A PROJECT REPORT

on

Dharan Khanepani App: A Web and Mobile Platform for Public Administration and Utility

Submitted To:

Department of Information Technology

Central Campus of Technology

Hattisar, Dharan

In partial fulfilment of the requirements for the degree of Bachelor's of Information Technology (BIT)

Submitted By:

Arpan Shrestha [BIT 340/077]

Arun Khatri [BIT 341/077]

Rijan Rai [BIT 355/077]

[February, 2025]

A PROJECT REPORT

on

Dharan Khanepani App: A Web and Mobile Platform for Public Administration and Utility

Submitted To:

Department of Information Technology

Central Campus of Technology

Hattisar, Dharan

In partial fulfilment of the requirements for the degree of Bachelor's of Information Technology (BIT)

Submitted By:

Arpan Shrestha [BIT 340/077]

Arun Khatri [BIT 341/077]

Rijan Rai [BIT 355/077]

[February, 2025]

SUPERVISOR RECOMMENDATION

This is to recommend that **Mr. Arpan Shrestha**, Symbol No. BIT 340/077, **Mr. Arun Kc**, Symbol No. BIT 341/077 and **Mr. Rijan Rai**, Symbol No. BIT 355/077, has carried out Project Work (BIT404) on the topic of "**Dharan Khanepani App: A Web and Mobile Platform for Public Administration and Utility.**" in partial fulfilment of the requirements for the degree of Bachelors of Information Technology (BIT) under my supervision in the Department of Information Technology, Central Campus of Technology, Institute of Science and Technology (IoST), Tribhuvan University (T.U.), Nepal.

To the best of my knowledge, this work has not been submitted for any other degree. They have fulfilled all the requirements introduced by Institute of Science and Technology (IoST), Tribhuvan University (T.U.), Nepal for the submission of the project work for the partial fulfillment of Bachelor of Information Technology.

Mr. Prakash Neupane
Supervisor
Department of Information Technology
Central Campus of Technology
Dharan-14, Sunsari

DECLARATION

This project work entitled " **Dharan Khanepani App: A Web and Mobile Platform for Public Administration and Utility.** " is being submitted to the Department of Information Technology, Central Campus of Technology, Institute of Science and Technology (loST), Tribhuvan University (T.U.), Nepal for the partial fulfillment of the requirement to the project work in Bachelor of Information Technology (BIT) degree. This project work is carried out by us under the supervision of Mr. Prakash Neupane T.U., Department of IT, Central Campus of Technology, Institute of Science and Technology (loST), Tribhuvan University (T.U.), Nepal.

This work is original and has not been submitted earlier in part or full in this or any other form to any university or institute, here or elsewhere, for the award of any degree.

Arpan Shrestha [BIT 340/077]
Arun Khatri [BIT 341/077]
Rijan Rai [BIT 355/077]
Department of Information Technology
Central Campus of Technology
Dharan-14, Sunsari

CERTIFICATE OF APPROVAL

This project work entitled " **Dharan Khanepani App: A Web and Mobile Platform for Public Administration and Utility** " by Mr.Rijan Rai [BIT 355/077], Mr.Arun Kc [BIT 341/077] and Mr.Arpan Shrestha [BIT 340/077] and T.U. under the supervision of Mr.Prakash Neupane in the Department of Computer Science And Information Technology, Central Campus of Technology, Institute of Science and Technology (IoST), Tribhuvan University (T.U.), is hereby submitted for the partial fulfillment of the Bachelor of Information Technology (BIT) degree. This report has been accepted and forwarded to the Exam Section, Office of the Dean, Institute of Science and Technology, Tribhuvan University, Nepal for the legal procedure.

Mr. Prakash Neupane
Supervisor
Department of IT
Central Campus of Technology, Dharan
IOST, Tribhuvan University

Mr. Balabhadra Bhandari
Program Director/HOD
Department of IT
Central Campus of Technology, Dharan
IOST, Tribhuvan University

External Examiner

Mr. Sanjay Niroula
Internal Examiner
Department of IT
Central Campus of Technology, Dharan
IOST, Tribhuvan University

ACKNOWLEDGEMENT

First and foremost, we are extremely thankful to our supervisor, Mr. Prakash Neupane, for his invaluable guidance, unwavering support, and patience throughout our Project Work. We are truly grateful for the time and effort he has dedicated to our growth and development, and we will always cherish the lessons that we have learned from him.

Moreover, we would like to thank the members of the Department of IT, CCT, Dharan for their time, insights, constructive comments and suggestions from the beginning. We would like to express our appreciation to our colleagues and fellows who have provided us with intellectual stimulation and constructive feedback throughout this project. Their contributions to our learning have been immeasurable. We also indeed enjoy sitting in a vibrant geographical topology and a peaceful green campus amidst the foothills of Dharan sub-metro city with our great colleagues. Furthermore, We would like to appreciate help from each and every member of the Department.

We also offer our special thanks to the Office of Municipal Commission, Dharan Sub Metropolitan City for their encouragement. As well we are also deeply grateful to our family, and dear friends, who have supported us through thick and thin, and always encouraged us to pursue our dreams.

Finally, We are humbled and honored by the support and encouragement we have received, and we are grateful to all those who have contributed throughout this journey. We could not have done it without your help.

We always had persistent reminiscence.

Arpan Shrestha [BIT340/077]
Arun Khatri [BIT341/077]
Rijan Rai [BIT355/077]

ABSTRACT

This report presents the development and implementation of the Dharan Khanepani App, a digital platform designed to modernize and streamline the operations of the Dharan Water Supply Management Office. The system bridges the gap between traditional methods and digital solutions by integrating a web application for administrative tasks and a mobile application for customer interaction. The web application is tailored for the water management board, offering features such as water distribution scheduling, customer and employee record management, enquiry handling, billing, and monitoring reservoir water levels. The mobile application, on the other hand, provides customers with access to personalized water distribution schedules and a platform to report issues directly to the water office.

The system aims to enhance communication, improve record-keeping, and optimize operational efficiency, creating a more organized and user-friendly approach to water supply management. The web application was developed using Laravel for backend functionality and Tailwind CSS for styling, while the mobile application was created using Android Studio with Java and XML for development. The Dharan Khanepani App represents a significant step toward digitizing municipal services, fostering better interaction between the water office and its customers.

Keywords: Laravel, Java, XML, Android Studio, Dharan Water Supply Management Board, Tailwind CSS

CONTENTS

LIST OF ACRONYMS AND ABBREVIATIONS

ADB Asian Development Bank

API Application Programming Interface

CSS Cascading Stylesheet

DWSS Department Of Water Supply And Sewerage

DWSSM Department of Water Supply and Sewerage Management

DWSMB Dharan Water Supply Management Board

GoF Government of Finland

GoN Government of Nepal

GSM Global System For Mobile Communication

HTML Hypertext Markup Language

IDE Integrated Development Environment

IoT Internet of Things

JS Javascript

MySQL My Structured Query Language

PHP Hypertext Preprocessor

REST Representational State Transfer

RWT Remote Water Tank

SCADA Supervisory Control And Data Acquisation

SDK Software Development Kit

STWSSSP Small Town Water Supply and Sanitation

SUSWA Sustainable Wash For All

WASH Water, Sanitation And Hygiene

XML Extensible Markup Language

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

1.1 Introduction

In 21st century, urban water supply has become one of the most pressing issue, driven by various factors such as increasing population, changing lifestyle, economic growth and development[1].According to the Department of Water Supply and Sewerage Management (DWSSM), only 51.69% of the population have piped water coverage, while 48.31% rely on non-piped locally and privately managed systems such as private tube wells (Department of Water Supply and Sewerage Management (DWSSM) 2019 [2].
The Dharan Water Supply Management Board (DWSMB) is responsible for overseeing all water distribution and administration in Dharan. Dharan is one of the most populous cities in Nepal, with an estimated population of 1,73,096, divided into 20 different wards [3]. Being a populated city, the initiative of building a digital platform for the mass audience seems viable. Dharan Khanepani application is an innovative and creatively designed prototype that can be beneficial in easily maintaining customer and employee records and providing timetable information regarding the availability of water in different wards of the Dharan. Residents in different wards can access the water availability schedule on their respective mobile devices. The customer may face different types of problems or may have different types of questions, queries, and complaints about the water supply. The proposed application can improve communication, as each user can send the message to the board through their mobile devices. This reduces communication gaps between the residents and the water supply authority board of Dharan.
Our Khanepani app consists of a web-based application and a mobile-based application. The Web-based version supports the administrative task for the respective authority, while the mobile-based version is intended for customers who consume water on a daily basis. In our Khanepani App, registration is a must for both admins and customers. After successfully registering, they can use their registered account to log in to the system and subsequently access the functions of our application. Since our system runs on the Web and Android [4], it can be accessed from anywhere in the world. Our system provides the feature of viewing the total revenue. Customers are required to enter their meter ID and phone number to log in, after which the system automatically fetches the water schedule for their specific ward. This ensures the better usability and user friendliness as the customer does not need to have any technical knowledge or experience to use it. All they need is an Internet connection. With this system, the user with administrative privileges can analyze the customer distribution across various wards.This allows them to analyze which ward consumes more water, thus simplifying the water scheduling task.

1.2 Problem Statement

With modernization of the world, everything is updated to digital form. In developing countries like Nepal itself, the IT sector has been a crucial field for development. Traditional methods of managing water supply in Dharan have led to inefficiencies, poor communication, and inadequate record keeping. Dharan has been lacking a digital platform that provides valuable and precise information to the respective authority about the entire water distribution scenario in the various wards. There is no such digital platform that allows the customer to communicate their issues with the authority quickly and easily. The people of Dharan frequently face problems with the irregular water supply schedule. These issues highlight the need for a digital system or application that can address these issues.

Our Dharan Khanepani App is intended to assist the respective authority in assessing the water distribution pattern, managing the time-table for water supply, and addressing the problems issued by the customers. It provides the authority with a diagrammatic representation of the number of customers residing in various wards,thus helping the authority identify the dense consumption area and schedule the water as needed. Our system allows the authority to view a generalized overview of the water distribution across various wards in a digital format. With our Khanepani App, customers can easily view the schedule of water, issue the problems directly into the admin panel's inbox section. Our system allows the customers to get in touch with the authority all the time, fostering better communication.This helps prevent the misunderstanding among the customers about the water routine as each customer gets the information from the authority in their respective mobile devices.

1.3 Objectives

The main objective of our proposed system/application is to provide residents with reliable water supply information. Along with it some of the main objectives of our application are mentioned below:

1. To digitalize administrative process.
2. To provide personalized water schedules.
3. To enhance customer interaction
4. To improve operational efficiency, accountability and water management.

1.4 Scope and Limitation

The system we build aims to provide reliable information and schedule water supply.The initiative focuses on the following key areas:

1. Keeping records of employees and customers
2. Billing and displaying the statistics in charts.
3. Monitoring the reservoir, sharing the scheduled time of water availability to consumers.
4. Public communication,engagement and awareness.

The system developed is simply a partial study for the fulfillment of Bachelor's degree, which has to be finished within limited time period. Although, the significant effort has been gone to preparing this research report, there are certain limitations that have impacted the study. Some of these are outlined below:

1. **Absence of IoT Devices, sensors or Supervisory Control And Data Acquisation (SCADA) Systems**: The project does not incorporate Internet of Things (IoT) devices, sensors, SCADA, or other technological instruments that can measure and transmit real-time water levels. The software is designed based on manually entered data rather than real-time inputs from water tanks or distribution points.
2. **Dependence on Assumed and Provided Data**: The time frames for the water distribution are also determined based on the population and demand assumptions, which may vary from the actual consumption patterns, and the water distribution logic is based on estimates rather than real-time consumption data. Additionally, the incoming and outgoing water flow data are based on historical records or estimates provided by the local water supply board, which may not always reflect real-time conditions. The price per unit of water displayed on the app may not always reflect the actual price.
3. **Geographical factors**: The study is general and may not cover the details of each area or ward of the Dharan.
4. **Policies**: The report may only lightly cover the laws and policies regarding water distribution.

1.5 Development Methodology

For the development of our app, we used the iterative development model [5].It is a part of an agile framework. The iterative development model was chosen because our project

required flexibility and adaptability. The system we developed has gone through multiple iterations and cycles, with each iteration refining or updating the existing system. The requirement was not well understood at the beginning of our project. So, the use of this model in our project development has been beneficial, as it was initially understood that the requirements would evolve over time. Use of this approach has allowed us for making incremental and continuous development of the system.
The iterative model [5] allows for user testing and validation in each iteration, ensuring that the final product meets the expected outcomes. The major phases and stages involved in the iterative development model are explained below in a figure:

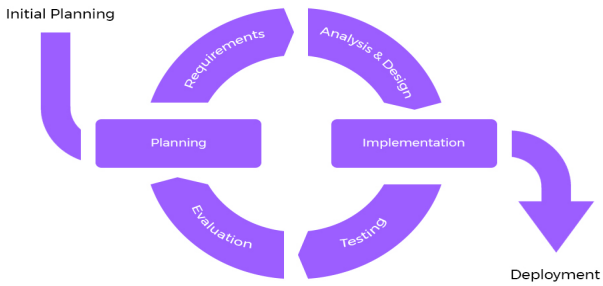


Figure 1.1: Iterative Development Model

1. **Initial Planning and Analysis**: In this stage, developers, along with the respective clients, identify the necessary requirements for the development of the system. After each of the requirements are well understood and collected, the analysis is conducted to ensure faster and simpler flow of the development processes.

2. **Design**:This stage includes the development of user interfaces, layouts, along with the structuring of the back-end system. A better and a simpler UI define if a system is easier to navigate or use.

3. **Implementation**: This stage is where the actual implementation of the proposed system takes place. Processes like coding, programming usually lies within this stage. The programmers follow different coding guidelines for the coding phase.

4. **Testing**: This phase is crucial in the system development process. Different testing methodologies help developers debug errors like usability issues and eliminate these errors out of the system ensuring a development of the reliable system software.

5. **Evaluation**: This phase is the final step in iterative development methodologies that

help developers identify areas for improvement to make the system more effective and reliable.

1.6 Report Organization

The report is organized into six chapters. The first chapter introduces our project, covering the problem statement, objectives, and limitations, as well as the organization of the report and the development methodology. The second chapter focuses on background studies and the literature review. The third chapter addresses the requirement analysis and feasibility analysis which further covers the details about the functional requirements and non-functional requirements. The fourth chapter explains the algorithms used in the project. Chapter five details the implementation and testing of the system, while the final chapter, chapter six, presents the conclusion and future recommendations.

CHAPTER 2: BACKGROUND STUDY AND LITERATURE REVIEW

2.1 Background Study

For many years, the Dharan city has faced many problems related to water supply. With more increase of the population and increasing of the number households it has become more challenging to the the Dharan water supply board to overcome the problem [3]. The board has been using outdated technology which has been a main issue with the communication between the public and the administrative board which has lead to public dissatisfaction.

With no any modern application systems the public has been remained unaware about the reliable information such as water schedule. The public has been using out dated system or technology to file a complain to the respective board. With the presence of this challenges and problems the board has recognized the need of an integrated system to overcome these challenges. The implementation of a Dharan Khanepani App that includes the features like managing customers according to their specific ward , managing all the employees, providing a schedule, the system aims to ensures the proper schedule along with reliable information to the customers.

2.2 Literature Review

In this subsection literature from various journals, articles, books, and seminar papers has been studied and reviewed.

1. **Sustainable WASH for all:** Sustainable Wash For All (SUSWA) is a human rights progressive bilateral project building on and continuing the long-term Water, Sanitation And Hygiene (WASH) sector cooperation of the Government of Finland and the Government of Nepal [6]. Sustainable WASH for all, SUSWA, is a bilateral human rights progressive Water, Sanitation, and Hygiene project funded by the Government of Nepal (GoN) and Government of Finland (GoF), as well as the European Union [6].

    - **Impact Statement:** The SUSWA project aims to enhance well-being and foster inclusive communities by promoting sustainable WASH services and behaviors while strengthening local governments' capacity to ensure equal WASH rights for everyone.
    - **Outcome Statement:** The project seeks to improve equitable access to safe drinking water, sanitation services, and hygiene practices, with a particular

focus on the needs of women, girls, and vulnerable groups, supported by the municipalities involved.

- **Sensor Dashboard Tank Status Overview:** SUSWA is implementing sensor dashboards to enable municipalities to remotely monitor the functionality of water systems and determine the necessary technical or financial support. These sensors, utilizing ultrasonic technology, measure water levels in tanks and transmit data via the local Global System For Mobile Communication (GSM) network. At regular intervals, they automatically report the water level in the tanks, which is then displayed live on a web-based interactive dashboard in both volume and percentage.
- The dashboard, accessible publicly through the provided link, offers detailed insights, including the number of monitored tanks, their locations, capacities, current water levels, and usage trends. Additionally, the dashboard features an Open Street Map with Remote Water Tank (RWT) location points, color-coded according to hourly data updates.
- The limitation of the SUSWA system is that it serves as a collective platform for multiple small municipalities with a broad focus on water, sanitation, and hygiene. As a result, it requires substantial funding for its development and implementation. In contrast, our system is designed to be more cost-effective, with a primary focus on optimizing water distribution. This targeted approach allows for significant cost savings while ensuring efficient and equitable water allocation.

2. **Small Town Water Supply and Sanitation Projects-I, Lekhnath Water Supply:** [7] This project aimed to provide high standards of water supply services to about 240,000 people in about 29 small towns. The Lekhnath Small Town Water Supply Project, known as Lekhnath Khanepani, is one of the eight pilot projects proposed in the first phase of Small Town Water Supply and Sanitation (STWSSSP) implemented by Department Of Water Supply And Sewerage (DWSS). It focuses on improving Water Supply Management with an Innovative System. In the town of Lekhnath in western Nepal, a GeoViewer asset management system has been installed, allowing the Water Users Committee to effectively manage and monitor their resources, plan maintenance and repairs, and keep track of asset performance in real-time.

    - The Lekhnath utility meter-readers visited nearly 16,000 customers and geo-positioned every meter, including the meter account number. This allows the asset management system to be linked to the utility's billing system. The asset management system has drastically cut the time it takes to address payment,

water quality, pipe leakage, and meter issues.

- Asian Development Bank (ADB) and the Government of Nepal are working together to supply water and sanitation services to small towns in Nepal. The asset management system has been financed by the High-Level Technology Fund, which brings in innovative and digital technologies into ADB projects. A cloud-based SCADA system has also been installed to monitor tank levels.
- Another feature of GeoViewer is the Customer App, allowing customers to report issues from their smartphones, such as a pipe leak, burst, or meter fault, with the accurate location and the option to upload a photo.
- The Lekhnath system enhances asset management but does not focus on ensuring equitable distribution of limited water resources, especially during periods of scarcity.

CHAPTER 3: SYSTEM ANALYSIS

3.1 System Analysis

3.1.1 Requirement Analysis

i. **Functional Requirements**

(a) Admin Dashboard

- Summarizes the entire application data and presents it in a dashboard including:
  - Water level
  - Incoming rate
  - Outgoing rate
  - Total number of enquiries
  - Total number of customers and employees
  - Revenue generated

(b) User Registration and Login/Logout

- Users can create accounts and log in securely.

(c) Message Inbox/Outbox Section

- Users can receive and send messages related to complaints and inquiries within the app.

(d) Billing

- Automatically calculates customer bills.

(e) Customer and Employee Management

- Manage customer profiles.
- Manage employee details, such as roles and responsibilities.

(f) Schedule Management

- Plan and manage schedules for water distribution.
- Provide updates to customers about scheduled timetables.

(g) Displaying Statistics in Graphs

- Visualize population across different wards and toles.

ii. **Non-Functional Requirements**

(a) User-Friendly Interface

- Clear navigation menus, search functionality, and a clean layout enhance usability.

(b) Availability

- Ensure 24/7 availability with minimal downtime.

(c) Security

- Allowing only authenticated users.

3.1.2 Feasibility Analysis

i. **Technical** :Following technologies have been chosen for the project:

(a) Laravel - For the development of necessary Application Programming Interface (API)s and other backend related works the Hypertext Preprocessor (PHP) [8] framework Laravel has been used for this project.

(b) MySQL -To record the necessary information, data or records the system is intended to used to store a database technology MYSQL has been used for this project.

(c) Android Studio - The project also includes the mobile application for the client to interact with the board and to know the water supply timetable. For the development of the mobile application Android Studio(Java) has been utilized and it includes Android Software Development Kit (SDK) which provides all the necessary tools and APIs(RESTful) [9] to build android apps.

(d) Github -For the project Github technology has been used to manage and for the version control of the application.

(e) VS Code - Primary editor for coding and development tasks.

(f) LaTeX - For preparing the documentation and reports.

ii. **Operational**: The absence of a digital application at the Dharan Water Supply Management Board DWSMB presents an opportunity for significant improvement in daily operations. By integrating this system into the office, staff will benefit from an intuitive user interface, making the application easy to operate without requiring intensive training. The mobile app also enables customers to access schedules and submit inquiries effortlessly, promoting user engagement. This streamlined digital solution will enhance operational efficiency and support effective administration, ensuring smooth management of water supply-related tasks.

iii. **Economic**: From a development perspective, the primary cost was not financial, but rather the technical expertise, hard work, and effort of the development team,

consisting of three members. Key costs included investing in decent laptops, ensuring good internet speed for development, and dedicating time and resources to overcome technical challenges. Furthermore, expenses related to traveling around Dharan for data collection and attending meetings for discussions contributed to the overall costs.

In terms of return on investment (ROI), one potential revenue model is selling the software to the Dharan Municipality, which could generate revenue from the system's deployment. Also we could charge the recurring maintenance fees for updates and additional services.

iv. **Schedule**: The project is expected to last for 16 weeks, starting from the 1st week of Bhadra and ending in the last week of Mangshir. The complete schedule of how the system is being is developed is shown below with the help of Gnatt chart.

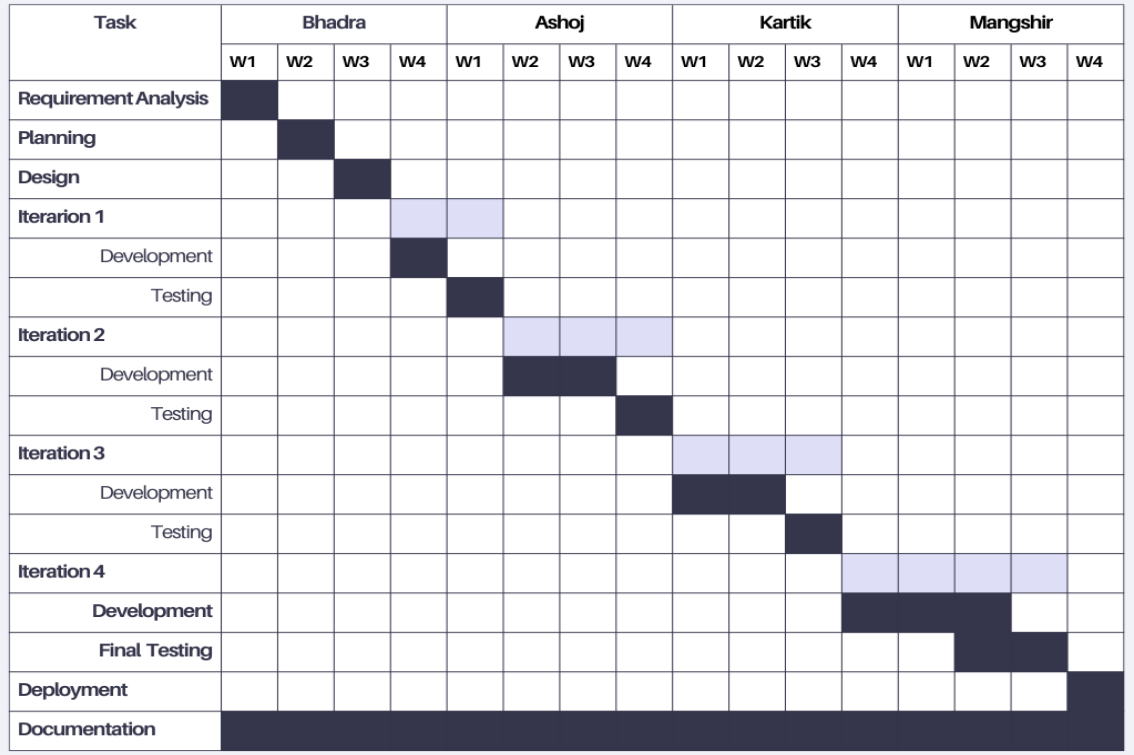


Figure 3.1: Gnatt chart

3.1.3 Analysis

The analysis section of this prepared report shows the overall structure of the Dharan Khanepani App. This section provides the detail working and it's overall implementation process by using different modeling tools which is explained below along with the necessary diagrams of each.

i. **Object modeling**: The purpose of object modeling is to visually show the structure of the overall system and the relationships of different object with the system. This model helps to depict how the data flows between different entities of the system.

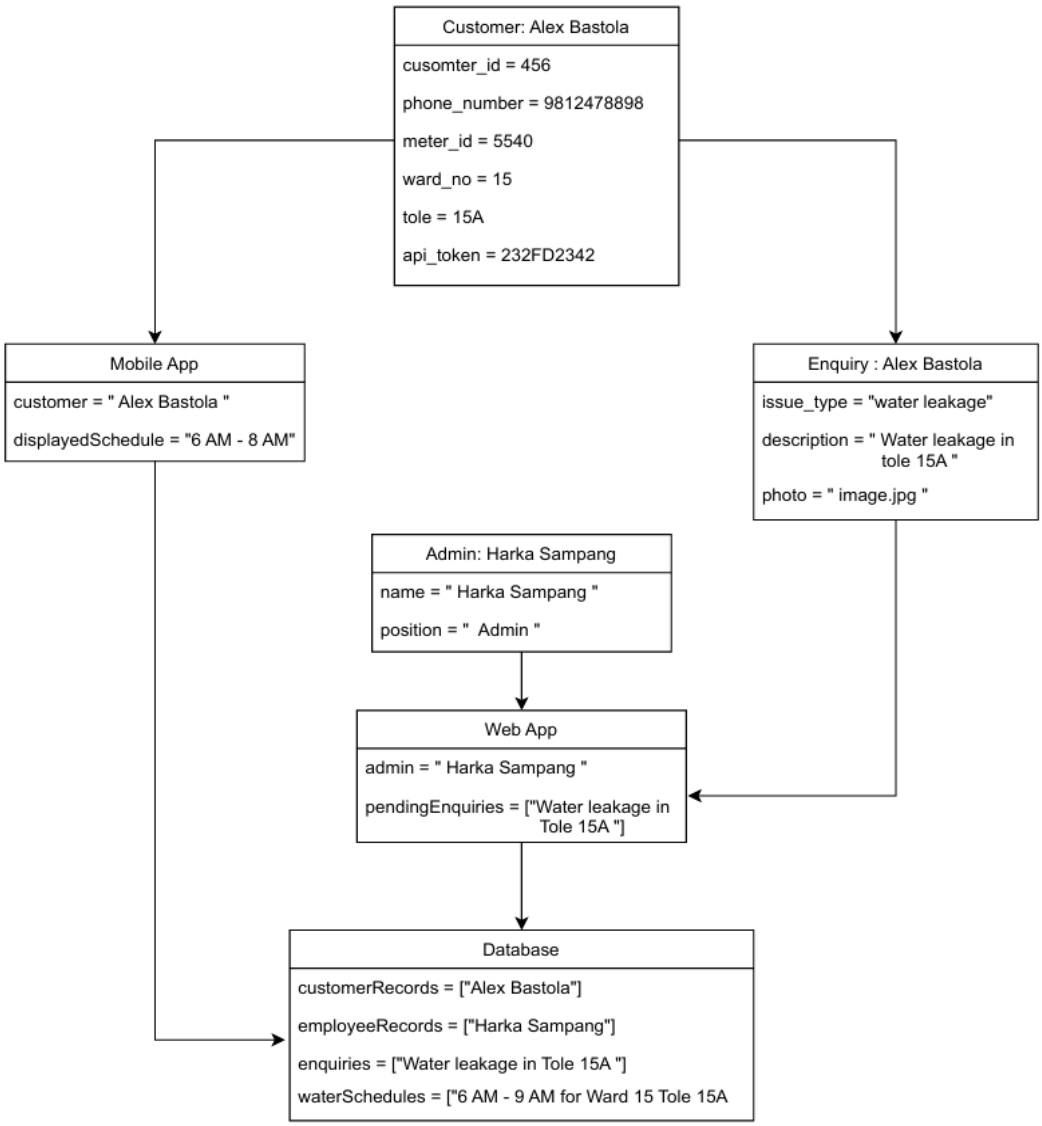


Figure 3.2: Object modeling

ii. **Sequence Diagram**: The sequence diagram shows different process interactions which are arranged according to the time sequence. This diagram shows the flow of different process and messages between entities to perform a certain functionality. Since this project includes both mobile app version and web application, the sequence diagram for both platform is shown below:

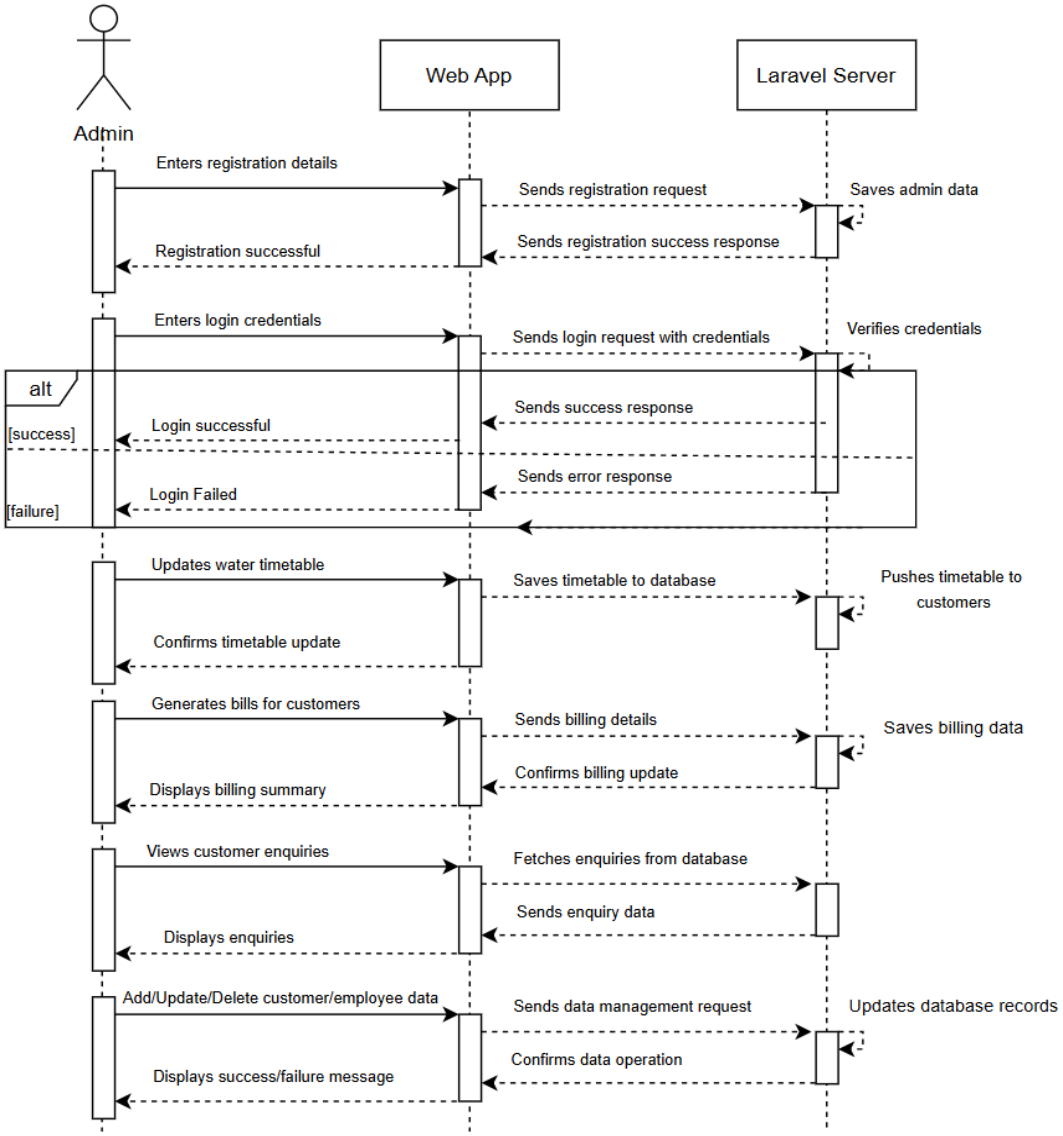


Figure 3.3: Sequence Diagram For Web App

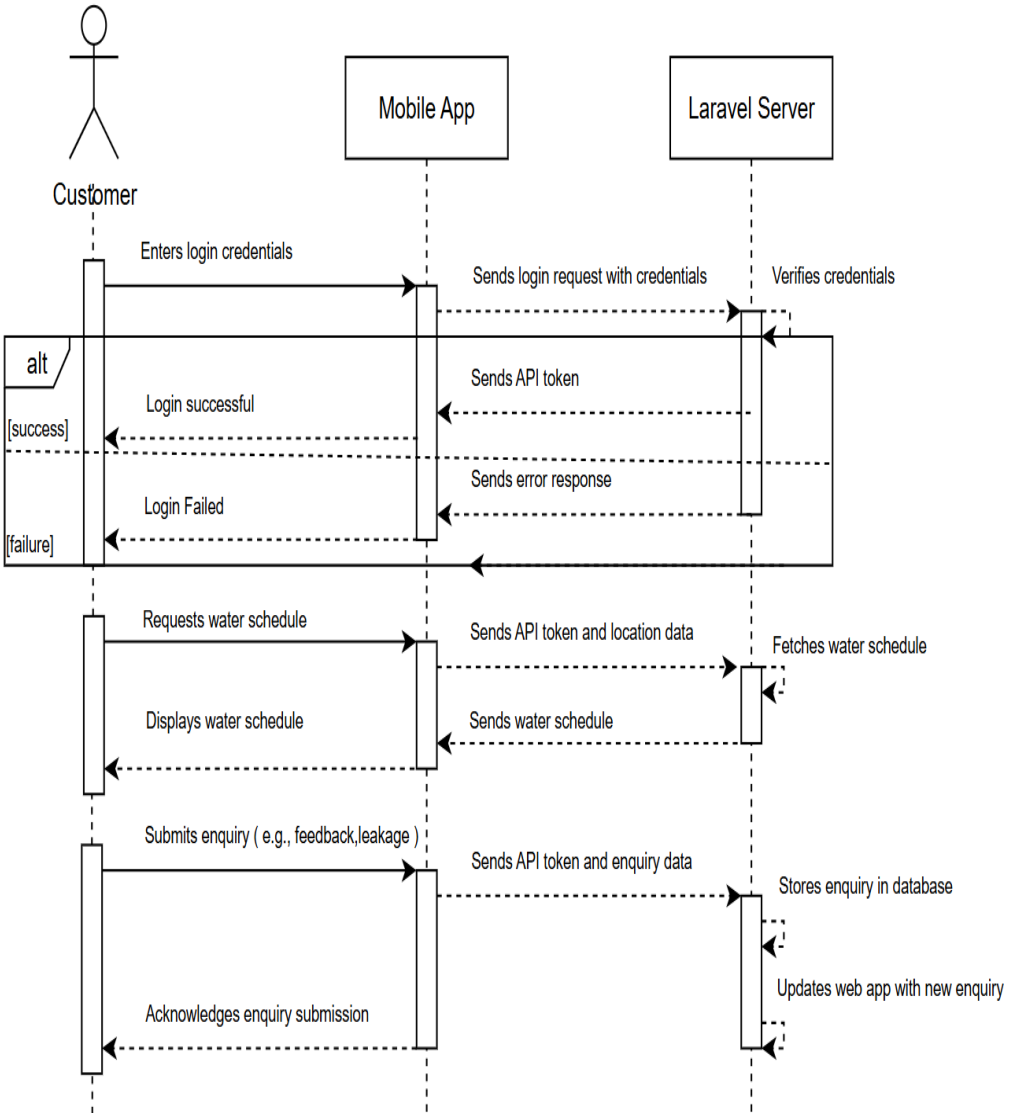


Figure 3.4: Sequence Diagram For Mobile app

iii. **Activity diagram**: An Activity Diagram is designed in such a way that it illustrates the way activities or processes flow between different system components. It aids by including information about how various tasks or activities are performed and how these are coordinated in the system. The activity diagram for the web application and the mobile application is shown below:

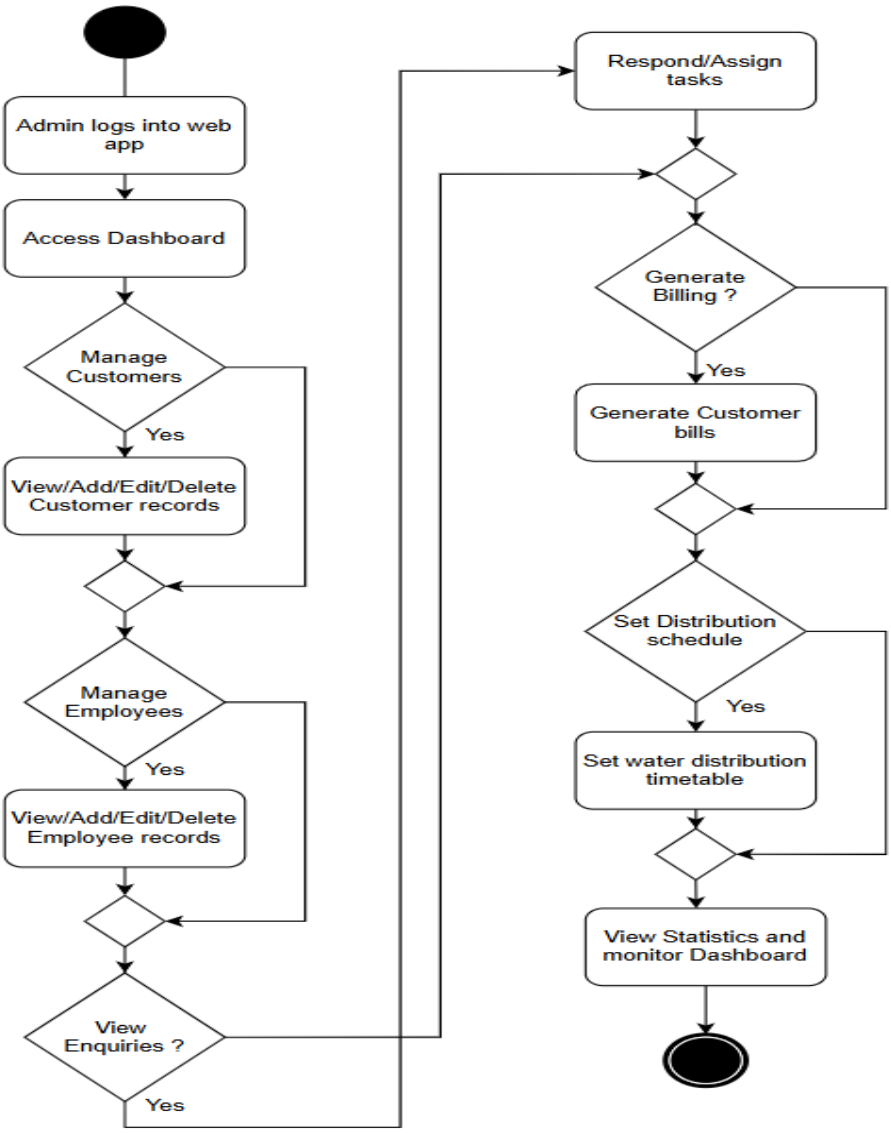


Figure 3.5: Activity Diagram For Web App

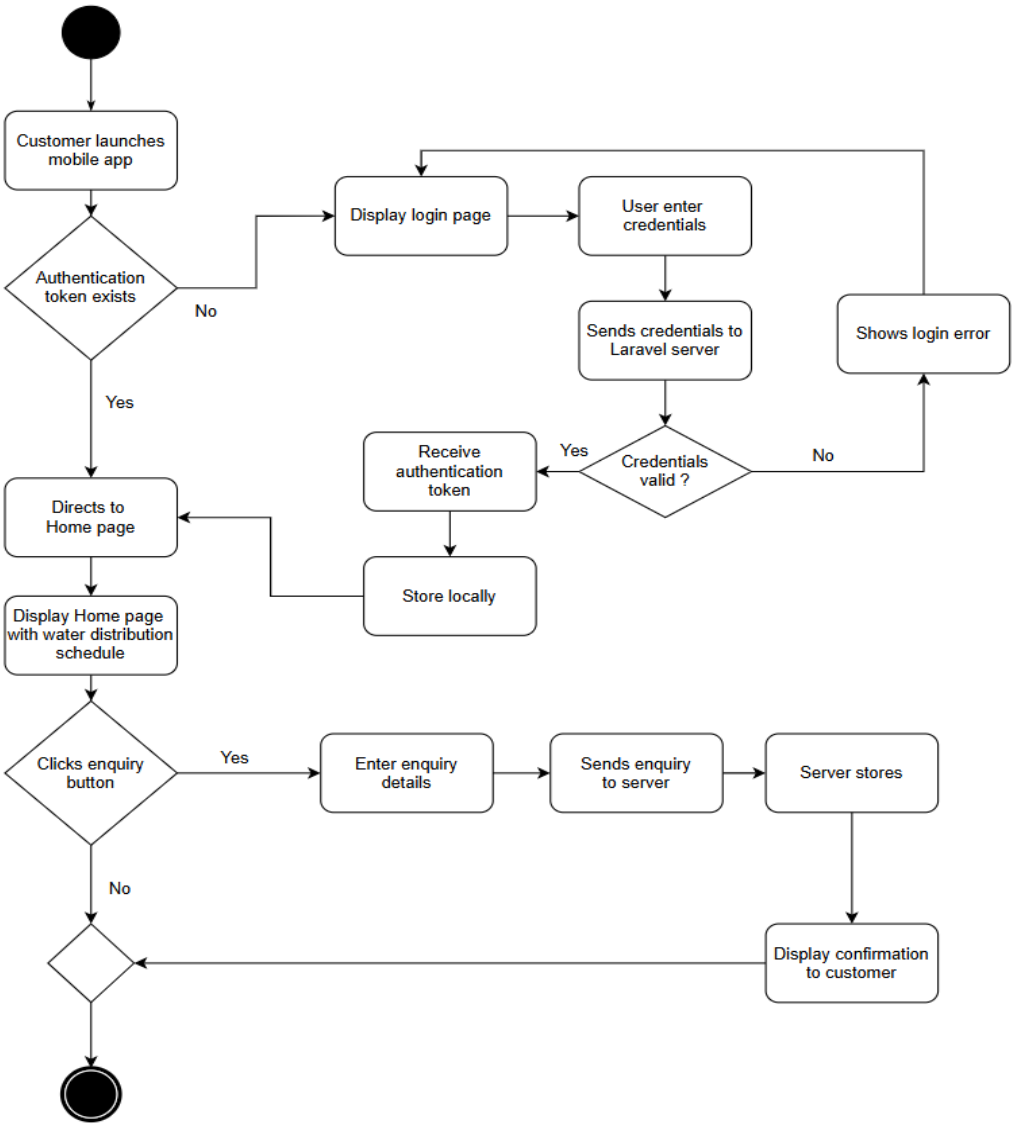


Figure 3.6: Activity Diagram For Mobile App

CHAPTER 4: SYSTEM DESIGN

4.1 Design

The design section describes the overall design pattern of system. Different kinds of visualizing tool like class diagram, use case diagram, state diagram have been use to properly show the design of the system. some of the tools used in this project is explained below along with their respective diagrams.

i. **Class Diagram**: It defines the overall system design by including all the classes along with the methods used ,relationships between classes and their attributes. The class diagram includes different components like:

- Classes
- Attributes
- Methods
- Relationships

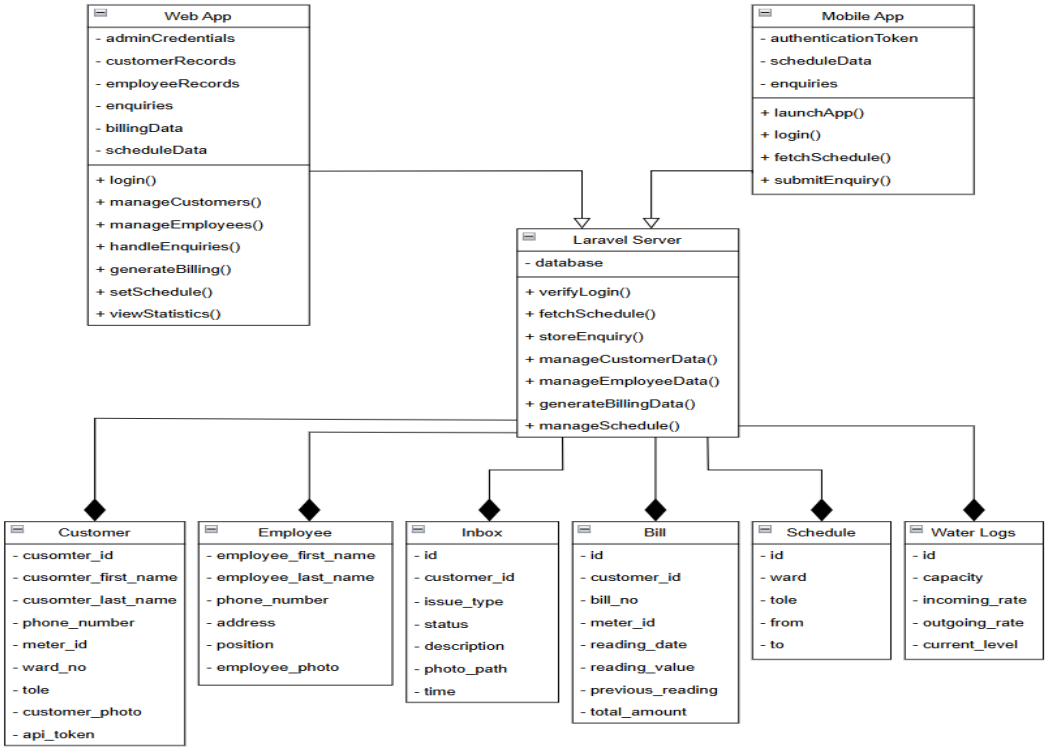


Figure 4.1: Class Diagram

ii. **Database schema**: The database diagram shows the complete structure of the underlying database. It explains the structure of the database schema along with the

relationships between different tables. The key components of the Database schema are:

- Tables
- Columns
- Primary Keys
- Foreign Keys

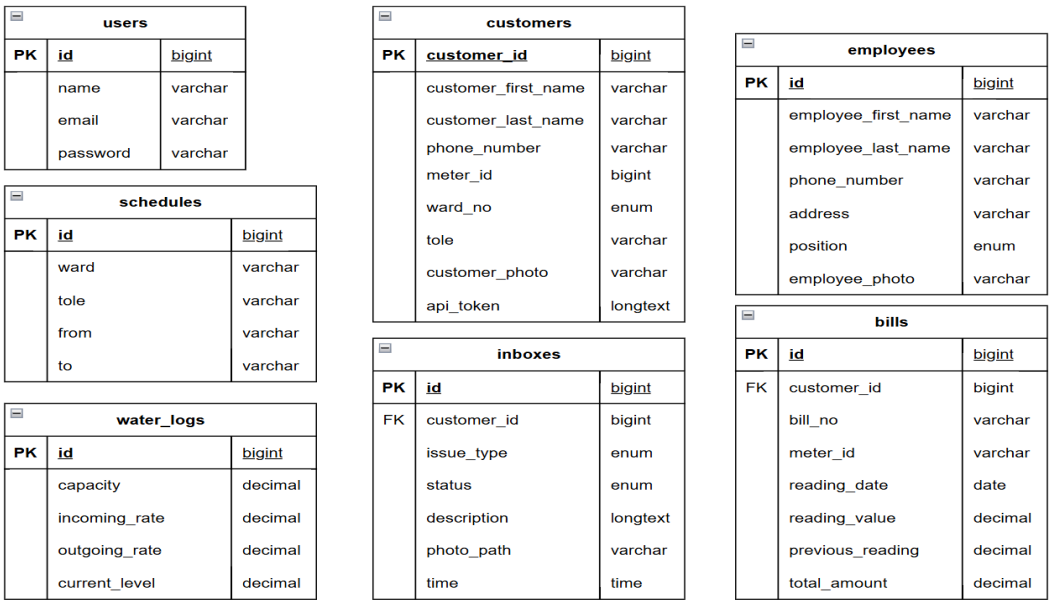


Figure 4.2: Database Schema

iii. **Use case Diagram**: The use case diagram shows how our proposed systems interacts with it's external environment. This diagram visually shows the interaction of the system with the users. It also shows various types of users having different privileges that tries to interact with the system. The different components of the use case diagram are:

- Use cases
- Actors
- System Boundary
- Relationships/Associations

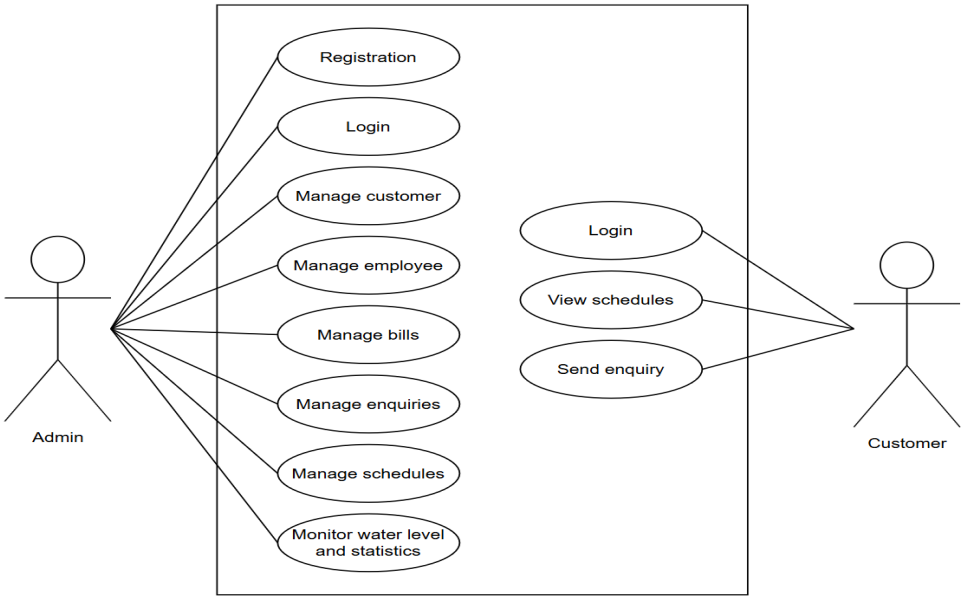


Figure 4.3: Use Case Diagram

iv. **State Diagram**: State diagram shows the behavior of the system. It explains that the system is composed of finite number of states. It is used to represent the condition of the system at a finite instances of the time. It defines how an underlying object of the system transition from one state to another state depending upon the events that occurs. Different components of the state diagram are:

- Initial state
- Final state
- Sub-states
- Transitions
- Events
- Conditions

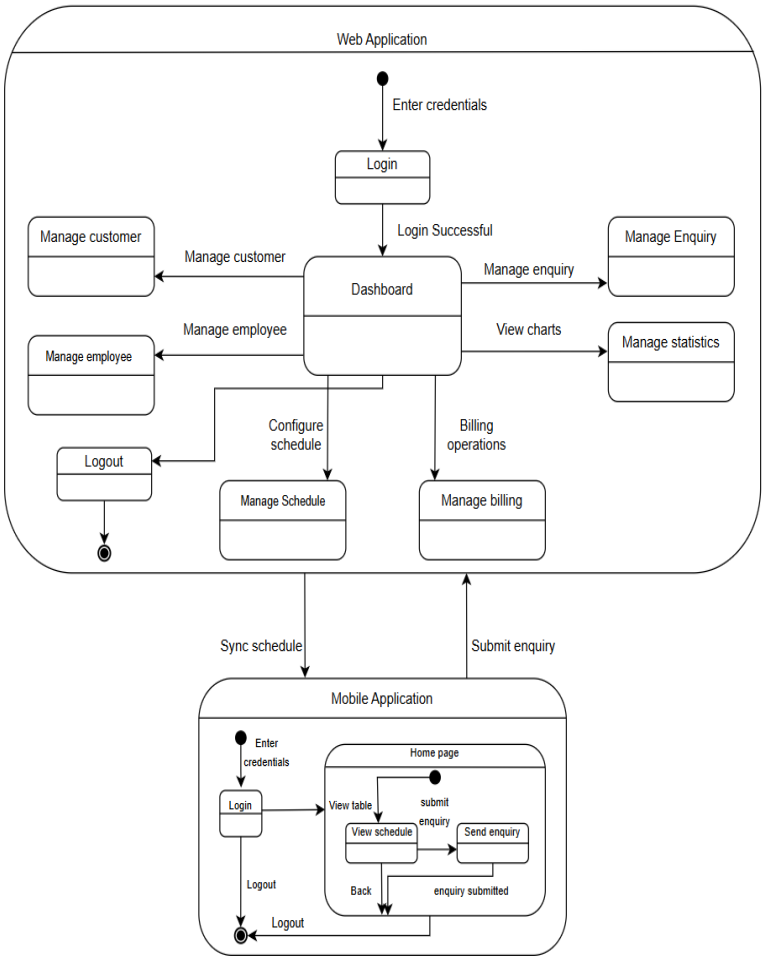


Figure 4.4: State Diagram

v. **Component Diagram**: This diagram shows all the software components of the system and also assists developers to understand how different parts of the system interact with each other. Different components of the component diagram are:

- Components
- Interfaces
- Dependencies

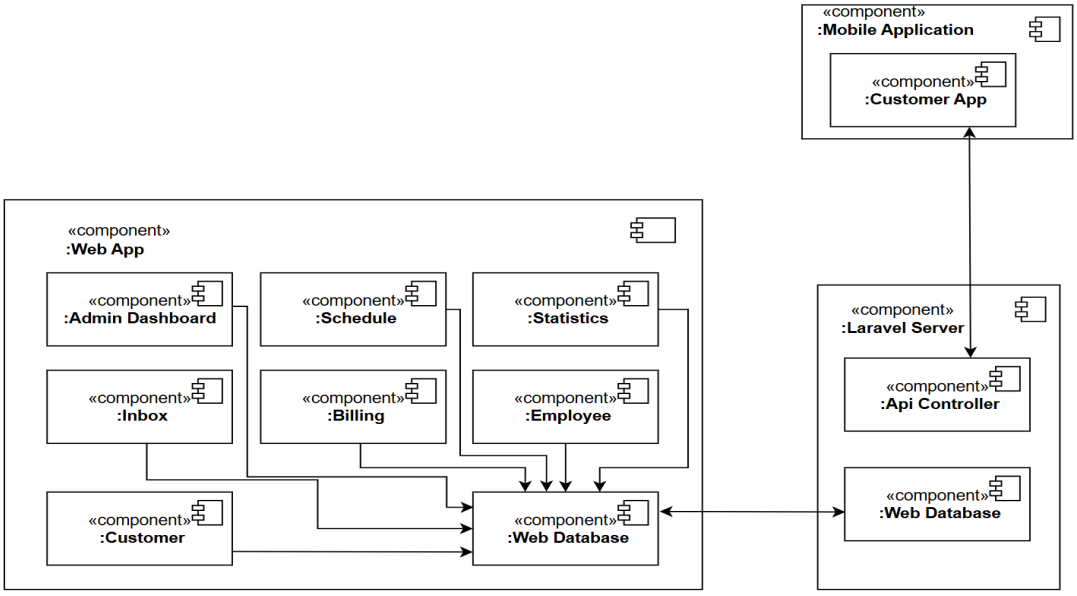


Figure 4.5: Component Diagram

vi. **Deployment Diagram**: The deployment diagram shows the overall physical architecture of the system including the hardware and the software components. It also helps developers to manage or configure the servers and networks. Key components of the deployment diagram are:

- Nodes
- Artifacts
- Connections

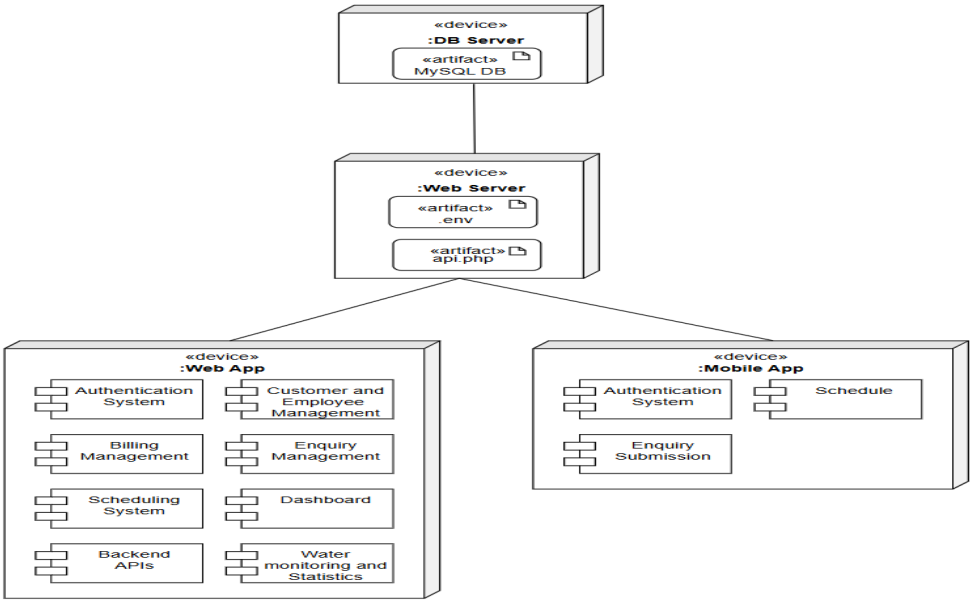


Figure 4.6: Deployment Diagram

4.2 Algorithm Details

All the algorithm used in this project is explained below with their details:

1. **Token Validation Algorithm**

   (a) Check the presence of the Token:

   - If a token already exists :
     {
     The user is already logged in.
     }
   - Otherwise:
     {
     Redirect to the login form.
     }

   (b) Check the Credentials of the user:

   - User Inputs their Login Credentials:
     {
     The user enters their login ID and their password.
     }
   - If the credentials are valid:
     {
     Sends an authentication token to the mobile app and log in.
     }
   - If the credentials are not valid::
     {
     Display an "Invalid credentials" message.
     }

   (c) Logout Action:

   - When the user clicks the logout button:
     - The authentication token is removed session.
     - The user is logged out.
     - Redirect to the login form.

2. **Bcrpt- cryptographic hash function**
   In Laravel, hashing of passwords is a crucial process to securely store the user credentials in respective database. The Laravel technology uses the Bcrypt algorithm [10] for password hashing. The user hash a password using the Hash::make()

method. Bcrypt runs a complex hashing process. During the process users password is transformed into a fixed-length thread of characters. It uses a one-way hash function, meaning that once the password is hashed, it cannot be roll-back to its original form.

To hash passwords, bcrypts password hashing algorithm combines the password string, salt, and the cost to derive a 24-byte hash using base 64 encoding. The algorithm is explained below with the steps that is involved in the process.

(a) Registration (Storing the Password)

i. User enters: password123

ii. Laravel hashes it:
$hashedPassword = Hash::make('password123');
Example hash: $2y$10$eDi6K1RoOQbc1uF4F9Z
EUOvJFc1uBf7GUjGFlRgHYzTk7.q3NghE6

iii. 2y: bcrypt identifier.

iv. $10: The cost factor (workload).

v. eDi6K1Ro...: Salt + hashed password.

vi. This generated hashed password is stored in the database.

(b) Login (Verifying the Password)

i. User enters: password123

ii. Laravel retrieves the stored hash: $2y$10$eDi6K1RoOQ
bc1uF4F9ZEUOvJFc1uBf7GUjGFlRgHYzTk7.q3NghE6

iii. Laravel checks the password: Hash::check('password123',
$storedHashedPassword);

- Extracts the salt from the stored hash.
- Hashes the input password (password123) with the same salt.
- Compares the resulting hash with the stored hash.

iv. If the hashes match: Log In successful

v. If the hashes dont match: Log In Failed.

3. **Linear Search Algorithm**:

A linear search algorithm [11] is one of the basic searching techniques for finding a specific element from among groupings of elements (such as an array, a list, or a table). In this algorithm, each element is either checked or analyzed until the required one is found or until every element in the collection has been traversed. In this system, the search feature allows users to find specific data in a table by entering search terms like username, phone number, or any other detail. The feature

is implemented using the DataTables plugin with Tailwind Cascading Stylesheet (CSS) [12] for styles and for frontend interactivity. The Working of the linear search algorithm is explained below:

(a) Begin with the first element in the collection.

(b) Compare the current element with the search key (the value you're looking for).

(c) If a match is found, return the index or the element itself.

(d) If no match is found after checking all elements, return a result indicating the item does not exist in the collection.

CHAPTER 5: IMPLEMENTATION AND TESTING

5.1 Implementation

5.1.1 Tools Used

Following are the list of tools that are used to develop this project:

(a) **Front-end tools**

i. Blade

ii. Tailwind CSS

iii. Javascript (JS) [13]

iv. Java [14]

v. Extensible Markup Language (XML) [15]

(b) **Backend tools**

i. Laravel [16]

(c) **Database**

i. My Structured Query Language (MySQL) [17]

(d) **Integrated Development Environment (IDE)**

i. Android Studio

(e) **Code-editor**

i. Visual Studio Code

(f) **Report editor**

i. LaTex

(g) **Version Control**

i. Github [18]

(h) **Data visualization tool**

i. ChartJS [19]

(i) **UI/UX Design**

i. Figma

5.1.2 Implementation Details of Modules

The modules are the units and the partition of an project to make the development process more easier. The partitioning of the modules also eases the debugging process which helps in finding any bugs and errors present in the system. The modules used in this system is explained below with their implementation methods:

(a) **Admin Dashboard Module**

i. Purpose: Displays real-time data such as water levels, revenue, and enquiry/customer/employee counts.

ii. Implementation: Developed using Laravel for backend logic and Tailwind CSS for the frontend.

(b) **Enquiry Management Module**

i. Purpose: Allows residents/consumers to send queries or complains to water authority board through app and Stores messages in the MySQL database.

ii. Implementation: Complains are submitted through a form in the mobile app. Built using Laravels backend logic and API's, integrated with the frontend for sending/receiving messages.

(c) **Employee Management Module**

i. Purpose: Allows admin to keep or maintain the record of each employees.

ii. Implementation: Backend was implemented using Laravel and frontend was designed using Hypertext Markup Language (HTML) ,Tailwind CSS, Blade.

(d) **Consumers Management Module**

i. Purpose: Allows admin to keep or maintain the record of each consumers of different wards.

ii. Implementation: Backend was implemented using Laravel and frontend was designed using HTML [20],Tailwind CSS, Blade

(e) **Billing Module**

i. Purpose: Automatically calculates bills based on water usage and stores data in the database.

ii. Implementation: Build using Tailwind css to design the frontend layout and Laravel with MySQL were used to handle all the necessary backend data.

(f) **Schedule Module**

i. Purpose: Enables admins to create and manage water distribution schedules.

ii. Implementation: Sends updates to customers via the mobile app.The mobile app communicates with the web applications backend via Representational State Transfer (REST)ful API's.

(g) **Statistics Visualization Module**

i. Purpose: Displays the graphical representation of population across different wards and toles.

ii. Implementation: Uses Chart.js to display graphs for population distribution

5.2 Testing

5.2.1 Test Cases for Unit Testing

Testing the login, registering functionality for the admin(user). Each of the operation is tested with the valid and invalid data to see if the system acts as it suppose to act and also to find any bugs and errors that may present in the system. All test cases for this operation are shown below:

a) SignUp

Table 5.1: Sign-up testing for the admin

Test Case ID	Test Scenario	Test Steps	Input Test Data	Result	Pass/Fail
1.	Check registering With Valid Data	1) Open the webpage and go to register page. 2) Filled up the form with necessary relevant data. 3) Register	a) Name:Rijan Rai b) Email:rijan2@gmail.com c)Password:123456 d)Confirm password:123456	As expected the user is registered and redirected to login page.	Pass
2.	Check registering with invalid data.	1) Open the webpage and go to register page. 2) Fill the form with wrong data. 3) Register	a)Name:Arun Khatri b) Email:arun2@gmail.com c)password:122334 d)Confirm password:12345	As expected the user will get invalid password message.	Pass

b) Login

Table 5.2: Testing login for the admin

Test Case ID	Test Scenario	Test Steps	Input Test Data	Result	Pass/Fail
1.	Login With Valid Data	1) Open the webpage and go to login page. 2) Filled up the form with necessary relevant data. 3) click Login	a) Email:rijan2@gmail.com b)Password: 123456	As expected the user is logged in and redirected to Dashboard.	Pass
2.	Log in with false data.	1) Open the webpage and go to login page. 2) Fill the form with wrong data. 3) click Login	a) Email:rijan2@gmail.com b)password:12346	As expected the user will get invalid password message.	Pass

Table 5.3: Testing login for the residents

Test Case ID	Test Scenario	Test Steps	Input Test Data	Result	Pass/Fail
1.	Login With Valid Data	1) Open the mobile App . 2) Fill up the form with necessary relevant data. 3) click Login	a) meterId:01553 b)Phone Number: 9842177533	As expected the user is logged in and redirected to Dashboard.	Pass
2.	Log in with false data.	1) Open the mobile app and go to login page. 2) Fill the form with wrong meter ID. 3) click Login	a) meterId:01443 b)Phone Number: 9842177533	As expected the user will get invalid Meter ID message.	Pass

c) Billing

Table 5.4: Testing the billing operation.

Test Case ID	Test Scenario	Test Steps	Input Test Data	Result	Pass/Fail
1	Adding the billing details of the existing customer with valid data	1) Open the webpage and login. 2) Click the Billing schedule menu in the sidebar. 3) Click the "New Bill" option. 4) Enter the meter Id and insert the reading units. 5) Click Save	a) Meter Id: 01553 b) Previous reading:100 units c) Current reading: 300 units d)Total amount:Rs 6000	As expected the billing details of the customer having meter Id 01553 is added to the list.	Pass
2	Adding the billing details of the non-existing customer.	1) Open the webpage and login. 2) Click the Billing schedule menu in the sidebar. 3) Click the "New Bill" option. 4) Enter the meter Id and insert the reading units. 5) Click Save	a) Meter Id: 01552 b) Previous reading:100 units c) Current reading: 300 units d)Total amount:Rs 6000	As expected it throws the error.	Pass

d) Employee unit

Table 5.5: Testing the employee functionality.

Test Case ID	Test Scenario	Test Steps	Input Test Data	Result	Pass/Fail
1	Adding a new employee details	1)Login. 2) Click the Employee menu from the sidebar. 3) Click the "Add Employee" option. 4) Fill the form. 5) Click Add	a) First Name: Rijan b) Last Name:Rai c)Address:Dharan-10 d)Phone:981567823 e) Position: Plumber	As expected new employee details is added to the list	Pass
2	Updating the existing employee details	1) Login 2) Click the employee menu from the sidebar. 3)click Edit from the Action field. 4) Change the data. 5) Click Save all	a) First Name: Rijan b) Last Name:Rai c)Address:Dharan-11 d)Phone:981567823 e) Position: Plumber	As expected the address is changed and is saved	Pass
3	Deleting employee details	1) Login 2) Click the employee menu from the sidebar. 3)Click Delete from the Action field.	-	As expected the employee details is removed from the list	Pass

e) Customer unit

Table 5.6: Testing the of customer functionality.

Test Case ID	Test Scenario	Test Steps	Input Test Data	Result	Pass/Fail
1	Adding a new customer details	1)Login. 2) Click the customer menu from the sidebar. 3) Click the "Add customer" option. 4) Fill the form. 5) Click Add	a) First Name: Rijan b) Last Name:Rai c)Meter ID:0115 d)Phone:981567823 e)ward no.:7 f)Tole:7A g)Dharan-7	As expected new customer details is added to the list	Pass
2	Updating the existing customer details	1) Login 2) Click the customer menu from the sidebar. 3)Click Edit from the Action field. 4) Change the data. 5) Click Save all	a) First Name: Rijan b) Last Name:Rai c)Meter ID:0115 d)Phone:981567824 e)ward no.:7 f)Tole:7A g)Address: Dharan-7	As expected the phone number is changed and is saved	Pass
3	Deleting customer details	1) Login 2) Click the customer menu from the sidebar. 3)Click Delete from the Action field.	-	As expected the customer details is removed from the list	Pass

5.2.2 Test Cases for System Testing

The system testing accepted the entire system evaluation once all the modules have been developed and put together for the purpose of verifying the accurate working and efficiency, meeting all specifications. This testing phase was undertaken to check the application thoroughly and look for possible bugs. Data provided by users was utilized to perform the tests, which helped uncover issues and gaps in the system requirements since real-world data interacts with the system differently compared to the test data used earlier. From the user's perspective, testing began with registering a new user. Upon successful registration, the user could log in to their account using valid credentials and perform the different activities which the user is allowed to perform. As well as for the admin, testing began with registering a new admin. Upon successful registration, the admin could log in to their account using valid credentials and perform the different activities which the admin is allowed to perform.

5.3 Result Analysis

From the start of the project through its development, implementation, and refinement, extensive research and analysis were conducted, including a review of prior

studies. While we encountered many significant challenges during development, we successfully resolved them through necessary adjustments and additional research, utilizing online resources and internet. In the end, we were satisfied with the project's outcome, as it performed just as we expected.

We visited the Dharan khanepani office to get the real time data about the reservoir capacity and fed those data in our system.We have used an API as a mediator between our web app and mobile app.Through web app,we can manage all the necessary records of each customers and employees. The admin dashboard also includes a section to view the total enquiries from the customers so that it can be easy to manage and maintain the queries from customers. The web app also provides a graph that can help in effective decision making.Also the web app provides a feature of analyzing the incoming and outgoing rate of water in reservoir.We have employed single factor authentication on admin side ensuring authenticity for admin. On the other hand, mobile app provides the features for consumers like viewing water schedule, sending feedback about the irregularity in water schedule to admin side.We have made simple and user-friendly mobile interface so that the consumer can easily understand without any complexity.

CHAPTER 6: CONCLUSION AND FUTURE RECOMMENDATIONS

6.1 Conclusion

The development of the software that we have included in this project report is a breakthrough in water supply management modernization and customer engagement. The mobile app acts as an easy access point for users to view real-time schedules and submit inquiries. It also acts as a bridge of communication between the municipality and the citizens it serves . The web application simultaneously introduces an unparalleled model for record-keeping, including the added functionality of graphical representation of statistics that target lamps on projected population distribution and water demand over different regions. The integrated billing feature in the web application also aids in the smooth functionality of financial transactions by promoting accountability and transparency.

The team of three worked in collaboration to develop a very robust and user-centered solution specifically for the needs of Dharan Municipality. This software enables DWSMB to work more efficiently and deliver better and quicker services. The development project employed an iterative model that, instead of crunching out a solution towards the tail end of the service life cycle, allowed any enhancements needed to improve the viability of the system to be affected throughout the complete life cycle of the project. The project's successful closure thus provides a platform to bring in the return on investment and create the next set of scalable public utility solutions..

6.2 Future Recommendations

Even though a excessive amount of time has been invested for building this system, there can still be some recommendations for the future to make the system more reliable and efficient. Some of the future recommendation are listed below:

(a) **Integration Of IoT and Real-Time Data** :Use IoT based sensors for real-time monitoring current water level present in the reservoir, identify consumption patterns and detect leaks.

(b) **Integration with Online Payment Systems**: Incorporate online payment gateways to enable customers to pay water bills directly via the app or web platform, enhancing convenience and efficiency.

(c) **Policy Support**: Collaborate with local government bodies to establish policies that regulate water use, ensure funding, and provide institutional support for long-term sustainability.

(d) **Advanced Data Analytics**: Expand data analytics capabilities to identify trends, predict future water demand, and optimize resource allocation across wards and toles.

REFERENCES

[1] J. P. Yadav, K. P. Sah, K. R. Karki, M. N. Jha, A. Mishra, *et al.*, "Dharan water supply system–alarming issues and future," *J. Adv. Res. Civ. Environ. Eng*, vol. 10, pp. 1–11, 2023.

[2] G. Maskey, C. L. Pandey, and M. Giri, "Water scarcity and excess: water insecurity in cities of nepal," *Water Supply*, vol. 23, no. 4, pp. 1544–1556, 2023.

[3] Dharan Municipality, "Dharan Municipality Official Website," 2025. Accessed: Feb. 1, 2025.

[4] Android Developers, "Android Studio," 2025. Accessed: Feb. 1, 2025.

[5] S. M. Khan, "Iterative model used in software development," 06 2023.

[6] SUSWA, "Suswa - sustainable water and sanitation in nepal." Website, 2024. Accessed: September 8, 2024.

[7] Lekhnath Water Supply, "Lekhnath Water Supply Official Website," 2025. Accessed: Feb. 1, 2025.

[8] PHP Documentation Contributors, "PHP Manual," 2025. Accessed: Feb. 1, 2025.

[9] RESTful API Contributors, "RESTful API Design A Quick Reference Guide," 2025. Accessed: Feb. 1, 2025.

[10] Laravel Documentation, "Hashing - Laravel 5.0 Documentation," 2025. Accessed: Feb. 1, 2025.

[11] E. Hossain, S. B. Hossain, M. S. A. Tutul, and S. Nahar, "Optimized search functionality with linear search algorithm," 11 2021.

[12] Tailwind CSS Developers, "Tailwind CSS: Rapidly Build Modern Websites Without Ever Leaving Your HTML," 2025. Accessed: Feb. 1, 2025.

[13] Mozilla Contributors, "JavaScript - MDN Web Docs," 2025. Accessed: Feb. 1, 2025.

[14] Oracle Corporation, "Java Programming Language," 2025. Accessed: Feb. 1, 2025.

[15] World Wide Web Consortium (W3C), "XML: eXtensible Markup Language," 2025. Accessed: Feb. 1, 2025.

[16] Laravel Developers, "Laravel: The PHP Framework for Web Artisans," 2025. Accessed: Feb. 1, 2025.

[17] MySQL Developers, “MySQL: The World’s Most Popular Open-Source Database,” 2025. Accessed: Feb. 1, 2025.

[18] GitHub, Inc., “GitHub: Where the World Builds Software,” 2025. Accessed: Feb. 1, 2025.

[19] Chart.js Contributors, “Chart.js Documentation,” 2025. Accessed: Feb. 1, 2025.

[20] Mozilla Contributors, “HTML - MDN Web Docs,” 2025. Accessed: Feb. 1, 2025.

APPENDIX

Table 7.1: Log of visits to supervisor

Date	Time	Purpsose of visit	Duration	Feedback	Next Step
2081/05 /02	12 PM	Discussion on project idea and requirement gathering.	40 Minutes	Supervisor approved the project idea.	Prepare flowchart and algorithm for approval.
2081/05 /20	2 PM	Presentation of design and use case diagram.	1 hour	Refine the system architecture.	Update the use case diagram
2081/06 /14	2 PM	Reveiw of intial Implementation.	1 Hour	Suggested testing edge cases.	Test the registration module for all possible edge cases
2081/10 /05	3 PM	Review the Report.	1 Hour	Suggestions for report formatting.	Make corrections on the report and prepare for mid-defence.

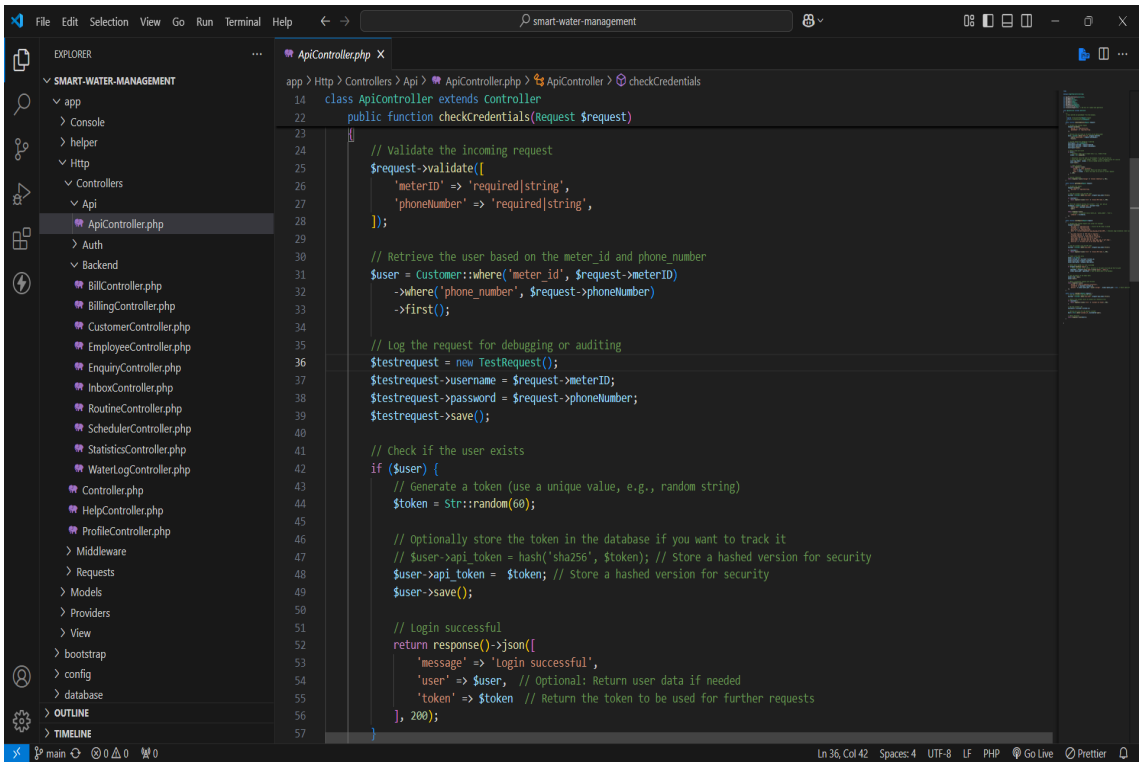


Figure 7.1: API Controller

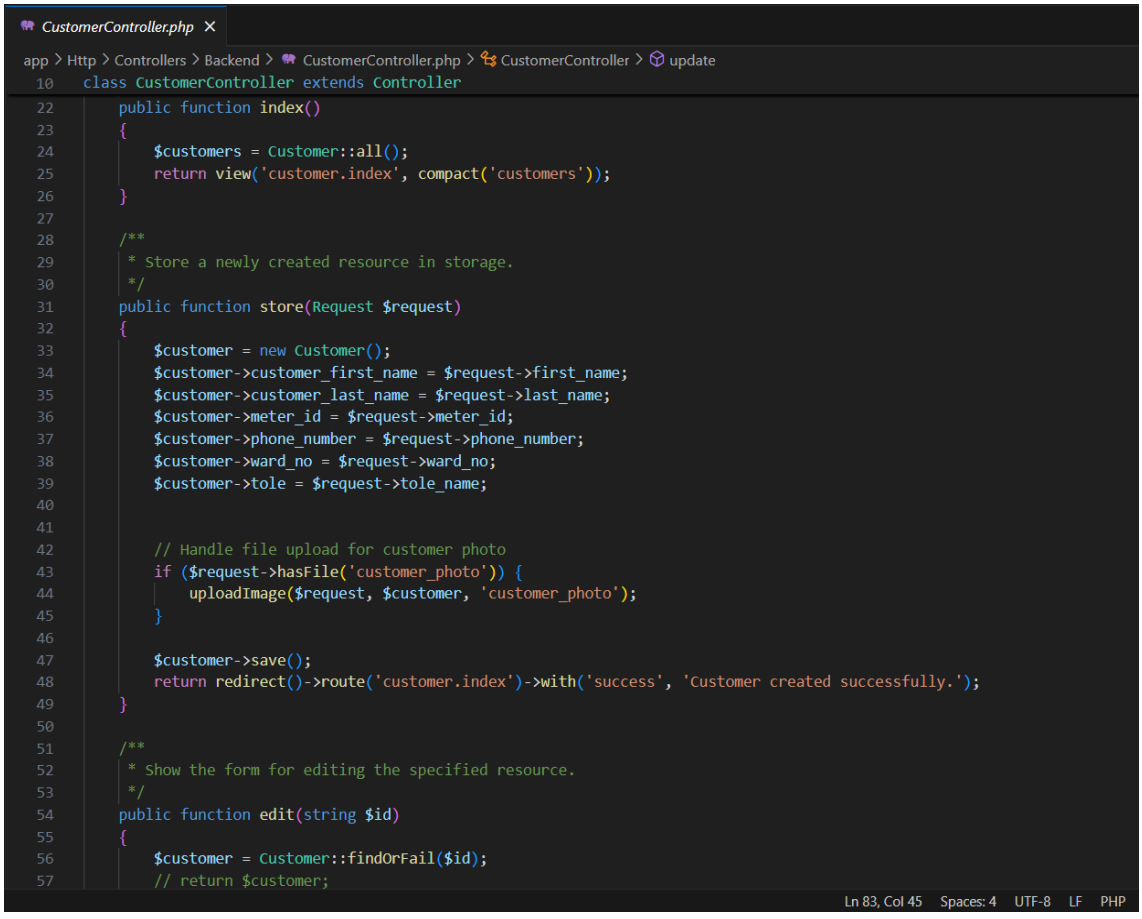
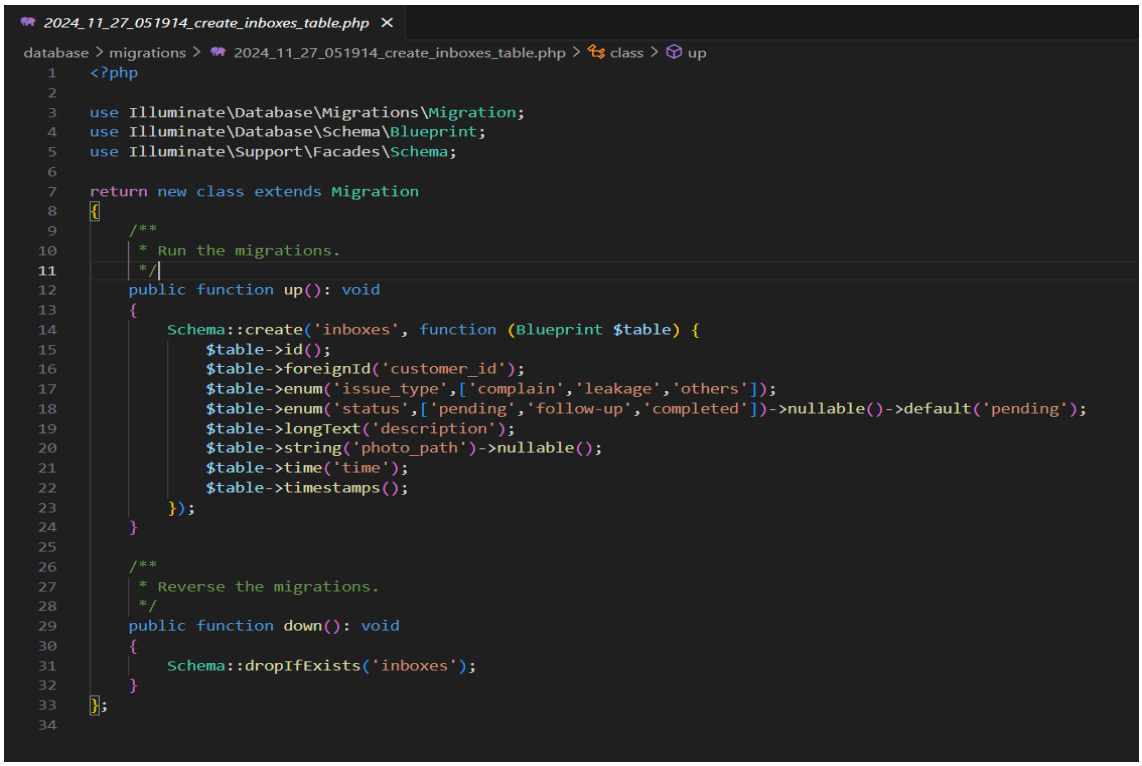


Figure 7.2: Customer Controller

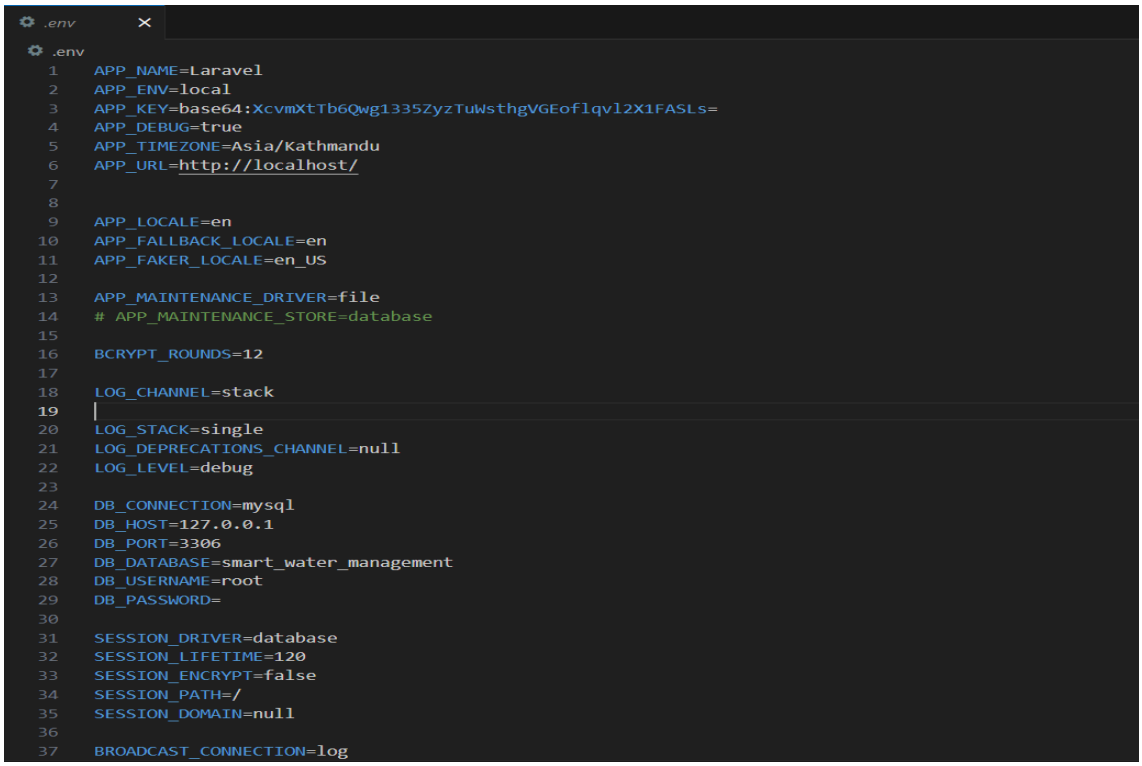


2024_11_27_051914_create_inboxes_table.php

database > migrations > 2024_11_27_051914_create_inboxes_table.php > class > up

```php
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

return new class extends Migration
{
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::create('inboxes', function (Blueprint $table) {
            $table->id();
            $table->foreignId('customer_id');
            $table->enum('issue_type',['complain','leakage','others']);
            $table->enum('status',['pending','follow-up','completed'])->nullable()->default('pending');
            $table->longText('description');
            $table->string('photo_path')->nullable();
            $table->time('time');
            $table->timestamps();
        });
    }

    /**
     * Reverse the migrations.
     */
    public function down(): void
    {
        Schema::dropIfExists('inboxes');
    }
};

```

Figure 7.3: Database Model



.env

```
APP_NAME=Laravel
APP_ENV=local
APP_KEY=base64:XcvmXtTb6Qwg1335ZyzTuWsthgVGEoflqvl2X1FASLs=
APP_DEBUG=true
APP_TIMEZONE=Asia/Kathmandu
APP_URL=http://localhost/


APP_LOCALE=en
APP_FALLBACK_LOCALE=en
APP_FAKER_LOCALE=en_US

APP_MAINTENANCE_DRIVER=file
# APP_MAINTENANCE_STORE=database

BCRYPT_ROUNDS=12

LOG_CHANNEL=stack

LOG_STACK=single
LOG_DEPRECATIONS_CHANNEL=null
LOG_LEVEL=debug

DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=smart_water_management
DB_USERNAME=root
DB_PASSWORD=

SESSION_DRIVER=database
SESSION_LIFETIME=120
SESSION_ENCRYPT=false
SESSION_PATH=/
SESSION_DOMAIN=null

BROADCAST_CONNECTION=log
```

Figure 7.4: Environent Variables

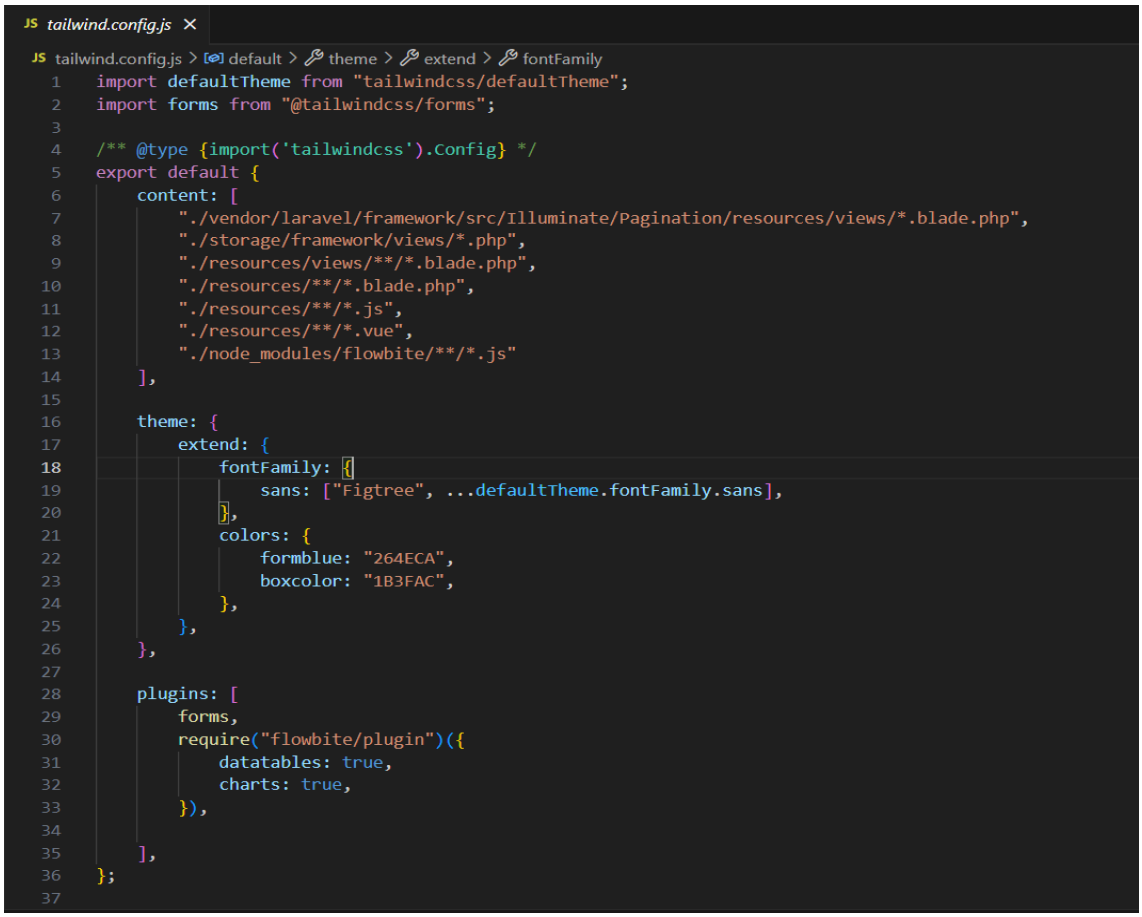


Figure 7.5: Tailwind Configuration

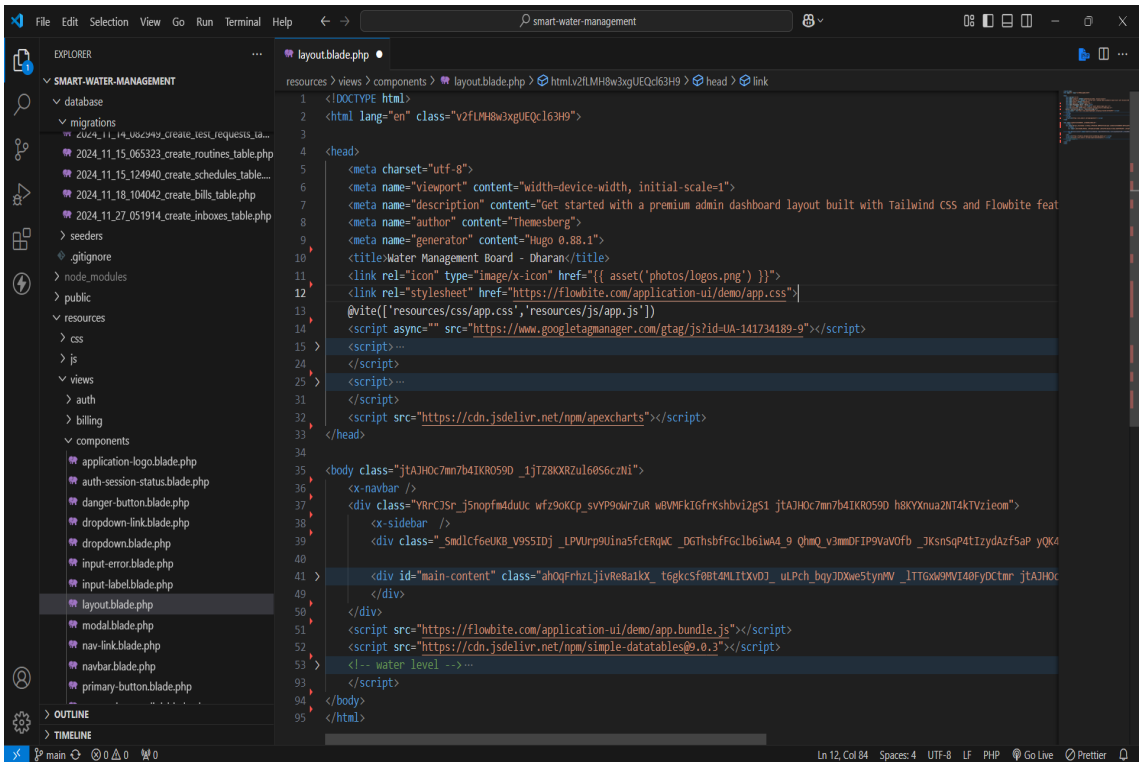


Figure 7.6: Layout Component

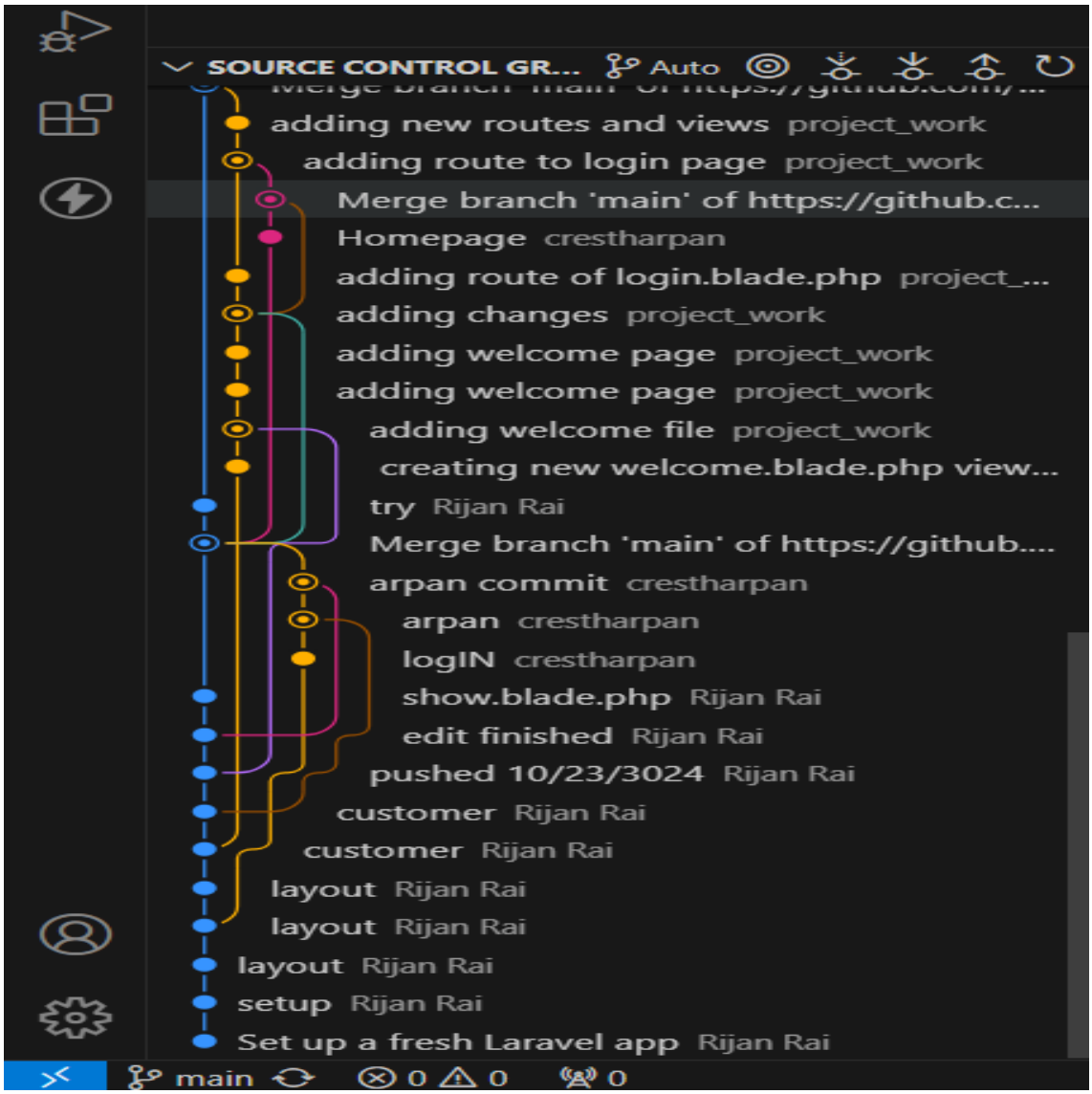


Figure 7.7: Github Logs

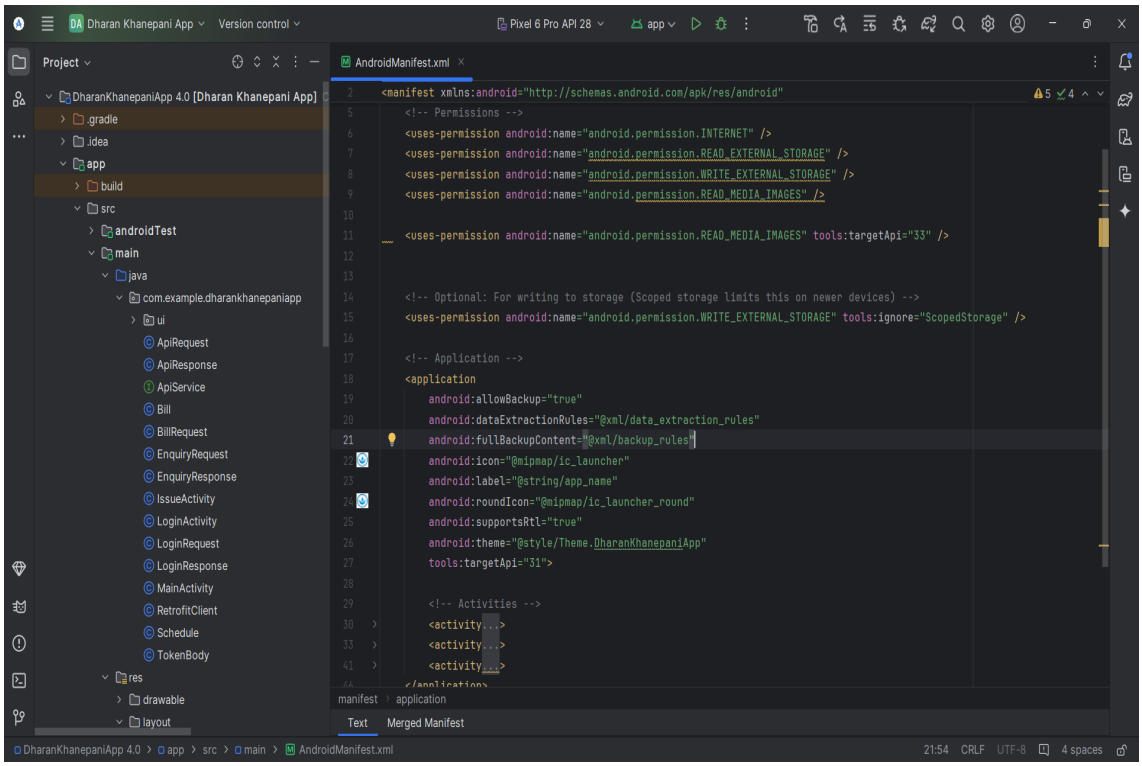


Figure 7.8: Application Configuration and Permissions for Mobile App

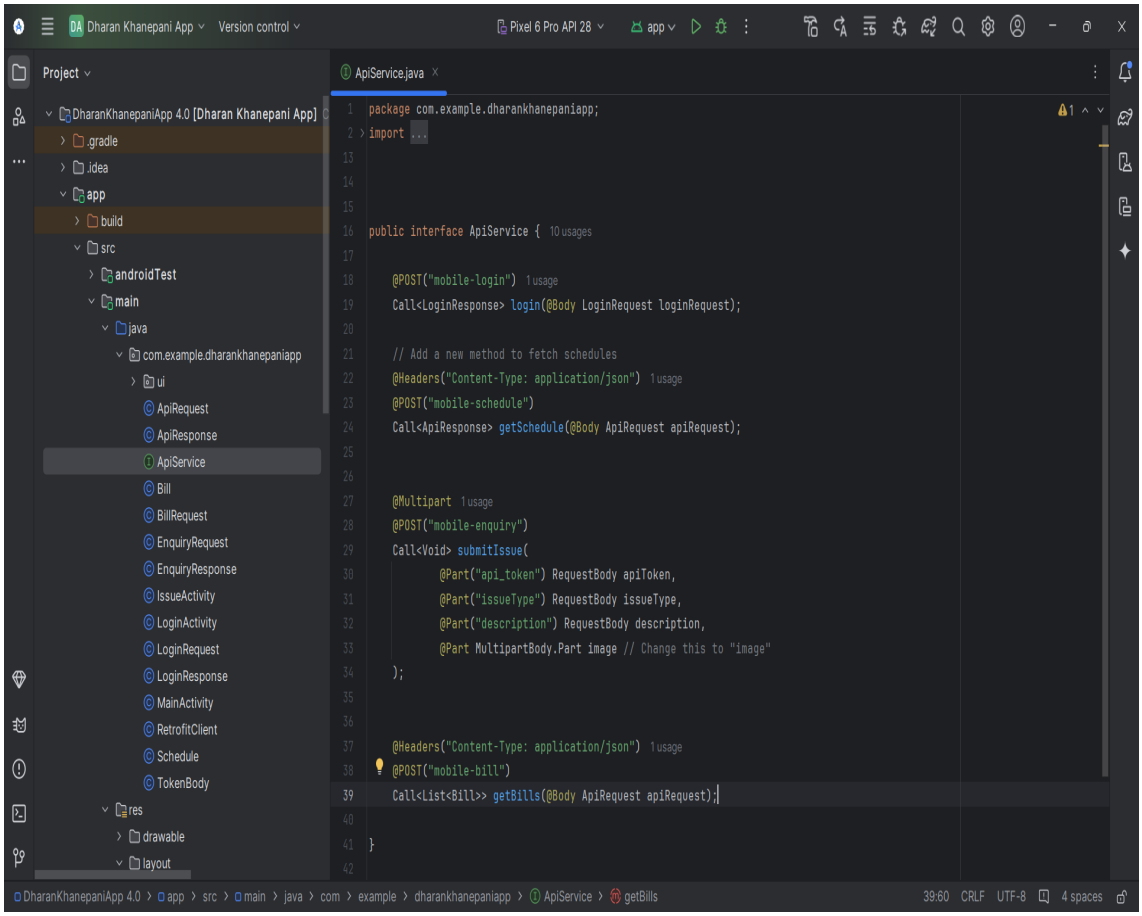


Figure 7.9: Defining API Endpoints of Mobile App

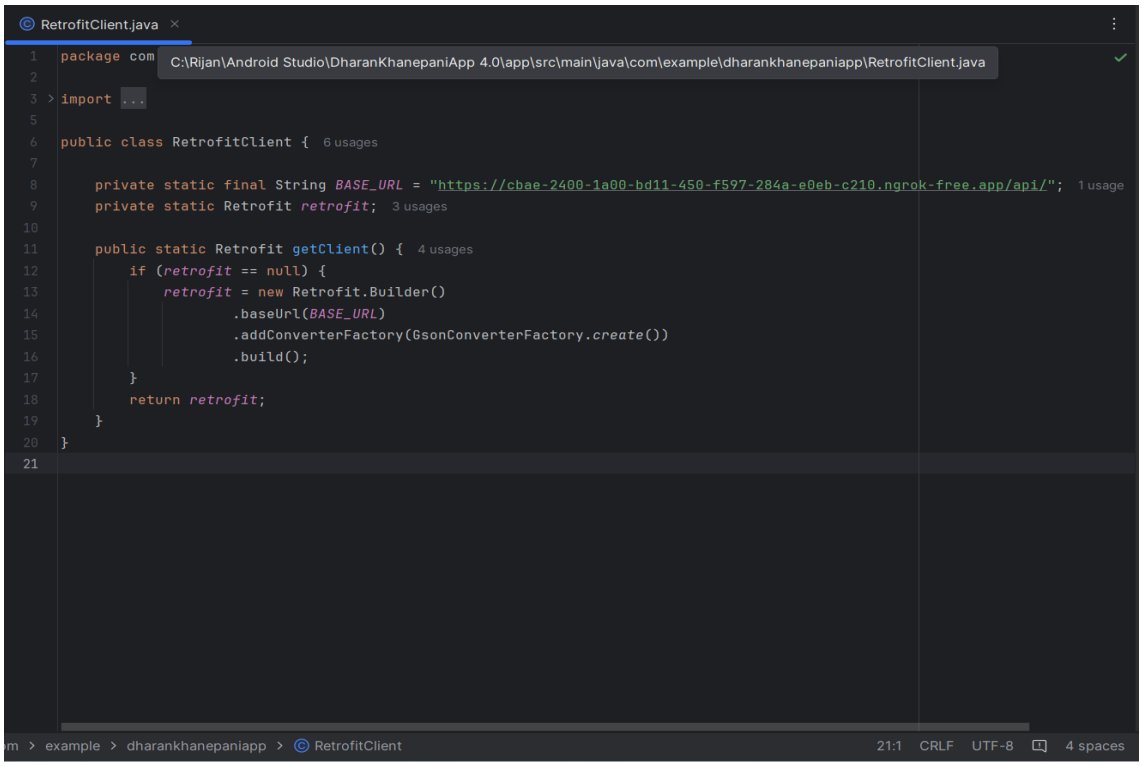


Figure 7.10: Configuring API Calls with Retrofit

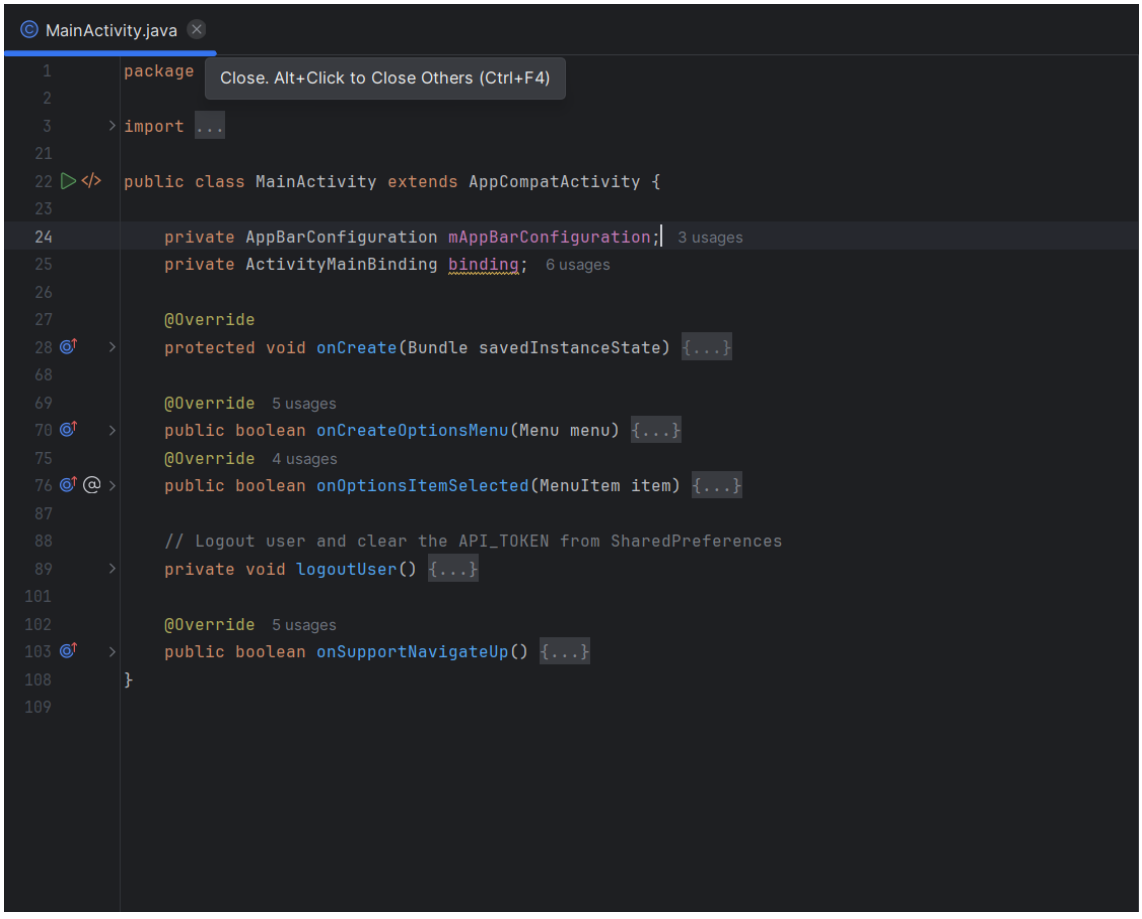


Figure 7.11: Primary Activity of Mobile App

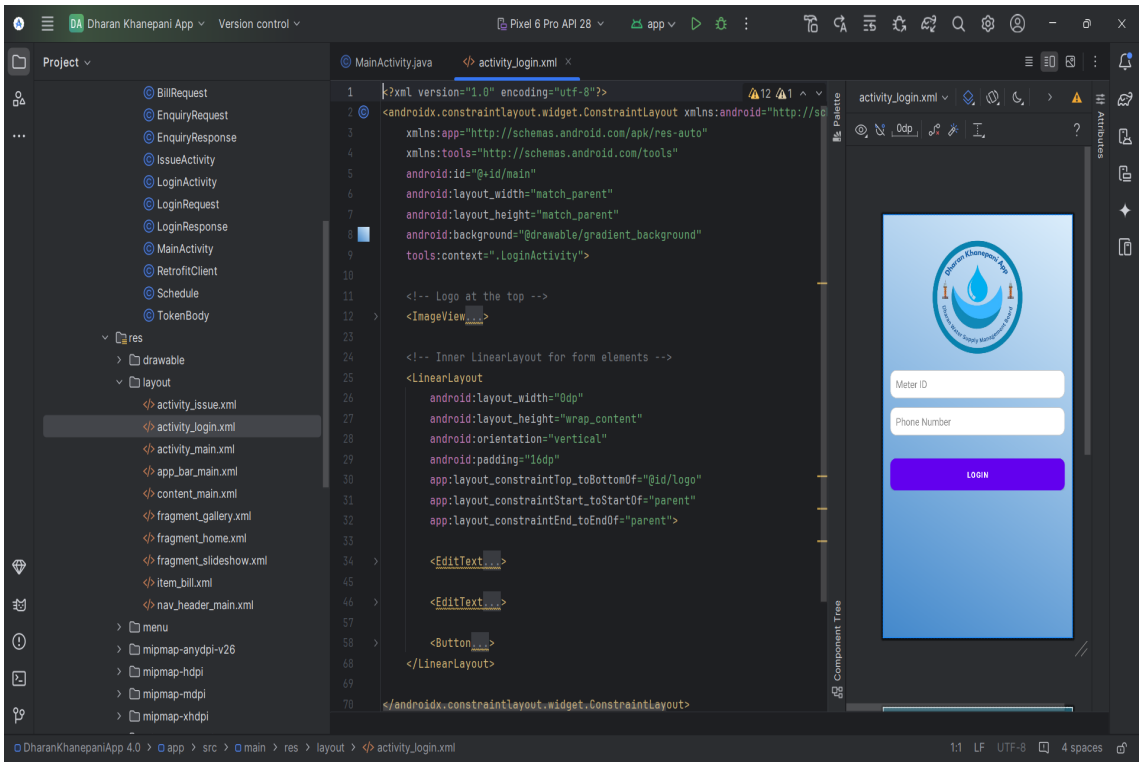


Figure 7.12: Log In Page Layout for Mobile App

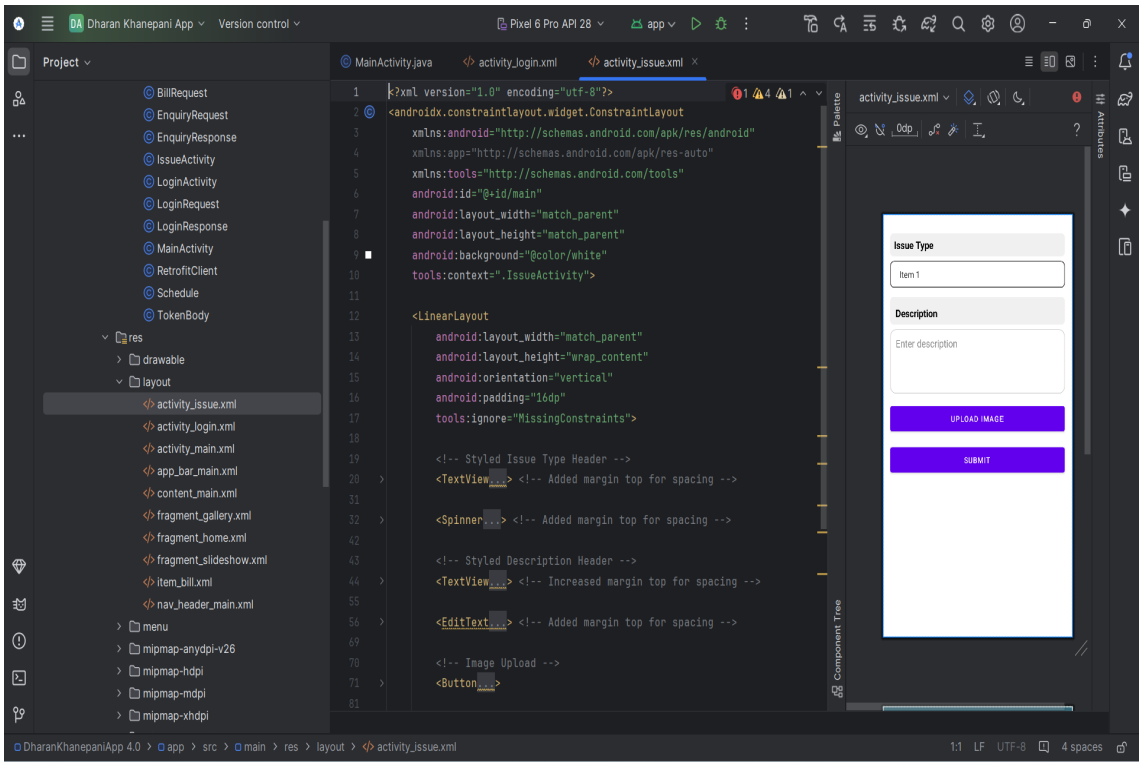


Figure 7.13: Issue Query Page Layout for Mobile App