AIRPORT FEEDBACK APP

A

Project Report

submitted

in partial fulfillment

for the award of the Degree of

Bachelor of Technology

in Department of Computer Science and Engineering

SUBMITTED TO:
Dr. Nilam Choudhary
Dept. of Computer Science & Engineering

SUBMITTED BY:
Akshat Gadodia (19ESKCS021)
Akshita Sharma (19ESKCS027)

Department Of Computer Science & Engineering

Swami Keshvanand Institute Of Technology, M & G, Jaipur

Rajasthan Technical Kota, Jaipur

Session 2022-23

Swami Keshvanand Institute of Technology, Management & Gramothan, Jaipur

Department of Computer Science and Engineering

CERTIFICATE

This is to certify that ……**Mr. Akshat Gadodia (19ESKCS021)**……, a student of B.Tech (Computer Science & Engineering) …..VII….. Semester has submitted his Project Report entitled "…..**Airport Feedback App**….." under my guidance

Mentor:
Dr. Nilam Choudhary
Dept. of Computer
Science & Engineering

Coordinator:
Mr. Sumit Mathur
Dept. of Computer
Science & Engineering

Swami Keshvanand Institute of Technology, Management & Gramothan, Jaipur

Department of Computer Science and Engineering

CERTIFICATE

This is to certify that ……**Ms. Akshita Sharma (19ESKCS027)**……, a student of B.Tech (Computer Science & Engineering) …..VII….. Semester has submitted his Project Report entitled "…..**Airport Feedback App**….." under my guidance

Mentor:
Dr. Nilam Choudhary
Dept. of Computer
Science & Engineering

Coordinator:
Mr. Sumit Mathur
Dept. of Computer
Science & Engineering

Swami Keshvanand Institute of Technology,

Management & Gramothan, Jaipur

Department of Computer Science and Engineering

DECLARATION

We hereby declare that the report of the project entitled **"AIRPORT FEEDBACK APP"** is a record of an original work done by us at **Swami Keshvanand Institute of Technology, Management & Gramothan, Jaipur under the mentorship of Dr. Nilam Choudhary (Dept. of Computer Science & Engineering) and coordinator Mr. Sumit Kumar (Dept. of Computer Science & Engineering).** This project report has been submitted as the proof of original work for the particular fulfillment of the requirements for the award of the degree **Bachelor of Technology (B.Tech) in the Department of Computer Science.** It has not been submitted anywhere else, under any other program to the best of our knowledge.

Team Members: **Signature:**

Akshat Gadodia (19ESKCS021)

Akshita Sharma (19ESKCS027)

Acknowledgement

A project of such 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 Dr. Nilam Choudhary. She has been a guide, motivator and source of inspiration for us to carry out the necessary proceedings for the project to be completely successful. We would also like to thank Mr. Sumit Mathur for his 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. Mukesh Kumar Gupta, HOD, Department of Computer Science and Engineering, 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 & 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: **Signature:**

Akshat Gadodia (19ESKCS021)

Akshita Sharma (19ESKCS027)

TABLE OF CONTENTS

INDEX OF FIGURES

INDEX OF TABLES

ABBREVIATIONS USED

1.	STLSD	Switching Theory & Logic System Design
2.	MIS	Management Information System
3.	DBMS	Database Management System
4.	OS	Operating System
5.	UML	Unified Modeling Language
6.	XML	Extensible Markup Language
7.	HTTP	Hypertext Transfer Protocol
8.	JSON	JavaScript Object Notation
9.	NPM	Node Package Manager
10.	API	Application Program Interface
11.	JSX	JavaScript XML

PROJECT CHAPTER

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.

Solution: 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:

- SATS gives the facility to provide co-ordination between airport staff and passengers.
- Also allows the passengers to access their belongings.

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 android app using which the airport authority can manage the database and passengers can access for their belonging.

1.2 Investigation & Analysis

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

Passengers

- Can submit their feedback according their preference.
- They can provide their experience based feedback after using airport provided services.
- 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 Solutions

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 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.

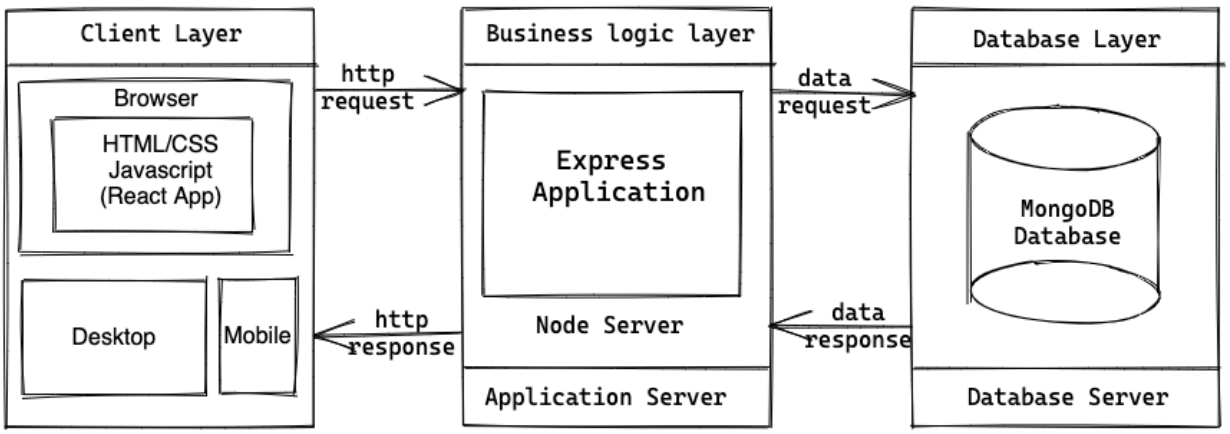

Figure 1.1: Architecture Diagram of the Proposed System Methodology

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.

SYSTEM REQUIREMENTS SPECIFICATIONS

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. These details provide valid login to user after getting registered.
- **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

Minimum Requirements:

Client Side			
	Processor	**RAM**	**Disk Space**
Google Chrome	Intel Pentium III or AMD - 800 MHz	128 MB	100 MB

Table 2.1 Minimum Client Side Hardware Interface

Server Side			
	Processor	**RAM**	**Disk Space**
Node.js	Intel Pentium III or AMD - 800 MHz	1 GB	3.5 GB
MongoDB	Intel Pentium III or AMD - 800 MHz	256 MB	500 MB (Excluding Data Size)

Table 2.2 Minimum Server Side Hardware Interface

Recommended Requirements:

Client Side			
	Processor	**RAM**	**Disk Space**
Google Chrome	All Intel or AMD - 1 GHZ	256 MB	100 MB

Table 2.3 Recommended Client Side Hardware Interface

Server Side			
	Processor	**RAM**	**Disk Space**
Node.js	All Intel or AMD - 2 GHZ	2 GB	3.5 GB
MongoDB	All Intel or AMD - 2 GHZ	512 MB	500 MB (Excluding Data Size)

Table 2.4 Recommended 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.

Minimum Requirements:

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 2.5 Minimum Software Interface

Recommended Requirements:

Software Tool	Version	Purpose of Use
Operating System	Windows 8 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 2.6 Recommended Software Interface

2.1.1.5 Communication 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 2048 MB of RAM and 20 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:

a) A Login facility for enabling only authorized access to the system.
b) Passenger can view their flight details and can give feedbacks.
c) 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 & 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 username, 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.

2.2.2.3 Portability

The application will be easily portable among any windows or Linux based systems that have Apaches web server and MySQL database installed.

SYSTEM DESIGN SPECIFICATIONS

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

The proposed system can be decomposed into following major modules:

1. Login
2. Passenger
3. Admin
4. Database

Module - 1:

Login Page:

This is a common page for both the users, can either be passenger or admin.

It has this basic layout where the passenger needs to enter their name, email, number and PNR. For admin they have to enter their email and password.

There are dynamic pages linked to user of every profile.

Module - 2:

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 & time, flight arrival date & time, etc. on this page.

Feedback Page:

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 in 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 Page:

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.

View Feedback Page:

This page displays the feedbacks summary of the respective department in the form of circular progress bar. This page also displays the entire feedback message given by the passengers

Module - 4:

Database:

The overall objective in 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.

Database is an integrated collection of data. The most significant form of data as seen by the programmers is data as stored on the direct access storage devices. This is the difference between logical and physical data.

Database files are the key source of information into the system. It is the process of designing database files, which are the key source of information to the system. The files should be properly designed and planned for collection, accumulation, editing and retrieving the required information.

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

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

The proposed system stores the information relevant for processing in the MySQL SERVER database. This database contains tables, where each table corresponds to one particular type of information. Each piece of information in table is called a field or column.

A table also contains records, which is a set of fields. All records in a table have the same set of fields with different information. There are primary key fields that uniquely identify a record in a table.

3.3 High Level Design Diagrams

3.3.1 Use – Case Diagrams

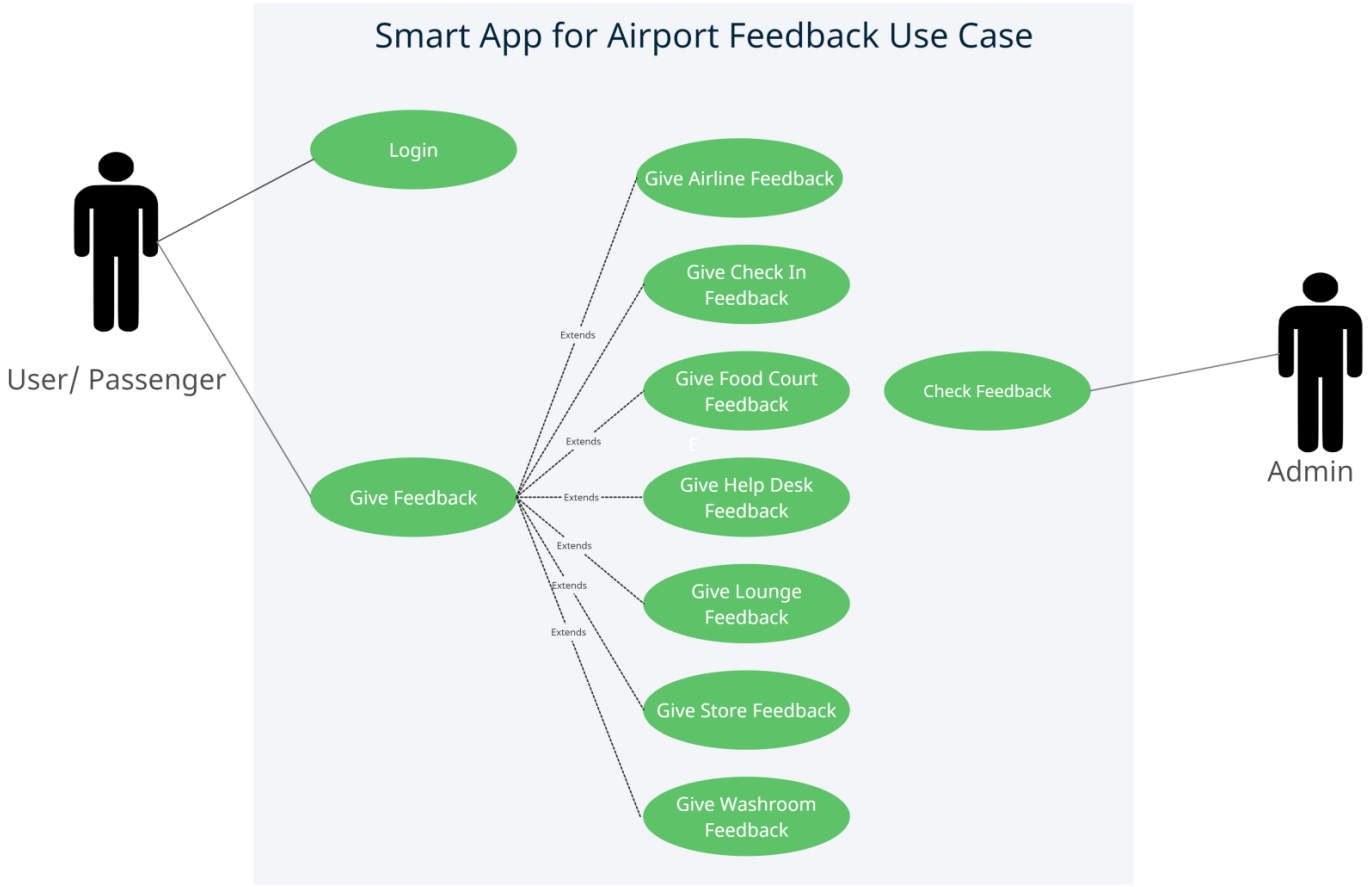

Fig 3.1: Use - Case Diagram

3.3.2 Activity Diagram

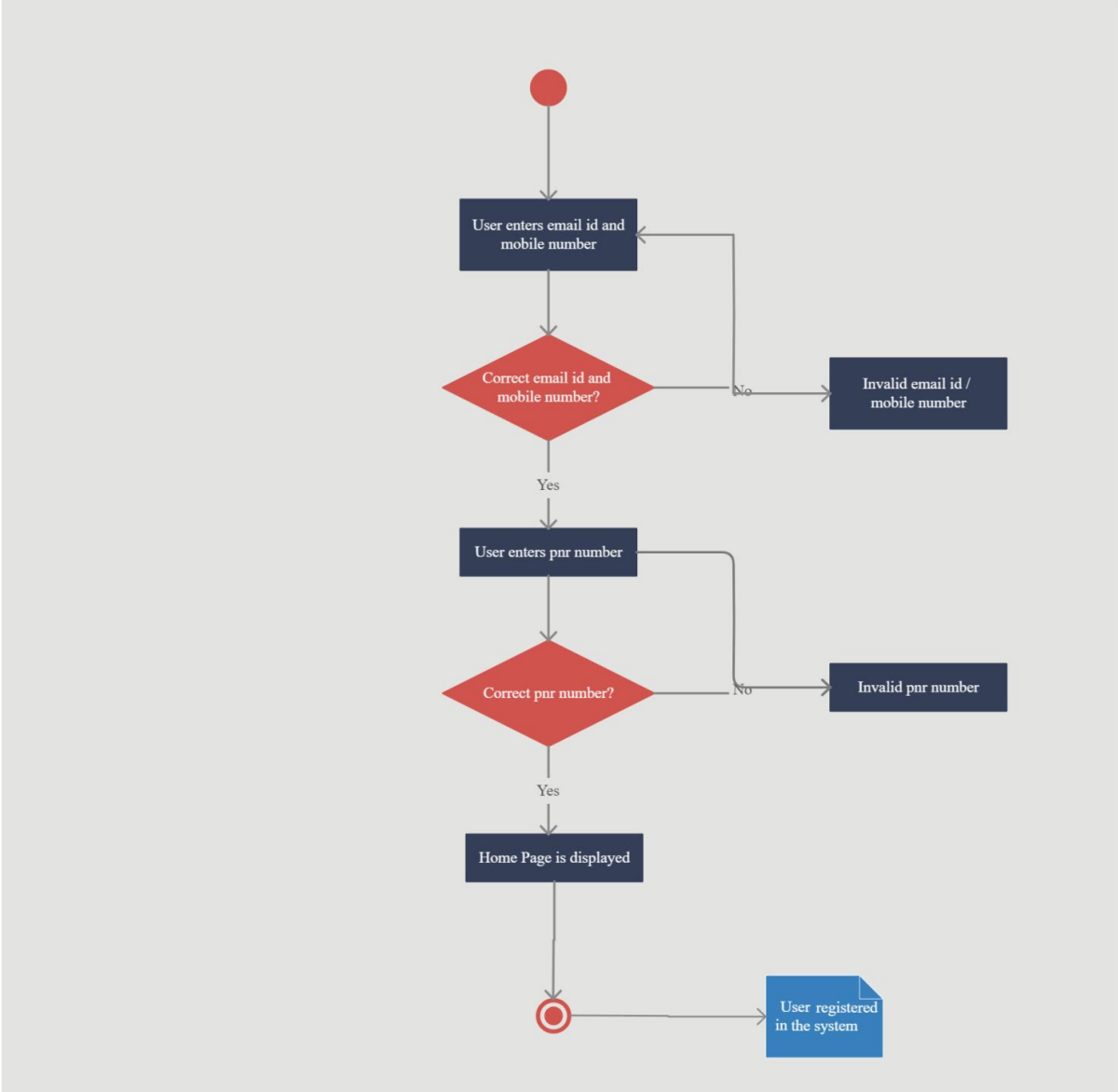

Fig 3.2: Activity Diagram Representing Passenger Registration

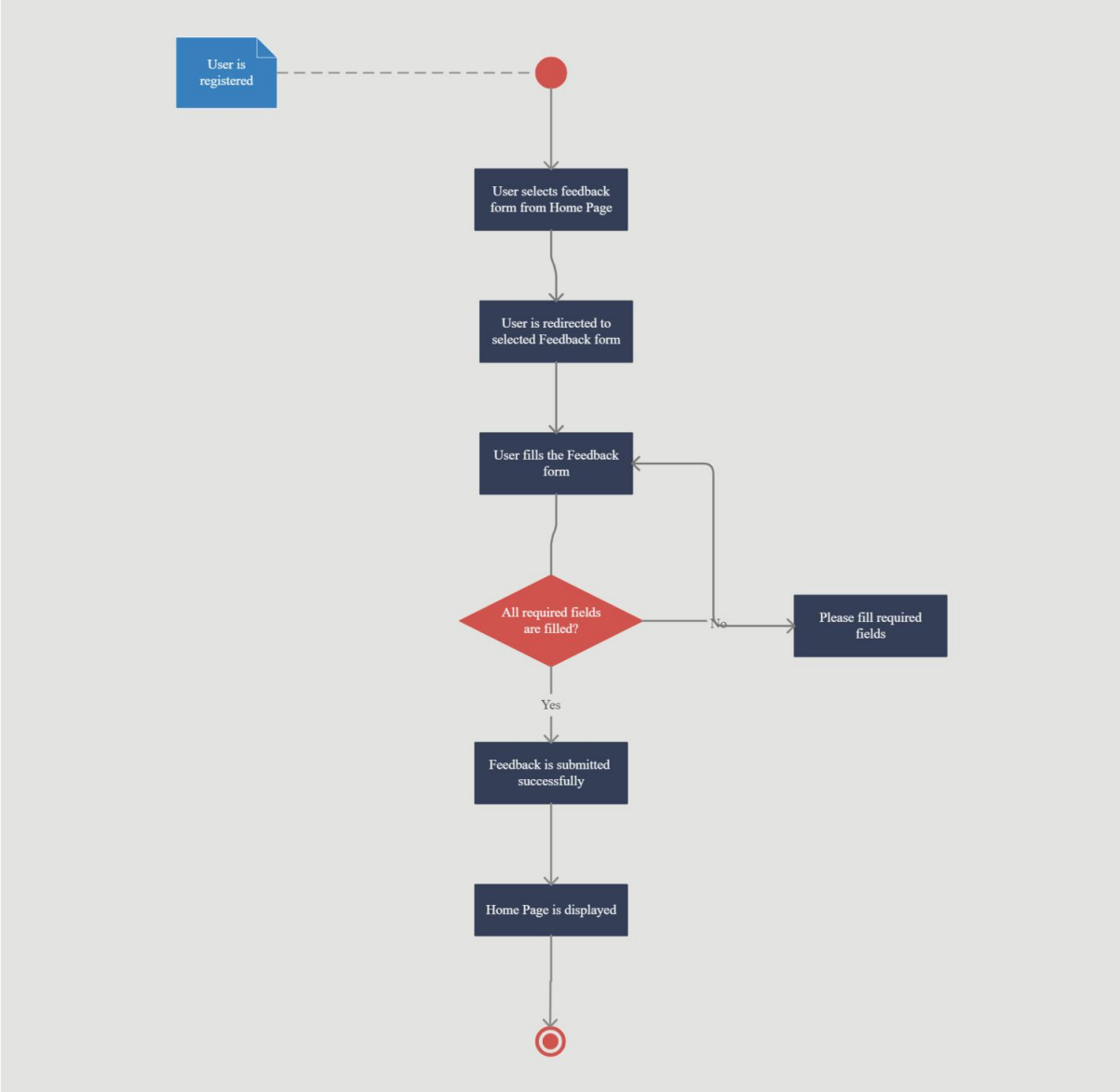

Fig 3.3: Activity Diagram Representing Passenger Feedback

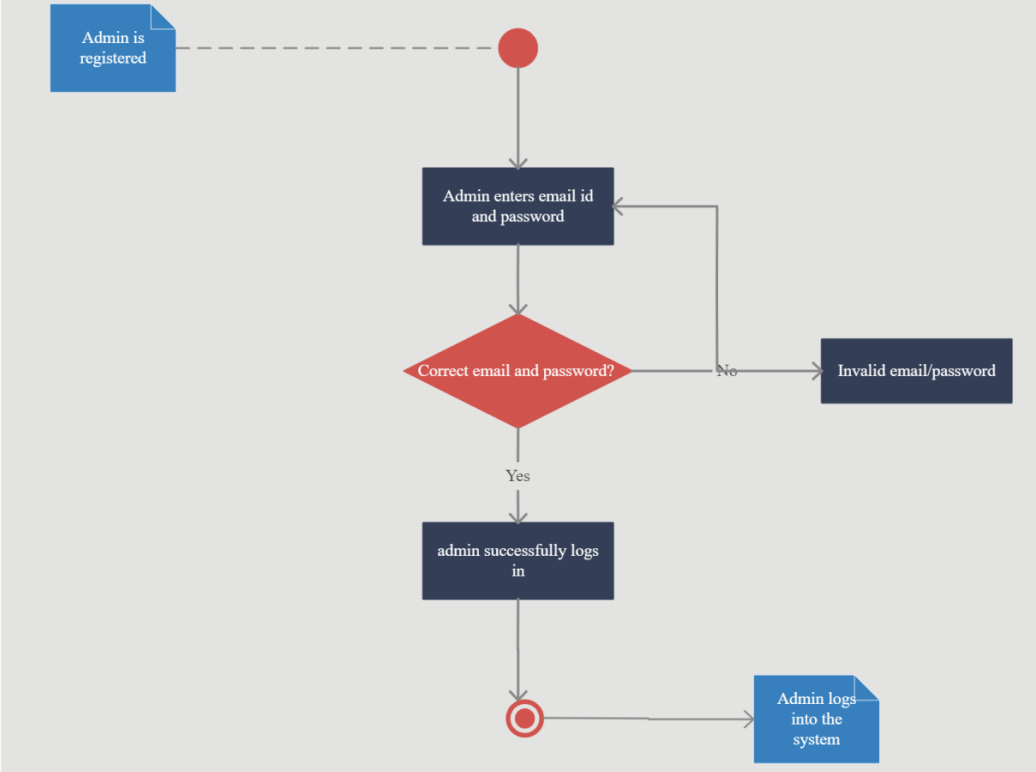

Fig 3.4: Activity Diagram Representing Admin Login

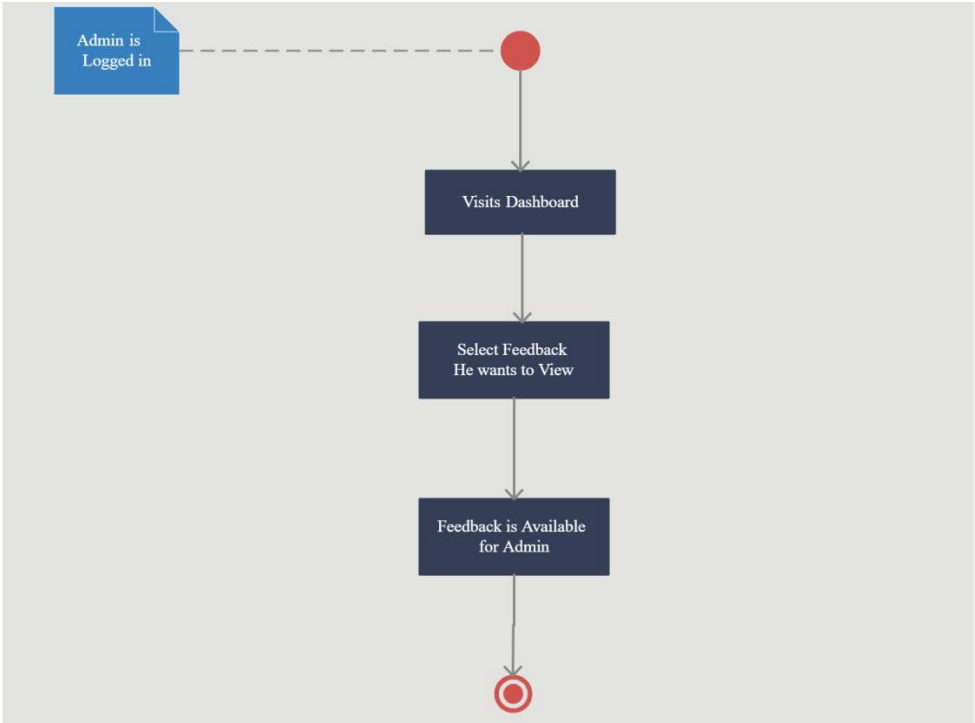

Fig 3.5: Activity Diagram Representing Admin View Feedback

3.3.3 Sequence Diagram

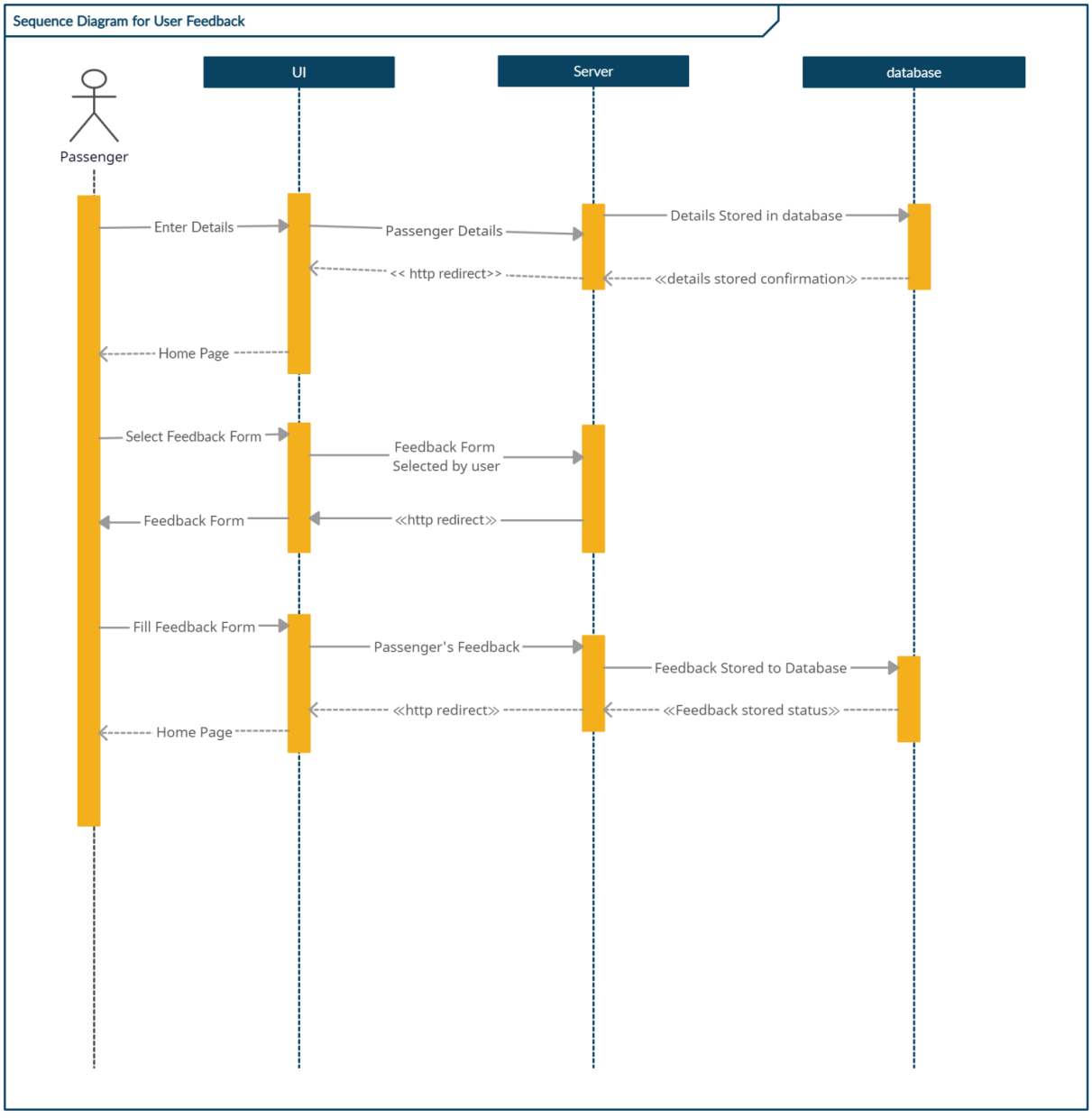

Fig 3.6: Sequence Diagram Representing Passenger

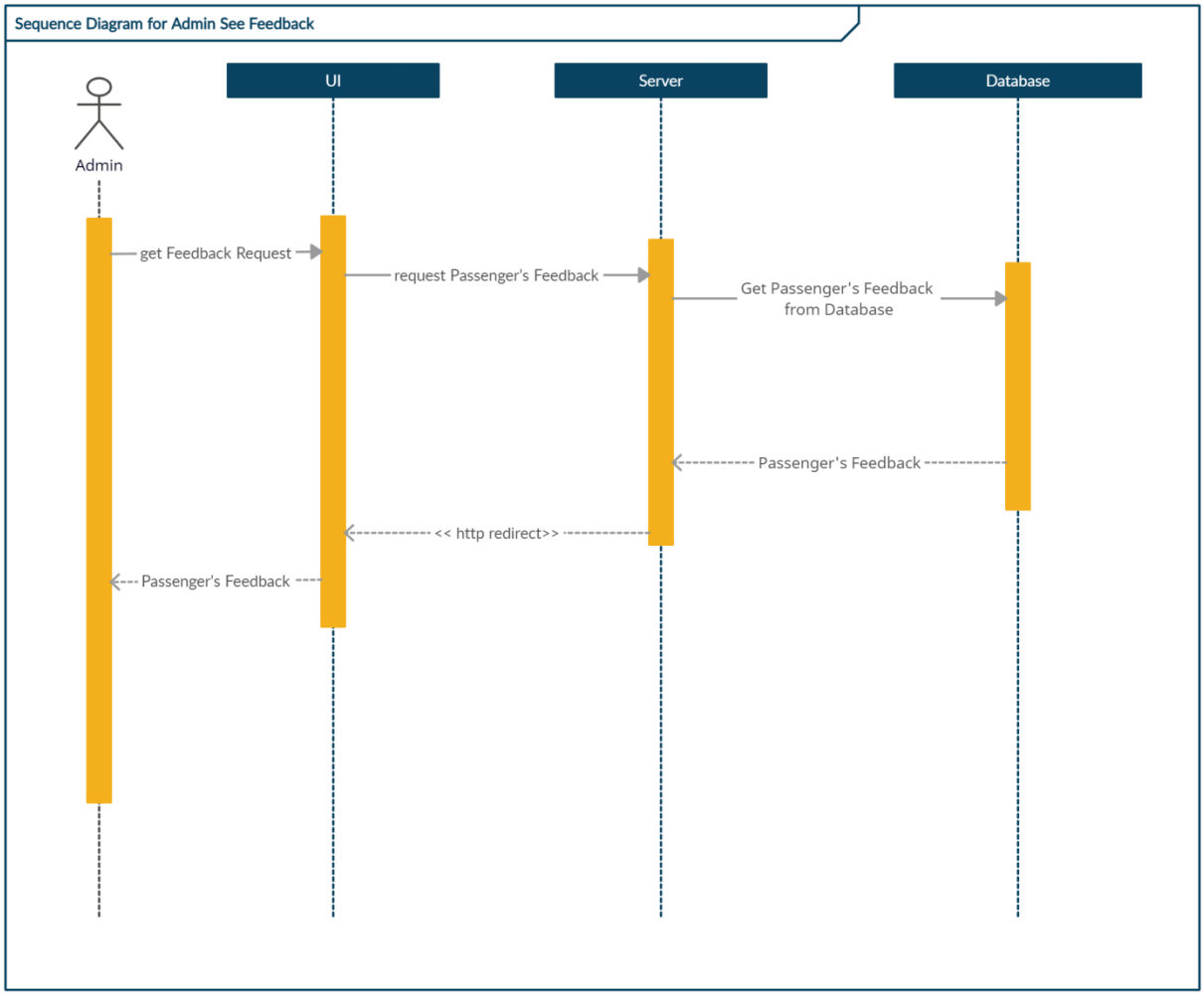

Fig 3.7: Sequence Diagram Representing Admin View Feedback

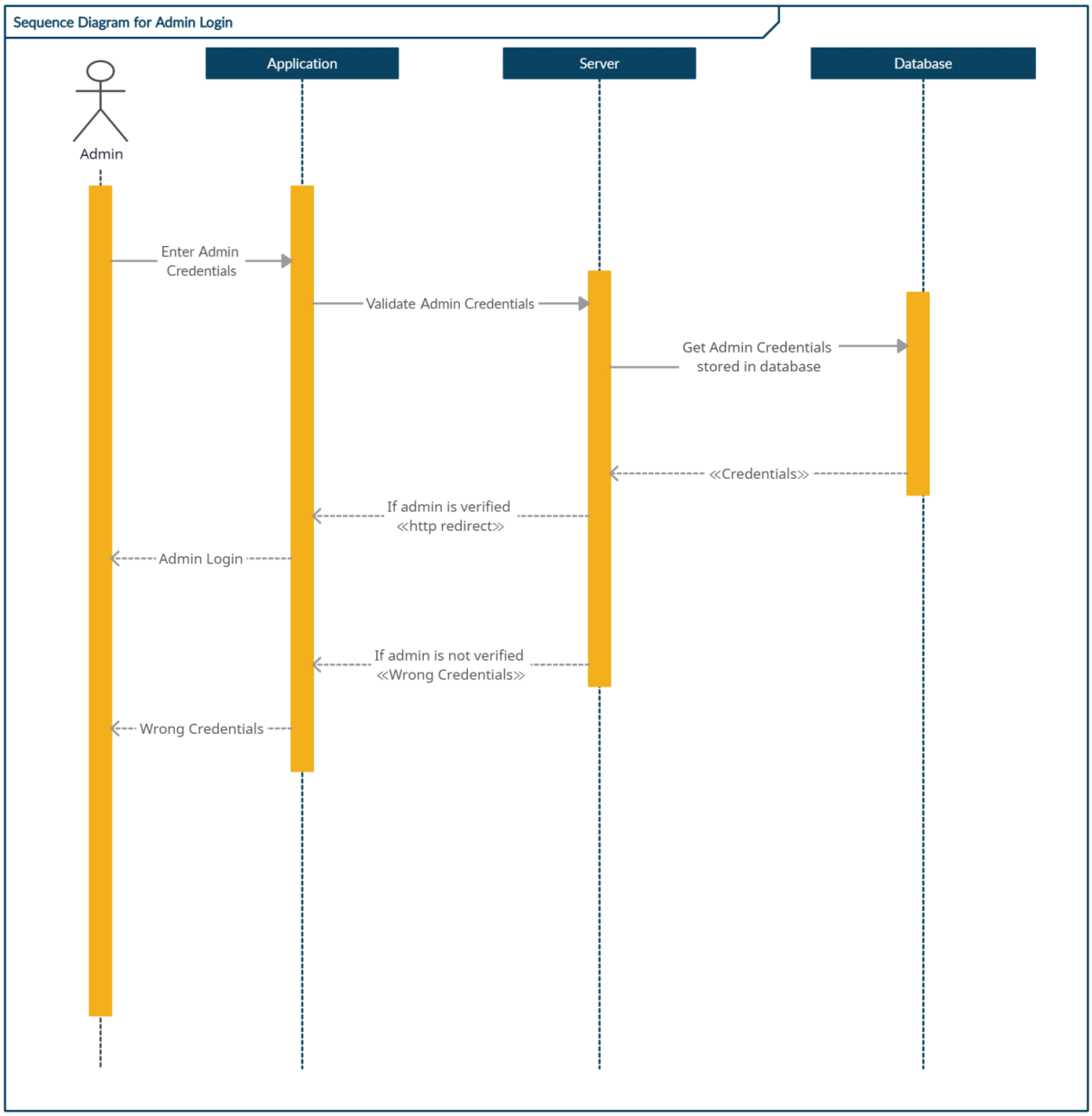

Fig 3.8: Sequence Diagram Representing Admin Login

3.3.4 Database Diagram

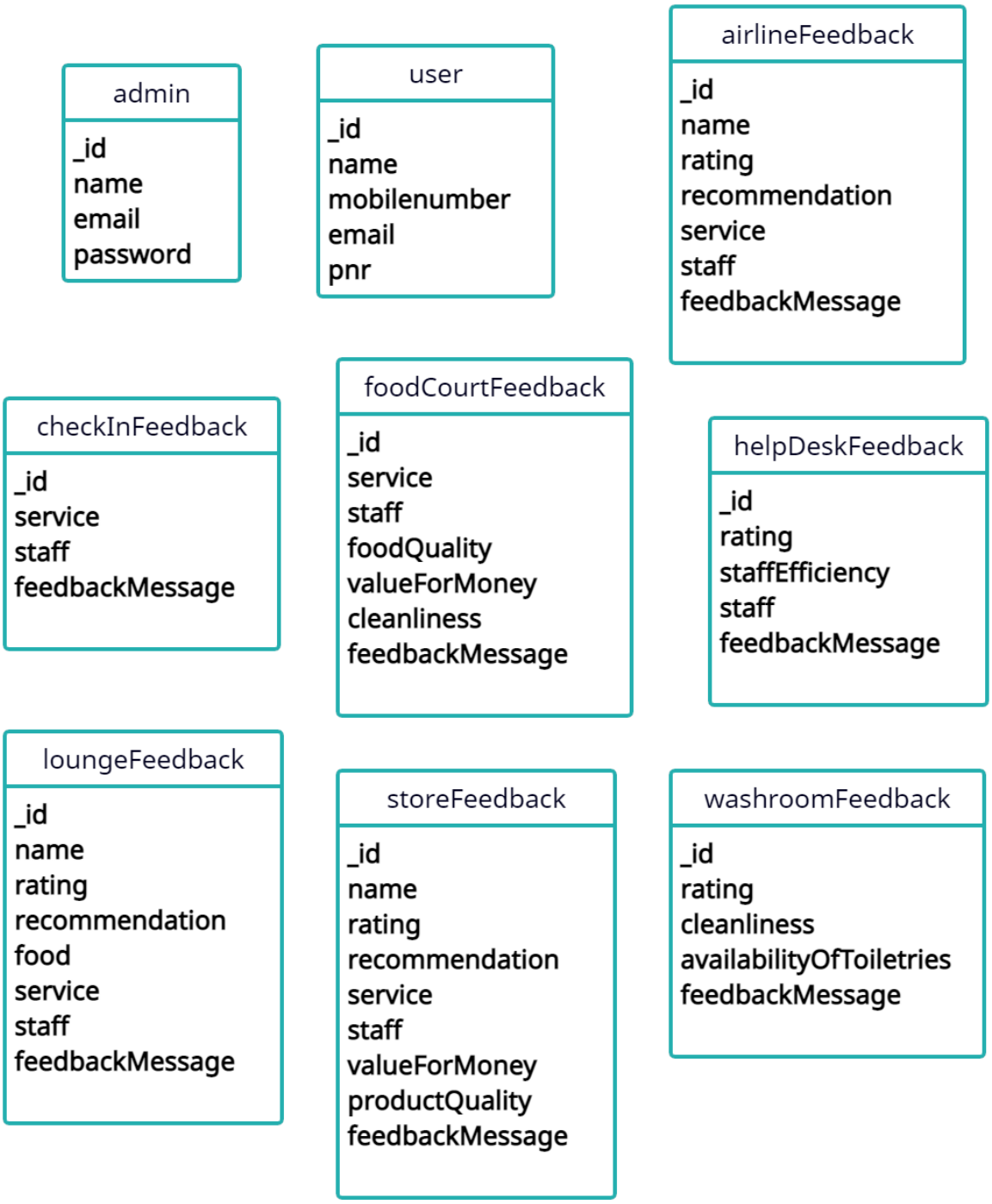

Fig 3.9: Database Diagram

3.3.5 Entity Relationship Diagram

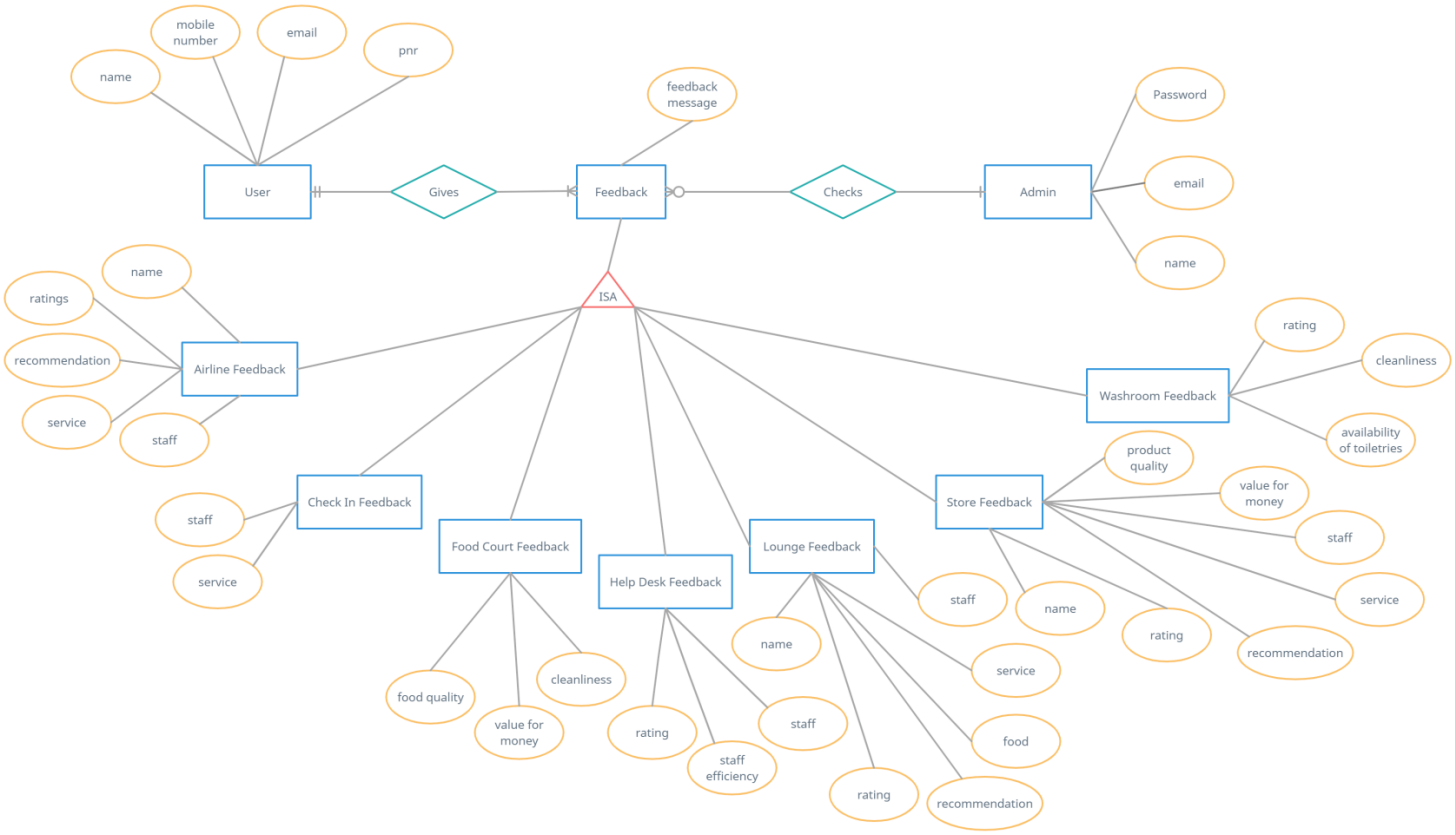

Fig 3.10: Entity Relationship Diagram

METHADOLOGY & TEAM

4.1 Introduction to Waterfall Model

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 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, typically, the outcome of one phase acts as the input for the next phase sequentially.

Following is a diagrammatic representation of different phases of waterfall model.

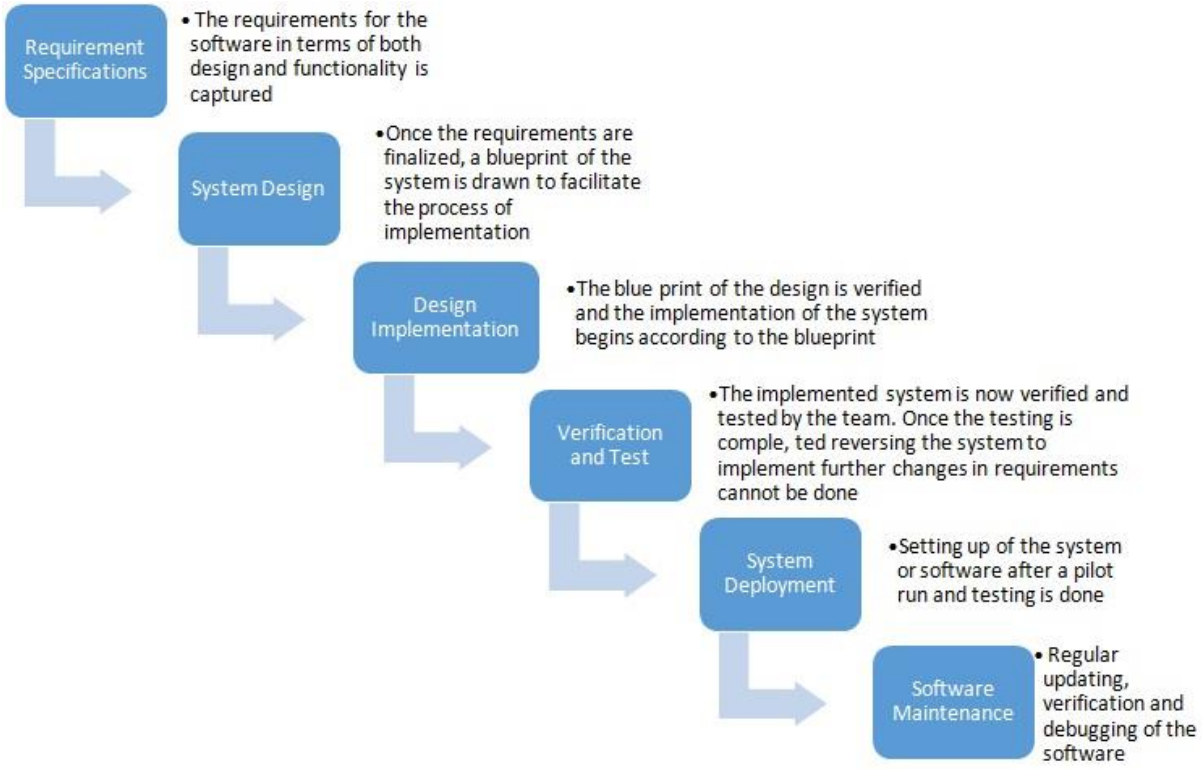

Fig 4.1: Waterfall Model with Feedback

The sequential phases in Waterfall model are:

- **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.
- **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.
- **Implementation:** With inputs from system design, the system is first developed 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.
- **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.
- **Deployment of system:** Once the functional and non-functional testing is done, the product is deployed in the customer environment or released into the market.
- **Maintenance:** There are some issues which come up in the client environment. To fix those issues patches are released. Also to enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment.

All these phases are cascaded to each other in which progress is seen as flowing steadily downwards 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". Every software developed is different and requires a suitable SDLC approach to be followed based on the internal and external factors.

Some situations where the use of Waterfall model is most appropriate are:

- Requirements are very well documented, clear and fixed.
- Product definition is stable.
- Technology is understood and is not dynamic.
- There are no ambiguous requirements.
- Ample resources with required expertise are available to support the product.

Waterfall Model Pros & Cons

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.

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

Team Members	Project Role	Responsibilities
Akshat Gadodia	Team Member	Designing Database, Admin Module, Integration of all Modules, Authentication
Akshita Sharma	Project Manager	Frontend, Passenger Module, Styling

Table 4.1: Roles and Responsibilities

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:

- **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 in 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) Mail Links:

No mail links are provided in our website till this stage but this is also a future enhancement of our website to trigger mails to people for keeping them updated about the online registration dates, the polling dates and other details.

d) Broken Links:

Broken link are those links which so not divert the page to specified 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.

- **Forms**

a) Field validation

Checks on dates have been applied. For e.g. The Date of birth should be less than the current date and after that we have checked the age to be greater than or equal to 18 years (the eligible age for casting vote). Checks have been applied on starting and ending dates, i.e. the

starting dates of elections or registrations should always fall before their respective ending dates.

b) Error message for wrong input

Error messages have been displayed as and when we enter the wrong details (e.g. Dates), and when we do not enter any detail 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.

c) 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.

- **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.

Till now it is done using the null modem on two systems.

The system load includes:

a) What is the number of users per time?
b) Checking for peak loads and how system behaves.
c) 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.

a) Ease of learning
b) Navigation
c) Subjective user satisfaction
d) General appearance

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

5.4 Server Side Interfacing

In this we tested the server side interface. This was done by verifying that communication is done properly. Also the compatibility of server with software, hardware, network and database was tested

5.5 Client Side Compatibility

The client side compatibility is also tested using various browsers like Google Chrome, Mozilla Firefox and Internet Explorer.

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 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 = O
5. Status of Test Cases = Passed

S. No.	Test Case ID	Test Case Description	Expected Outcome	Test Case Status	No. of Resources Consumed
1.	PROI	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

PROJECT SCREEN SHOTS

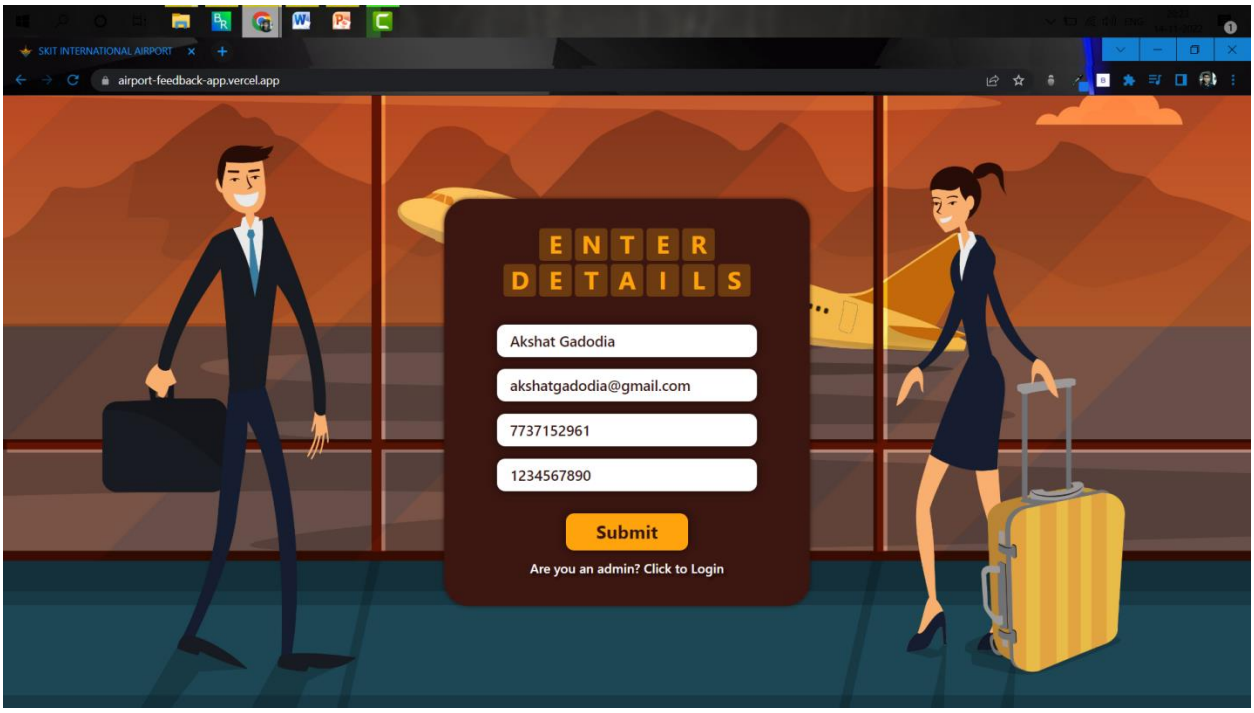

Fig 7.1: Passenger Login Screen

Fig 7.2: Passenger Home Displaying Flight Details

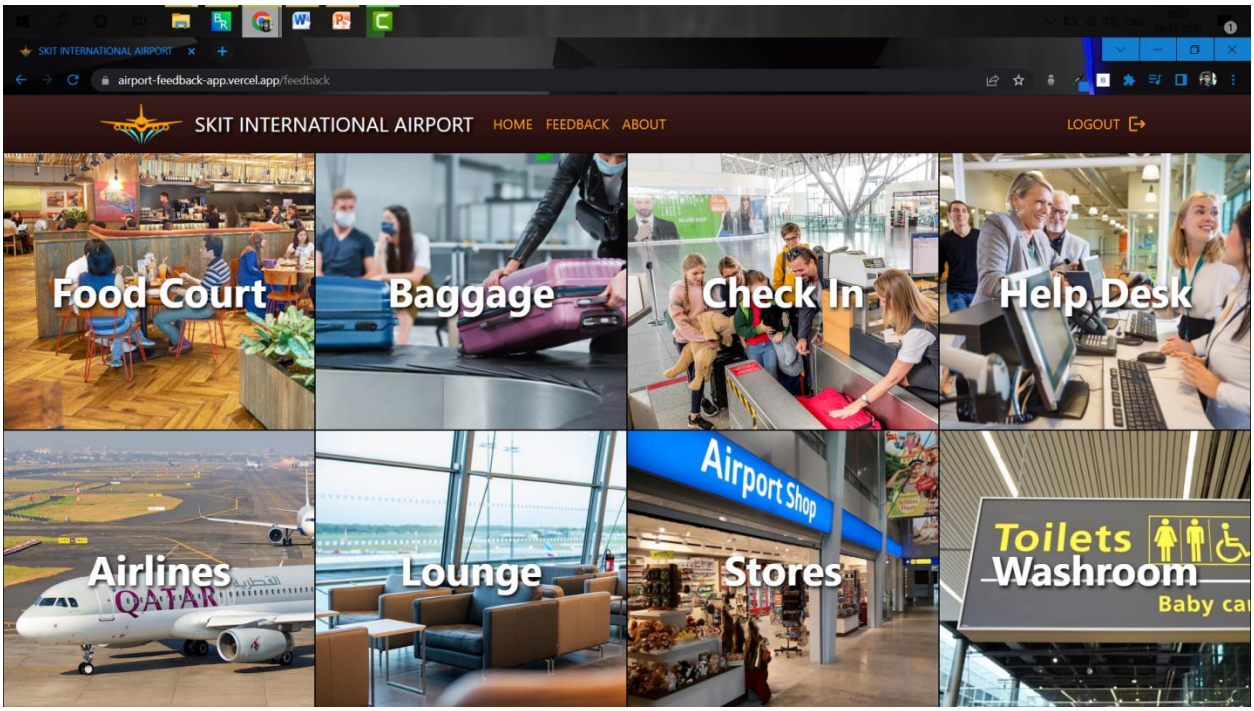

Fig 7.3: Feedback Page (Common for both Passenger & Admin)

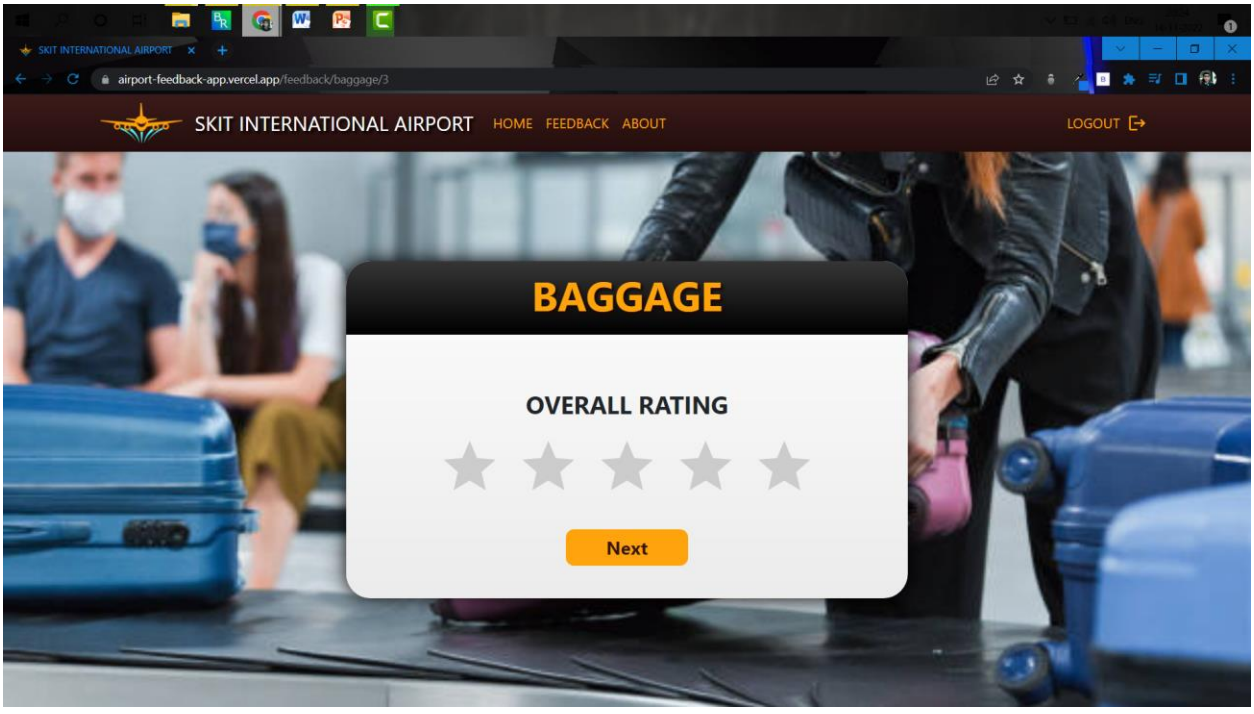

Fig 7.4: Passenger Rating Form

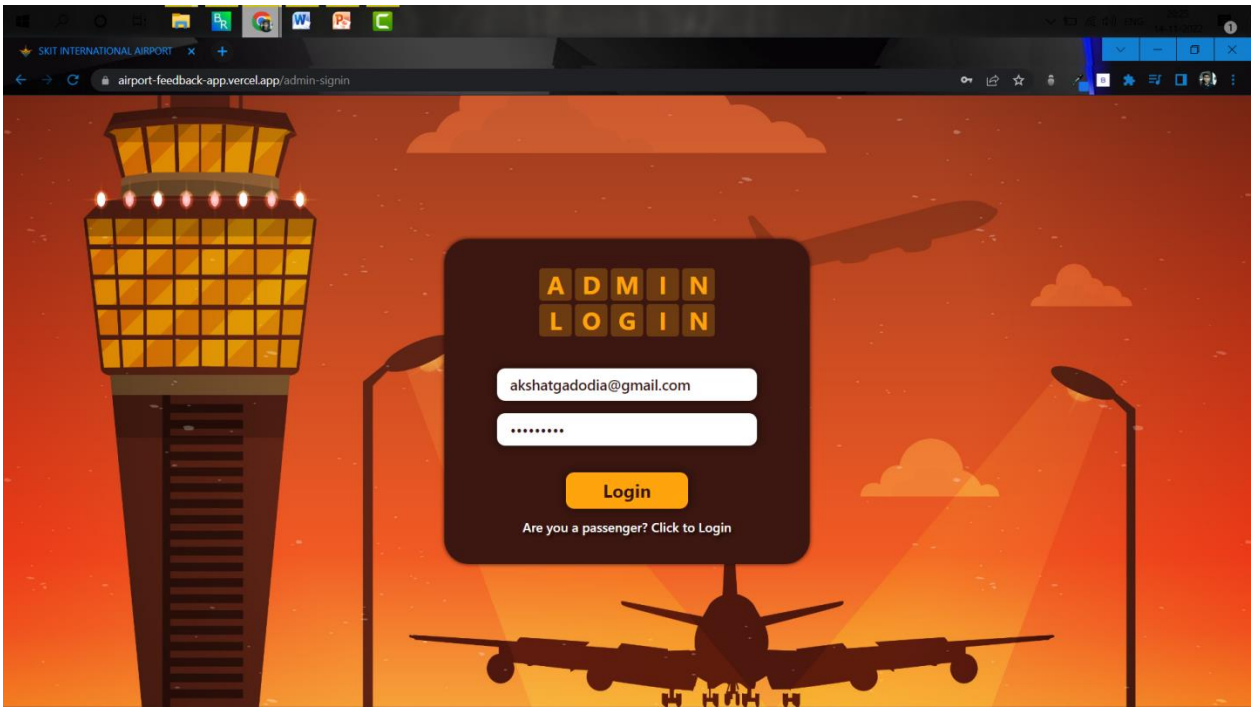

Fig 7.5: Admin Login Page

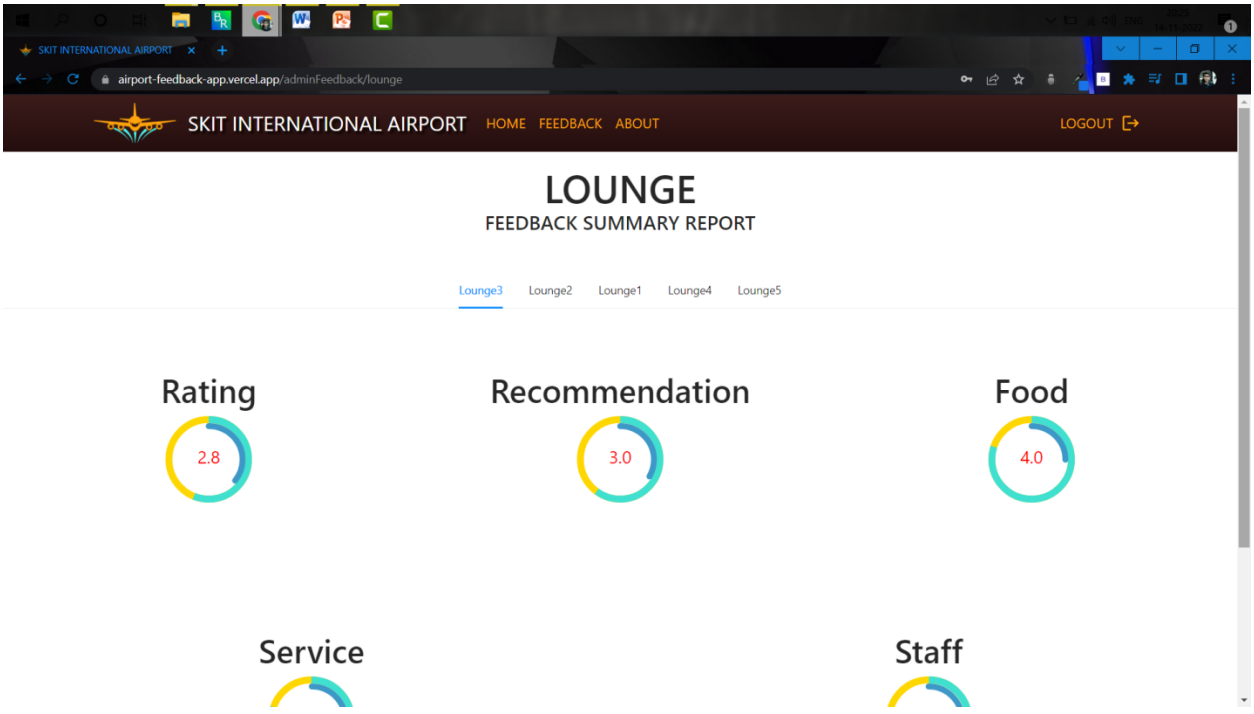

Fig 7.6: Lounge View Feedback Page

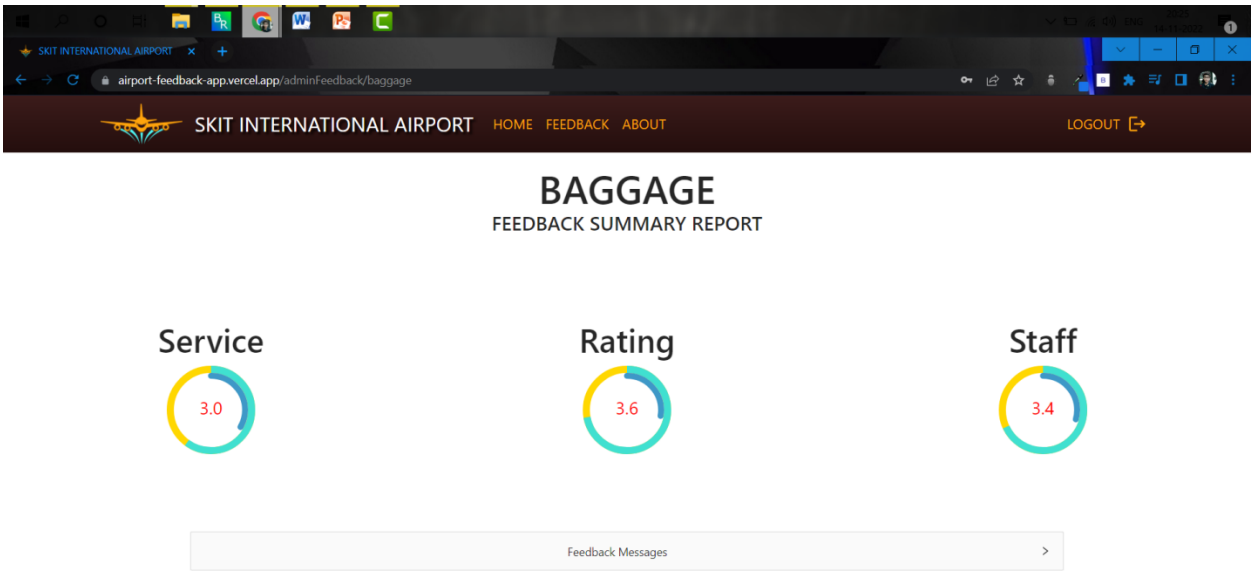

Fig 7.7: Baggage View Feedback Page

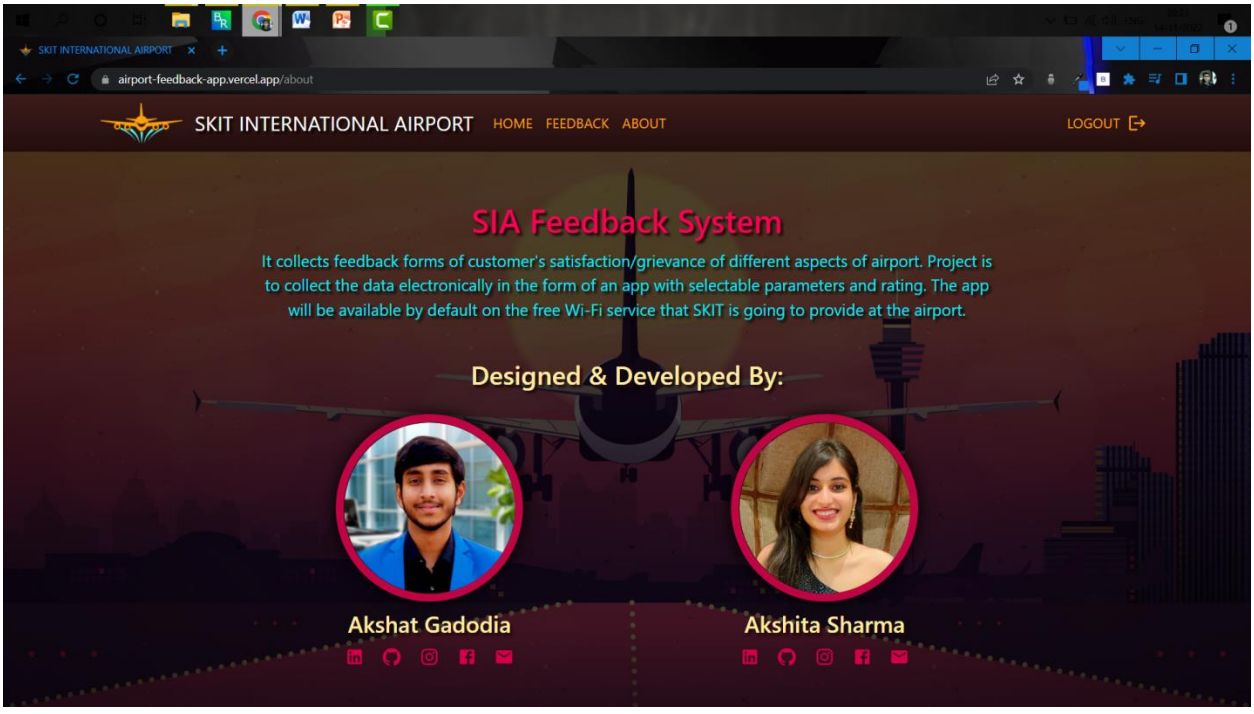

Fig 7.8: About Page

SUMMARY AND CONCLUSIONS

"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.

Most progressive organizations have skill evaluation as a formal program. An employee's customized development plans are made reflecting his/hers skills and talents. These exercises are conducted in addition to the informal programs. Most large it companies maintain a central skills inventory of their employees. Companies are currently conducting these programs from time to time as and when required, but the IT industry is looking at this with complete focus.

Many companies are even introducing web-based tracking system, which is more prevalent in the west." employee-skill inventory is becoming a major source for designing training and development programs, as well as becoming an integral part of HRM systems—whether automated or otherwise— in it companies. For organizations where formal programs are still a distant reality on account of resource constraints, conduction of informal exercises at least is becoming an acceptable norm.

Limitations

- It is limited up to process department of SAAF group only.
- Machine Learning is not included

FUTURE SCOPE

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

- SAAF will be enhanced up to all departments of SAAF group.
- 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 Rambaugh.
- Software Engineering, Seventh Edition, Ian Sommerville.
- Wikipedia - *www.wikipedia.com*
- MongoDB - *www.mongodb.com*
- Node.js- *www.w3schools.com*
- Express.js -*www.tutorialspoint.com*
- React.js - *www.reactjs.org*

PROJECT LINKS

- **GitHub Repository**
  https://github.com/akshatgadodia/airport-feedback-app
- **Deployed Application**
  https://airport-feedback-app.vercel.app/