FUEL CITY EXPRESS

A PROJECT REPORT

Submitted by

ARYA M.S (TKM21MCA-2012)

to

The APJ Abdul Kalam Technological University

In partial fulfillment of the requirements for the award of the degree of

MASTER OF COMPUTER APPLICATION

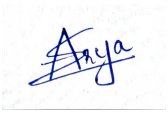
Thangal Kunju Musaliar College of Engineering
Kerala

DEPARTMENT OF COMPUTER APPLICATION

MAY 2023

DECLARATION

I undersigned hereby declare that the project report on **FUEL CITY EXPRESS**, submitted for partial fulfillment of the requirements for the award of degree of Master of Computer Application of the APJ Abdul Kalam Technological University, Kerala is a bonafide work done by me under supervision of **Dr. Nadera Beevi S**. This submission represents my ideas in my own words and where ideas or words of others have been included.I have adequately and accurately cited and referenced the original sources. I also declare that I have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in our submission. I understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree, diploma or similar title of any other University.



Kollam

16-05-2023

ARYA M.S

DEPARTMENT OF COMPUTER APPLICATION

TKM COLLEGE OF ENGINEERING, KOLLAM

2022 - 2023

CERTIFICATE

This is to certify that the report entitled **FUEL CITY EXPRESS** submitted by **ARYA M.S** (TKM21MCA-2012) to the APJ Abdul Kalam Technological University in partial fulfillment of the Masters degree in Computer Application is a bonafide record of the project work carried out by her under our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

Internal Supervisor Head of the Department External Examiner

Acknowledgement

First and foremost, I thank GOD almighty and my parents for the success of this project. I owe sincere gratitude and heart full thanks to everyone who shared their precious time and knowledge for the successful completion of my project.

I am extremely grateful to **Dr. Fousia M Shamsudeen**, Head of the Department, Department of Computer Application, for providing me with the best facilities.

I would like to thank my project guide **Dr. Nadera Beevi S**, Department of Computer Application, who motivated me throughout the project.

I extend profound thanks to my project coordinator **Prof Vaheetha Salam**, Department of Computer Application for their exceptional support and coordination

I would like to express heartfelt gratitude to my advisor **Prof Nadthera Beevi M** ,Department of Computer Application, for the unwavering support throughout the project.

I profusely thank all other faculty members in the department and all other members of TKM College of Engineering, for their guidance and inspiration throughout my course of study.

I owe my thanks to my friends and all others who have directly or indirectly helped me in the successful completion of this project.

ARYA M.S

ABSTRACT

FUEL CITY EXPRESS, is a system that deals with fuel stations and provides a web application for users to search for the nearest pump using Google Maps. The pumps are to be registered to the site with their name, license number, location etc.The location is mapped using Google map. Customer who registers to the site can search for the nearest pump using the location in the map. They can send a request to the nearest pump found in the list. This request is processed by the pump itself. If the pump is unable to handle the request, the update is to be updated in the request. They can pass the request to the next nearest pump. This will solve the current problem.

We may be in emergency for fuel in some situations when we are travelling in a remote rural area. Getting fuel from nearest location will be a tedious task at that time. So, in order to help people this web application is developed.By using Google Maps and real-time updates, the app provides users with reliable information, simplifies the process of requesting fuel, and ensures safety.

Contents

List of Figures

Chapter 1

Introduction

FUEL CITY EXPRESS, Fuel City Express is a mobile app that delivers fuel on demand to vehicle owners who need to refuel their vehicles anytime and anywhere. The cost of creating such an app depends on multiple factors such as the features, technology, and operating system chosen. As the number of automobiles increases, fuel consumption also rises. When a vehicle runs out of petrol, pushing it to the nearest fuel station becomes an arduous task. Moreover, when people travel to new locations, they may not know where the nearest stfuelation is. Fuel City Express aims to solve these problems by providing a reliable and efficient fuel delivery service.

The app has three modules: user, fuel station, and admin. Before allowing users to access the user module, the admin verifies the details of the fuel station to ensure that only reliable and verified stations are available on the app. This approach ensures the safety and security of the users.

However, the safety of both the users and fuel delivery personnel is a significant concern when developing an on-demand fuel delivery app. Proper safety measures need to be implemented to prevent accidents that could have serious consequences.Fuel City Express is a convenient and reliable solution to the challenge of fuel delivery in remote areas. The app has the potential to transform the fuel delivery industry and make refueling more convenient for vehicle owners.

1.1 Problem Statement

Accessing fuel in remote rural areas, particularly during emergencies, by providing users with information on the nearest available fuel stations to save time and effort.

1.2 Objectives

Project deliverable includes:

- Create a user-friendly platform
- Cost-effective solution for vehicle owners
- Offer door to door delivery
- Implement a Complaint system
- Integrate secure payment gateways
- Offer robust customer support
- Enhance the overall refueling experience for vehicle owners

Chapter 2

Literature Survey

A literature survey, also known as a literature review, involves analyzing scholarly sources related to a particular subject. Examining the available literature, it provides a comprehensive overview of the state of the field, allowing you to identify relevant theories, approaches, and gaps in the existing body of knowledge. When conducting a literature review from an audit perspective, the main focus is on evaluating the relevant literature. This process covers information that has been published in a specific field of study and sometimes includes information published within a specific time frame. The literature review is an indispensable tool for conducting research and is frequently used as a starting point for delving into a specific subject area. In addition to identifying important theories and concepts, a literature review can also pinpoint gaps in current knowledge and draw attention to areas where further research is necessary. By scrutinizing multiple sources, a literature review can provide a more comprehensive understanding of a given topic or issue. A well-crafted literature review can also enhance the credibility and authority of the author, as it demonstrates their familiarity with the current research and debates in the field. In certain cases, a literature review may include a meta-analysis, which involves analyzing the findings of numerous studies to uncover common patterns or trends. It is important to keep in mind that a literature review is distinct from a research paper or an argumentative essay; it is instead a focused examination of the existing research and literature on a specific topic.

2.1 Purpose of the Literature Review

1. It chooses top-notch research papers or studies that are pertinent, significant, important, and valid and summarises them into a single comprehensive report to provide readers with quick access to information on a certain issue.

2. By requiring them to describe, assess, and compare original research in this particular field, it gives researchers who are starting their research in a new area a great place to start.

3. It makes sure that researchers don't repeat already completed studies.

4. It can indicate potential directions for future research or suggest topics to concentrate on.

5. It emphasises the important findings.

6. It points up gaps, discrepancies, and inconsistencies in the literature.

7. It offers a helpful critique of the methods and strategies used by other researchers.

2.2 Related Works

Battin et.al[1] focuses on the development of an Android application that utilizes the Google Maps API to provide location-based reminders. The application aims to assist users in setting reminders based on specific locations, where the reminders are triggered when the user enters or exits the defined geographical area.

By integrating the Google Maps API, the application leverages the geolocation services provided by Google to track the user's location. It enables users to set reminders tied to specific places such as home, work, or any custom location. When the user approaches or leaves the designated area, the application generates an alert or notification to remind the user of a task or event.

Pejic et.al[2] focuses on the development of an expert system specifically designed for tourists, which utilizes the Google Maps API. The objective of the system is to assist tourists in planning their trips and exploring new destinations by providing them with relevant information and recommendations.By integrating the Google Maps API, the expert system leverages the mapping and geolocation capabilities offered by Google. It enables users to search for various

points of interest such as tourist attractions, restaurants, hotels, pumps and other relevant places. The system then uses the obtained location data and additional parameters to generate personalized recommendations and suggestions for the users.

Li et.al[3] focuses on conducting an empirical study of three PHP frameworks. PHP frameworks are tools that provide a structured and efficient way to develop web applications using the PHP programming language. The study aims to evaluate and compare the performance and features of three specific PHP frameworks, although the exact frameworks are not mentioned in the provided information.presents the findings of the study, including the strengths and weaknesses of each framework based on the empirical analysis. It may discuss factors such as ease of use, code maintainability, community support, available resources, and extensibility. The authors might also provide recommendations or insights for developers who are considering using PHP frameworks for their web application projects.

Bai et.al[4]provides an introduction to the Apache NetBeans Integrated Development Environment (IDE) in the context of SQL Server database programming with Java. The Apache NetBeans IDE is a popular open-source development tool widely used for Java development and offers comprehensive features to support database programming.Also provide an overview of the Apache NetBeans IDE, discussing its key features, user interface, and capabilities specifically related to SQL Server database programming. This might include features like SQL code editing and execution, database connectivity, SQL syntax highlighting, query optimization, and other relevant functionalities.It also covers setting up the Apache NetBeans IDE for SQL Server database development, configuring database connections, and managing database-related projects within the IDE.

Liu et.al[5] focuses on the research and development of Android applications. Android is an open-source mobile operating system developed by Google, and this paper likely explores various aspects related to creating applications for the Android platform.Android software development kit (SDK), which provides the necessary tools, libraries, and APIs for developing Android applications. It may discuss the programming languages, such as Java or Kotlin, commonly used for Android app development.It also address user interface design for Android applications, including the use of XML-based layout files, widgets, and event handling. It may discuss best practices and design patterns for creating intuitive and visually appealing user interfaces.

Latif et.al[6]focuses on providing a survey of cross-platform approaches for mobile

application development. Cross-platform developmentpresent an overview of various cross-platform development frameworks, tools, and technologies available at the time of publication refers to the process of creating mobile applications that can run on multiple platforms, such as iOS and Android, using a single codebase.The authors likely discuss the challenges and advantages of cross-platform development compared to platform-specific development. They may highlight the potential cost savings, time efficiency, and code reusability that can be achieved through cross-platform frameworks.It present an overview of various cross-platform development frameworks, tools, and technologies available at the time of publication. The authors might discuss the features, capabilities, and suitability of each framework for different types of mobile applications.Additionally, the paper may explore the technical aspects of cross-platform development, including the underlying architecture, development workflow, and performance considerations.

Wanganoo et.al [7] presents the challenges faced by cities in delivering goods to customers efficiently, particularly in the last-mile delivery stage. The solution consists of several components, including an IoT-based tracking system for monitoring delivery vehicles, a route optimization algorithm, and a customer feedback system. The authors provide a detailed description of each component and how they work together to improve delivery efficiency.Tt was able to reduce delivery times by up to 25 percentage and improve overall delivery efficiency.Highlight the potential benefits of IoT technologies in improving last-mile delivery services in smart cities, such as reducing traffic congestion, improving delivery times, and enhancing customer satisfaction.

Shibghatullah et.al[8] presents a vehicle tracking application that utilizes real-time traffic data to provide accurate and timely information to users. The authors propose a system that employs Global Positioning System (GPS) technology and real-time traffic data to monitor the location of vehicles and calculate the expected time of arrival (ETA) for each vehicle. The paper describes the design and implementation of the proposed system, highlighting its key features and functionalities.The use of real-time traffic data for vehicle tracking applications, offering a comprehensive analysis of the system's design and performance. A valuable contribution to the field of vehicle tracking applications, offering insights that could potentially transform the transportation industry's approach to tracking vehicles and optimizing traffic flow.

Vuksanovic et.al[9] the advantages of web application frameworks, including faster development, improved security, access to standardized libraries, easier organization of work

in development teams, and a more structured codebase due to the adoption of conventions and design patterns that promote the separation of domain logic, user interface, and data processing.Despite these advantages, many developers tend to avoid using frameworks for small applications due to perceived disadvantages such as increased complexity, overhead from framework code, a learning curve associated with the framework, and concerns about potential security vulnerabilities.To address this issue, the paper compares the development process of two versions of the same application. The first version is developed using pure PHP, while the second version utilizes the CodeIgniter web application framework.It provides insights into the benefits and trade-offs of using frameworks in small-scale projects, helping developers make informed decisions when choosing their development approach.

Sarkar et.al[10] provides a concise overview of the Android platform and its evolution in terms of security systems. It aims to give readers a comprehensive understanding of the Android development process and the security measures employed in Android applications.Introducing the Android platform, highlighting its open-source nature, widespread adoption, and the vast number of available applications on the Google Play Store.Then delves into the security aspects of Android applications. It explores the evolution of security systems in Android, starting from the early days when security was limited to permissions and application sandboxing.It covers more advanced security mechanisms, such as the Android Security Model, which encompasses secure inter-process communication, secure storage, and secure networking. It also highlights the importance of secure coding practices and the use of secure libraries in Android application development.

Chapter 3

Methodology

FUEL CITY EXPRESS system requires a structured methodology that includes planning, design, development, testing, deployment, and maintenance. During the planning phase, the scope of the project is defined, requirements are identified, and a timeline is set for completion. In the design phase, a detailed blueprint of the application is created, including the user interface, database schema, and API architecture, which takes into account the requirements identified in the planning phase. Development involves coding the frontend and backend components of the application in parallel and testing them thoroughly to ensure they function as intended. Once development is complete, testing is performed to ensure the application meets all requirements before deployment. Finally, a maintenance plan is put in place to ensure the application remains reliable and secure over time.

Structured methodology can help ensure the success of the project and create a reliable, scalable, and secure application that meets the requirements of all stakeholders. With careful planning and a detailed design, developers can create an application that meets all identified requirements. Thorough testing ensures the application works as intended before deployment, and a maintenance plan keeps the application reliable and secure over time.

3.1 Module Description

3.1.1 User Module

- Register: The user creates a new account on the platform by providing personal information such as name, contact information, etc.
- Login: The user logs into their account using their registered email address and password.
- Search Fuel Station: The user searches for a fuel station based on their location or other relevant criteria such as fuel type or pricing.
- Place Order: The user selects a fuel station and places an order for a specific amount of fuel, either by providing the amount in liters or by specifying the amount in monetary value.
- Make Payment: The user submits a payment for their order through the platform using their preferred payment method, such as credit card, debit card, or online payment services.
- Get Delivery: Once the payment is confirmed, the user receives the delivery of fuel at their specified location, either by providing the delivery address or by using the location-based services of the platform.

3.1.2 Fuel Station Module

- Register: The Fuel Station registers on the platform by providing its business information such as name, address, contact information, and payment details.
- Login: The Fuel Station logs into its account using its registered email address and password.
- Create Fuel Station: The Fuel Station creates a new fuel station profile by providing its location, fuel types, fuel pricing, and other relevant details. This action enables the Fuel Station to be visible to users searching for fuel stations.
- Update/Delete Fuel Station: The Fuel Station can update or delete its fuel station profile to reflect any changes in its location, fuel types, pricing, or other relevant details.

- Receive Order: The Fuel Station receives an order from a user and verifies the order details such as the amount of fuel requested, the location of the delivery, and the payment status.

- Approve Status: The Fuel Station approves the status of the order, which may include confirming the payment, verifying the delivery location, and ensuring that the requested fuel is available. Once the Fuel Station approves the order, it initiates the delivery process.

3.1.3 Admin Module

- Login: The Admin logs into their account using their registered email address and password.

- Approve/Reject Fuel Station: The Admin reviews the details of a new fuel station profile created by a Fuel Station on the platform. Based on the review, the Admin can either approve or reject the Fuel Station profile. Approval means the Fuel Station becomes visible to users searching for fuel stations, while rejection means the Fuel Station profile is not published on the platform.

3.2 System Requirements

The application development architecture recognized for this project is specified in this section on the basis of requirements.

3.2.1 Hardware Specification

- Processor : intel i3 or above
- RAM : 8 GB or above
- Hard Disk : 10 GB or above
- Internet Connection

3.2.2 Software Specification

- Frontend : Android
- Programming language : Java
- IDE : Android studio,NetBeans
- Web server : Glassfish server , XAMPP server
- Web Browser : Any web browser
- Database : MySQL
- Backend : PHP

3.2.3 Software Tools and Platform

Android

Android is a Linux-based operating system designed primarily for touch screen mobile devices such as Smartphone and tablet computers.Android is open source and Google releases the code under the Apache License. This open-source code and permissive licensing allows the software to be freely modified and distributed by device manufacturers.Wireless carriers and enthusiast developers. Additionally, Android has a large community of developers writing

applications (”apps”) that extend the functionality of devices, written primarily in a customized version of the Java programming language.

Android Studio is Android’s official IDE. It is purpose-built for Android to accelerate your development and help you build the highest-quality apps for every Android device.

Android Studio’s Instant Run feature pushes code and resource changes to your running app. It intelligently understands the changes and often delivers them without restarting your app or rebuilding your APK, so you can see the effects immediately.The code editor helps you write better code, work faster, and be more productive by offering advanced code completion, refactoring, and code analysis.

The Android Emulator installs and starts your apps faster than a real device and allows you to prototype and tests your app on various Android device configurations: phones, tablets, Android Wear, and Android TV devices. You can also simulate a variety of hardware features such as GPS location, network latency, motion sensors, and multi- touch input. Android Studio includes project and code templates that make it easy to add wellestablished patterns such as a navigation drawer and view pager. You can start with a code template or even right-click an API in the editor and select Find Sample Code to search for examples. Moreover, you can import fully functional apps from GitHub, right from the Create Project screen.

Android Studio provides a robust static analysis framework and includes over 280 different lint checks across the entirety of your app. additionally, it provides several quick fixes that help you address issues in various categories, such as performance, security, and correctness, with a single click. Android software development is the process by which new applications are created for the Android operating system. Applications are usually developed in the Java programming language using the Android Software Development Kit.

ADT (Android Development Tools) is the software used to develop android apps. It basically encases Eclipse IDE, which is a multi-language Integrated development environment (IDE) comprising a base workspace and an extensible plug-in system for customizing the environment... The latest version comes with ADT plug-in preinstalled and bundled to the IDE.

Application programming interface (API) specifies how some software components should interact with each other. In practice in most of the cases an API is a library that usually includes specification for routines, data structures, object classes, and variables.

An API specification can take many forms, including an International Standard such as POSIX, vendor documentation such as the Microsoft Windows API, the libraries of a

programming language, e.g., Standard Template Library in C++ or Java API. Google APIs can be downloaded from Google Code, Google's site for developer tools, APIs and technical resources. The Google Data API] allow programmers to create applications that read and write data from Google services. Currently, these include APIs for Google Apps, Google Analytics, Blogger, Google Base, Google Book Search, Google Calendar, Google Code Search, Google Earth, Google Spreadsheets, Google Notebook, and Picasa Web Album.

Features:

- Code editing: Android Studio includes a powerful code editor with a range of features, such as syntax highlighting, code completion, and code refactoring tools.
- Debugging: Android Studio includes a range of debugging tools that enable developers to identify and fix issues in their code. The IDE supports both local and remote debugging, and includes features such as breakpoint management, variable inspection, and memory profiling.
- Project management: Android Studio includes a range of project management tools that enable developers to manage their projects efficiently. The IDE provides features such as version control integration, project templates, and build and deployment tools, making it easy to create and deploy Android apps.

Java

Java is a prevalent programming language that utilizes object-oriented design and is known for its ability to function on different platforms and high level of security. It is frequently utilized for constructing applications on various platforms including desktop, web, and mobile, and is also widely employed in enterprise software development. Java's source code is converted into bytecode, which can be executed on any system with a Java Virtual Machine (JVM). The language is famous for its resilience, expandability, and vast range of libraries, making it a preferred option for developing large-scale software. The popularity of Java has resulted in a dynamic network of tools, frameworks, and communities that support its development.

Features:

- Object-Oriented Design: Java is an object-oriented programming language, which means it allows developers to model real-world entities as objects with properties and behaviors. This feature makes it easier to write and maintain complex software.

- Security: Java has built-in security features that prevent unauthorized access and protect against malicious code. It includes a security manager, a security policy, and a sandbox environment that restricts the actions of untrusted code.

- Rich API: Java provides a vast collection of libraries and APIs that offer a wide range of functionalities, including networking, database connectivity, graphical user interface (GUI), and more. These libraries make it easier to develop complex applications by providing pre-built components that can be used with minimal effort.

- Editor :There are several editors available for Android development, including Android studio.

MySQL

MySQL is an effective data storage and management system. Typically, a database stores data in an organised format. It has been tested with a number of compilers to check for flaws and inconsistencies.

Features:

- Open-Source : MySQL is an open-source application that anybody may download, use, and edit. It is straightforward and free to use. The MySQL source code may be studied and altered to meet specific requirements. The application is governed by the GNU General Public License, or GPL, which specifies what may and cannot be done with it.

- Quick and Reliable :MySQL effectively saves data in memory, ensuring that it is consistent and not duplicated. Therefore, MySQL allows for rapid data access and manipulation.

- Scalable : Scalability is the capacity of a system to run without issue with little quantities of data, large amounts of data, clusters of machines, etc. The MySQL server facilitates the management of large databases.

- Data Types :It includes a variety of data types, including unsigned and signed integers, floats, doubles, characters, variable characters, text, blobs, dates, times, datetimes, timestamps, years, and more.

- Character Sets : It supports a variety of character sets, such as German, Ujis, latin1 (cp1252 character encoding), and other Unicode character sets.

PHP

PHP is a server-side scripting language used to develop web applications and dynamic websites. It is open-source, which means it is free to use and modify. PHP code is embedded in HTML and executed on the server, generating dynamic web pages for the client. PHP supports various databases, including MySQL, Oracle, and PostgreSQL, making it a powerful tool for web development. Its simplicity, ease of use, and wide adoption have made PHP one of the most popular programming languages for web development.

Features:

- Dynamic Web Page Generation: PHP allows developers to generate dynamic web pages by embedding PHP code within HTML, making it easier to create interactive and personalized websites.
- Database Connectivity: PHP supports various databases, including MySQL, Oracle, and PostgreSQL, allowing developers to easily connect their web applications to a database and manipulate data.
- Flexibility: PHP is a highly flexible language that can be easily integrated with other technologies, such as web servers, content management systems, and other programming languages, providing a wide range of options for web development.

NetBeans

NetBeans IDE is a free and open-source integrated development environment (IDE) for software developers. It provides developers with a range of tools for creating and managing applications, including support for multiple programming languages and platforms. NetBeans IDE offers features such as code editing, debugging, and project management tools, as well as support for third-party plugins. The ID is widely used for developing desktop, web, and mobile applications, and is particularly popular among Java developers.

Features:

- Source code development:NetBeans IDE includes a powerful code editor with a range of features, such as syntax highlighting, code completion, and code refactoring tools.
- Troubleshooting:NetBeans IDE includes a range of debugging tools that enable developers to identify and fix issues in their code. The IDE supports both local and remote

debugging, and includes features such as breakpoint management, variable inspection, and memory profiling.

- Software development management:NetBeans IDE includes a range of project management tools that enable developers to manage their projects efficiently. The IDE provides features such as version control integration, project templates, and build and deployment tools, making it easy to create and deploy applications across multiple platforms.

GlassFish Server

GlassFish is an open-source application server for Java-based web applications. It's developed by Oracle and provides a platform for hosting, deploying, and managing web applications. GlassFish supports the Java EE platform and includes features such as web services, security, and database connectivity. The application server is widely used for developing and deploying Java-based web applications, and is known for its ease of use and scalability.

Features:

- Database Connectivity: GlassFish includes built-in support for connecting to a variety of databases, including Oracle, MySQL, and PostgreSQL.
- Security: GlassFish offers a range of security features, including support for SSL/TLS encryption, authentication, and authorization.
- Web Services: GlassFish provides a comprehensive set of web services tools, including support for RESTful web services, SOAP-based web services, and Java API for XML Web Services (JAX-WS). These tools enable developers to create and consume web services in a variety of formats.

XAMPP Server

XAMPP can be considered a server. It is a software package that includes the necessary components to set up a local web server environment on your computer. It includes the Apache HTTP Server, which acts as the web server software, and MariaDB or MySQL, which serve as the database management systems. These components work together to allow you to host websites and web applications locally on machine.

3.3 System Design

System design is the process of developing specifications for a candidate system that meet the criteria established in the system analysis.Inventing and developing input, data bases, off line files, method and procedures, for processing data to get meaning full output that satisfy the organization objectives.

3.4 Input Design

Input design is the process of converting user-oriented input to a based format. Inaccurate input data are the most common cause of errors in data processing. Errors entered by data entry operators can be controlled by input design. The goal of designing input data is to make data entry as easy, logical and free from errors. When we approach input data design; we design the data source documents that capture the data and then select the media used to enter them into computer.

User interface design is very important for any application. The interface design describes how the software communicates within itself, to system that interpreted with it and with humans who use it. The input design is the process of converting the useroriented inputs into the computer-based format. Input design is a part of overallsystem design, which requires very careful attention. If data going into the system is correct, then the processing and output will magnify these errors. Thus, the designer has a number of clear objectives in the different stages of input design

- To produce a cost-effective method of input
- To achieve the highest possible level of accuracy
- To ensure that input is acceptable to and understand by the user

input data is most common cause of data processing errors. If poor input design-particularly where operators must enter data from source documents-permits bad data to enter a computer system, the outputs produced are of little value. The input design process was initiated in the study phase were, as a part of the feasibility study:

- Input data were found to be available for establishing and maintaining master and transaction files and for creating output records

- The most suitable types of input media, for either off-line or on-line devices were selected after a study of alternative data capture techniques

In this Fuel City Express Project's , all the text boxes are validated. If any mandatory field is not filled then it will display the message.

The features of the input design of Fuel City Express Project are:

- Input design is carried out in order to have the most effective means of user and system interaction.
- There have been steps done to reduce user inputs.
- The process is simplified for various project users and extra steps are removed.
- By using a password, the user is given additional security.

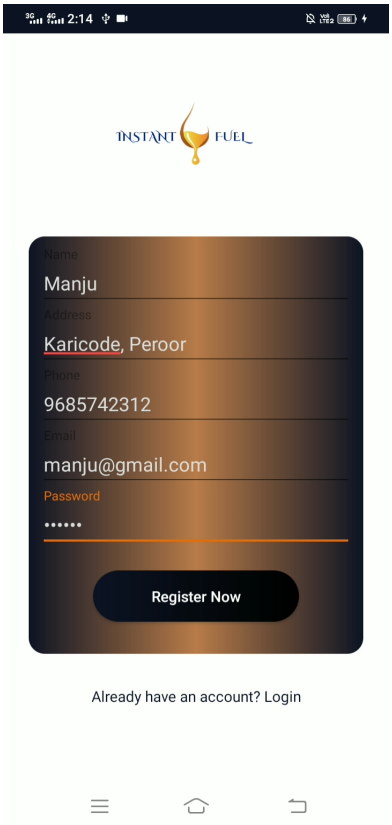


Figure 3.1: User Registration

user registration form in Fuel City Express enables new users to create an account by providing necessary information such as their name, contact details and a secure password.

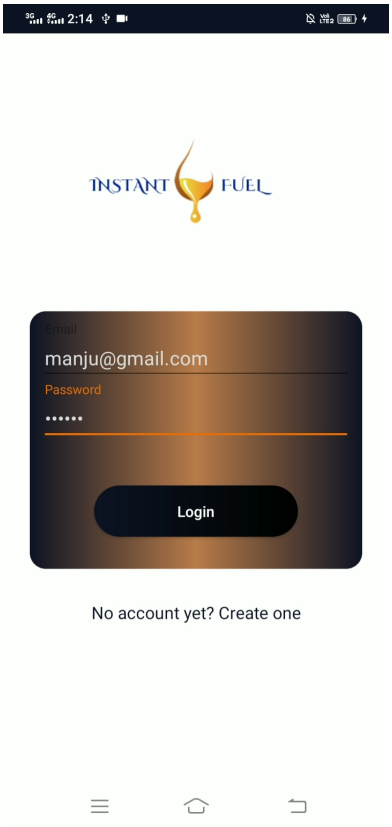


Figure 3.2: Login

The login form in Fuel City Express provides a secure interface for users to access their accounts by entering their registered email or username and password.

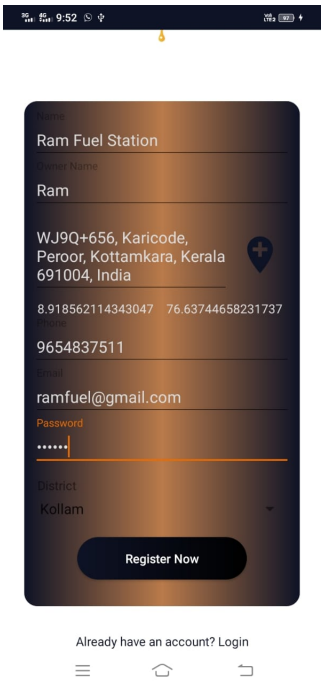


Figure 3.3: Fuel station registration

The fuel station registration form in Fuel City Express allows fuel station owners to submit their details of the fuel station.

3.5 Output Design

Output design is one of the, most important features of the information system. The log- ictal design of an information system is analogous to an engineering blue print of an automobile. It shows the major features and how they are related to one another. The outputs, inputs and databases are designed are in this phase. At the beginning of the output design various types of outputs such as external, internal, operational, and interactive and turnaround are defined. Then the format, content, location, frequency, volume and sequence of the outputs are specified. The content of the output must be defined in detail. The system analysis has two specific objectives at this stage.

- To interpret and communicate the results of the computer part of a system to the users in a form, which they can understand, and which meets their requirements

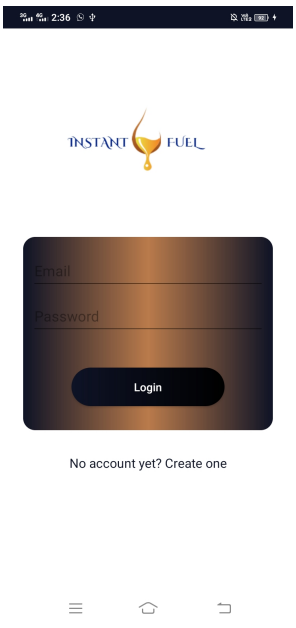


Figure 3.4: Login Form

The login form in Fuel City Express provides a secure interface for users to access their accounts by entering their registered email or username and password.

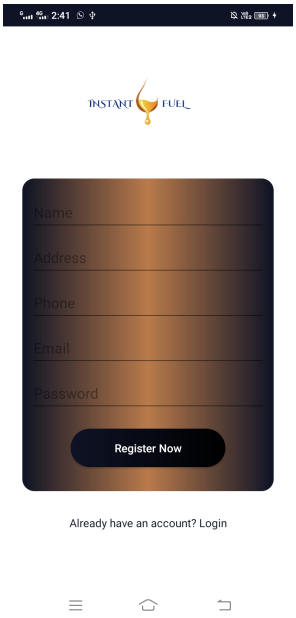


Figure 3.5: User Registration Form

The user registration form in Fuel City Express enables new users to create an account by providing necessary information such as their name, contact details and a secure password.

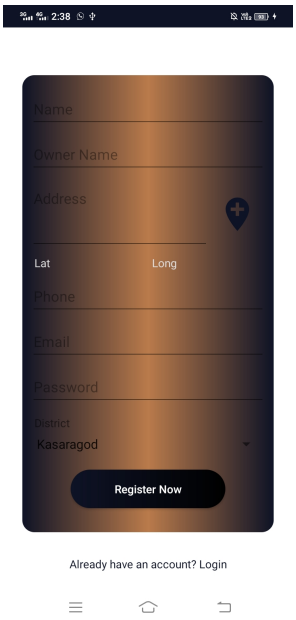


Figure 3.6: Fuel Station Registration Form

The fuel station registration form in Fuel City Express allows fuel station owners to submit their station details, including location, contact information.

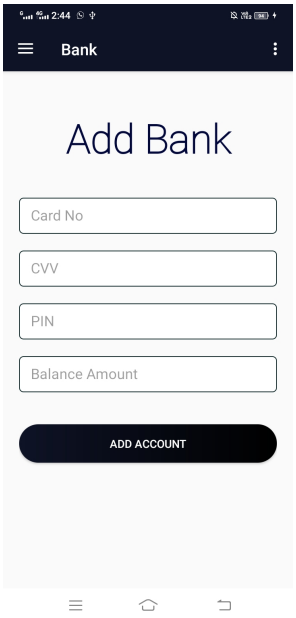


Figure 3.7: Bank Account Form

Add bank form enables users to securely add their bank account details, including account number and associated information, for seamless payment processing during fuel delivery transactions. It ensures that users can conveniently make payments for fuel services directly from their registered bank accounts.

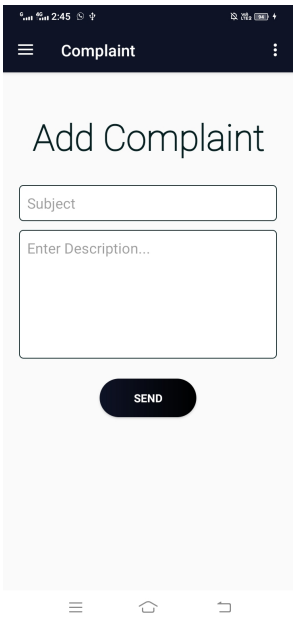


Figure 3.8: Complaint Form

Complaint form allows users to submit complaints or report issues related to their fuel delivery experience, providing a platform for users to express concerns and seek resolution.

3.6 DataBase Design

The Database design is the process of producing a detailed data model of a database. The logical data model contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in a Data Definition Language, which can then be used to create a database. A fully attributes for each entry. The term database design can be used to describe many different parts of the design of an overall database system. Principally, and most correctly, it can be thought of as the logical design of the base data structures used to store the data. In the relational model, these are the tables and views. In an object database, the entities and relationships map directly to object classes and named relationships. However, the term database design could also be used to apply to the overall process of designing, not just the base data structures, but also the forms and queries used as part of the overall data base applications within the database management system.

3.6.1 Tables

Database tables are crucial for organizing and storing data in a structured manner. They allow for efficient data retrieval, manipulation, and analysis, forming the foundation of many applications and systems that rely on data storage and retrieval.

The database tables are,

1.Login Table

The login table typically contains the following fields, each with their corresponding data types:

Field	Data Type	Comment
Lid	int (20) NOT NULL	Primary Key
Email	varchar (100) NULL	
Password	varchar (100) NULL	
Type	varchar (100) NULL	
Status	varchar (100) NULL	

Table 3.1: Login Table

2.Fuel Station Registration Table

The fuel station registration table contains the following fields, each with their corresponding data types:

Field	Data Type	Comment
stationid	int (20) NOT NULL	Primary Key
name	varchar (100) NULL	
owner	varchar (100) NULL	
address	varchar (100) NULL	
phone	varchar (100) NULL	
district	varchar (100) NULL	
email	varchar (100) NULL	
status	varchar (100) NULL	

Table 3.2: Fuel Station Registration Table

3.Customer Table

The customer table contains the following fields, each with their corresponding data types:

Field	Data Type	Comment
customerid	int (20) NOT NULL	Primary Key
name	varchar (100) NULL	
address	varchar (100) NULL	
phone	varchar (100) NULL	
email	varchar (100) NULL	
status	varchar (100) NULL	

Table 3.3: Customer Table

4.Bank Table

The Bank table contains the following fields, each with their corresponding data types:

Field	Data Type	Comment
bankid	int (20) NOT NULL	Primary Key
urid	varchar (100) NULL	
cardno	varchar (100) NULL	
cvv	varchar (100) NULL	
pin	varchar (100) NULL	
balance	varchar (100) NULL	
status	varchar (100) NULL	

Table 3.4: Bank Table

5.Feedback Table

The Bank table contains the following fields, each with their corresponding data types:

Field	Data Type	Comment
feedbackid	int (20) NOT NULL	Primary Key
aid	varchar (100) NULL	
subject	varchar (100) NULL	
description	varchar (100) NULL	
status	varchar (100) NULL	
type	varchar (100) NULL	

Table 3.5: Feedback Table

6.Fuel Request Table

The fuel request table contains the following fields, each with their corresponding data types:

Field	Data Type	Comment
requestid	int (20) NOT NULL	Primary Key
customerid	int (20) NOT NULL	Foreign Key
said	varchar (100) NULL	
fuel	varchar (100) NULL	
restatement	varchar (100) NULL	
status	varchar (100) NULL	
stationrate	varchar (100) NULL	
restate	varchar (100) NULL	
rustling	varchar (100) NULL	

Table 3.6: Fuel Request Table

3.7 Data Flow Diagram

A Data Flow Diagram (DFD) or a bubble chart is a graphical tool for structured analysis.There are various symbols used in DFD. Bubbles represent the process. Named arrows indicate the dataflow. External entities are represented by rectangles and are outside the system such as vendors or customers with whom the system interacts. They either supply or consume data. Entities supplying data are known as sources and those that consume data are called sinks. Data are stored in a data store by a process in the system. Each component in a DFD is labelled in with a descriptive name. Process names are further identified with a number.

DFD can be hierarchically organized, which help in partitioning and analyzing large systems. As a first step, one Data Flow Diagram can depict an entire system. Which gives the system overview? It is called Context Diagram of level 0 DFD. The context diagram can be further expanded. The successive expansion of the DFD from the context diagram that giving more details is known as levelling of DFD. Thus, of top-down approach is used, starting with an overview and then working out the details.

3.7.1 DFD Symbols

Rectangle

A rectangle-shaped symbol is utilized to indicate the origin or end of the Data Flow Diagram. This symbol has rounded corners and is usually captioned with a brief description of the entity it represents.

Figure 3.9: Rectangle

Arrow

A data flow is a route, which enables packets of data to travel from one point to another. Data may flow from a source to a processor and from data store or process. An arrow lines depicts the flow, with arrowhead pointing in the direction of flow.

Figure 3.10: Arrow

Double Sided Arrow

The double-sided arrow in a Data Flow Diagram (DFD) denotes that a process has the ability to both receive and transmit data, indicating a two-way data flow between two processes. This symbol represents the bidirectional exchange of data, allowing data to move in both directions. In simpler terms, it shows that information can flow back and forth between two processes.

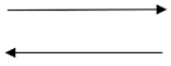


Figure 3.11: Double Sided Arrow

Circle

A Process represents transformation where incoming data flows are changed into outgoing data flows.

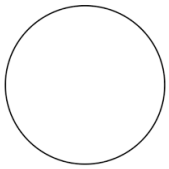


Figure 3.12: Circle

Open Rectangle

It represents the various processes or activities within a system that transform input data into output data. These rectangles are typically labeled with a verb that describes the specific activity or process being carried out by the system.

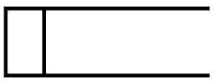


Figure 3.13: Open Rectangle

Oval

The oval shape in a Data Flow Diagram (DFD) is used to indicate the beginning or ending of a process. It shows where data enters or exits the system. The oval symbol is usually labeled with an action verb that describes the specific process that is taking place. It is also commonly known as the process symbol.

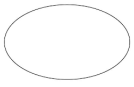


Figure 3.14: Oval

Diamond

The diamond shape serves as a representation of a point in the process where a decision needs to be taken, usually based on certain conditions or criteria that are commonly stated inside the diamond shape.

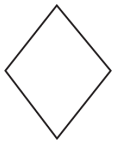
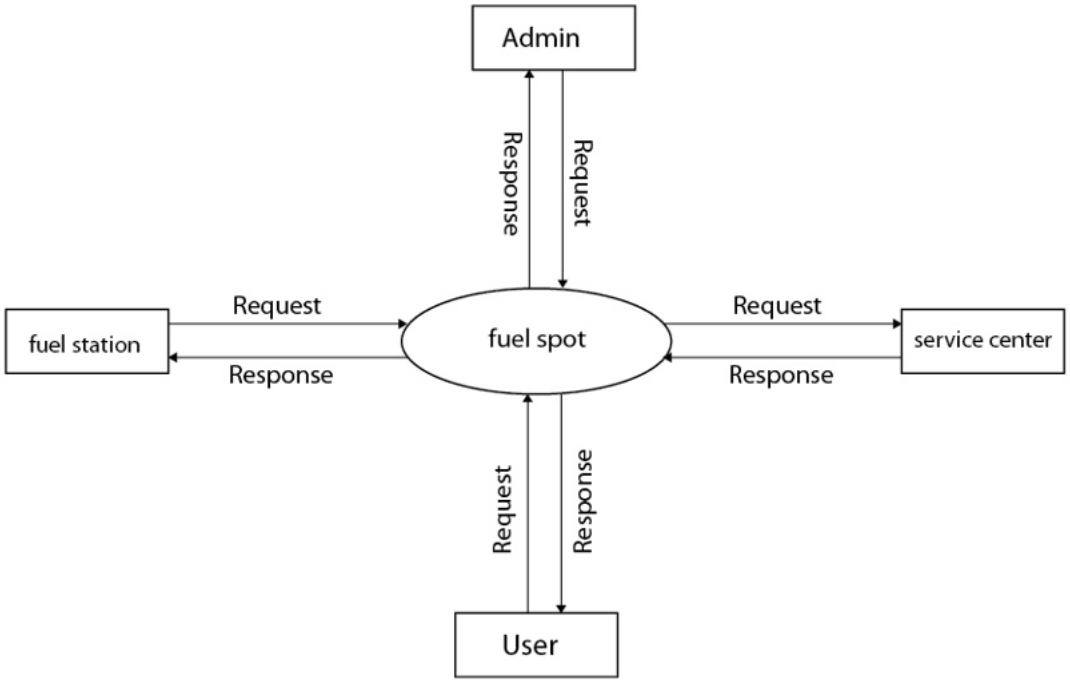


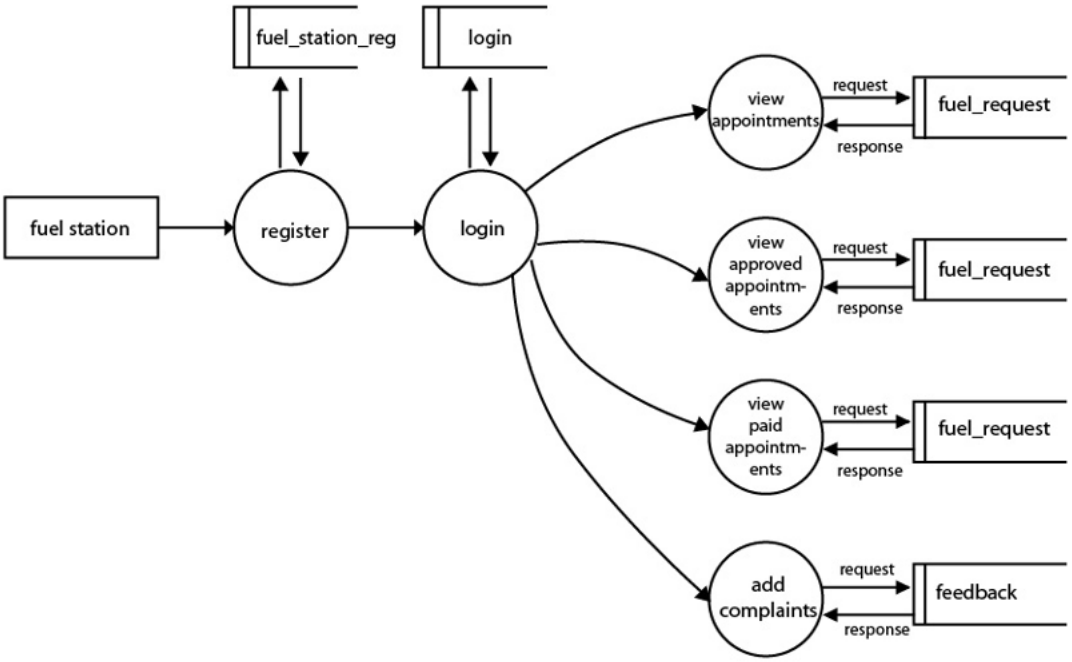
Figure 3.15: Diamond

3.7.2 Data Flow Diagrams

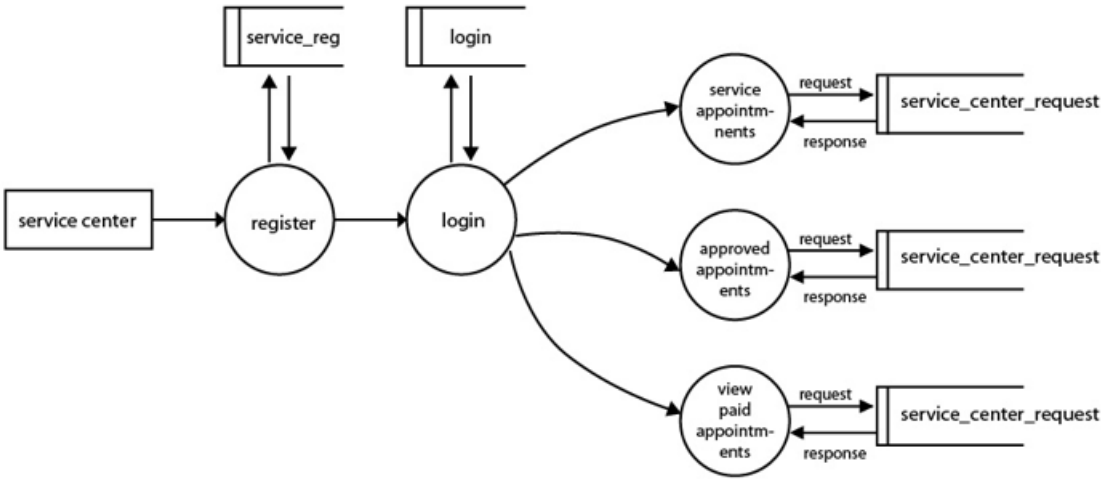
LEVEL 0



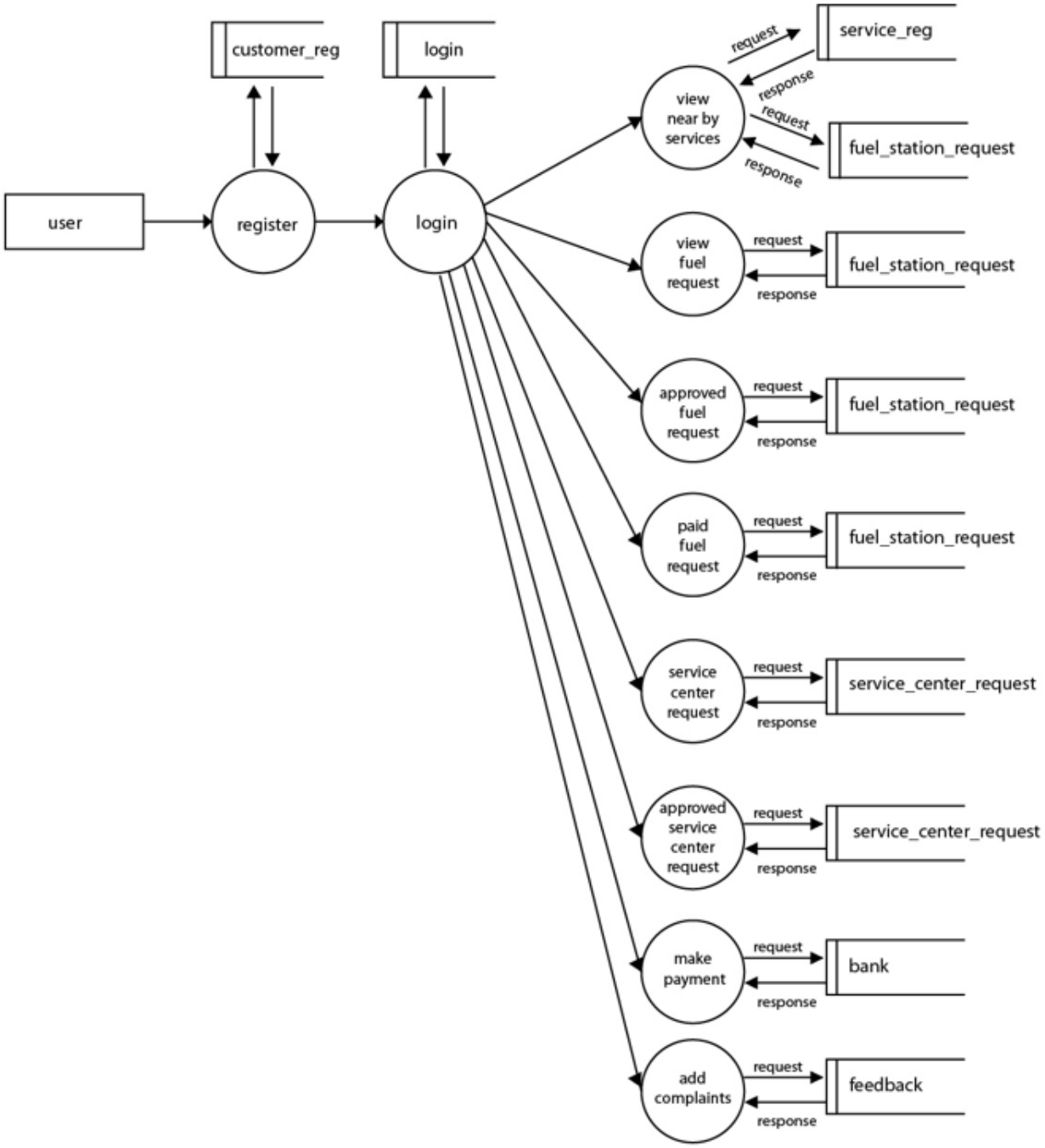
LEVEL 1 Fuel Station



LEVEL 1 Admin



LEVEL 1 User


customer_reg
login
service_reg
request
response
view near by services
request
response
fuel_station_request
user
register
login
view fuel request
request
response
fuel_station_request
approved fuel request
request
response
fuel_station_request
paid fuel request
request
response
fuel_station_request
service center request
request
response
service_center_request
approved service center request
request
response
service_center_request
make payment
request
response
bank
add complaints
request
response
feedback

Chapter 4

RESULT AND DISCUSSION

The primary quality control method used in software development is testing. Following the coding stage, testing purposes are served by running the accessible computer programmes. Testing must find flaws made during the earlier phase as well as those introduced during development. So, the purpose of testing is to find programme requirements, design, or coding flaws.

- A programme is tested by being run with the goal of identifying any errors.
- A excellent test case is one that has the highest chance of spotting an error that hasn't been identified yet.
- A test that finds an error that hasn't been found yet is successful.

Our objective is to develop tests that systematically uncover many sorts of issues with minimal time and effort. Testing indicates that software functionalities appear to operate as expected and that performance criteria appear to have been met. The information acquired during testing is an excellent predictor of programme reliability and a partial indicator of software quality as a whole. Testing has one drawback, however: it can only demonstrate the presence of software defects, not their absence.

4.1 Testing

Testing is an important stage in the software development life cycle. System testing is a critical element of a software quality assurance and represents the ultimate review ofspecification, design and coding. Importance of software testing and its implication with software quality cannot be over me- phased. Testing is one-way developers can validate the quality of a software product and verify that it fully meets the specification. During testing, the system is tested with a set of cases and checked whether the input of the program is performing as it is expected. The system tested and reviewed to ensure that the entire user requirement has being satisfied. Testing was done throughout the system development at various stages since it is always a good practice to test the system at many different levels at various intervals that issub systems, program modules as work progress and finally the system as a whole. If this is not done, then the poorly tested system can fail after installation. Testing is a very tedious and time-consuming job. For a test to be successful the tester should try and make the program file. Each test is designed with the intention of finding errors in the way system will process it. Though testing of a program doesn't guarantee the reliability of the system, it is done to assure that the system runs errors free. The Testing process begins by developing a comprehensive plan to test the general functionlaity and special features on a variety of platform combinations. Strict quality control procedures are used. The Process verifies that the application meets the requirements specified in the system requirements document and is bug free. At the End of each testing day, the summary of completed and failed tests is prepared. And the Application is redeveloped and retested until every item is resolved. All the changes and retesting are tracked through spread sheets. Applications are not allowed to launch until all identified problem are fixed. Finally, a report is prepared at the end of testing to show exactly what was tested and to list the final outcomes. The software testing methodology is applied in four distinct phases:

- Functional Testing
- Usability Testing
- Performance Testing
- Compatibility Testing
- Integration Testing

4.1.1 Functional Testing

The registration process for pumps is thoroughly verified to ensure that accurate and complete information, such as name, license number, and location, can be provided. This step ensures that pumps can successfully register on the site. The search functionality is then tested to guarantee that customers can accurately locate the nearest pump based on their own location. By validating this feature, users can easily find the most convenient pump for their needs. Another crucial aspect is validating the process of sending a fuel request to the nearest pump. This testing ensures that the request is properly processed and reaches the intended pump without any issues. Lastly, the handover functionality is thoroughly tested to verify that requests are effectively passed on to the next nearest pump when the initially contacted pump is unable to handle the request. By performing these tests, it provide a seamless and efficient experience for both pump registration and the fuel request process, ensuring a reliable service for users in need of fuel in various situations.

4.1.2 Usability Testing

In the testing of the application, particular attention is given to evaluating the user interface for ease of use, clarity, and intuitiveness. This involves assessing the overall design and layout of the application, as well as the intuitiveness of navigation and interaction elements. The registration process and fuel request process are thoroughly tested to ensure that they are straightforward and user-friendly, allowing users to easily complete these tasks without confusion or complications. Additionally, the application is validated to ensure that it provides clear and relevant information to users regarding pump locations and availability. This includes displaying accurate and up-to-date information about nearby pumps, their operating status, and any relevant details users may need. Feedback from users is actively sought to identify any usability issues and areas for improvement. By gathering feedback and incorporating user perspectives, the application can continually enhance its user interface, making it more intuitive, user-friendly, and efficient for users seeking fuel in emergency situations.

4.1.3 Performance Testing

The application is tested to ensure it can handle the expected user traffic, both during normal usage and peak load situations. This testing allows for the identification of any performance

bottlenecks or issues that may arise when a large number of users access the application simultaneously. The response time of the application is thoroughly evaluated during various operations, such as registration, search, and fuel request processing. This ensures that users experience optimal speed and efficiency when interacting with the application. Additionally, resource usage is measured, including CPU, memory, and network utilization, to ensure efficient utilization of system resources. This testing helps identify any resource-intensive processes or areas where optimization is needed, ensuring smooth performance and a seamless user experience. By conducting these performance tests and measuring resource usage, the application can deliver reliable and efficient performance, even during periods of high user demand.

4.1.4 Compatibility Testing

This testing phase aims to verify that the application functions correctly across a range of devices, ensuring a consistent user experience. Additionally, compatibility testing includes testing the application on different versions of Android and various web browsers to ensure seamless functionality across different platforms. By conducting these tests, any potential issues or inconsistencies can be identified and addressed, ensuring that users can access and use the application regardless of their device or preferred browser. Furthermore, the integration with Google Maps is thoroughly tested across different platforms to verify proper functionality. This testing ensures that location mapping and related features work reliably, providing users with accurate information regarding pump locations. By testing for compatibility and integration across different devices, screen resolutions, platforms, and Google Maps functionality, the application can provide a consistent and reliable experience for users across various environments.

4.1.5 Integration Testing

This testing verifies that users can rely on the application to provide correct and up-to-date information regarding pump locations. Secondly, the integration with external systems, such as the database for storing pump and request data, is verified. This ensures that data is seamlessly synchronized and stored correctly, allowing for efficient retrieval and processing.

4.2 Output Screens

Admin will have access to the following pages.

- Splash Page :

  This is the main landing page. It is mainly a splash page to see the logo of the application.

- Login Page

  Login as an admin by entering your email and password. Once authenticated, you will gain access to the admin pages.

- Welcome page

  This is the welcome page of the application. There are two choices available - fuel station and user. We have the flexibility to choose between them

- Admin Dashboard

  This is the main landing page for Admin. To view the admin dashboard, we need to click on the icon situated at the top left corner.

- View Users

  The admin has the ability to see a comprehensive list of all users registered on the platform, including their profiles

- Verify Fuel Station

  As part of their responsibilities, the admin can perform a verification process on fuel stations to ensure that they comply with specific requirements, including licensing and safety regulations. The admin can then take appropriate actions such as approving, rejecting, or removing the fuel station from the platform based on their verification.

- View Users Complaints

  The admin is authorized to review and address any complaints submitted by users on the platform. They can investigate the complaints, communicate with the users, and resolve the issues accordingly. Additionally, the admin can monitor and analyze the complaints to identify any potential trends or recurring problems that need to be addressed.

- View Fuel Station Complaints

  The admin is authorized to review and address any complaints submitted by users against fuel stations on the platform. They can investigate the complaints, communicate with the fuel station and user, and take necessary actions to resolve the issue and ensure user satisfaction. By monitoring these complaints, the admin can identify any recurring problems with fuel stations and implement measures to prevent them from happening again.

  User will have access to the following pages.

- Splash Page :

  This is the main landing page. It is mainly a splash page to see the logo of the application.

- Login Page

  Login as an user by entering your email and password. Once authenticated, you will gain access to the user pages.

- User Registration Page

  In case the user doesn't possess an account, there exists a provision to create one. Following the selection of the 'user' option, the user can proceed with the registration process.

- User Dashboard

  This is the main landing page for User. To view the admin dashboard, we need to click on the icon situated at the top left corner.

- Current Location

  This is the main landing page for User. To view the current location and nearest fuel stations in the map.

- Nearby Service Page

  By accessing the nearby service page, we can observe the service stations that are located in proximity to our current location

- Fuel Station List in the User Account

  The page displays a comprehensive list of fuel stations that have been approved by the admin.

- Fuel Requested

  This page displays the fuel requests that have been approved by the admin, including both past and present requests.

- Approved Fuel Appointments

  The approved fuel appointments for the fuel station can be viewed on this page.

- Paid Fuel Appointments

  Users can access the paid fuel appointment page to view a comprehensive record of their appointments for which they have already paid online. This includes appointments where the fuel station charged the user the corresponding fuel amount upon request

- Add user bank account

  Users can conveniently pay for their fuel expenses by adding their bank account information for online transactions

- Add user Complaints

  Users have the option to register complaints on the platform, and the admin will address them accordingly.

  Fuel station will have access to the following pages.

- Splash Page :

  This is the main landing page. It is mainly a splash page to see the logo of the application.

- Login Page

  Login as a fuel station entering the email and password. Once authenticated, you will gain access to the fuel station pages.

- Fuel Station Dashboard

  Dashboard in a fuel station can improve customer service by offering customers easy access to essential services, while allowing managers to efficiently manage various tasks.

- Add employee and view employee

  The fuel station administrator has the ability to add new employees to the system, which involves providing their personal information, job title, and other pertinent details.

  The fuel station administrator can access a list of employees who are currently registered in the system, which includes their job titles and other pertinent information.

- View Appointments

  This page displays a list of all appointments made by customers to refill their vehicles at the fuel station, including necessary details.

- Requested Appointments

  Requested appointments page of the fuel station dashboard displays fuel refill requests submitted by users

- Paid Appointments

  This page displays a list of appointments where the customers have paid for the fuel refilling service.

- Compliant Page

  This feature allows fuel station manager to file a complaint if they experience any issues while refilling vehicles.The administrator can view these complaints and take appropriate actions to resolve them.

Chapter 5

CONCLUSION

Fuel City Express app has addressed the main problems faced by individuals stranded with no petrol station nearby or those who have no time to drive to the gas station. With the app, users can conveniently order fuel via their phones and have it delivered right to their location, ensuring a seamless journey. This app is not just convenient but also an innovative solution to the age-old problem of refueling on the go. Fuel City Express has demonstrated the potential of technology to make our lives easier, safer and more efficient. It is a step towards a more convenient and sustainable future.

5.1 Future Enhancement

Fuel City Express has a great potential for growth and development in the future. One option is to broaden its range of services to incorporate other fuel types such as diesel or electric. Another possibility is to integrate with other transportation apps to offer a more complete solution to its users. The app's appeal to its users could be enhanced by improving its user interface and user experience. Additionally, offering more payment options and integrating with more payment systems would be a valuable improvement. Finally, by utilizing data analytics, the app could gain insights into user preferences and behavior, resulting in more efficient and personalized service.

REFERENCES

[1] Battin, Pradnya, and S. D. Markande. ”Location based reminder Android application using Google Maps API.” 2016 International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT). IEEE, 2016.

[2] Pejic, A., Pletl, S., Pejic, B. (2009, September). An expert system for tourists using Google Maps API. In 2009 7th International Symposium on Intelligent Systems and Informatics (pp. 317-322). IEEE.

[3] Li, Xiaosong, Sai Karnan, and Jahanzaib Ali Chishti. ”An empirical study of three PHP frameworks.” In 2017 4th International Conference on Systems and Informatics (ICSAI), pp. 1636-1640. IEEE, 2017.

[4] Bai, Ying. ”Introduction to Apache NetBeans IDE.” SQL Server Database Programming with Java: Concepts, Designs and Implementations. Cham: Springer International Publishing, 2022. 143-178.

[5] Liu, J., Yu, J. (2011, November). Research on development of android applications. In 2011 4th International Conference on Intelligent Networks and Intelligent Systems (pp. 69-72). IEEE.

[6] Latif, Mounaim, Younes Lakhrissi, and Najia Es-Sbai. ”Cross platform approach for mobile application development: A survey.” 2016 International Conference on Information Technology for Organizations Development (IT4OD). IEEE, 2016.

[7] Wanganoo, Leena, and Anukul Patil. ”Preparing for the smart cities: IoT enabled last-mile delivery.” 2020 Advances in Science and Engineering Technology International Conferences (ASET). IEEE, 2020

[8] Shibghatullah, Abdul S., et al. ”Vehicle tracking application based on real time traffic.” International Journal of Electrical and Electronic Engineering Telecommunications 11.1 (2022): 67-73.

[9] Vuksanovic, Irena Petrijevcanin, and Bojan Sudarevic. ”Use of web application frameworks in the development of small applications.” 2011 Proceedings of the 34th International Convention MIPRO. IEEE, 2011.

[10] Sarkar, Anirban, et al. ”Android application development: a brief overview of android platforms and evolution of security systems.” 2019 Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC). IEEE, 2019.

APPENDIX

Screenshots

[redacted]

Figure A.1: Splash Page

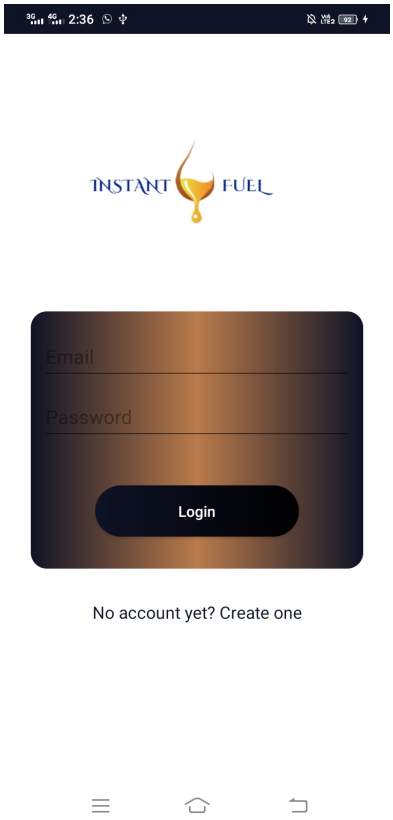


Figure A.2: Login Page

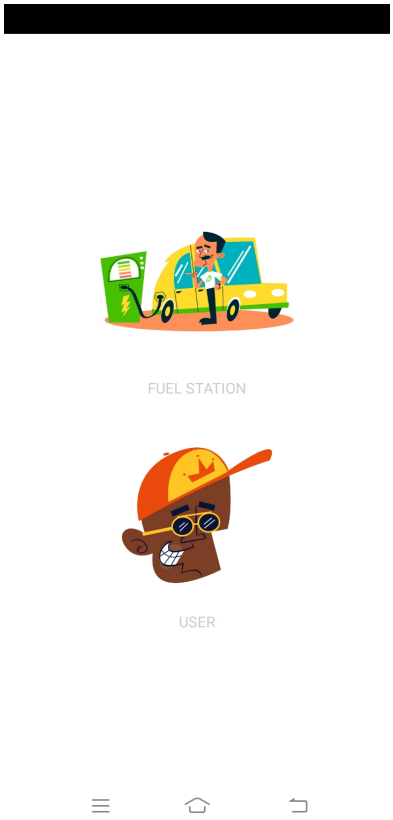


Figure A.3: Welcome Page

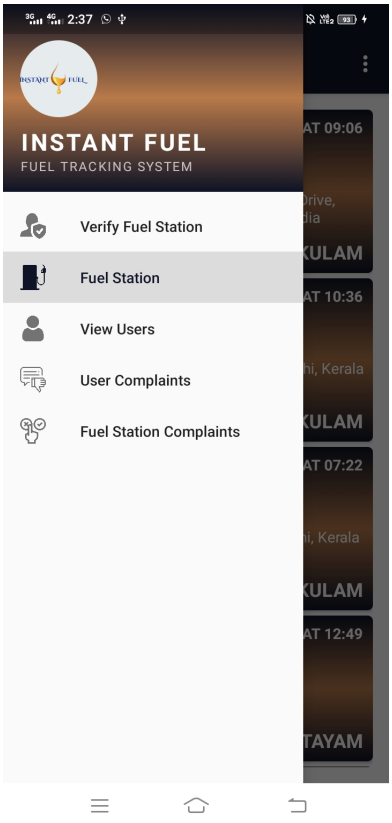


Figure A.4: Admin Dashboard

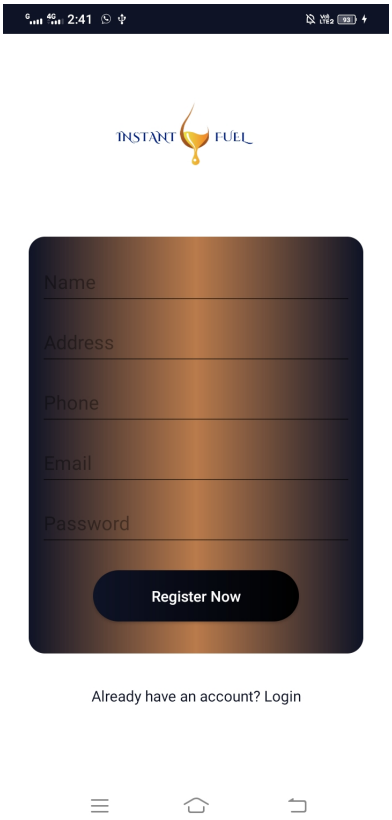


Figure A.5: User Registration Page

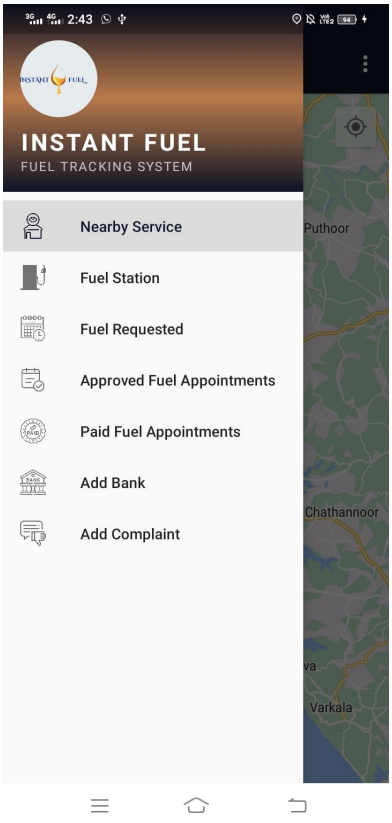


Figure A.6: User Dashboard

Figure A.7: Fuel stations List

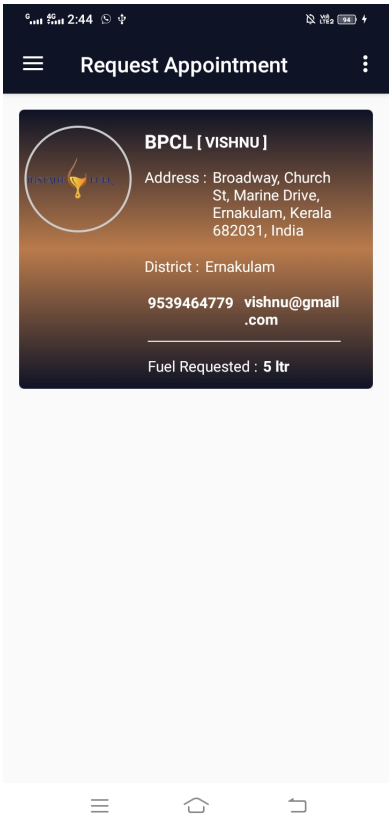


Figure A.8: Fuel Request Appointment

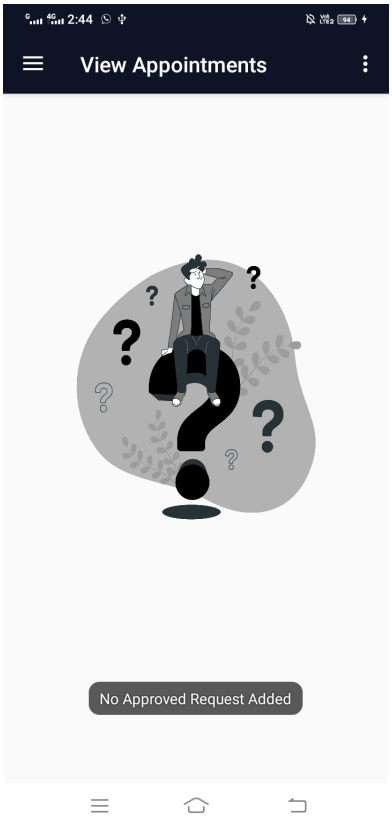


Figure A.9: View Appointments Page

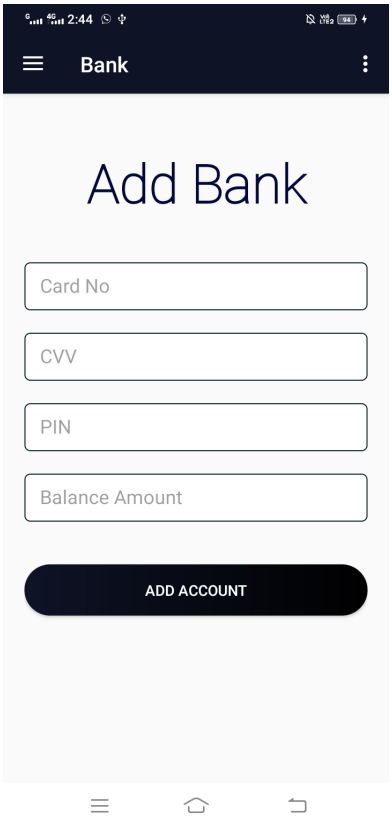


Figure A.10: Add User Bank Account

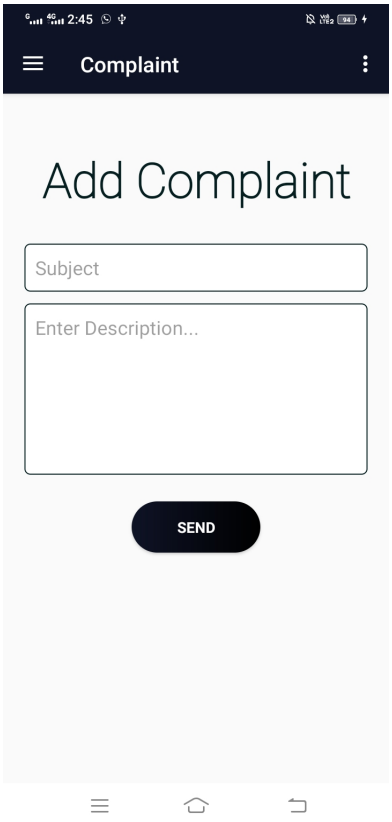


Figure A.11: User Complaint Page

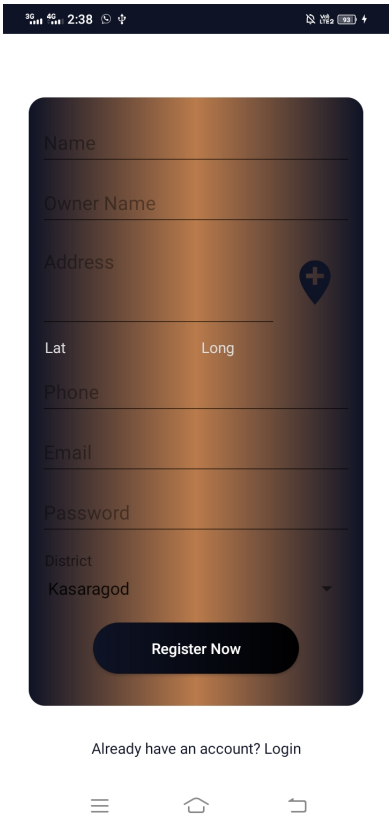


Figure A.12: Fuel Station Registration Page

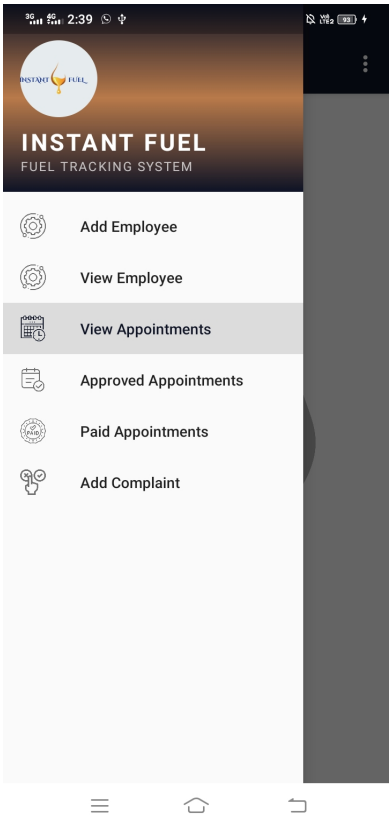


Figure A.13: Fuel Station Dashboard

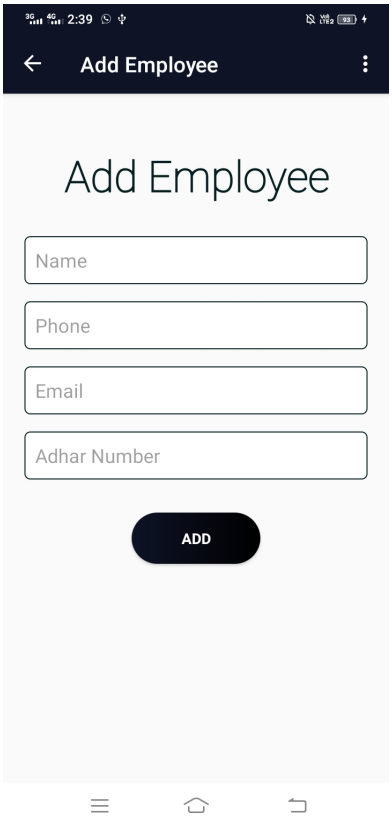


Figure A.14: Add Employee

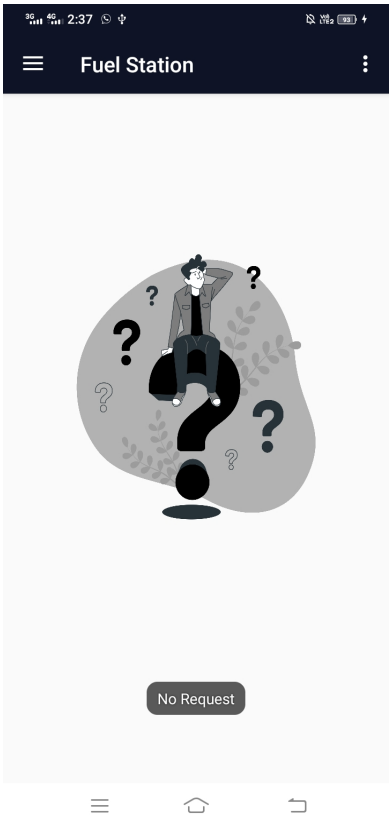


Figure A.15: View Request of user

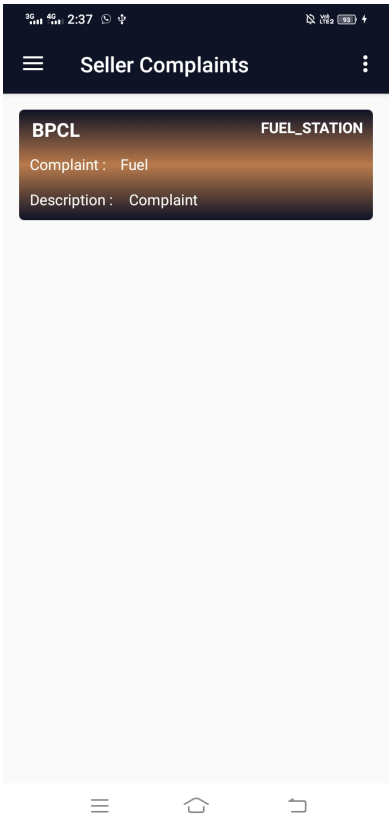


Figure A.16: Seller Complaint

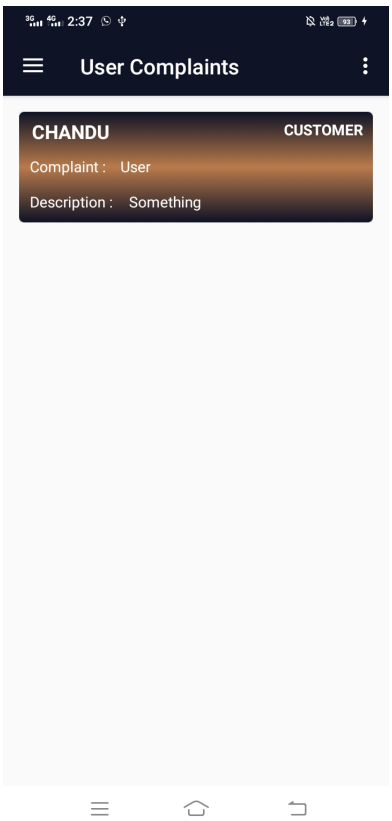


Figure A.17: User Complaint