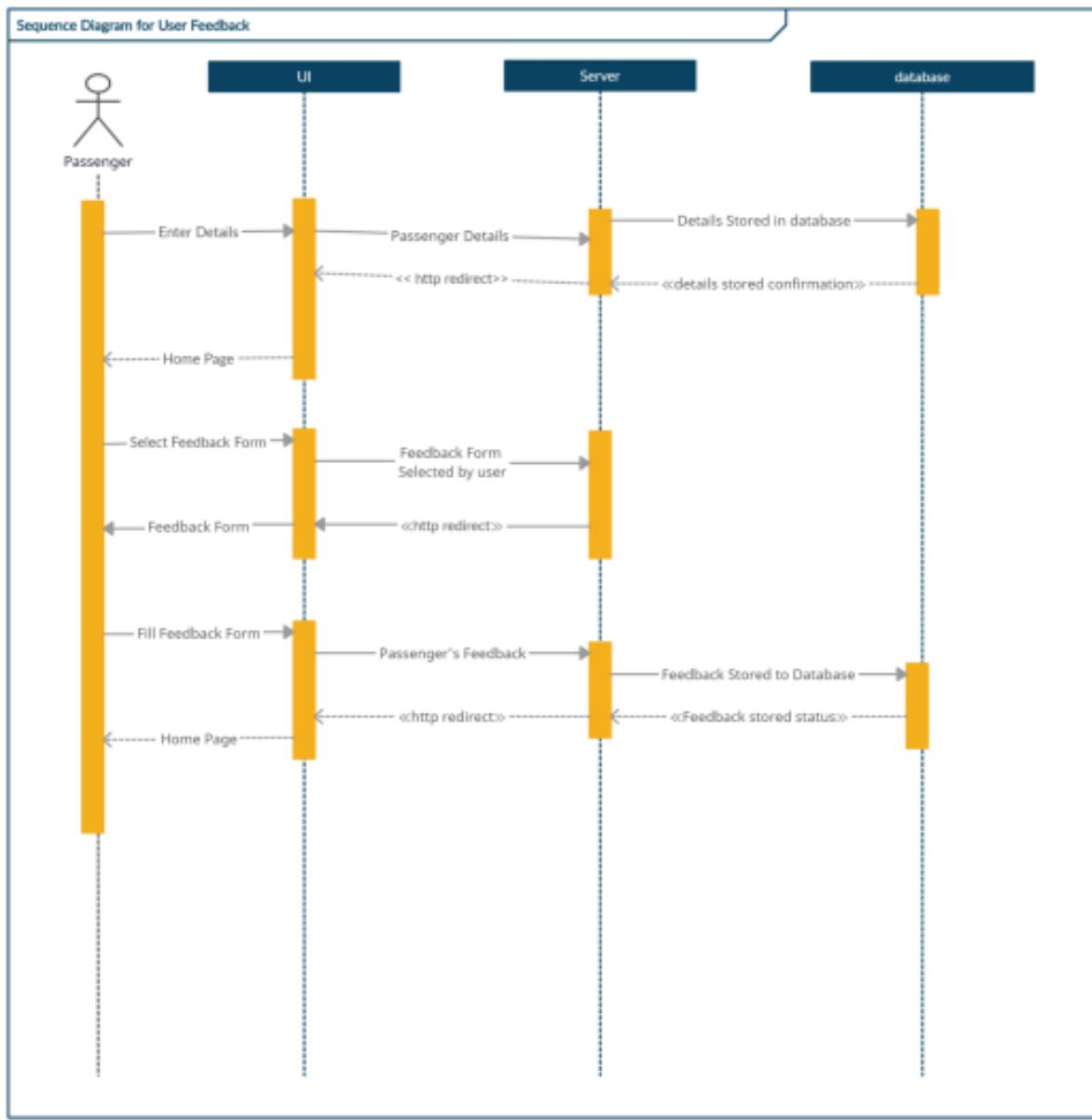


Figure 3.7: Sequence Diagram Representing Passenger

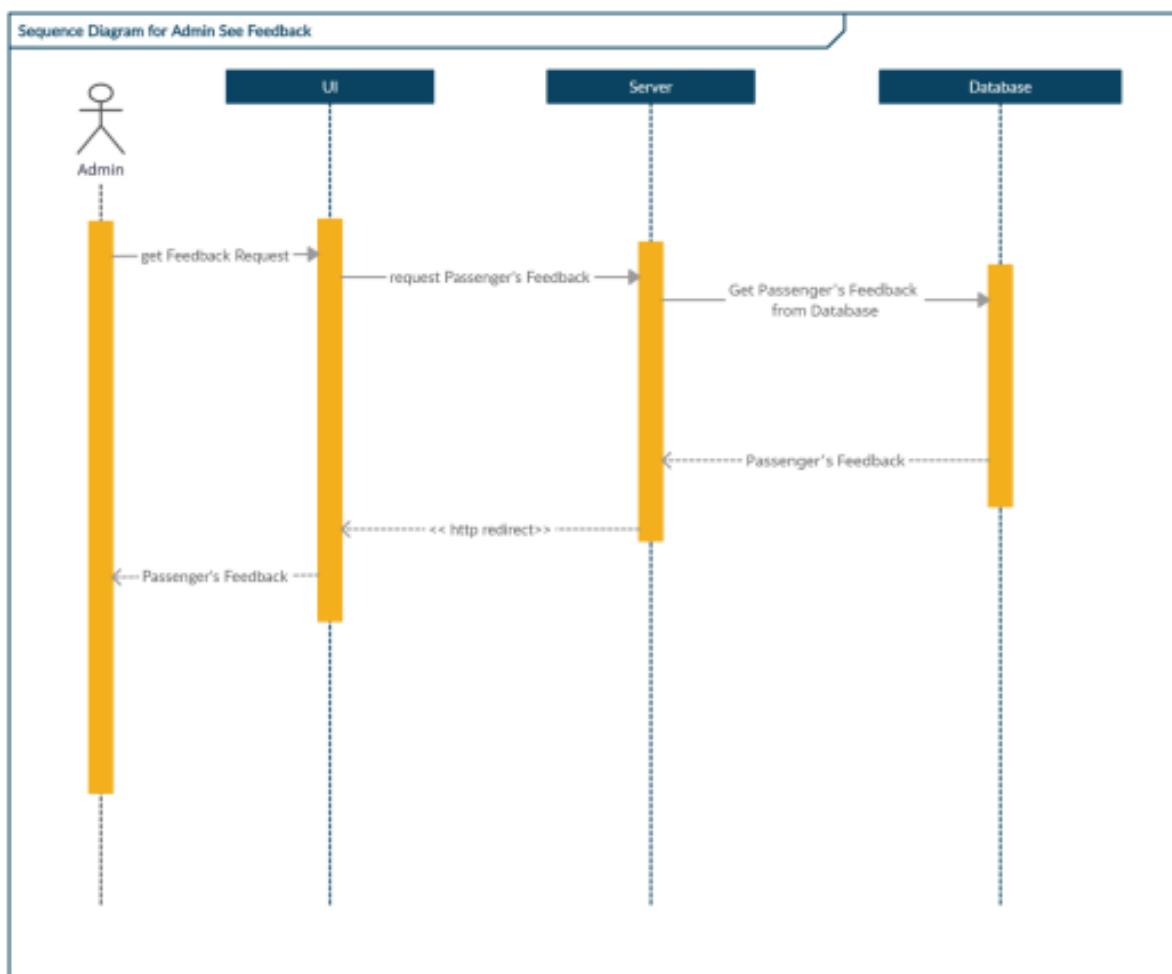


Figure 3.8: Sequence Diagram Representing Admin View Feedback

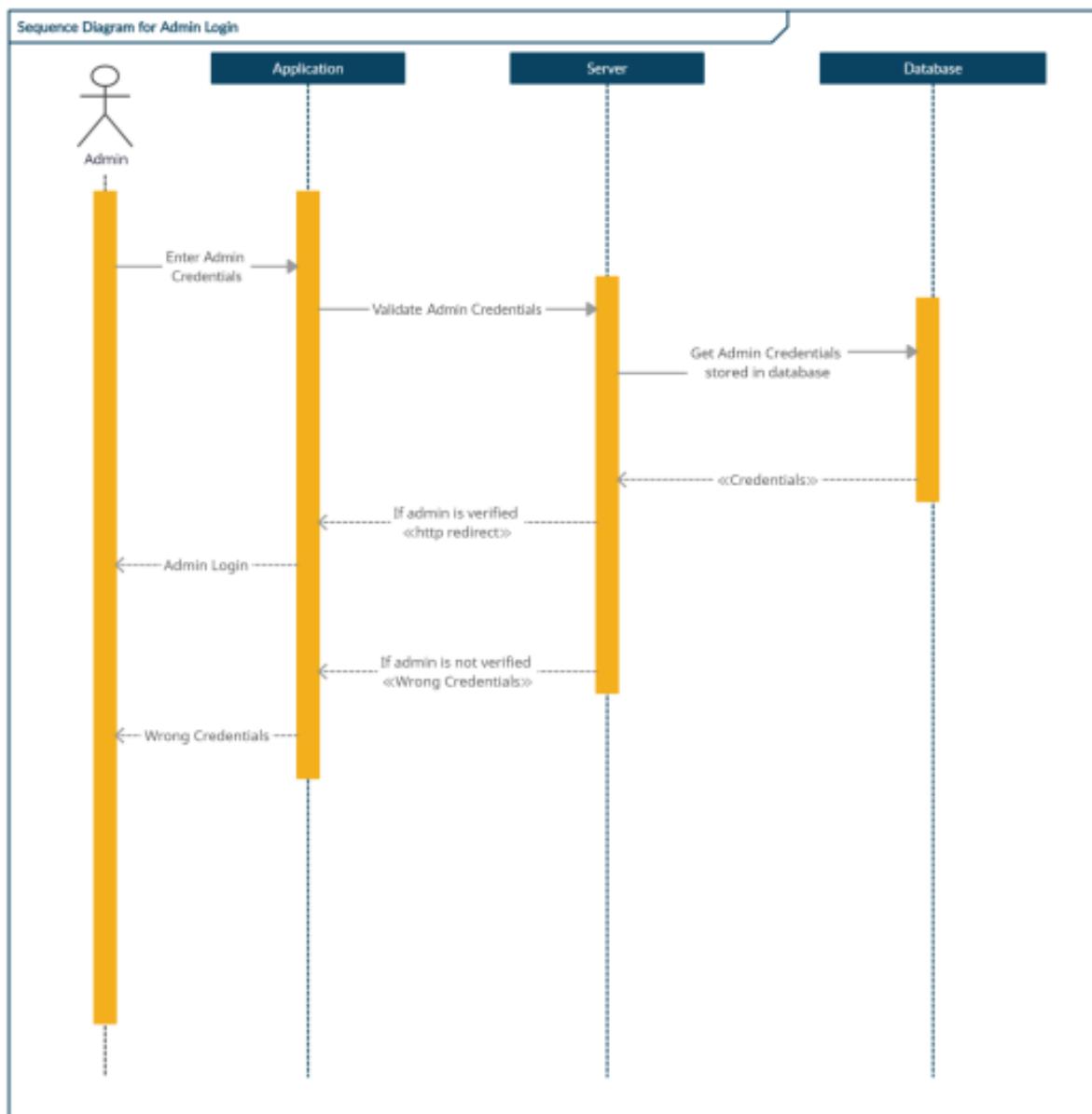


Figure 3.9: Sequence Diagram Representing Admin Login

3.3.5 Class Diagram

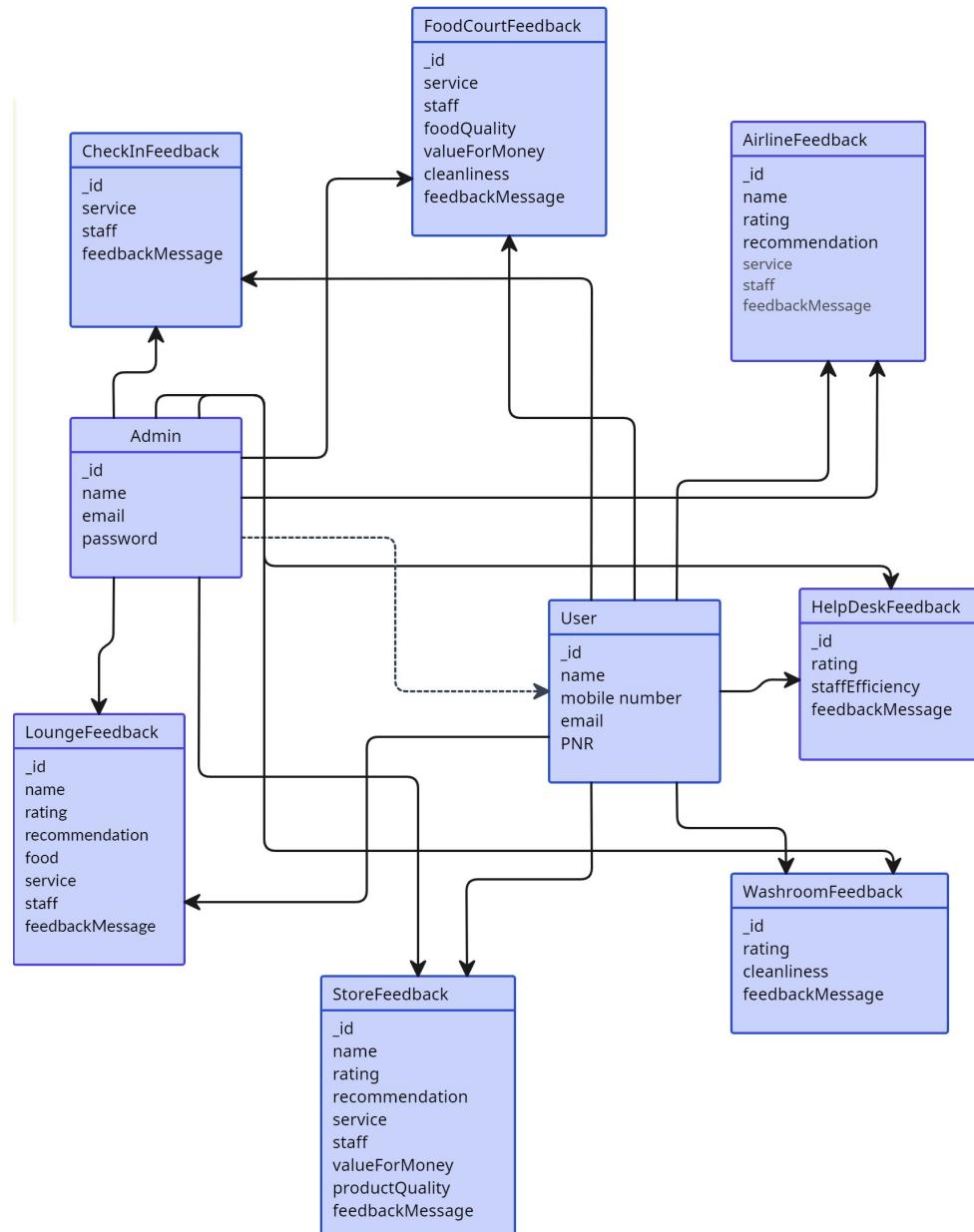


Figure 3.10: Class Diagram

3.3.6 ER Diagram

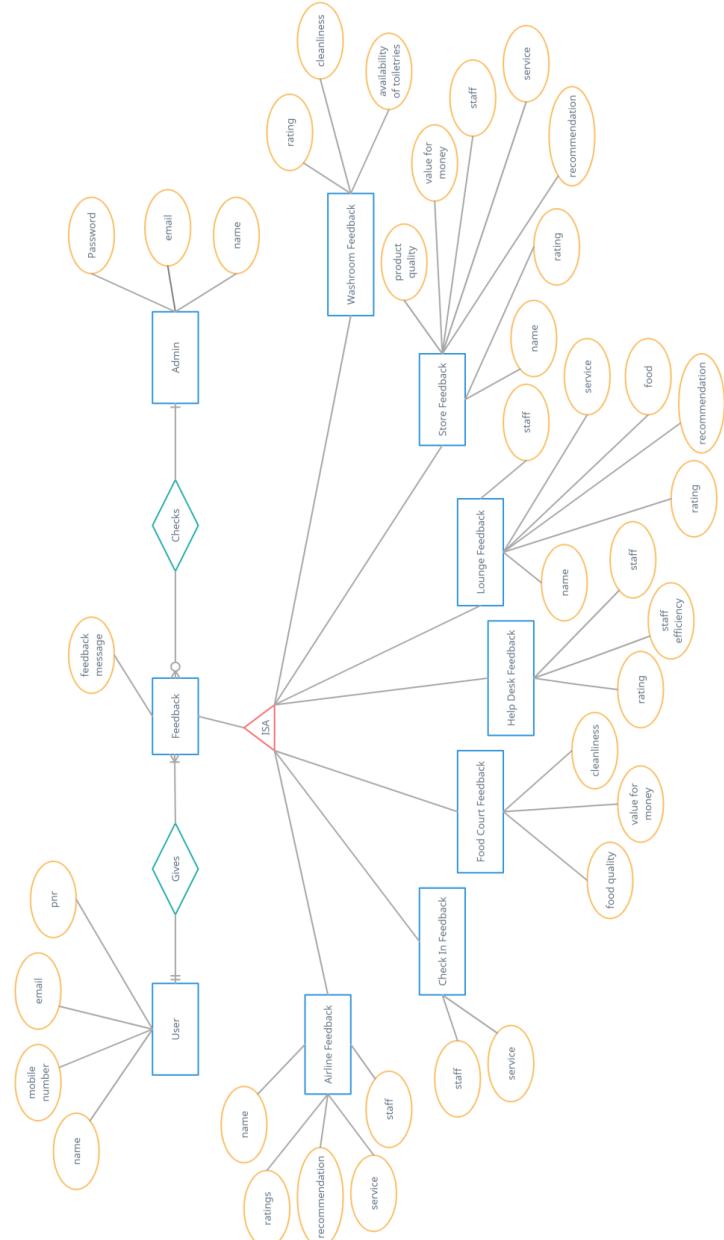


Figure 3.11: ER Diagram

Chapter 4

Methodology and Team

4.1 Introduction to Waterfall Framework

The Waterfall Model was first Process Model to be introduced. It is also referred to as a linear-sequential life cycle model. It is very simple to understand and use. In a waterfall model, each phase must be completed before the next phase can begin and there is no overlapping in the phases.

The Waterfall model is the earliest SDLC approach that was used for software development.

The waterfall model is a breakdown of project activities into linear sequential phases, meaning they are passed down onto each other, where each phase depends on the deliverables of the previous one and corresponds to a specialization of tasks. The approach is typical for certain areas of engineering design.

The Waterfall Model illustrates the software development process in a linear sequential flow; hence it is also referred to as a linear-sequential life cycle model. This means that any phase in the development process begins only if the previous phase is complete. In waterfall model phases do not overlap. In "The Waterfall" approach, the whole process of software development is divided into separate phases. In Waterfall model, typically, the outcome of one phase acts as an input for the next phase sequentially. Following is a diagrammatic representation of different phases of waterfall model.

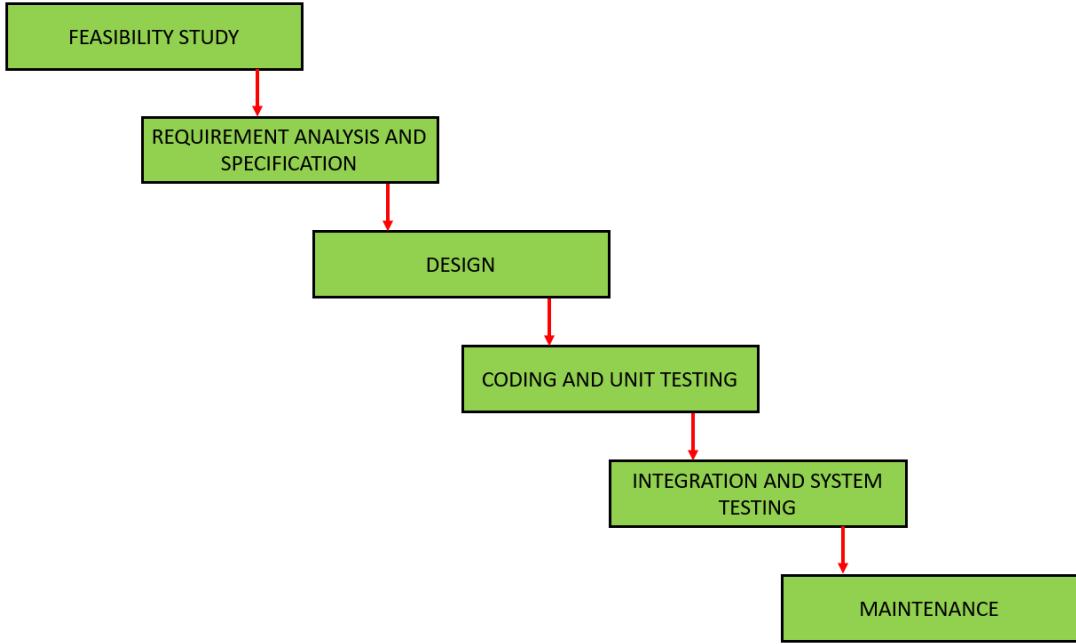


Figure 4.1: WaterFall Model

The sequential phases in Waterfall model are-

1. **Requirement Gathering and analysis:** All possible requirements of the system to be developed are captured in this phase and documented in a requirement specification doc.
Like in this phase we work on SRS (Software Requirement Specification), we gather project requirement like what is the project, what we need to do in the project, what feature we need to do this in project, how many module are there in project etc.
2. **System Design:** The requirement specifications from first phase are studied in this phase and system design is prepared. System Design helps in specifying hardware and system requirements and also helps in defining overall system architecture.
In this we created blue print for coding part like we select what programming language we require for project, which hardware and software are best for this project.
3. **Implementation:** With inputs from system design, the system is first devel-

oped in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality which is referred to as Unit Testing.

In this phase coding is start, we create small units like login page, different feedback forms and check that whether they are working fine or not.

4. Integration and Testing: All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures. In this phase we combine all the units and check that whether they are working as expected or not, like we integrate Passenger and feedback form to see whether passenger after login is able to submit the feedback

5. Deployment of system: All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.

In this phase all our project modules like Passengers, Feedback forms , Admin and Feedback Management are integrated and tested as whole, after successfull testing we deploy the project for use.

6. Maintenance: All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.

Like after deployment if we got some problem we will work on that problem in this phase and if we want to add extra feature we will add that feature in this phase also.

All these phases are cascaded to each other in which progress is seen as flowing steadily downwards (like a waterfall) through the phases. The next phase is started only after the defined set of goals are achieved for previous phase and it is signed off, so the name "Waterfall Model". In this model phases do not overlap.

Waterfall Model Pros & Cons

Advantage : The advantage of waterfall development is that it allows for departmentalization and control. A schedule can be set with deadlines for each stage of development and a product can proceed through the development process model phases one by one. Development moves from concept, through design, implementation, testing, installation, troubleshooting, and ends up at operation and maintenance. Each phase of development proceeds in strict order.

Disadvantage : The disadvantage of waterfall development is that it does not allow for much reflection or revision. Once an application is in the testing stage, it is very difficult to go back and change something that was not well-documented or thought upon in the concept stage.

4.2 Team Members, Roles & Responsibilities

S.No	Team Members	Responsibilities
1	Shashank Varshney	Designing Database, Admin Module, Integration of all Modules, Authentication, Feedback Management Module
2	Shivam Thakkar	Feedback Form, Passenger Module
3	Sujal Somanı	Feedback Form, Passenger Module, Documentation
4	Yuvraj Gakkhar	Feedback Management Module

Table 4.1: Roles and Responsibilities

Chapter 5

Centering System Testing

The designed system has been testing through following test parameters.

5.1 Functionality Testing

In testing the functionality of the web sites the following features were tested:

1. Links

- (a) Internal Links: All internal links of the website were checked by clicking each link individually and providing the appropriate input to reach the other links within.
- (b) External Links: Till now no external links are provided on our website but for future enhancement we will provide the links to the candidate's actual profile available online and link up with the elections updates online etc.
- (c) Broken Links : Broken links are those links which do not divert the page to specific page or any page at all. By testing the links on our website, there was no link found on clicking which we did not find any page.

2. Forms

- (a) Error message for wrong input : Error messages have been displayed as and when we enter the wrong details (eg. Dates), and when we do not enter any details in the mandatory fields. For example: when we enter wrong password we get error message for acknowledging us that we have entered it wrong and when we do not enter the username and/or password we get the messages displaying the respective errors.
- (b) Optional and Mandatory fields : All the mandatory fields have been marked with a red asterisk (*) and apart from that there is a display of error messages when we do not enter the mandatory fields. For example: As the first

name is a compulsory field in all our forms so when we do not enter that in our form and submit the form we get an error message asking for us to enter details in that particular field.

3. Database

Testing is done on the database connectivity.

In the database testing we included following cases:

- (a) Entries in database through frontend and checked for the same in the tables.
- (b) Checked for the data types.
- (c) Checked for range of each type of data.
- (d) If entries made in one table are affecting other tables then we have checked those entries also.
- (e) We not just added details in the database by default but also did the same using the frontend.

5.2 Performance Testing

Performance testing can be applied to understand the website's scalability, or to benchmark the performance in the environment of third party products such as servers and middleware for potential purchase. This can only be done once it is put into use on the actual internet server and tested by the users.

The system load includes:

1. What is the number of users per time?
2. Checking for peak loads and how system behaves.
3. Amount of data accessed by user.

This is done using only 2 systems for now so cannot be tested for load unless we deploy it on a real server machine.

5.3 Usability Testing

Usability testing is the process by which the human-computer interaction characteristics of a system are measured, and weaknesses are identified for correction.

1. Ease of learning
2. Navigation
3. Subjective user satisfaction
4. General appearance

As system is not put into the real time use so it's not yet tested for usability.

Chapter 6

Test Execution Summary

Execution Test Summary Report is an overall view of Testing Process from start to end. Test Plan comes at the starting of project while Test Summary Report comes at the end of the testing process. This report is given to the client for his understanding purpose. The Test Summary Report contents are :

1. Test Case ID generated = PRO1, PR02
2. Total number of resources consumed = 2
3. Passed Test Cases = 2
4. Failed Test Cases = 0
5. Status of Test Cases = Passed

S.No	Test Case Id	Test Case Description	Expected Outcome	Test Case Status	No. of Resources Consumed
1	PRO1	Login user id accepts only authorized users.	Accepts 10 alphanumeric keyword only	PASS	Monitor, Keyboard
2	PR02	Once logged out of the system user has to login again.	Logged Out	PASS	Monitor, Keyboard

Table 6.1: Test Case Summary

Chapter 7

Project Screen Shots

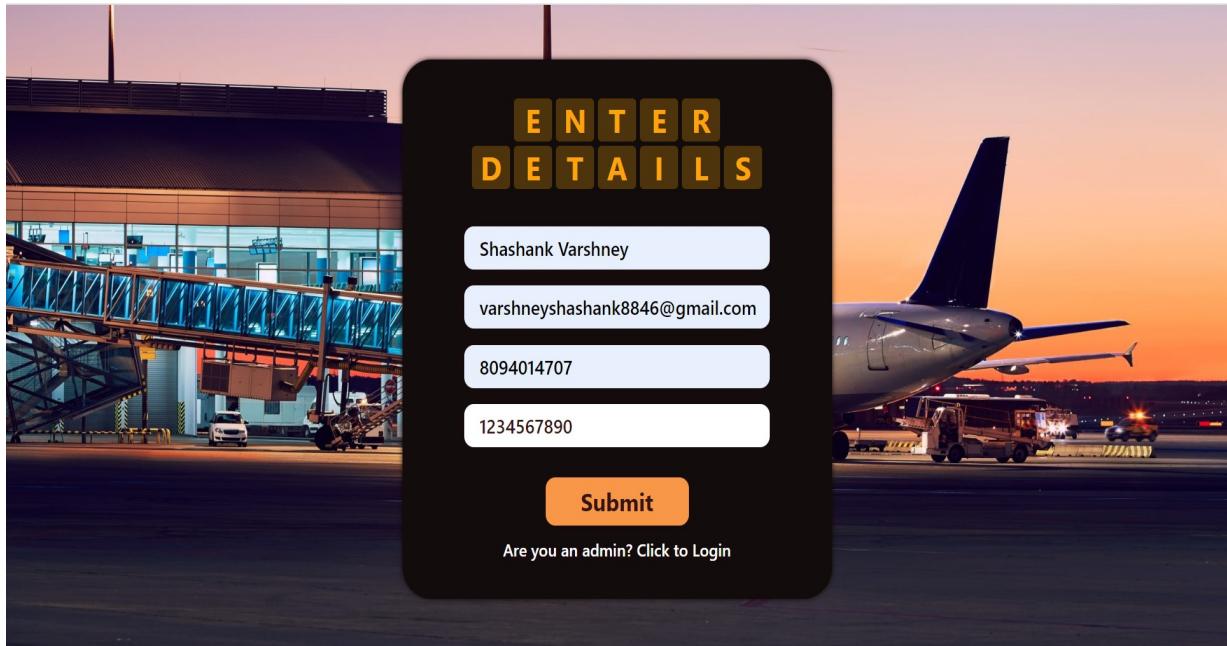


Figure 7.1: Passenger Login Screen



Figure 7.2: Passenger Home Displaying Flight Details

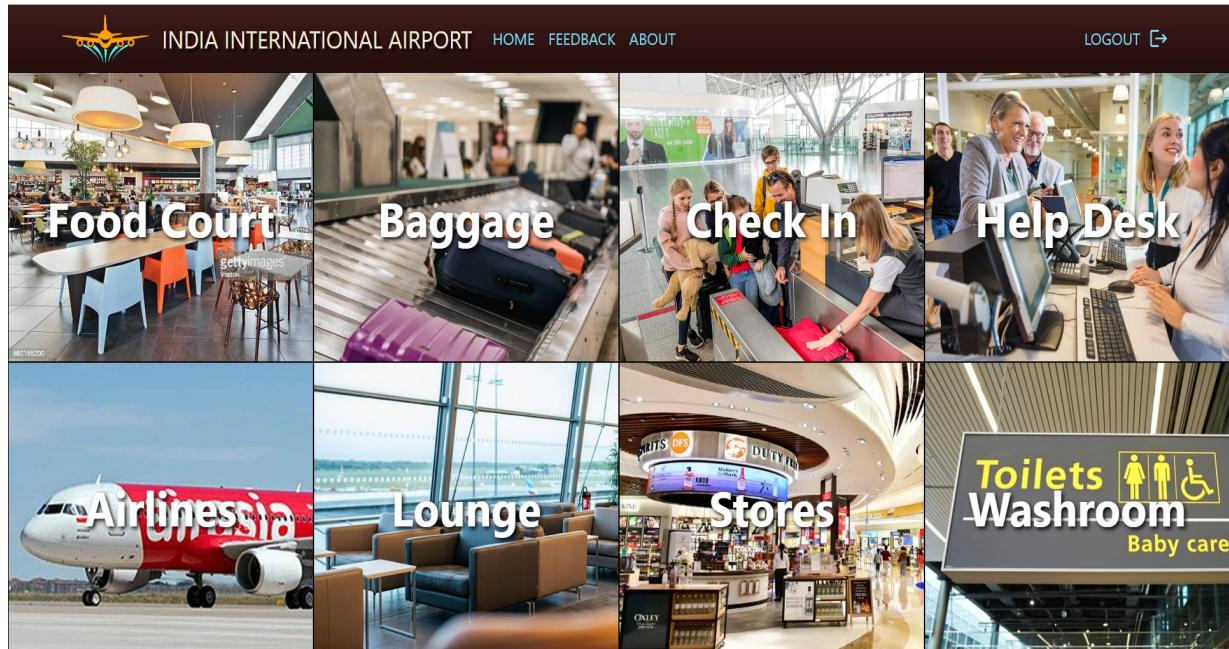


Figure 7.3: Feedback Page (Common For both Passenger and Admin)

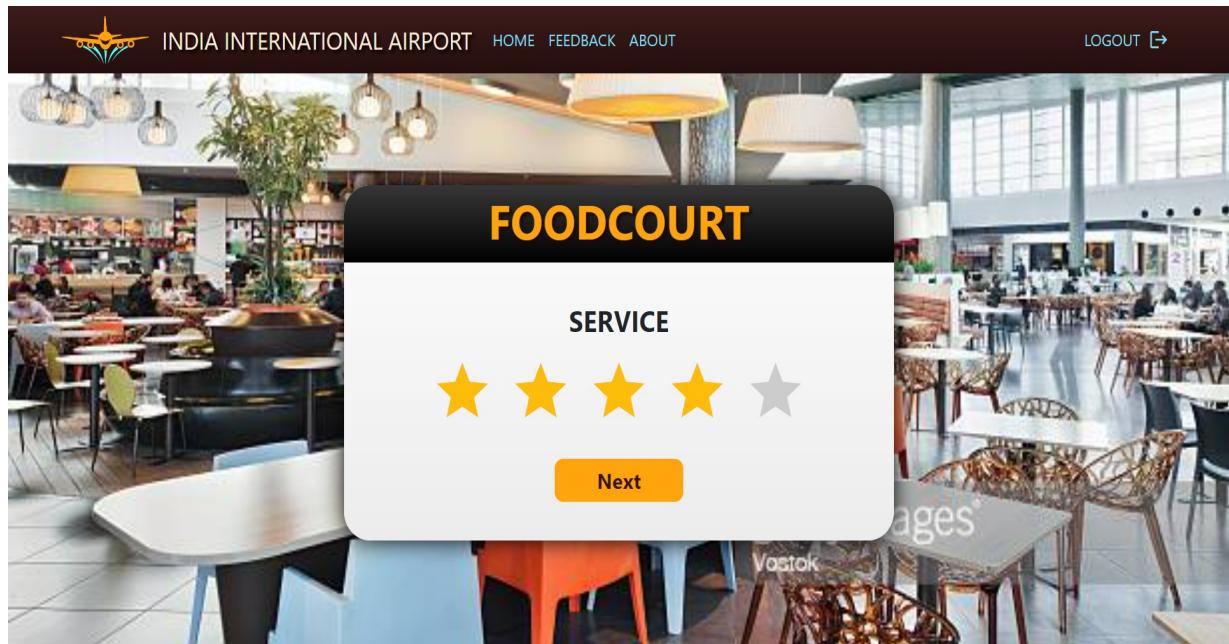


Figure 7.4: Passenger Rating Form

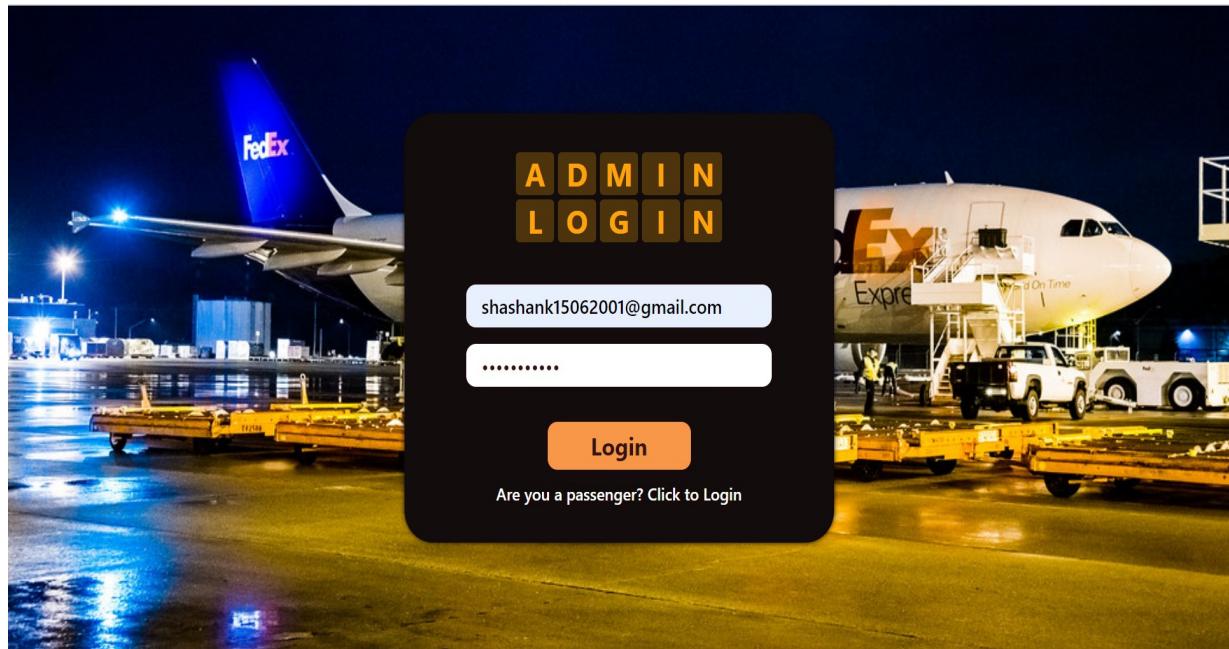


Figure 7.5: Admin Login Screen

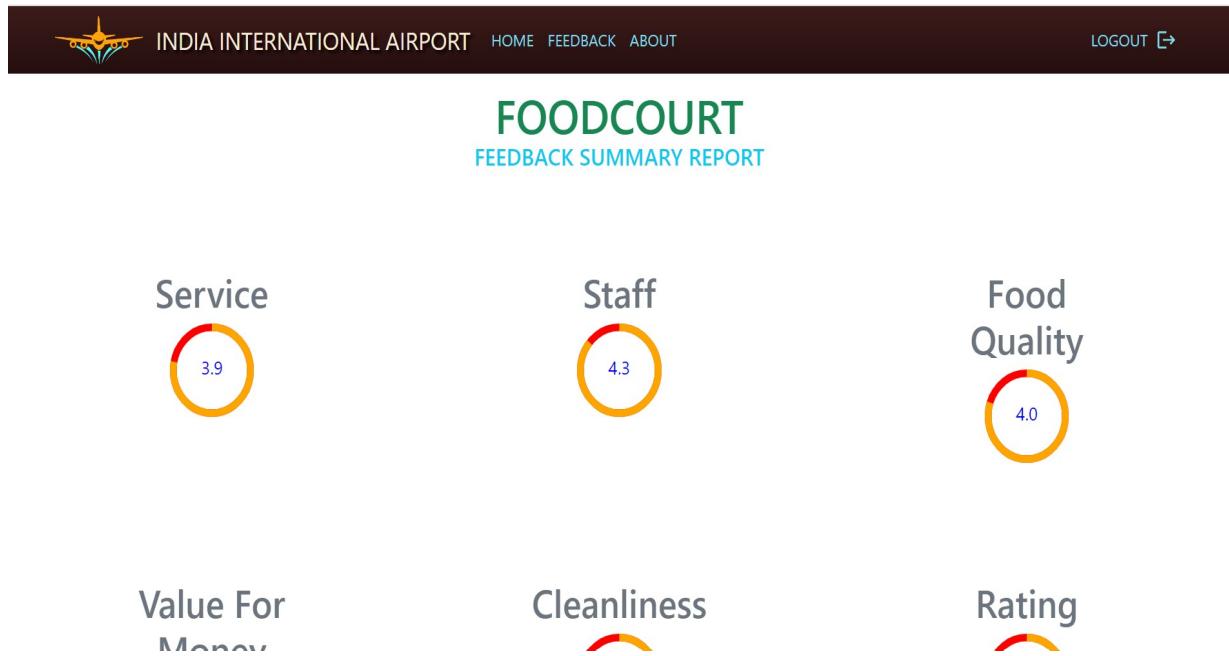


Figure 7.6: FoodCourt View Feedback Page

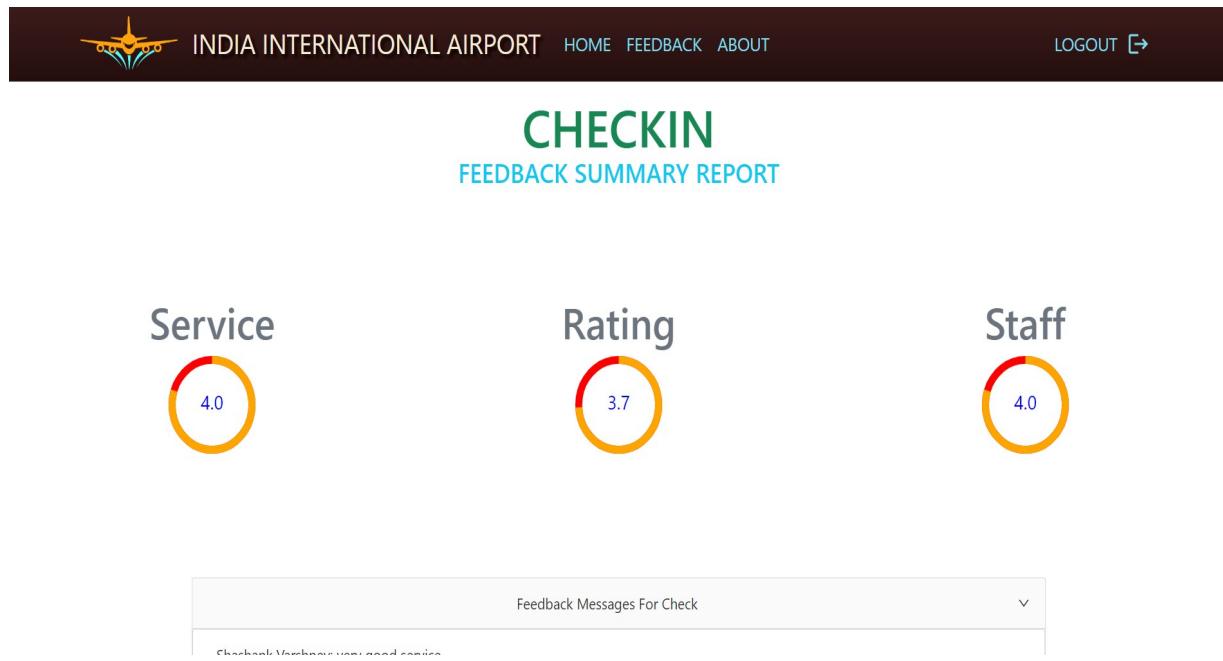


Figure 7.7: ChechIn FeedBack Page

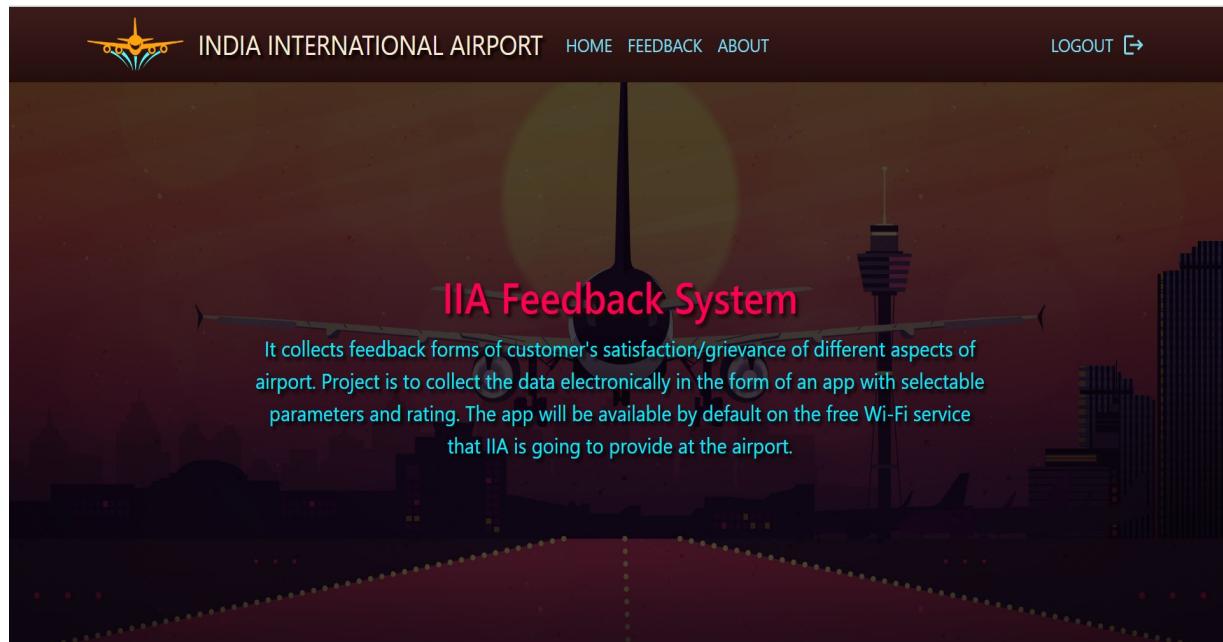


Figure 7.8: About Page

The screenshot shows the MongoDB Atlas Data Services interface. On the left, a sidebar lists various services and security options. The main area displays the 'test.admins' collection. It shows a storage size of 20KB, logical data size of 105B, and 1 total document. The document details are:

```

_id: ObjectId("640c74f2c6ef7084a877ac5e")
name: "Shashank"
email: "shashank15062001@gmail.com"
password: "Shashank123"

```

Figure 7.9: Admin Collection

The screenshot shows the MongoDB Atlas Data Services interface. On the left, a sidebar lists various services and security options. The main area displays the 'test.airlinefeedbacks' collection. It shows a storage size of 36KB, logical data size of 968B, and 6 total documents. The document details are:

```

_id: ObjectId("6470183adf8530a6ea1cebf7")
feedbackBy: ObjectId("6470181bd8530a6ea1cebd5")
name: "Air Arabia"
rating: 4
recommendation: 4
service: 4
staff: 4
feedbackMessage: "NA"
__v: 0

```

Figure 7.10: Airline Feedback Collection

The screenshot shows the MongoDB Atlas Data Services interface. On the left, the sidebar includes sections for Deployment, Database (with Data Lake PREVIEW selected), Services (Triggers, Data API, Data Federation, Search), Security (Backup, Database Access, Network Access, Advanced), and a central panel for the 'checkinfeedbacks' collection. The main area displays the collection's details: Storage Size: 36KB, Logical Data Size: 1.29KB, Total Documents: 11, and Indexes Total Size: 36KB. It features tabs for Find, Indexes, Schema Anti-Patterns, Aggregation, and Search Indexes. A search bar at the top right allows users to type a query. Below the search bar, the results are displayed with a count of 1-11 OF 11. One document is shown in expanded form:

```

_id: ObjectId('645a7bc7ade63d4bed53b4de')
feedbackBy: ObjectId('645a7ba6ade63d4bed53b4cc')
service: 4
rating: 4
staff: 4
feedbackMessage: "NA"
__v: 0

```

Figure 7.11: CheckIn Feedback Collection

This screenshot shows the MongoDB Atlas Data Services interface with the 'users' collection selected. The left sidebar remains the same as in Figure 7.11. The main area displays the collection's details: Storage Size: 36KB, Logical Data Size: 1.29KB, Total Documents: MANY, and Indexes Total Size: 36KB. It features tabs for Find, Indexes, Schema Anti-Patterns, Aggregation, and Search Indexes. A search bar at the top right allows users to type a query. Below the search bar, the results are displayed with a count of 1-20 OF MANY. One document is shown in expanded form:

```

_id: ObjectId('64463c518ff3b991459f587f')
name: "mike"
email: "pankaj@gmail.com"
mobileNumber: 8094014701

```

Navigation buttons for 'PREVIOUS' and 'NEXT' are visible at the bottom of the results pane.

Figure 7.12: User Collection

Chapter 8

Project Summary and Conclusions

8.1 Conclusion

”In the IT industry you are in the business of your employee’s Skills and Knowledge. This is the most important selling factor for grabbing a new business as well as for sustaining and growing the existing businesses,” competition is further forcing organizations to be 100 percent aware of their employee’s skills and abilities. With companies becoming aware of their employee’s skills, more than ever before the skills tracking exercise have achieved serious proportions, almost at par with other regular internal employee programmers.

While some progressive organizations had started the skills tracking exercise some time back, others are now following suit. ”These initiatives are becoming more integral to forward looking it companies, who have the tremendous need to tap into the employee talent reservoir to provide value add to their business operations and customers and, in turn, help employees to explore their potentialities.

The Airport Feedback App has successfully provided a platform for passengers to share their feedback and contribute to improving the overall airport experience. The app’s user-friendly interface, comprehensive feedback categories, and anonymity options have encouraged users to provide valuable insights. The feedback data collected through the app has helped airport authorities make data-driven decisions and implement necessary improvements. The app will continue to evolve based on user feedback and technological advancements to ensure a seamless and satisfactory experience for airport passengers.

Limitations

- This application can not be used outside the airport.
- Machine Learning is not included.

Chapter 9

Future Scope

The possible future scope of this application will be on mobile platform with following enhancements:

- Integrating leave management and attendance system.
- Apply machine learning to generate meaningful trends and results.
- Integrating complaint system.

References

- Object Oriented Modeling and Design with UML-Michael Blaha, James Rumbaugh.
- Software Engineering, Seventh Edition, Ian Sommerville.
- Wikipedia - www.wikipedia.com
- Node.js- www.w3schools.com
- Express.js -www.tutorialspoint.com
- React.js - www.reactjs.org