LONDON
METROPOLITAN
UNIVERSITY

ESOFT
Shaping Lives, Creating Futures.

Advance Software Engineering

Group Members :

Supervisor : Mis. Jiwanthi Marasinghe

Date : February.16.2023

Scenario

FuelIn is a responsible public company establish in Sri Lanka where they distribute and manage fuel Supply Island wide. They have several registered fuel stations at all the districts. FuelIn wishes to introduce an online fuel requesting and que management system for the requirement of fuel distribution and tracking. End consumers are to be given facilities to request a fuel based on the quota given for the vehicle registration number. The requesting customers can obtain a token with expected filling time and date with 3 hours of tolerance. If the fuel distribution cannot made available, the customer will be given a new scheduled period. If there is no any scheduled delivery to a particular filling station or scheduled fuel stock is finished, the necessary message will be given and requests are not permitted from those fuel stations. Once the delivery schedule to the given filling station is confirmed by the dispatch office in the head office, notification SMS/email will be sent to all token holders to make the payment. If the customer unable to satisfy the payment and obtain fuel, quota will be deducted. As tokens are issued only for the scheduled delivery amount of liters, the non-committed or unpaid fuel stocks can be made available to request on demand after the scheduled payment time, those tokens are issued for the same day collection. End consumers can take a token from the filling station also using the same system when scheduled delivery is available. Twelve hours before the delivery happens to the filling station, end consumers will receive sms/email as a reminder with amount requested. Each vehicle receive a new quota after in each week’s period. The filling station orders are to be taken separately using the system. The fuel stocks are to be distributed according to population density within the area of filling stations

Acknowledgement

The work on this assignment has been an inspiring, often exciting, sometimes challenging, but always an interesting experience. This assignment would not have been possible without the support of many people. First, I would like to express my deep and sincere thankfulness to my lecturer, Ms. Jiwanthi Without his guidance and support this assignment cannot be done properly like this. He always gives us her support and, guides us on doing our assignment in purpose to produce a good outcome. And I take this opportunity to thank my institute, ESOFT Metro Campus Kurunegala, for providing us with the necessary facilities. Especially I would like to thank my parents, family members and colleagues for whom I have great regard and I wish to extend my beneficent thanks to all those who have helped me.

Thank you

Abstract

FuelIn wishes to introduce an online fuel requesting and que management system for the requirement of fuel distribution and tracking... This system is enabled by the internet it is the internet that connects a company on one hand, and the customer on other hand. This report is based on the project about the development of the online fuel requesting and que management system for the requirement of fuel distribution and tracking. A brief description about the project is cited below.

Initially, the background of the project and the company environment had been explained, which include a bit about the company itself, the aim of the company and the scope of the project. After that, the deliverables provided with the system are started. Afterwards, the project constraints and the system specification are started. Afterwards, the project constraints and the system specification are started and explained. At the end of the introduction section, features and limitations of the existing system and the advantages of the caused by the proposed system are mentioned.

Project management falls under the next chapter, where the project management process and the project management types are explained. The chapter after that is the analysis. In this chapter the literature review, as in which other system were examined prior to developing the hotel reservation system has been started, which also includes the requirements analysis, feasibility study carried out for the system followed by the SWOT analysis carried out for the system. The use case diagram, the ER diagram, and the class diagram falls under the next chapter, which is system design.

The next, as in the fourth chapter, include the screenshots of the system developed for the requirement of fuel distribution and tracking system. The design of the project is very associated with object-oriented concepts with the utilization of UML (Unified Modeling Language). The system has been developed with python. And the database is handled by MySQL. The chapter after that is testing and evaluation, which includes an explanation of the testing strategies, testing levels and so on. This chapter also include all the tests held for the system including functionality testing, usability testing, interface testing, performance testing, compatibility testing, and performance testing security testing and so on.

Finally, a conclusion of the whole system has been included to justify the effort of the team and conclude that all the specified user requirements are successfully included in the system.

Introduction

A reputable public company called FuelIn with headquarters in Sri Lanka has decided to roll out an online fuel ordering and queue management system. This system is made to make it easier to track and distribute gasoline supply over the island and to give end users a quick and easy way to order fuel. The system consists of two parts: a client web application with a database server and a web server with a database. For data operations, the mobile app can make use of the same application services that the server offers.

Objectives:

This system's primary objective is to provide customers with an easy way to request fuel and receive a token with an estimated fill time and date. As soon as the customer pumps the fuel, the station managers should be able to check the tokens and information about their vehicles, marking the status of their requests. Based on the fuel distribution for the outlets, the head office should also be able to see the status of each outlet.

1. Multiple registrations for the same car number should not be permitted. Consumers should be able to register in the system.

2. Managers of gas stations should be able to confirm the validity of the tokens and the specifics of the customers' vehicles.

3. As soon as the fuel is pumped, filling station management should be able to record the requests' status.

4. Based on the allocation of fuel for the outlets, the head office should be able to see the status of each outlet.

Problem Statement

FuelIn, a responsible public company in Sri Lanka, faces challenges in managing and distributing fuel efficiently to its registered fuel stations island-wide. Fuel distribution is inefficient since the company lacks a centralized mechanism to order and monitor fuel delivery. There is no way for clients to get a token with the anticipated filling time and date, so they cannot request fuel based on the limit assigned for their car registration number. The lack of a reliable tracking system makes it difficult to plan gasoline delivery and make sure that consumers can pay for their fuel and get it when they need it. Customers become dissatisfied as a result, and fuel distribution is inefficient.

FuelIn needs a solution that can effectively manage the fuel distribution and tracking process, and provide end consumers with the necessary facilities to request fuel based on their quota. The solution should also provide a mechanism for customers to obtain a token with the expected filling time and date and allow for the scheduling of fuel deliveries. A reminder system should also be in place to ensure that customers can pay for and obtain their fuel within the expected time. The solution should be able to provide real-time data on fuel distribution and tracking and allow the head office to view the status of each outlet.

Project deliverables

1. Web application: An easy-to-use web application that enables end users to sign up, request fuel, and receive a token with an anticipated filling time and date
2. Mobile app: End users can make fuel requests through a mobile app, get tokens in return, and check the status of their requests while on the road.
3. Database: To manage and store customer registration data, fuel orders, payment data, delivery schedules, and other pertinent data, a safe database must be used.
4. Admin panel: An administration panel enables filling station managers to record the status of requests, monitor the status of each outlet, and verify tokens and information about customers' vehicles.
5. Head office portal: A portal that allows the head office to track each outlet's status and control gasoline distribution based on the population density in the vicinity of the filling stations.
6. Notifications: Automated SMS/email messages to end users about the status of their requests, reminders for payments, and delivery dates.
7. A thorough user manual is provided to help end users and station managers operate the system.
8. Technical documentation: Comprehensive technical documentation for the system, with code samples, architecture diagrams, and data flow diagrams included.
9. Testing and quality assurance: Thorough testing and quality assurance to ensure the system meets the outlined requirements and performs optimally.
10. Deployment and maintenance: Deployment of the system to production servers and ongoing maintenance to ensure the system continues to perform optimally over time.

Purpose

The online fuel ordering and queue management system is designed to streamline the tracking and distribution of fuel for FuelIn, a Sri Lankan fuel distribution company. The system intends to offer customers a quick and easy way to request fuel based on their quota and receive a token with the anticipated time and date of filling. Additionally, the system gives managers of gas stations the ability to check tokens, track requests, and see how each outlet is doing. Based on the population density in the vicinity of the filling stations, the head office can monitor the condition of each outlet and control fuel distribution. The system streamlines and simplifies the fuel distribution process, improving the overall customer experience and ensuring fuel efficient and fair distribution.

Project Scope

- Objectives: The main objective of this project is to design and create an online and mobile application that will let users order fuel, get a token with an estimated time and date of delivery, and check the status of their order. While the head office may observe each outlet's status and control fuel distribution depending on population density in the filling station area, the system also gives filling station managers the ability to check tokens, mark the status of requests, and view each outlet's status.
- Functionality: The system will provide end consumers with the ability to register and request fuel based on their quota, receive a token with an expected filling time and date, and view the status of their request. Filling station managers will be able to verify tokens and details of consumers' vehicles, mark the status of requests, and view the status of each outlet. The head office will be able to view the status of each outlet and manage fuel distribution based on population density within the area of the filling stations.
- User Roles: The system will have three user roles: end consumers, filling station managers, and head office administrators. Each role will have specific functionality and permissions within the system.
- Data Management: The system will store and manage consumer registration information, fuel requests, payment information, delivery schedules, and other relevant data in a secure database.
- Automated Notifications: The system will send automated SMS/email notifications to end consumers regarding their request status, payment reminders, and delivery schedules.
- Technology Stack: The system will be developed using a modern technology stack, including HTML, CSS, and JavaScript. A database management system such as MySQL or MongoDB will be used for data storage.
- Testing and Quality Assurance: To make sure the system satisfies the above standards and operates at peak efficiency, it will go through extensive testing and quality assurance.
- Deployment and Maintenance: To guarantee the system continues to function at its peak over time, it will be deployed to production servers and will receive regular maintenance.
- Timeline: The client and the development team will choose the project's timeline after discussing the project's scope, budget, and resources.
- Budget: The client and the development team will decide on and agree upon the project budget while considering the project's scope, timetable, and resource availability into account.

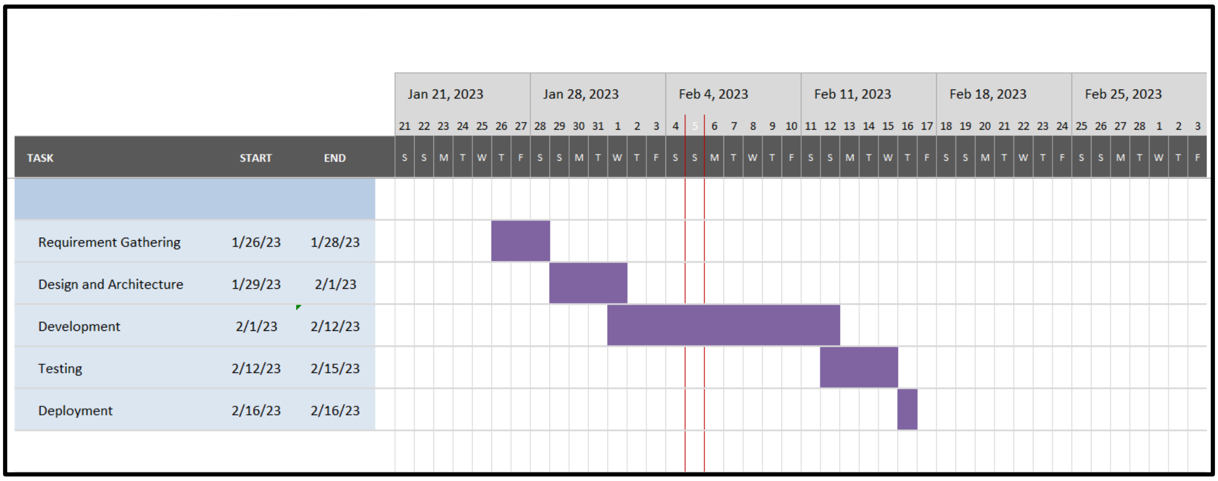
Methodology

A combination of Waterfall and Agile project management techniques may be used for the FuelIn online fuel ordering and queue management system project methodology.

1. Initiation: The project charter, which describes the project objectives, timeframe, budget, and stakeholders, is created during this phase. The project is started along with the formation of a project team.
2. Gathering requirements: The project team will collect requirements from all parties involved during this phase, including the head office, filling station managers, and end users. This will make it easier to make sure that the finished system satisfies the requirements of all parties.
3. Design: During this phase, the system architecture and design will be created and documented. This will include the user interface design, database design, and data flow diagrams.
4. Development: The online and mobile applications will be created, tested, and connected to the database during this stage.
5. Testing: To make sure the system is operating as intended, a thorough test will be conducted at this step. Any problems discovered during testing will be dealt with and fixed.
6. Deployment: During this stage, the system will be put into use and deployed to the production environment. To guarantee that the system is being utilized properly, the project team will offer training and support.
7. Monitoring and maintenance: The system will be checked to make sure it is operating properly after deployment. Any problems that develop will be dealt with and fixed.

Throughout the project life cycle, regular check-ins and retrospectives will be conducted to assess the project's progress and to make any necessary adjustments. This combination of Waterfall and Agile methodologies will help to ensure that the project is delivered on time and within budget, while also allowing for flexibility and adaptability as requirements evolve.

Timeline



TASK	START	END
Requirement Gathering	1/26/23	1/28/23
Design and Architecture	1/29/23	2/1/23
Development	2/1/23	2/12/23
Testing	2/12/23	2/15/23
Deployment	2/16/23	2/16/23

1. Requirements Gathering: This phase involves understanding the system requirements, including business requirements, functional requirements, and non-functional requirements.
2. Design and Architecture: In this phase, the system design and architecture are developed. This includes the design of the user interface, the database structure, and the overall architecture of the system.
3. Development: During this phase, the actual development of the system takes place. This includes writing code, testing, and fixing any bugs.
4. Testing: In this phase, the system is thoroughly tested to ensure it meets the requirements and works as intended.
5. Deployment: Once the system has been thoroughly tested and all bugs have been fixed, it can be deployed for use.
6. Maintenance and Support: After deployment, the system will need ongoing maintenance and support to ensure it continues to work as intended and to address any issues that arise.

Proposed system

The proposed system, FuelIn, is designed to meet the needs of a responsible public company that distributes and manages fuel supplies across Sri Lanka. The system aims to provide an online platform for consumers to request fuel based on the quota assigned to their vehicle registration number. The consumers will receive a token with the expected filling time and date, along with a 3-hour tolerance period. If the delivery of fuel cannot be made available, the customer will be given a new scheduled period. If there is no scheduled delivery to a particular filling station or the scheduled fuel stock is finished, customers will be informed and requests will not be permitted from those fuel stations.

The FuelIn system will consist of two components: a web server with a database and a client web application with a database server. A mobile app will also be available, using the same application services provided by the server for data operations. The consumers will be required to register in the system, with multiple registrations per vehicle number prevented. The filling station manager will be able to verify the tokens and details of the consumers' vehicles and mark the status of the requests upon pumping the fuel. The head office will be able to view the status of each outlet based on the distribution of fuel for the outlets. The deliverables for the project may include a functional web server, client web application, mobile app, user documentation, and training for end-users. The purpose of the system is to provide an efficient and reliable platform for the management of fuel distribution and tracking, allowing customers to request fuel and receive notifications regarding their delivery.

Advantages of system

1. Convenience for customers: The system offers customers a simple way to order fuel and receive a token with the time and date of the anticipated fill-up.

2. Enhanced productivity: The system streamlines the distribution and monitoring of fuel supplies, resulting in a decrease in human work and a boost in productivity.

3. Real-time visibility: Based on the distribution of gasoline for the outlets, the head office may see in real-time how each outlet is doing.

4. Accurate tracking: The system tracks fuel orders, payments, and delivery times correctly, lowering the possibility of human error and inconsistencies.

5. Improved customer service: By ensuring that consumers receive timely and effective service, the system helps to lower the risk of frustration and discontent.

6. Enhanced transparency: The system enhances accountability and trust by increasing transparency in the distribution and management of fuel supply.

7. Time and money savings: By automating many manual operations and lowering the demand for human labor, the system saves time and lowers expenses.

8. Better fuel management: Based on population density in the vicinity of the filling stations, the system helps to optimize gasoline distribution.

Aim and Objectives

1. To roll out an online fuel ordering and queue management system for tracking and distributing fuel.
2. To give customers the option to order fuel based on the number of their vehicle's registration.
3. To send consumers tokens that include information about predicted fill times and dates as well as alerts for payment via SMS and email.

4. To monitor and control the fuel delivery to the filling stations based on population density.
5. To stop repeated registrations using the same car number and to authenticate vehicle details and tokens for customers.
6. To enable fueling station managers to categorize requests after fuel is pumped.
7. To make it possible for the main office to view each outlet's status.
8. To set up a mechanism that will allow customers to request a token from the gas station. To send a reminder SMS/email to end consumers twelve hours before the delivery.
9. To provide a new quota to each vehicle every week.
10. To allow filling station orders to be taken separately using the system.

Project Management

Project management process

The project management process for the FuelIn system will involve several key steps to ensure the successful development and deployment of the system.

First, a project charter should be created to define the scope, goals, stakeholders, and timeline for the project. This will serve as a contract between all parties involved and provide a clear understanding of expectations for the project.

Next, a project team will be assembled, including individuals with the necessary skills and expertise to develop the web and mobile apps, manage the database, and provide technical support. The project manager will lead the project team and be responsible for ensuring that the project is completed on time and within budget.

The project management process will also involve creating a detailed project plan, including tasks, timelines, and resource requirements. This plan will be used to monitor progress, track costs, and make any necessary adjustments to keep the project on track.

The methodology used for the project will play a critical role in its success. Agile project management methodologies, such as Scrum or Kanban, may be used to allow for flexible and iterative development of the system.

The project will also require a robust risk management plan to identify and mitigate potential risks to the project timeline and budget. Finally, the project will require ongoing communication and collaboration between the project team and stakeholders to ensure that the system is being developed according to requirements and that any issues are addressed promptly.

The project management process for FuelIn can involve several stages, including initiation, planning, execution, monitoring and controlling, and closing. The following is a high-level description of each stage:

- Initiation: This stage involves defining the scope and objectives of the project, as well as identifying stakeholders and key resources. This stage may also involve conducting a feasibility study to determine if the project is viable and aligns with the company's overall strategy.
- Planning: This stage involves creating a detailed project plan, including a project timeline, budget, resource allocation, and risk management plan. The project plan should outline the tasks, milestones, and deliverables that will be necessary to complete the project.
- Execution: This stage involves putting the project plan into action, including assembling the project team, allocating resources, and executing the various tasks and activities outlined in the project plan.
- Monitoring and controlling: This stage involves regularly monitoring and evaluating the project's progress to ensure it stays on track and within budget. This stage may also involve making adjustments to the project plan as needed to address any issues that arise.
- Closing: This stage involves wrapping up the project, including documenting lessons learned, releasing resources, and closing out any open project tasks. The closing stage is also an opportunity to conduct a final review of the project's results and evaluate its overall success.

In conclusion, the project management process for the FuelIn system will involve a combination of project planning, team management, methodology, risk management, and ongoing communication and collaboration to ensure the successful development and deployment of the system.

Project Management Types

Several different project management methodologies can be used in the development of a system.

1. Waterfall Methodology: This is a traditional approach to project management where each phase of the project is completed before moving on to the next. It is characterized by a sequential and rigid approach to project delivery. (Source: "Waterfall Model - Project Management Methodology" by Mind Tools, mindtools.com)

2. Agile Methodology: This is a flexible and adaptive approach to project management that emphasizes collaboration, continuous improvement, and response to change. It is well suited to projects where requirements may change frequently or where the end goal is unclear. (Source: "Agile Project Management" by Agile Alliance, agilealliance.org)
3. Scrum Methodology: This is an iterative and incremental approach to project management that is based on the Agile methodology. It involves a cross-functional team working together to deliver incremental releases of a product in a series of sprints. (Source: "Scrum Framework" by Scrum Alliance, scrumalliance.org)
4. Kanban Methodology: This is a visual approach to project management that uses a board and cards to represent the different stages of a project. It is based on the principles of Lean manufacturing and emphasizes continuous improvement and just-in-time delivery. (Source: "Kanban Method" by LeanKit, leankit.com)

How each management type can be used?

1. Kanban Methodology: Kanban can be used to control the flow of work related to fuel demand, distribution, and tracking in the context of the fuel distribution system previously outlined. As part of the process, a visual board is used to depict the many stages of a project and the progression of work through each stage. This board can be used to monitor the status of each request and make sure that each step is carried out in the proper order.

One of the key benefits of using Kanban in this system is that it allows for a flexible, pull-based approach to project management. Instead of working on a set schedule, work is started when the previous task has been completed. This allows for a more responsive and efficient approach to project management, as work can be prioritized based on the current status of the project.

Additionally, Kanban also provides a clear visual representation of the flow of work, making it easier for stakeholders to understand the status of each request and for team members to coordinate their efforts. This can help to improve communication and collaboration within the project team and ensure that all requests are fulfilled in a timely and efficient manner.

Overall, the use of Kanban methodology in the fuel distribution system can help to optimize the flow of work, improve communication and collaboration, and ensure that requests are fulfilled in a timely and efficient manner.

2. Waterfall methodology: The Waterfall methodology can be used in the development of the FuelIn online fuel requesting and queue management system in the following way.

- Requirements Gathering and Analysis: The first stage in the Waterfall methodology involves defining the requirements for the system. This would involve understanding the requirements of both the end consumers and the fuel stations, and defining the functional and non-functional requirements for the system.

- Design: In this stage, the design of the system is created. This would include creating the architecture of the system, defining the modules and components of the system, and creating the user interface design.
- Implementation: This stage involves the actual implementation of the system. This would involve coding, testing, and debugging the system.
- Testing: This stage involves thorough testing of the system to ensure that it meets all the requirements and works as expected.
- Deployment: In this stage, the system is deployed to the production environment, and is made available to the end users.
- Maintenance: This stage involves providing ongoing support and maintenance to the system, fixing bugs, and adding new features as required.

The Waterfall methodology is a sequential approach to software development and is best suited for projects where requirements are well-defined and are unlikely to change during the development process. The methodology provides a structured approach to software development and is ideal for projects with a high level of predictability and stability.

(Source: "Waterfall Model." Wikipedia, Wikimedia Foundation, 27 Jan. 2023, en.wikipedia.org/wiki/Waterfall model.)

3. Scrum methodology: Scrum is an Agile methodology that is commonly used in software development projects. It is well suited for projects that require flexibility and quick adaptation to changing requirements. It is an incremental and iterative technique to software development that works effectively for difficult, adaptive projects where it is challenging to describe the requirements precisely at the outset.

According to the Scrum approach, the development process is divided into sprints, which are time-boxed intervals of work that are typically completed over the course of two to four weeks. At the end of each sprint, the Scrum team which consists of a Product Owner, Scrum Master, and Development Team delivers a potentially shippable increment of the product after prioritizing the work that needs to be done and estimating the time needed to do it.

The Scrum Master is responsible for facilitating the process, ensuring that the rules of Scrum are followed, and removing any obstacles that may impede the progress of the Development Team. The Product Owner is responsible for defining the priorities for the work to be done and making sure that the Development Team has a clear understanding of what is expected of them. The Development

Team is responsible for delivering the work, breaking it down into manageable tasks, and estimating the effort required to complete each task.

During each sprint, the Scrum team holds a Sprint Planning meeting to define the work to be done in the next sprint, a Daily Scrum to discuss progress and identify any obstacles, a Sprint Review to demonstrate what has been delivered and get feedback from stakeholders, and a Sprint Retrospective to reflect on the process and identify areas for improvement.

4. Agile methodology:

Agile methodology can be used in the given fuel distribution and management system development by following these steps:

- Gathering and analyzing requirements: The development team will work closely with fuelIn stakeholders to gather requirements, comprehend their needs, and pinpoint the main goals of the system.
- Iterative development: To produce manageable chunks of the system, the development team will work in brief sprints. Each sprint will utilize stakeholder feedback to make the necessary adjustments and enhancements.
- Continuous integration and delivery: To guarantee the system's quality, the development team will continuously integrate and test the code.
- Flexible planning: Flexible planning is made possible by the Agile methodology, which is crucial in the system at hand since changes in requirements and other circumstances may have an impact on the timeframe.
- Collaboration and communication: Collaboration and communication are highly valued aspects of the agile process between the development team, stakeholders, and end users. To make certain that everyone is on the same page and to address any concerns that may occur, frequent meetings and demos will be held.
- User participation: The Agile methodology promotes active user participation in the development process, which will assist guarantee that the system satisfies their needs and expectations.

Continuous improvement and adaptation are valued by the Agile methodology. The system will be regularly monitored by the development team, and adjustments will be made as needed to optimize performance and user experience.

The development team will be able to produce a top-notch fuel distribution and management system that satisfies the requirements and standards of FuelIn and its stakeholders by adopting an Agile approach.

Analysis

Literature Review

By processing safe online FuelIn is a responsible public company establish in Sri Lanka where they distribute and manage fuel supply island wide. They have several registered fuel stations at all the districts. FuelIn wishes to introduce an Online fuel requesting and que management system for the requirement of fuel distribution and tracking.

The information is then transferred to a backend platform that Online fuel requesting and que management system for the requirement of fuel distribution and tracking.

It might also include additional features, like automated reservation confirmation emails. Technology is essential for FuelIn is a responsible public company establish in Sri Lanka where they distribute and manage fuel supply island wide. They have several registered fuel stations at all the districts. FuelIn wishes to introduce an Online fuel requesting and que management system for the requirement of fuel distribution and tracking.

que management system for the requirement of fuel distribution and tracking. Not only can your current website and que management system for the requirement of fuel distribution and tracking be synced, but also Facebook. consumers are to be given facilities to request a fuel based on the quota given for the vehicle registration number. The requesting customers can obtain a token with expected filling time and date with 3 hours of tolerance. If the fuel distribution cannot made available, the customer will be given a new scheduled period Online reservation systems are frequently offered for sale as cloud-based software-as-a service. If necessary, the system is integrated into both your current website and your social media pages. The customer-facing end and the back end are two separate components of a booking system.

FuelIn wishes to introduce an Online fuel requesting and que management system for the requirement of fuel distribution and tracking. End consumers are to be given facilities to request a fuel based on the quota given for the vehicle registration number. The requesting customers can obtain a token with expected filling time and date with 3 hours of tolerance. If the fuel distribution cannot made available, the customer will be given a new scheduled period. If there is no any scheduled delivery to a particular filling station or scheduled fuel stock is finished, the necessary message will be given and requests are not permitted from those fuel stations. Once the delivery schedule to the given filling station is confirmed by the dispatch office in the head office, notification SMS/email will be sent to all token holders to make the payment.

As tokens are issued only for the scheduled delivery number of liters, the non-committed or unpaid fuel stocks can be made available to request on demand after the scheduled payment time, those tokens are issued for the same day collection. End consumers can take a token from the filling station also using the same system when scheduled delivery is available. Twelve hours before the delivery happens to the filling station, end consumers will receive SMS/email as a reminder with amount requested. Each vehicle receives a new quota after in each week's period. The filling station orders are to be taken separately using the system. The fuel stocks are to be distributed according to population density within the area of filling stations.

Here is a quick summary of the benefits an Online fuel requesting and que management system before we look more closely at specific features and how to choose one.

- **Increased sales**: At the risk of stating the obvious, Online fuel requesting and que management system for the requirement of fuel distribution and tracking. End consumers are to be given facilities to request a fuel based on the quota given for the vehicle registration number Once the delivery schedule to the given filling station is confirmed by the dispatch office in the head office, notification SMS/email will be sent to all token holders to make the payment. and web system that is multilingual and enables customers to pay in their local currency depending on your target markets.
- **Internet-based reservation management systems can lower your staffing costs**: Through a reduction in the number of staff required to secure a booking.
- **Standardized and streamlined payments**: As soon as a customer makes an online reservation, money is sent right to your account.
- **Better process management Automated**: business operations by doing away with the need for manual data entry, booking management, inventory management, report creation, and payment processing.

User Requirements

1. A customer may make, change, or cancel reservations via the website. When making a reservation through a reservation clerk, a customer gives personal details, states the room type, number of occupants, and dates of arrival and departure.
2. A reservation can be done by providing credit card details or without providing the details. Reservations that are not provided credit card details are automatically cancelled at 7 PM daily.
3. No-show customers should pay for the reservation. A billing record is created for each no-show reservation by 7:00 PM daily. Along with this, a report is also produced providing the total occupancy and revenue for the previous night.
4. Reservation clerk can check in a customer (with or without a prior reservation), change the checkout date, and check out the customer.

5. A room is assigned to the customer at check-in time and a customer record is created.

6. At the checkout, customer may pay by cash or with credit card. Customer billing record is created and then customer receives a check out statement. Customer who does not check out by the checkout time is charged for an additional night. Other optional charges include restaurant charges, room service and laundry, telephone service, automatic issuing of keys, and a club facility.

7. The manager views reports, such as hotel occupancy to identify present or past dates bookings, projected occupancy for future dates, and view financial information, including room revenue information.

8. Travel companies can block bookings at a discounted rate for one or more nights for three or more rooms. Bills are charged directly to the travel company.

9. If needed, customers can perform reservation of residential suites instead of hotel rooms, where a guest can occupy a suite for a week or month at a time, paying a weekly or monthly rate.

System Requirements

Cloud-based

Payment options (credit card, bank transfer, PayPal, deferred payment processing, cash)

Inventory system

Real-time booking

User-friendly interface

Customer support

Integrations with an email marketing software

SWOT Analysis

SWOT analysis is a framework used to assess a company's competitive position and to create strategic planning. It stands for strengths, weaknesses, opportunities, and threats. The SWOT analysis evaluates internal and external variables as well as present and potential future situations.

A SWOT analysis is intended to help you take a practical, fact-based, and data-driven look at the advantages and disadvantages of a company, its initiatives, or its sector. The organization must avoid preconceived notions or gray areas and concentrate on real-life contexts to maintain the analysis' accuracy. Companies should use it as a reference rather than a strict prescription.

A SWOT analysis' main goal is to assist organizations in fully understanding all the variables that go into choosing a course of action. Before you decide to take any company action, whether you are investigating new initiatives, updating internal policies, considering opportunities to pivot, or changing a plan midway through its execution, conduct a SWOT analysis. With a focus on utilizing strengths and opportunities to overcome weaknesses and threats, use your SWOT analysis to find recommendations and strategies.

SWOT Analysis Related of the System

There are broken down each category and given some examples to make it easier for you to conduct your own SWOT analysis for your hotel reservation system. This is by no means a comprehensive list, and every hotel reservation system will have different priorities. Simply put, each section is meant to get you going. Following that, we'll offer some suggestions on how to start a SWOT analysis for your hotel and how to involve others in the process.

Strengths

An easy-to-use online fuel requesting and que management system for the requirement of fuel distribution and tracking

Remote fuel requesting and que management software is accessible anytime, anywhere.

To simplify operations, manage each aspect of the business from a single dashboard.

Mobile housekeeping reports in real-time with checklists to increase output

Online check-in and registration reduce workload and are well-liked by visitors.

Weaknesses

Determine if the SWOT has any drawbacks or if hotel needs a feature that is missing from it. Ask such questions.

Is the price too high?

Does it integrate with the programs and apps you already use?

Will the current staff have a learning curve?

Have you got a reliable Wi-Fi connection?

Opportunities

Identifying external industry trends and how the SWOT might be used to give you a competitive edge is helpful in this situation. Personalization, for instance, is becoming more crucial in both hotel marketing and customer service.

To create customer segments for specialized marketing campaigns, use guest data.

SMS text messaging

Initiatives to surprise and delight customers to increase loyalty and repeat bookings.

Email marketing automation for the entire customer lifecycle.

Integration of smart room technology and sustainability initiatives.

Expand the potential for ancillary income.

Threats

Threats typically originate outside of hotel reservation system and are beyond your control. Threats can come in the form of calamities like pandemics and business interruptions, changing technologies, connection errors or even hacking. Consider going a step further and asking the salesperson how the SWOT can defend you from dangers. For instance, in the event of a pandemic, a SWOT could support contactless transactions and remote management in the event of staffing shortages while also assisting in maintaining sanitation standards through mobile housekeeping reports.

Examine potential obstacles and consider how a SWOT might or might not support your hotel's economy. For example, does the SWOT provide a framework for diversified hotel revenue streams? (e.g., vacation or activity packages)

Connection errors

3.5 Feasibility Study

As the name implies, a feasibility study seeks to determine the viability of a project or plan. It is a determination of how feasible a project or plan is.

Any project or plan's initial design phase includes a feasibility study. It is carried out to unbiasedly identify the advantages and disadvantages of a proposed venture or an established company. The resources needed for the project, the likelihood of success, and the opportunities and threats present in the surrounding environment can all be identified and evaluated. The following questions are being investigated to obtain answers:

Does the business have the necessary tools and technology?

Will the company's investment yield a high enough return?

Step for feasibility study

Conduct the feasibility study involves the below steps.

Conduct preliminary analyses.

Make a statement of projected income. What kind of revenue could the project possibly bring in?

Make a market analysis. Does the project produce a good or service that the market needs? What will customers spend on the product or service?

Prepare the new project's organizational structure. What are the personnel needs?

How many employees are required? What additional resources are required?

Create an opening day balance of the anticipated costs and income.

Review and examine the project's internal points of vulnerability that can be mitigated or eliminated.

Choose whether to move forward with the project or plan.

Five areas of project feasibility

1. Technical feasibility

Is the project technically feasible? An overview of the organization's technical prowess is provided, along with information on the availability of qualified personnel and suitable facilities.

2. Economical

 feasibility

Is the project affordable given the economic resources available? Is the anticipated return on investment (ROI) sufficient, even if it can be afforded? A project is not feasible if it is simply too expensive and doesn't provide enough economic benefits.

Legal

Does the proposed business comply with all applicable rules and laws?

Operational

Is the project consistent with the organization's goals and operations?

5. Schedule

Is it reasonable to expect that the project will be finished on time given the suggested schedule? The project's overall feasibility may suffer if it takes too long to finish, which could increase costs.

3.6 Requirement Analysis

3.6.1 Functional and Non-functional Requirements of the System

Functional requirements

Give each online reservation a corresponding account.

Each account should only have one user.

Make it possible for users to search for and select the best booking options.

Accept the date and time to check the rooms that are available at that time.

The provided contact information should receive a booking confirmation.

Calculate and display the cost of utilities and lodging.

Book cancellations.

Display and modify visitor records.

changing areas.

Non-functional requirements

To stop bots from making reservations, use encryption.

Search results should appear in a reasonable amount of time.

If a user enters an invalid value, they should be given the proper assistance to complete the required fields.

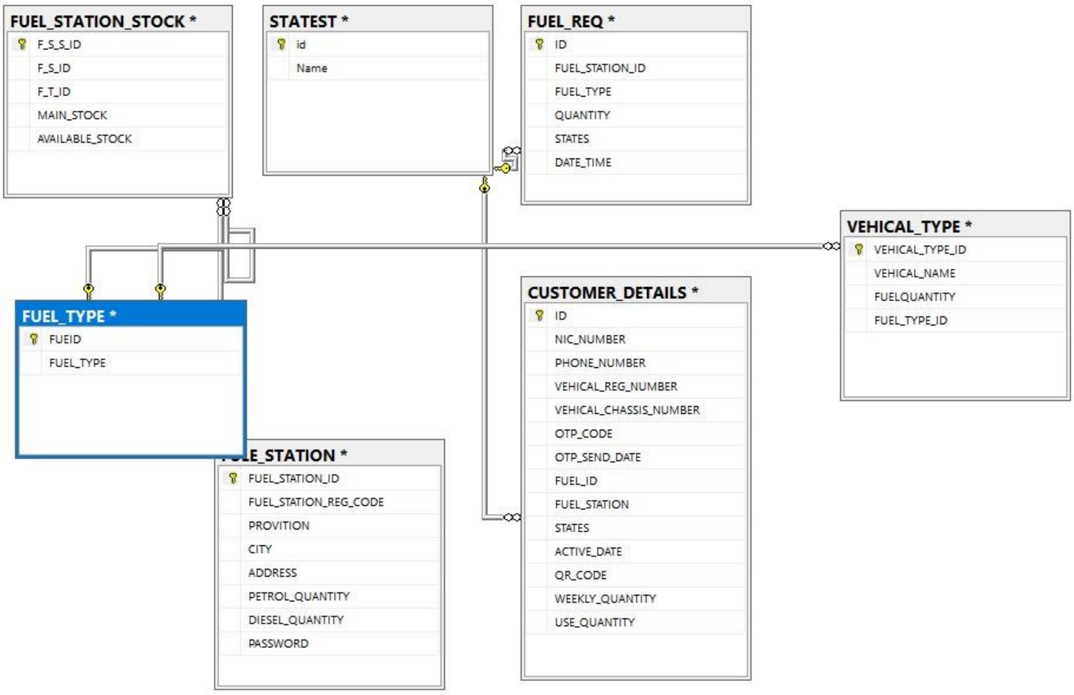
The system ought to accept payments made using a variety of methods.

Accessible, effective, and simple to use.

Record all activities, documentation, and responses.

4.0 System Design

4.1 ER Diagram



Implementation of System

Database System

Data are the most crucial component of a system. Therefore, the database needs to be very effective. We have applied the normalization principle throughout the database to achieve that.

In this database is developed by using MS SQL.

Testing

6.1 Purpose

This test plan describes the testing approach that will drive the testing of member division. The objective of the test is to verify that the functionality of member module work according to the specifications. The test will execute and verify the test cases, identify, fix and retest all high and medium severity defects per the entrance criteria, prioritize lower severity defects for future fixing.

The final product of the test is:

- A production-ready member module.

6.2 Scope

Under the scope of testing section will elaborate the scope that is going to be tested for member module. Mainly ensure proposed module will manage the member relationships according to the gathered requirement and features in a proper manner.

6.3 Test Coverage

Test coverage measure of the effectiveness of the testing by providing data on different items. Test coverage can be best measured with the following 2 things:

By mapping the requirements to the test cases.

By the test case status.

Test Coverage can be implemented by static testing techniques. For example these test techniques include peer review. And also testers can convert the defects into test cases and analyses test coverage.

6.4 Assumptions

6.4.1 Functional Testing

Exploratory testing would be carried out once the build is ready for testing.

Performance testing is not considered for this estimation.

Most of the defects would come along with a snapshot JPEG format.

The test team assumes all necessary inputs required during test design and execution will be supported by development/business analyst appropriately.

Test case design activities will be performed by QA

Test environment and preparation activities will be owned by development team.

Business analyst will review and sign-off all test cases prepared by test team prior to start of test execution.

The defects will be tracked.

Project Manager/ Business analyst will review and sign-off all test deliverables.

The project will provide test planning, test design and test execution support.

Data Validations

Conclusion

Here, the definition of a hotel reservation system, its operation, and its advantages for the hotel sector are discussed. An online reservation system has many advantages, such as lowering human error rates, increasing productivity, simplifying data collection and reward administration, and giving you a competitive edge.

With the development of technology, one of the simplest changes a hotel can make to reap the benefits mentioned in this report is to install a hotel reservation system software. It enables you to maintain occupancy and make sure everything is going according to plan with little assistance from hotel staff. Staff can devote more time to providing the best service possible if they have more time to spare to hale administrative tasks.

Technical Feasibility

Technical feasibility refers to the assessment of whether a proposed system or project can be successfully developed and implemented using existing technology and infrastructure. In the context of the fuel distribution and tracking system, technical feasibility can be evaluated by considering the following factors

Hardware Requirements:

Server: A high-performance server with sufficient processing power and memory to support the web server and database server.

Storage: Adequate storage capacity to accommodate the growing data needs of the system.

Network: High-speed internet connectivity to ensure fast and reliable communication between the server and client systems.

Security: Firewall, anti-virus, and anti-malware software to protect the system from cyber threats.

Backup: An efficient backup and disaster recovery system to ensure the safety of the data.

Software Requirements

Operating System: A modern and supported operating system, such as Windows Server or Linux, to run the web server and database server.

Web Server: Microsoft IIS or Apache HTTP Server to host the web applications and API.

Database Server: Microsoft SQL Server manage the data storage and retrieval.

.NET Framework: The latest version of the .NET framework to support the development of the ASP.NET web API.

Entity Framework: The latest version of Entity Framework to support data access and management in the ASP.NET web API.

Angular: The latest version of Angular to support the development of the client-side web application.

Payment Gateway Integration: Integration with payment gateway APIs to support online payments.

SMS and Email Services: Integration with SMS and email services to send notifications to customers and managers.

Following are the minimum server requirements for the Fuel Distribution and Tracking System.

Hardware Requirements	**Software Requirements**
RAM(32GB) 2933 MT/s DDR4	Windows or Linux OS
12G drives 4800MB/s,	MS-SQL
Intel Xeon W-2145	Chrome/internet explorer or opera browser latest version
Server Requirements	
Hardware Requirements	**Software Requirements**
RAM 4GB	Windows OS / Linux OS / Mac OS
128 GB SSD || 500GB	
i3 1th generation	Chrome/internet explorer or opera browser latest version
Client Requirements	

Economic Feasibility

Economic feasibility is an important consideration when developing a new system, as it helps determine whether the costs of developing and deploying the system are outweighed by the potential benefits.

Some key factors to consider when evaluating the economic feasibility of the Fuel Distribution and Tracking System include:

Development Costs: The costs of developing the system, including software development, hardware procurement, and any other expenses incurred during the development process.

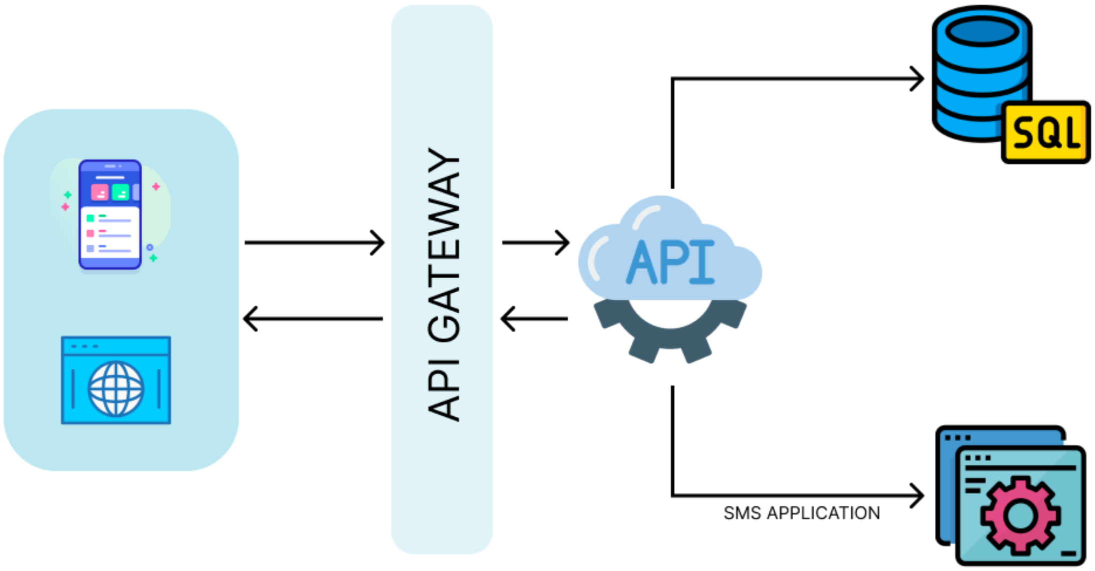
Deployment Costs: The costs of deploying the system, including server hardware and hosting expenses, licensing fees, and any other costs associated with getting the system up and running.

Maintenance Costs: The ongoing costs of maintaining and updating the system, including software upgrades, hardware replacements, and other costs associated with keeping the system running smoothly.

Fuel Station Pc cost	
I3 4th generation/ 4GB Ram / 500Gb Hard	Rs.30000
Printer	Rs.25000
Operating System	Rs.2000
Antivirus guard	Rs.500
	Rs.57,500 * 60

Head office Pc cost	
I3 4th generation/ 4GB Ram / 500Gb Hard	Rs.30000
Printer	Rs.25000
Operating System	Rs.2000
Antivirus guard	Rs.500
	Rs.57,500 * 10

System Architecture design



The system architecture design for the FuelIn online fuel management system would involve the following steps:

Define the overall system architecture: This involves identifying the key components of the system, the relationships between them, and the interfaces between the components.

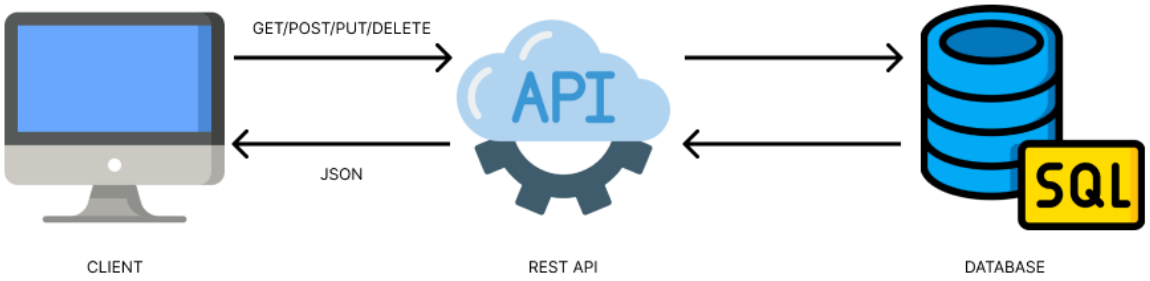
Determine the functional requirements: This involves analyzing the requirements specified in the project brief and deriving the detailed functional requirements for each component of the system.

Design the system components: This involves defining the structure and behavior of each component, and determining how they will interact with each other to provide the desired functionality.

Determine the data architecture: This involves defining the data model, the data stores, and the relationships between the data elements.

Define the security architecture: This involves identifying the security risks associated with the system and determining the appropriate measures to mitigate those risks.

REST API Architecture



There are several advantages of using a REST API in the FuelIn online fuel management system:

Scalability: REST APIs are designed to be scalable, making it easier to add more users, features, and functionalities to the system.

Interoperability: REST APIs use standard HTTP protocols and methods, making it easier to integrate the FuelIn system with other systems and technologies.

Flexibility: REST APIs provide a flexible and efficient way to access and manipulate data, allowing for a wide range of use cases.

Portability: REST APIs can be accessed from any platform that supports HTTP, making it easier to build cross-platform mobile and web applications.

Ease of use: REST APIs use simple, straightforward methods and return data in a standard format, making it easier for developers to understand and use the API.

Performance: REST APIs can be optimized for performance, making it possible to handle large amounts of data and traffic.

By using a REST API in the FuelIn system, the system can be designed to be scalable, flexible, and easy to use, while also providing robust security and performance.

Security of the REST API

To ensure the security of the REST API in the FuelIn online fuel management system, the following measures should be taken:

Authentication: Implement authentication mechanisms, such as API keys, OAuth, or JWT, to ensure that only authorized users have access to the API.

Authorization: Use role-based access controls to restrict access to sensitive data and functionality to only those users who need it.

Input validation: Validate all incoming data to ensure that it is in the expected format and does not contain any malicious content.

Logging and monitoring: Keep detailed logs of API activity, and monitor the logs for any suspicious activity.

Security updates: Regularly apply security updates to the API server and underlying infrastructure to protect against known vulnerabilities.

By implementing these security measures, the REST API in the FuelIn system can be protected against common threats, such as unauthorized access, data breaches, and denial-of-service attacks.

JWT Authentication

JWT (JSON Web Token) authentication is a common way of authenticating users in modern web applications. JWT is a compact, URL-safe means of representing claims to be transferred between two parties.

- The basic flow of JWT authentication involves the following steps:
- The user sends a login request to the server with their credentials (e.g., username and password).
- The server validates the credentials and generates a JWT token if the credentials are correct.
- The server sends the JWT token back to the client as a response.
- The client then sends the JWT token with every subsequent request to the server.

- The server verifies the JWT token on each incoming request and grants or denies access based on the claims contained in the token.
- The JWT token consists of three parts separated by dots (.): the header, the payload, and the signature. The header contains information about the type of token and the signing algorithm used. The payload contains the claims (e.g., user ID, expiration time, etc.). The signature is used to verify that the token was generated by the server and has not been altered in transit.

JWT tokens are signed using a secret key shared between the server and the client. This secret key is used to encrypt and decrypt the token, allowing the server to verify that the token was generated by it and has not been tampered with.

Advantages of using JWT authentication in a REST API:

Statelessness: JWT tokens are self-contained and carry all the necessary information about the user. This allows for a stateless architecture, where the server does not need to keep track of user sessions.

Scalability: JWT authentication allows for horizontal scaling as there is no need for a centralized session store.

Easy to implement: JWT authentication is relatively simple to implement and does not require complex infrastructure.

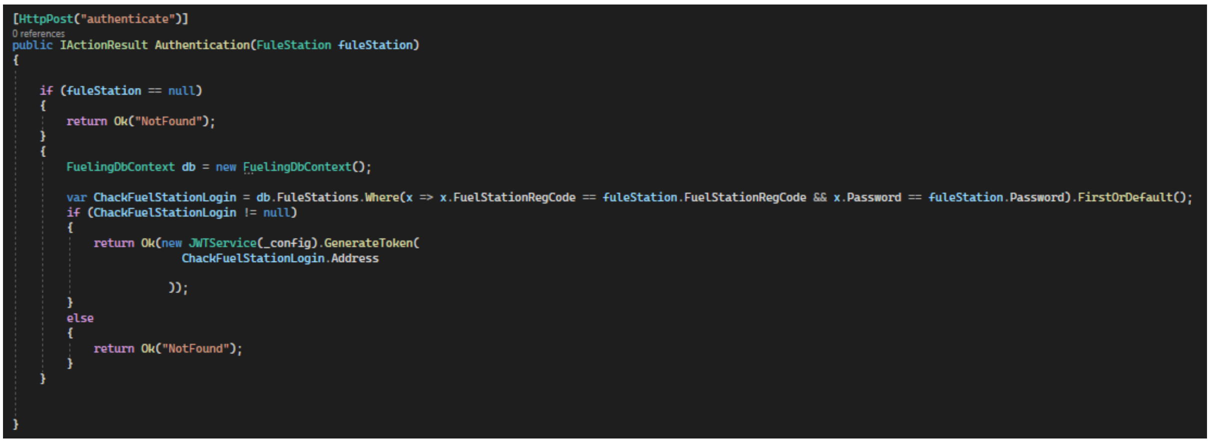
Cross-domain compatibility: JWT tokens can be easily passed between different domains, making it suitable for modern web applications.

Improved security: JWT tokens can be encrypted and signed, providing a secure way of transmitting user information.

JWT token generate

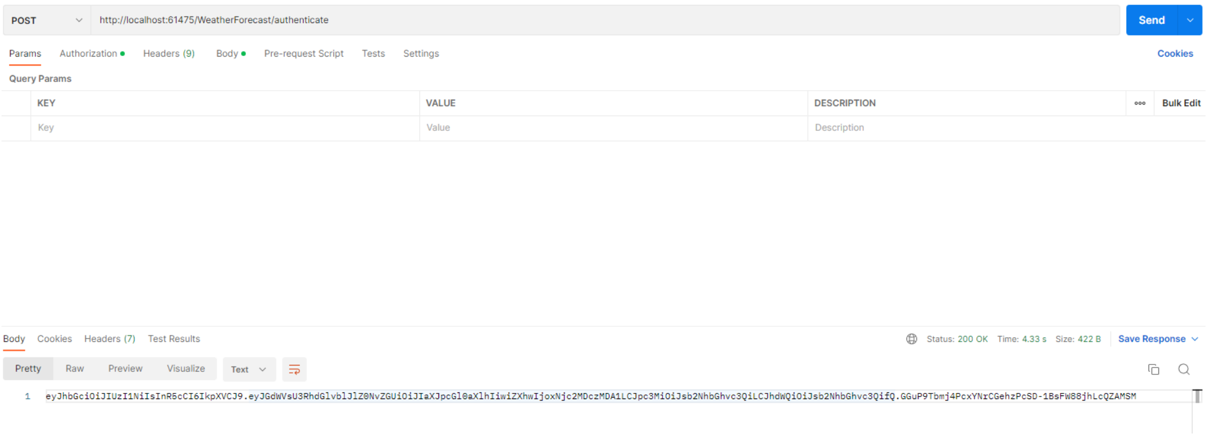
JWT (JSON Web Token) is a JSON data structure that is used for authentication purposes. The JWT token is generated on the server-side using a library or a custom implementation in C#.

Ex -



```
[HttpPost("authenticate")]
0 references
public IActionResult Authentication(FuleStation fuleStation)
{

    if (fuleStation == null)
    {
        return Ok("NotFound");
    }
    {
        FuelingDbContext db = new FuelingDbContext();

        var ChackFuelStationLogin = db.FuleStations.Where(x => x.FuelStationRegCode == fuleStation.FuelStationRegCode && x.Password == fuleStation.Password).FirstOrDefault();
        if (ChackFuelStationLogin != null)
        {
            return Ok(new JWTService(_config).GenerateToken(
                         ChackFuelStationLogin.Address

                       ));
        }
        else
        {
            return Ok("NotFound");
        }
    }


}
```

Testing POSTMAN

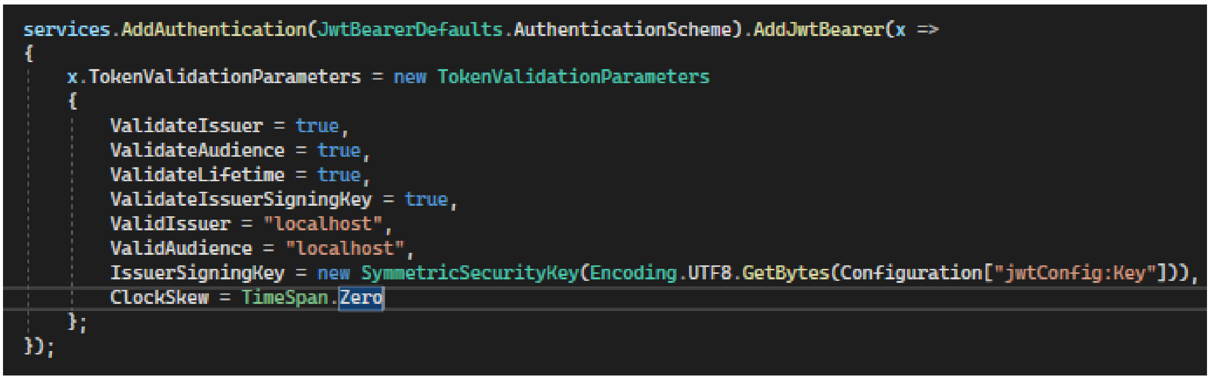


http://localhost:61475/WeatherForecast/authenticate

Status: 200 OK Time: 4.33 s Size: 422 B

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJGdWVsU3RhdGlvblJlZ0NvZGUiOiJIaXJpcGl0aXlhIiwiZXhwIjoxNjc2MDczMDA1LCJpc3MiOiJsb2NhbGhvc3QiLCJhdWQiOiJsb2NhbGhvc3QifQ.GGuP9Tbmj4PcxYNrCGehzPcSD-1BsFW88jhLcQZAMSM
```

Authenticate



```
services.AddAuthentication(JwtBearerDefaults.AuthenticationScheme).AddJwtBearer(x =>
{
    x.TokenValidationParameters = new TokenValidationParameters
    {
        ValidateIssuer = true,
        ValidateAudience = true,
        ValidateLifetime = true,
        ValidateIssuerSigningKey = true,
        ValidIssuer = "localhost",
        ValidAudience = "localhost",
        IssuerSigningKey = new SymmetricSecurityKey(Encoding.UTF8.GetBytes(Configuration["jwtConfig:Key"])),
        ClockSkew = TimeSpan.Zero
    };
});
```

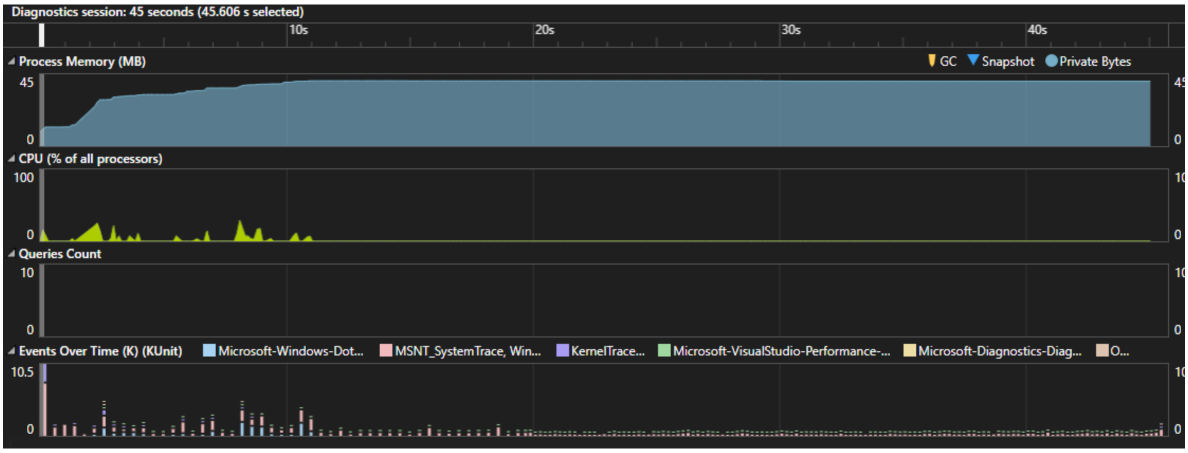
Entity Framework (EF)

There are several benefits of using Entity Framework (EF) in this system:

Productivity: EF simplifies the development of data-driven applications by reducing the amount of code required to perform common data access tasks. This increases the productivity of developers, as they can focus more on implementing business logic rather than writing complex data access code.

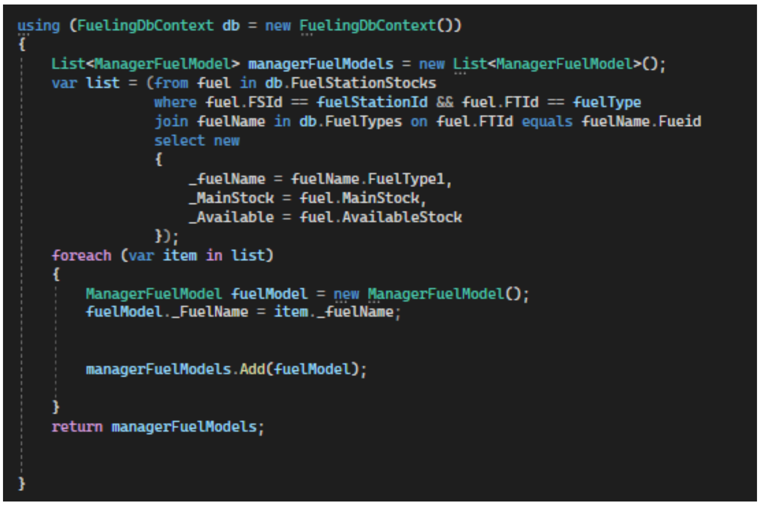
Abstraction: EF provides a higher level of abstraction over the underlying database, which helps to isolate the application from changes in the database structure. This makes it easier to maintain and evolve the application as the database evolves over time.

Performance: EF provides an optimized execution engine that can generate efficient SQL statements and minimize the number of database round-trips. This results in faster performance, even when working with large datasets.



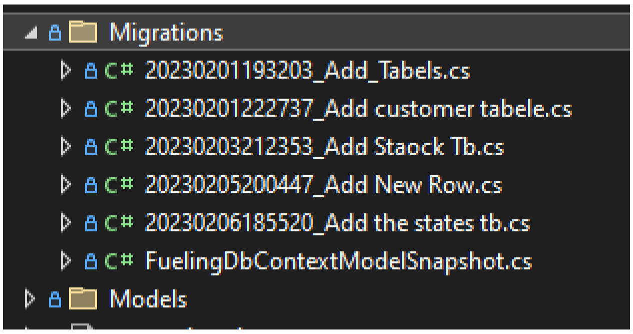
Portability: EF supports multiple database platforms, including Microsoft SQL Server, Oracle, and MySQL. This makes it possible to develop applications that can run on different database platforms without having to change the application code.

LINQ Support: EF provides support for LINQ, which is a powerful query language that allows developers to write type-safe, declarative queries against their data. This simplifies the process of querying and manipulating data in the application.



```
using (FuelingDbContext db = new FuelingDbContext())
{
    List<ManagerFuelModel> managerFuelModels = new List<ManagerFuelModel>();
    var list = (from fuel in db.FuelStationStocks
                where fuel.FSId == fuelStationId && fuel.FTId == fuelType
                join fuelName in db.FuelTypes on fuel.FTId equals fuelName.Fueid
                select new
                {
                    _fuelName = fuelName.FuelType1,
                    _MainStock = fuel.MainStock,
                    _Available = fuel.AvailableStock
                });
    foreach (var item in list)
    {
        ManagerFuelModel fuelModel = new ManagerFuelModel();
        fuelModel._FuelName = item._fuelName;


        managerFuelModels.Add(fuelModel);

    }
    return managerFuelModels;


}
```

Database Migrations: EF provides a built-in mechanism for managing database migrations, which makes it easier to evolve the database schema over time as the application evolves. This helps to avoid the risk of data loss and ensures that the application continues to work as expected even as the database changes.



Migrations
20230201193203_Add_Tabels.cs
20230201222737_Add customer tabele.cs
20230203212353_Add Staock Tb.cs
20230205200447_Add New Row.cs
20230206185520_Add the states tb.cs
FuelingDbContextModelSnapshot.cs
Models

Using Angular

Using Angular in the development of this fuel distribution and management system offers several benefits, including:

Modularity: Angular provides a modular architecture, which enables the development team to break down the system into smaller, manageable modules, making it easier to manage, test, and maintain.

Reusable Components: Angular provides a component-based architecture, where developers can build reusable components, reducing the development time and effort.

Two-Way Data Binding: Angular offers two-way data binding, allowing the automatic synchronization of data between the model and the view components, making the process of updating data smoother and faster.

Improved User Experience: Angular provides a smooth and fast user experience through its fast rendering, high performance, and intuitive design.

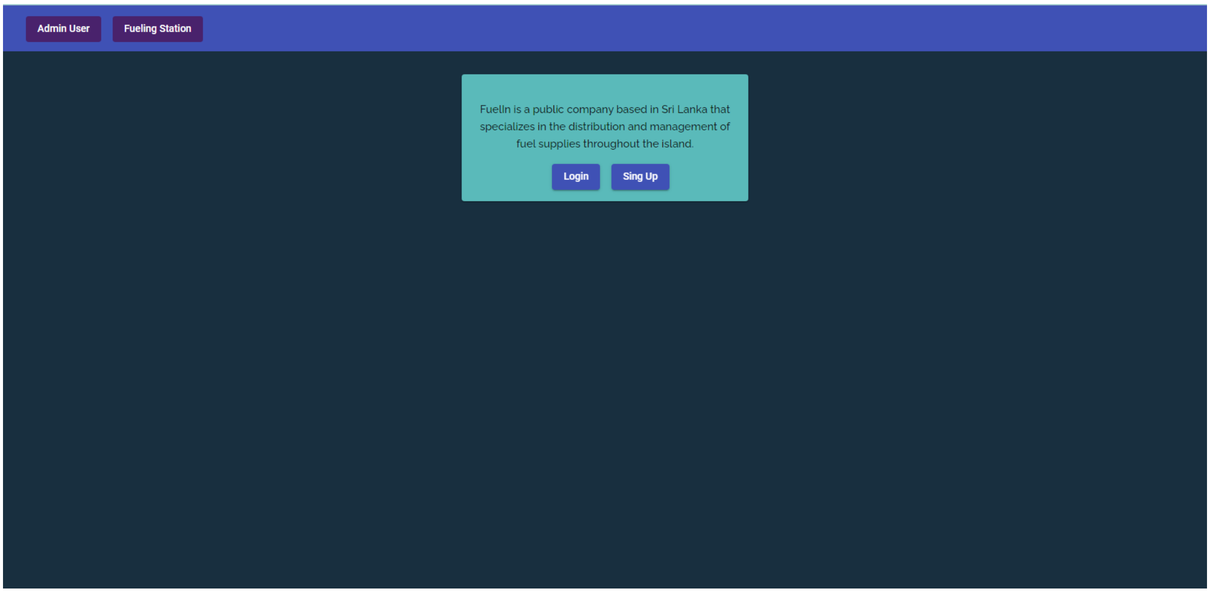
Easy Testing: Angular provides robust testing tools and frameworks, making it easier to test the system, reducing the risk of bugs and errors, and ensuring the quality of the system.

Better Maintainability: Angular offers improved maintainability through its modular and component-based architecture, making it easier to identify and fix issues as the system evolves.

Wide Community Support: Angular has a large and active community of developers, which provides support, guidance, and a wealth of resources for developers working with the framework. The angular was the client side rendering

User Interface

Home page



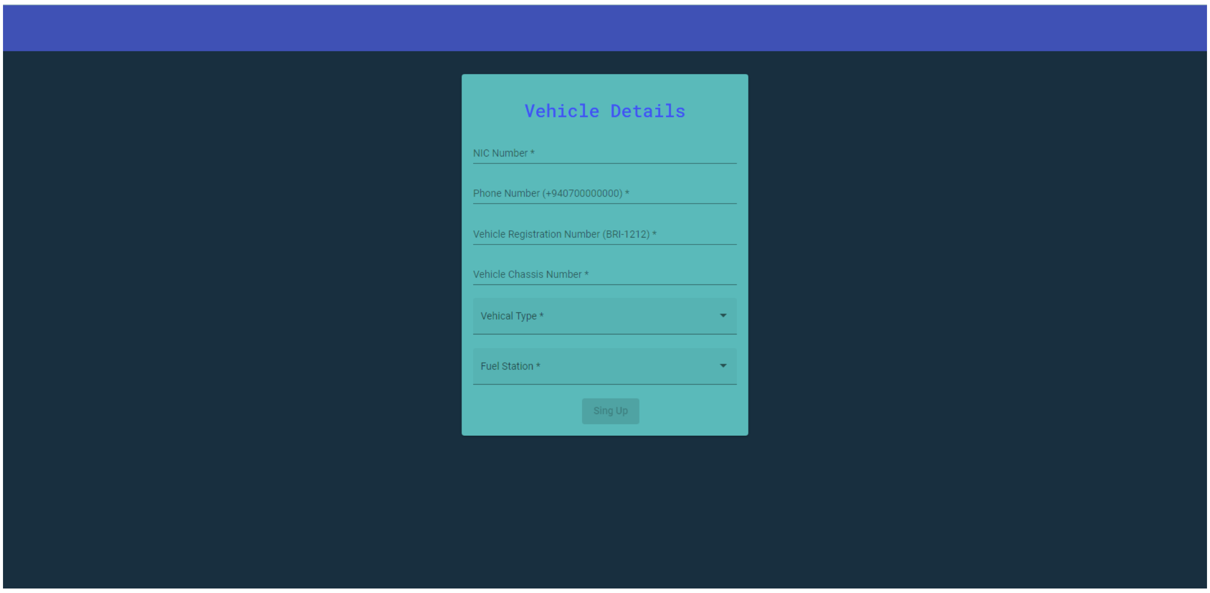
FuelIn is a public company based in Sri Lanka that specializes in the distribution and management of fuel supplies throughout the island.

The home page of the web application is the first page that the users will see when they visit the website. The page should have a clear and intuitive design, making it easy for users to navigate and understand its purpose. The home page should have two main buttons, "Sign In" and "Sign Up". The "Sign In" button is for existing users who want to log in to the system to access their account and request fuel. The "Sign Up" button is for new users who want to create an account to start using the system.

Additionally, there should be two separate logins, one for the admin and one for the filling station managers. The admin login will provide access to the back-end of the system, where the admin can view the status of each filling station and manage the distribution of fuel. The filling station manager login will provide access to the filling station's account, where the manager can verify tokens and mark the status of requests.

The home page should be visually appealing and easy to use, with a clean and modern design. This will help to create a positive first impression and make it easy for users to get started with the system.

Registration page

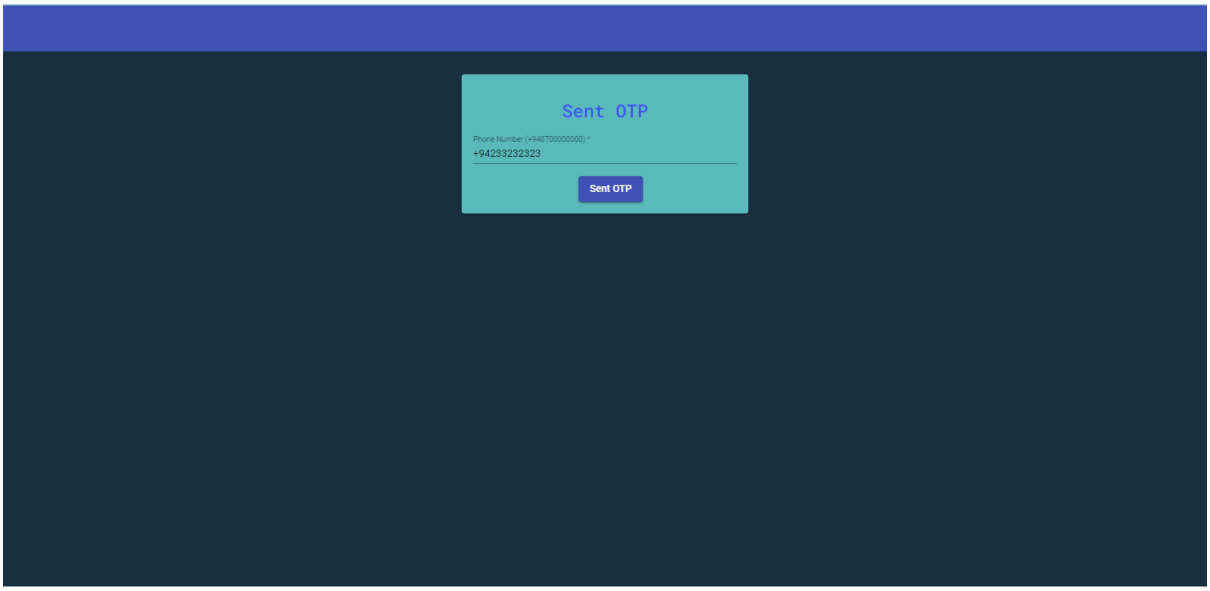


When a customer wants to register for the FuelIn fuel request and que management system, they will be directed to a user registration page. On this page, they will be asked to enter their national identity card number, phone number, vehicle registration number, and vehicle chassis number. The customer will also have the option to select the type of vehicle they own and the preferred fuel station for delivery.

This information will be used to verify the customer's identity and ensure that only one account is created per vehicle. The national identity card number and vehicle registration number will be used to check for duplicates in the system. The phone number will be used for notifications and SMS reminders, and the vehicle information will be used to determine the customer's quota for fuel.

Once the customer has completed the registration process, they will receive a confirmation message, and they will be able to log into the system and request fuel. The information provided during the registration process will be stored in a secure database, and the customer's privacy will be protected according to applicable laws and regulations.

Login Page



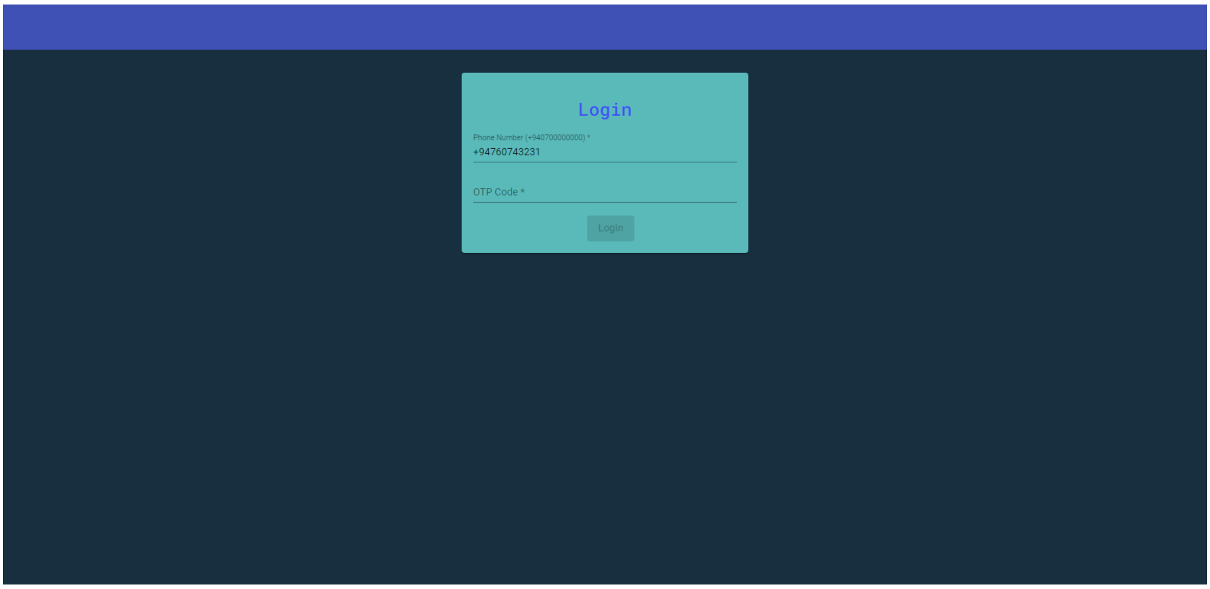
Sent OTP

Phone Number (+940700000000) *

+94233232323

The manager can approve the customer's request by logging into the system and verifying the customer's information such as their National Identity Card number, phone number, vehicle registration number, and vehicle chassis number. The manager can then confirm the customer's vehicle type and preferred fuel station. To authenticate the customer's identity, the back-end system will send an OTP (One-Time Password) code to the customer's registered phone number. The customer will need to enter this code on the website to verify their identity and complete the registration process. This ensures the security and accuracy of customer information in the system.

Login (Check the OTP)

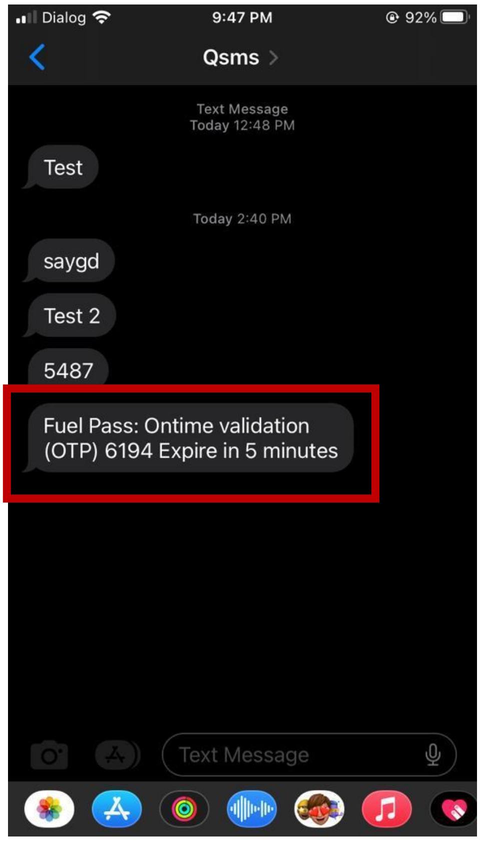


Login

Phone Number (+940700000000) *
+94760743231

OTP Code *

Yes, once the customer enters the OTP code, they can log into the system. After the successful login



Text Message
Today 12:48 PM

Test

Today 2:40 PM

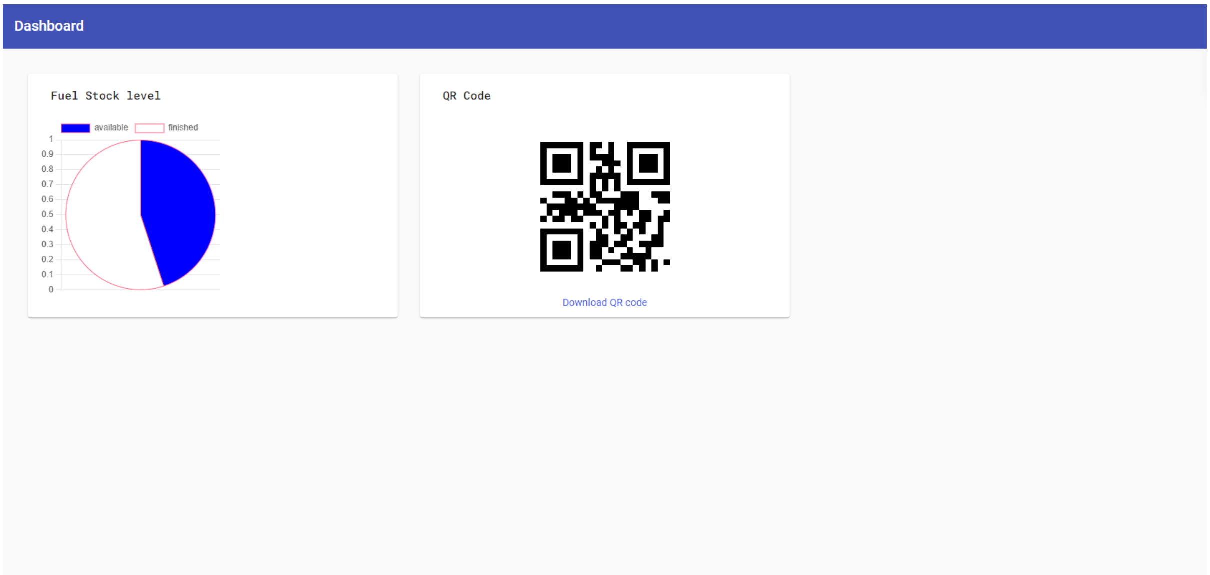
saygd

Test 2

5487

Fuel Pass: Ontime validation (OTP) 6194 Expire in 5 minutes

Customer Dashboard

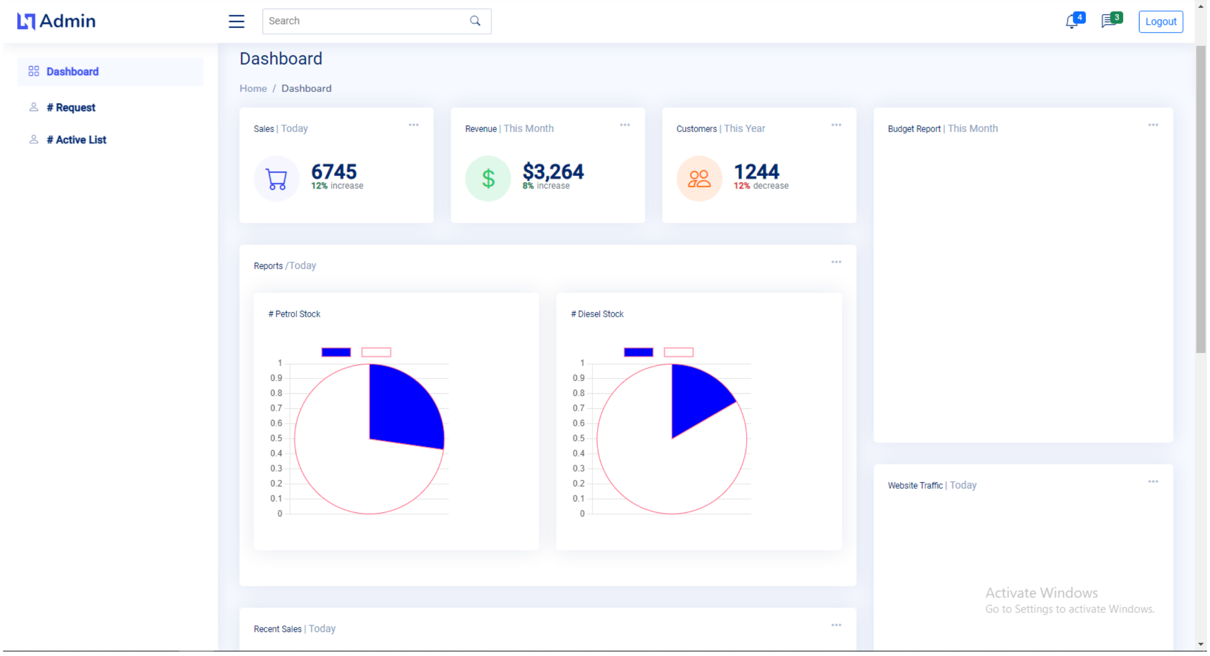


In the customer dashboard, the customer can view the current fuel stock levels of the fuel station they have selected. This information can be updated in real-time by the fuel station and can help the customer make informed decisions about when to refuel. The customer can also see the types of fuel that are currently available at the fuel station. In addition to this, the customer can also view a unique QR code, which acts as a token for them to claim their fuel once they arrive at the fuel station. This QR code can be scanned by the fuel station personnel to verify the customer's identity and to keep track of the fuel that has been dispensed to them. This system helps to make the refueling process more efficient and streamlined for the customer.

Token (QR Code)

The customer can then show the QR code to the fuel station attendant, who can use it to confirm the customer's fuel request and dispense the fuel. This system allows for a quick and efficient process for the customer to receive the fuel they need without having to go through a lengthy process of providing personal information or waiting in line.

Manager Dashboard



Dashboard

Sales | Today
6745
12% increase

Revenue | This Month
$3,264
8% increase

Customers | This Year
1244
12% decrease

Budget Report | This Month

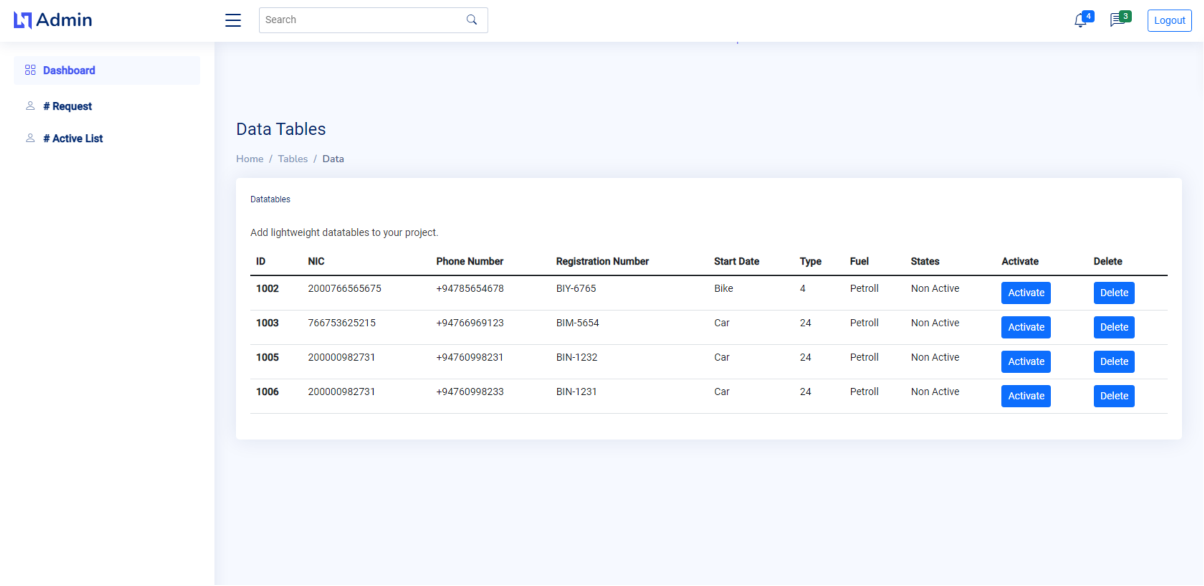
Reports /Today

Website Traffic | Today

Recent Sales | Today

The manager dashboard allows the manager to have a complete overview of the fuel stock for petrol and diesel. This helps them monitor the fuel levels and ensure that there are adequate supplies to meet the demand of customers. Additionally, the manager can also view the details of all the registered customers, including their national identity card number, phone number, vehicle registration number, vehicle chassis number, and selected vehicle type. This information can be used by the manager to analyze the customer behavior, purchase patterns, and make data-driven decisions. Overall, the manager dashboard provides the necessary tools and information to effectively manage the fuel station and ensure the smooth running of operations.

Customer request approval

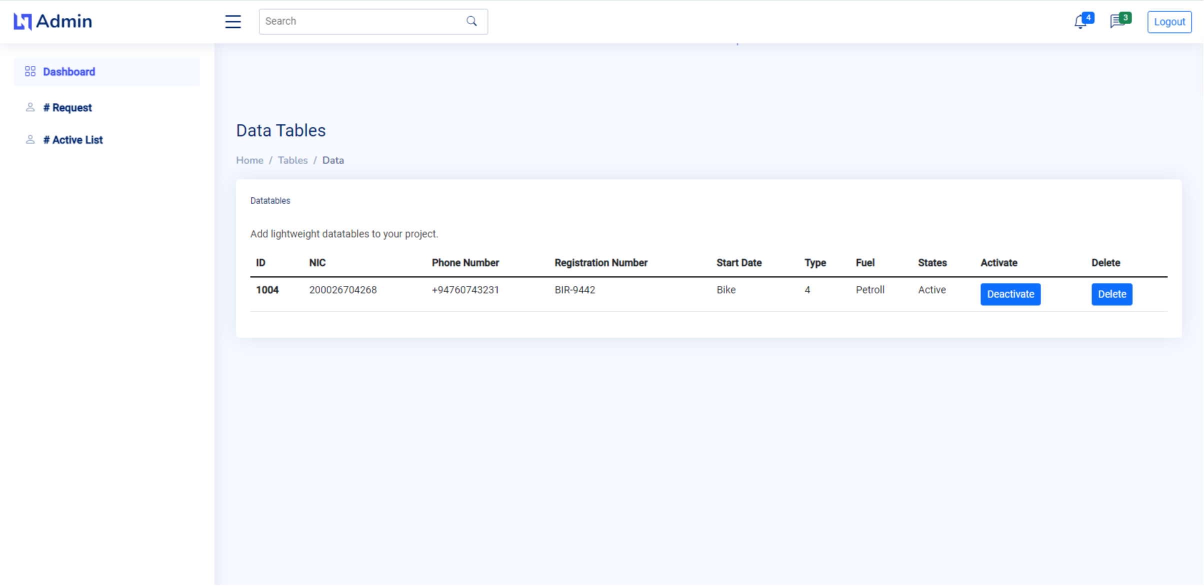


Add lightweight datatables to your project.

ID	NIC	Phone Number	Registration Number	Start Date	Type	Fuel	States	Activate	Delete
1002	2000766565675	+94785654678	BIY-6765	Bike	4	Petroll	Non Active	Activate	Delete
1003	766753625215	+94766969123	BIM-5654	Car	24	Petroll	Non Active	Activate	Delete
1005	200000982731	+94760998231	BIN-1232	Car	24	Petroll	Non Active	Activate	Delete
1006	200000982731	+94760998233	BIN-1231	Car	24	Petroll	Non Active	Activate	Delete

The manager can view all customer requests for registration and approve or delete them as needed. The manager has the capability to control the access of customers to the system and maintain the records of registered customers. By reviewing the customer requests and their associated information, the manager can ensure that only legitimate and authorized customers are granted access to the system. This helps in maintaining the security and privacy of the fuel station's information and resources. Additionally, the manager can also monitor the fuel stock levels and track the usage of fuel by customers to ensure that the station's resources are being utilized effectively.

Customer deactivate

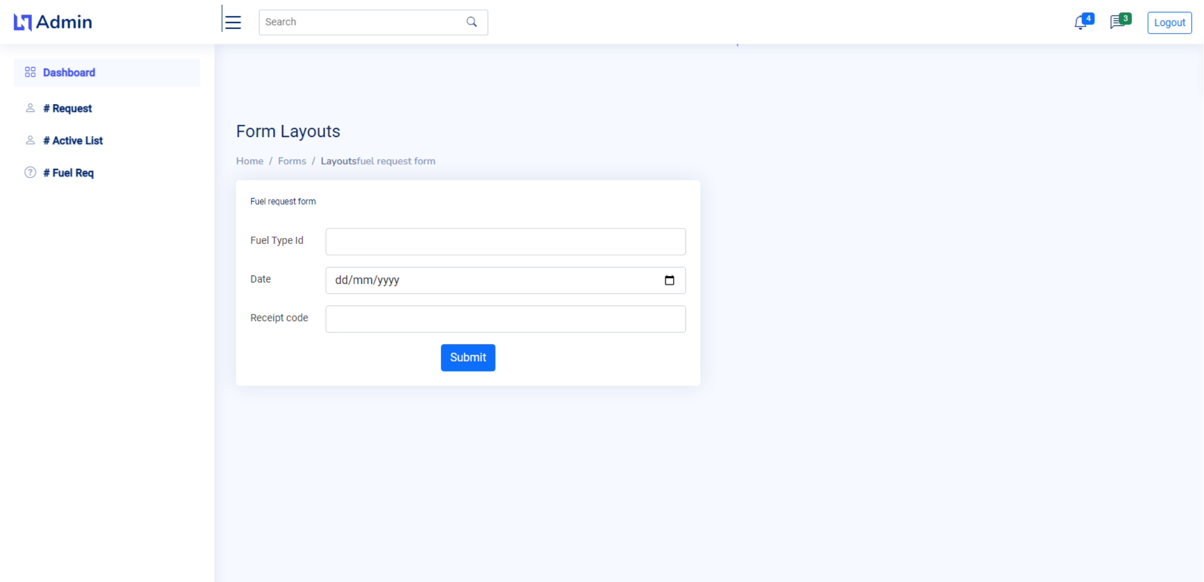


Add lightweight datatables to your project.

ID	NIC	Phone Number	Registration Number	Start Date	Type	Fuel	States	Activate	Delete
1004	200026704268	+94760743231	BIR-9442	Bike	4	Petroll	Active	Deactivate	Delete

In the manager dashboard, the manager has the ability to approve or decline customer requests, as well as deactivate or delete existing customer accounts. This provides the manager with complete control over the customer base, allowing them to ensure the security of the system and the accuracy of customer information. The manager can view the status of customer accounts and easily manage them, making the system efficient and user-friendly for both the manager and the customers.

Fuel Request Form



Form Layouts

Fuel request form

Fuel Type Id

Date dd/mm/yyyy

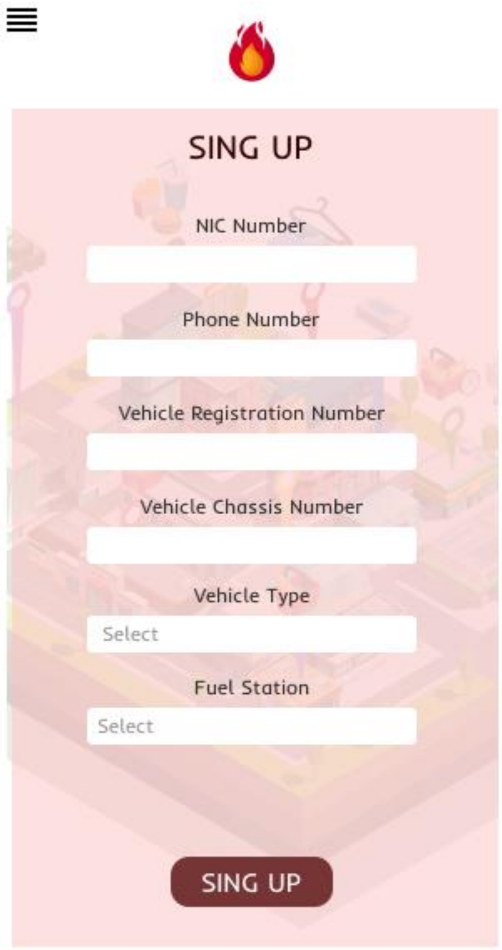
Receipt code

The manager can use the fuel request form to enter the fuel ID, date, and receipt code and submit it for review by the head office. The form should allow the manager to provide all necessary information about the fuel request, including the type and quantity of fuel needed, the date the fuel is needed, and any other relevant details. Once the form is submitted, the manager should be able to view its status and track its progress through the approval process. The head office can then review the request, approve or reject it based on availability and other factors, and take action to fulfill the request as needed. This process helps ensure that fuel is distributed efficiently and effectively, and helps the manager keep track of fuel usage and inventory levels at the fuel station.

Customer Mobile app

It's not specified in your question which platform the customer mobile app is being created for. However, if the app is being created using Java, it can be developed for both Android and other platforms. Java is a popular programming language that offers many features and benefits, including cross-platform compatibility, scalability, and a large community of developers. Additionally, Java has several libraries and frameworks available, which can make the development process more efficient and streamlined.

Mobile app Sing Up



SING UP

NIC Number

Phone Number

Vehicle Registration Number

Vehicle Chassis Number

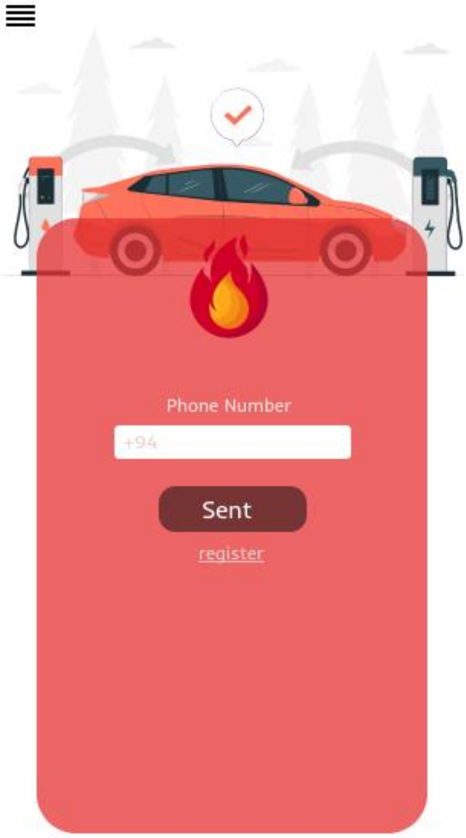
Vehicle Type

Select

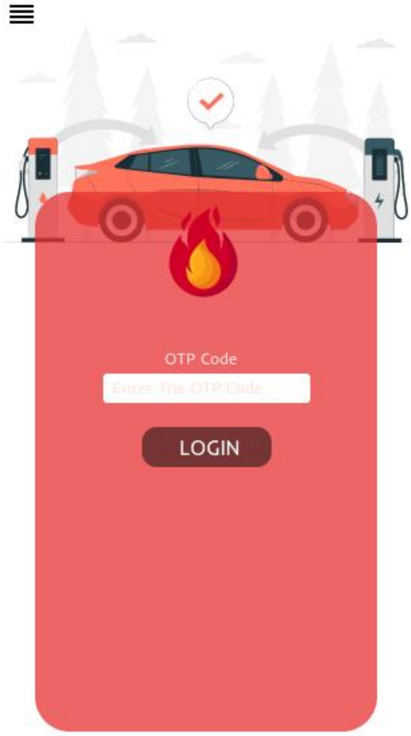
Fuel Station

Select

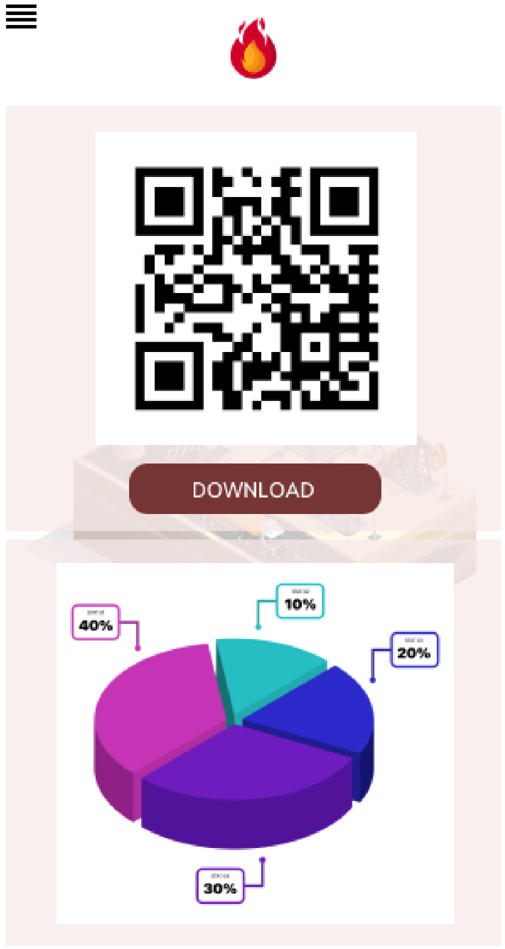
Mobile app Login Page (OTP Send Form)



Mobile app Login Page (OTP Login Page)



Mobile app Dashboard



Reference

"Waterfall Model." Wikipedia, Wikimedia Foundation, 29 Jan. 2021, en.wikipedia.org/wiki/Waterfall_model.

"What is the Waterfall Model?" Techopedia, 2 Jan. 2021, www.techopedia.com/definition/7726/waterfall-model.

Project Management Institute. (2017). A Guide to the Project Management Body of Knowledge (PMBOK Guide). 6th ed. Pennsylvania: Project Management Institute.

Scrum Alliance. (2021). Scrum Framework. Retrieved from https://www.scrumalliance.org/why-scrum/scrum-framework

Kanbanize. (2021). What is Kanban Methodology? Retrieved from https://kanbanize.com/kanban-methodology/