UNIVERSITY OF BUEA

P.O Box 63,
Buea, South West Region
CAMEROON
TEL: (237) 3332 21 34/3332 26 90

REPUBLIC OF CAMEROON

PEACE-WORK-FATHERLAND

FACULTY OF ENGINEERING AND TECHNOLOGY

DEPARTMENT OF COMPUTER ENGINEERING

DESIGN AND IMPLEMENTATION OF A PROPOSED EMPLOYEE ATTENDANCE MANAGEMENT SYSTEM

Internship Report Submitted to the Department of Computer Engineering. Faculty of Engineering and Technology, University of Buea, in Partial Fulfillment of the Requirements for the Award of the Bachelor of Engineering (B.Eng.) Degree in Computer Engineering

By:

QUINUEL TABOT NDIP-AGBOR

Matricule: *FE21A300*

Option: *SOFTWARE ENGINEERING*

Technical Supervisors:

ENGR. MANTOH NASAH KUMA
ENGR. CYPRIAN TINGIWEH

GO-GROUPS LTD

Academic Supervisor:

DR. SOP DEFFO Lionel

University of Buea

2024 - 2025 Academic Year

TABLE OF CONTENTS

DEDICATION

I dedicate this project to my lovely family.

CERTIFICATE OF ORIGINALITY

We the undersigned, hereby certify that this internship project entitled "**A PROPOSED EMPLOYEE ATTENDANCE MANAGEMENT SYSTEM**" presented by **QUINUEL TABOT NDIP-AGBOR,** matriculation number **FE21A300** has been carried out by her in Go-Groups LTD, Buea after a six-month internship from **6th August 2024 to 25th February 2025**, the University of Buea under the supervision of **Dr SOP DEFFO Lionel.**

In testimony, therefore, this is approved for being relevant to the industrial experience and keeping with academic and operational procedures.

Signature ____________________

Industrial Supervisor

Engr. Mantoh Nasah Kuma

Signature ___________________

Academic Supervisor

Dr. SOP DEFFO Lionel

Signature ____________________

Industrial Supervisor

Engr. Cyprian Tingiweh

ACKNOWLEDGEMENTS

I would like to express my heartfelt acknowledgments to the following individuals and entities who have played a significant role in my academic journey and the successful completion of my internship:

Firstly, I want to acknowledge the Faculty of Engineering and Technology and academic supervisor for their unwavering support and guidance throughout my academic years. The faculty's exceptional lecturers have imparted invaluable knowledge and skills, shaping me into a competent professional in my field. Their dedication to teaching and commitment to my growth have been instrumental in my development.

I extend my sincere appreciation to my course mates and schoolmates, who have been my companions throughout our academic journey. Together, we faced numerous challenges, and their constant encouragement and motivation have been pivotal in overcoming obstacles and fostering a collaborative and stimulating learning environment.

I am immensely grateful to my professional supervisors and Go-Groups Ltd. Their guidance and mentorship have been invaluable in shaping my understanding of the industry and providing me with the necessary skills to excel as a professional. Their expertise and unwavering support have been instrumental in my growth and development during this internship period.

Lastly, I would like to express my deepest gratitude to my family. Their unconditional love, unwavering belief in my abilities, and unwavering support have been my source of strength and inspiration. Their constant presence, encouragement, and provision of everything I needed have enabled me to focus on my studies and pursue my goals with determination.

To all those mentioned above, I am indebted for their contributions, support, and belief in my abilities. Their collective impact has shaped me both academically and personally, and I am truly grateful for their presence in my life.

LIST OF FIGURES

LIST OF ABBREVIATIONS

API	Application Programming Interface
CRUD	Create, Read, Update, and Delete
CSS	Cascading Style Sheet
DFD	Data Flow Diagrams
EMS	Employee Management System
ERD	Entity Relation Diagram
HR	Human Resource
HTML	Hypertext Markup Language
HTTPS	Hypertext Transfer Protocol Secure
JAR	Java Archive
JPA	Java Persistence API
SQL	Structured Query Language
UI	User Interface
UML	Unified Modelling Language

CHAPTER ONE: GENERAL INTRODUCTION

1.1 Definition of Internship

An internship is a professional learning experience that offers meaningful, practical work related to a student's field of study or career interest. An internship gives a student the opportunity for career exploration and development, and to learn new skills. It offers the employer the opportunity to bring new ideas and energy into the workplace, develop talent, and potentially build a pipeline for future full-time employees. A quality internship:

- Consists of a part-time or full-time work schedule that includes no more than 25% clerical or administrative duties.
- Provide a clear job/project description of the work experience.
- Orient the student to the organization, its culture, and proposed work assignment(s).
- Helps the student develop and achieve learning goals.
- Offers regular feedback to the student intern.

1.2 Goals and Objectives of the Internship

As a software student going for an internship, I had several goals in mind to make the most of my experience. These goals include:

1. **Applying theoretical knowledge from school (Faculty of Engineering and Technology):** One of my primary goals was to apply the theoretical knowledge I have gained during my coursework in the Faculty of Engineering and Technology to real-world projects and scenarios. I aimed to bridge the gap between academic concepts and practical software development by working on actual projects and utilizing programming languages, tools, and frameworks.
2. **Gaining hands-on experience:** I wanted to gain practical, hands-on experience in software development through my internship. Working on bigger projects than I did during my school period, collaborating with professionals, and learning about industry best practices, software development methodologies, and project management techniques are important aspects of this goal.
3. **Developing technical skills:** Enhancing my technical skills is a key goal for my internship. I planned to improve my proficiency in coding, debugging, testing, software documentation, version control systems, and other technical aspects relevant to my area of specialization.
4. **Learning new technologies:** I aimed to familiarize myself with new technologies and trends in the software industry during my internship. Whether it's machine learning, cloud computing,

mobile app development, or web development frameworks, learning these technologies can enhance my competitiveness in the job market and provide me with a broader perspective on software development.

5. **Collaborating in a team environment:** Working in a professional team setting is an important goal for me. I want to develop effective communication skills, learn how to work as part of a team

and understand the dynamics of collaborative software development.

6. **Networking and building professional relationships:** Building a strong professional network is crucial for future career opportunities. During my internship, I plan to connect with professionals in my field, seek mentorship, attend industry events, and establish connections that could lead to future job prospects.

7. **Gaining industry insights:** Gaining valuable insights into the software industry, its trends, challenges, and best practices is a key goal for me. Understanding the software development lifecycle, project management methodologies, quality assurance processes, and client interactions will contribute to my overall understanding of the industry and inform my career decisions.

8. **Enhancing problem-solving and critical thinking skills:** I aimed to enhance my problem-solving and critical thinking skills during my internship. By tackling real-world problems and challenges, I wanted to improve my ability to analyze complex issues, propose effective solutions, and iterate based on feedback.

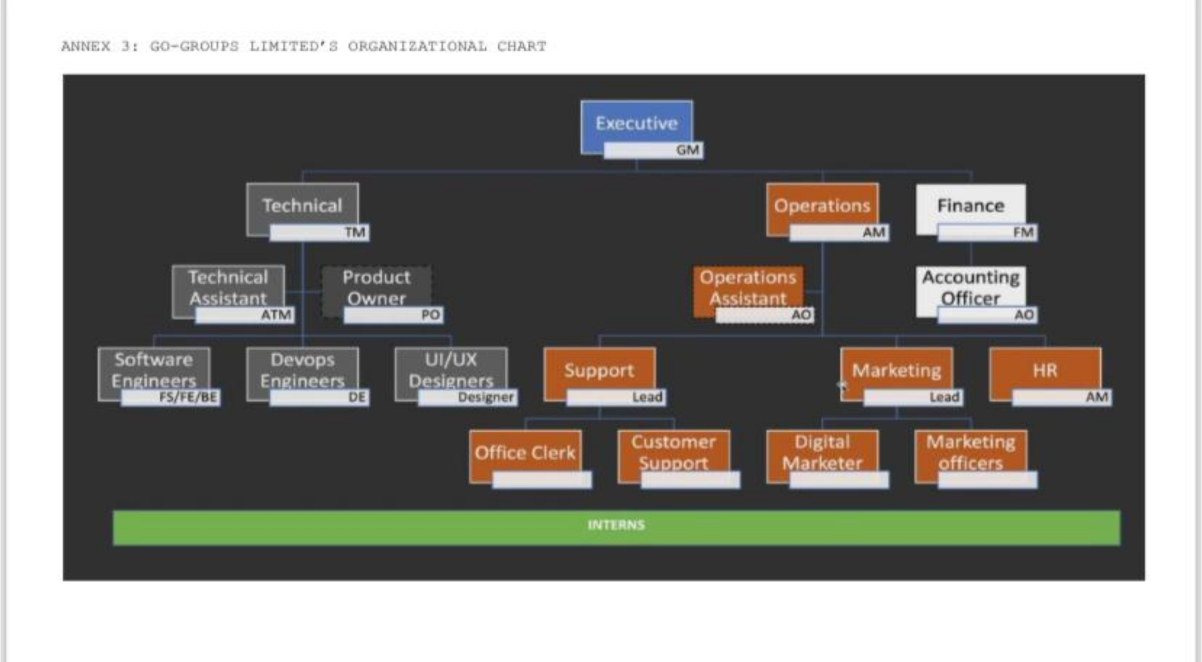
1.3 Presentation of the Company

1.3.1 Company Description of Groups LTD

Go-Groups LTD is a software company based in Buea, South West Region, Cameroon envisioned to advancing the use of computer technology in the face of scarce resources with the motto *"Be on the GO"*. It was founded in 2012 and provides Software as a Service (SaaS). Go-Groups Ltd has built on applications in the transport, e-commerce, taxation, and academic sectors such as Go-Student, GoJobs, GoWaka, and a few others.

1.3.2 Organizational Structure of Go-Groups LTD

The figure below is an organigram of Go-Groups LTD, it contains the hierarchical structure of the company which practicalities in the day-to-day running of the company. Being a software engineering student, I was under a technical team composed of Front-end engineers, Back-end engineers, and DevOps Experts:



ANNEX 3: GO-GROUPS LIMITED'S ORGANIZATIONAL CHART

Figure 1: Go-Groups LTD Organizational Chart

CHAPTER TWO: INTERNSHIP ACTIVITIES

2.1 Professional Supervisors

My technical supervisors, Eng. Mantoh Nasah and Eng. Cyprian Tingiwieh are graduates from the University of Buea, they played a crucial role in guiding me throughout my internship. Being well-versed in all the technologies used at the company for both frontend and backend development, my supervisors were able to provide me with comprehensive guidance and support.

Their strong technical backgrounds, skills, and knowledge allowed them to effectively mentor me in the technologies utilized by the company, including Angular.ts and Spring boot (Java). With their expertise, they were able to explain complex concepts, provide insights into best practices, and help me navigate any challenges or roadblocks I encountered during my internship.

Beyond technical proficiency, my supervisor's work ethic and dedication are very commendable, which was visible in their drive and cooperation to deliver high-quality results in their own work. This dedication also surfaced in their role as supervisors, where they took the time and effort to guide and mentor me effectively.

2.2 Tools Used During Internship

During my internship, I was an onsite intern with an HP Laptop using the Ubuntu OS under the software development team working on full-stack development-related tasks, the company utilized several technologies including Angular with Typescript, CSS, Spring Boot, and MySQL. Here's a brief description of each technology and why they were chosen:

1) Angular

Angular is a powerful and popular front-end framework for building dynamic and interactive web applications. It employs TypeScript, a superset of JavaScript, which enhances code maintainability and readability. Key features of Angular include:

- Component-based architecture: Applications are structured into reusable components, making development, testing, and maintenance more efficient.
- Two-way data binding: Seamlessly synchronizes data between the model and the view, simplifying data flow and reducing boilerplate code.

- Dependency injection: Promotes modularity and testability by managing dependencies between components.
- Reactive programming: Enables efficient handling of asynchronous events and updates to the user interface.

2) CSS

Cascading Style Sheets (CSS) is the fundamental language for styling the visual presentation of web pages. It controls the layout, colors, fonts, and other visual aspects of HTML elements.

- Selectivity: CSS rules can be applied to specific HTML elements, groups of elements, or even individual elements based on various attributes.
- Cascading: Styles are applied in a cascading order, allowing for more specific styles to override general ones.
- Flexibility: CSS offers a wide range of properties and values to customize the appearance of web pages to any desired style.

3) Spring Boot

Spring Boot is a powerful Java framework built on top of the Spring ecosystem. It simplifies and accelerates the development of Spring-based applications. Key features include:

- Auto-configuration: Automatically configures many aspects of the application, such as database connections and web servers, reducing the need for extensive manual configuration.
- Embedded servers: Includes embedded servers like Tomcat and Jetty, making deployment easier and faster.
- Opinionated approach: Provides a set of recommended best practices and conventions, leading to faster development and more consistent code.
- RESTful web services: Easily creates RESTful APIs for building modern, web-based applications.

4) Java

Java is a robust, platform-independent programming language known for its versatility and reliability. Key characteristics of Java include:

- Object-oriented programming: Supports key OOP concepts like encapsulation, inheritance, and polymorphism, leading to modular and maintainable code.
- Platform independence: "Write once, run anywhere" – Java code can be compiled into bytecode that runs on any platform with a Java Virtual Machine (JVM).

- Large ecosystem: A vast collection of libraries and frameworks, such as Spring, Hibernate, and Java EE, provides extensive support for various development needs.
- Strong community and support: A large and active community provides abundant resources, tutorials, and support.

5) MySQL

MySQL is a powerful open-source relational database system known for its robustness, reliability, and advanced features. Key features include:

- ACID compliance: Ensures data integrity and consistency through features like atomicity, consistency, isolation, and durability.
- SQL support: Utilizes the widely adopted SQL language for data querying, manipulation, and administration.
- Rich data types: Supports a wide range of data types, including arrays, JSON, and spatial/geographic data.
- Extensibility: Allows for user-defined functions, operators, and data types to customize the database system.
- Strong community and ecosystem: Benefits from a large and active community, providing extensive documentation, support, and a wealth of third-party tools and extensions.

6) Git Version System/GitHub

Git is a powerful version control system that tracks changes to files over time. It allows developers to collaborate effectively by enabling multiple developers to work on the same project simultaneously without overwriting each other's changes. GitHub is a popular platform that provides hosting for Git repositories, allowing developers to share their code with others, track issues, and collaborate on projects. In your project, Git and GitHub likely played a crucial role in managing the development process, ensuring code integrity, and facilitating collaboration among team members.

2.3 First Day and First Week Experience

During the first few days of my internship, I had the opportunity to meet the team and staff and get acquainted with the premises, rules, and regulations. Within these days, I was tasked to revise and test my level of strength with HTML, CSS and JavaScript by replicating the official Netflix website landing screen and it took about 10 days to realize the completion. This initial experience helped me polish my knowledge with respect to professional standards during my work review.

Overall, the experience of being introduced to the team, the technologies, and the tasks/projects was very insightful and exciting. It provided a foundation for my internship journey, allowing me to establish a common understanding of full-stack development and get a clear sense of my responsibilities and expectations. This initial exposure to the team dynamics, the product's requirements, and the company's business goals set the stage for my subsequent tasks and enabled me to make meaningful contributions to the project.

2.4 Internship Project

2.4.1 Context

This internship project focused on the design and implementation of a proposed Employee Attendance Management System (EMS) to streamline attendance registration within Go-Groups LTD. Utilizing a robust technology stack including Angular, Spring Boot, and MySQL, the EMS was designed to automate attendance recording, enhance data accuracy, and provide valuable insights into employee attendance patterns. Key features of the system include employee registration, attendance tracking. The project successfully proposes and demonstrates the effectiveness of modern technologies in improving efficiency and accuracy in employee management.

2.4.2 Problem Statement

Before the implementation of this Employee Attendance Management System, Go-Groups Ltd grappled with several significant challenges in managing employee attendance. Manual record-keeping using attendance sheets and cards checked by a controller at the door proved to be time-consuming and prone to human error, leading to inaccuracies in payroll processing and difficulties in generating accurate reports for HR analysis. The lack of a centralized system resulted in fragmented data across various spreadsheets and documents, hindering efficient data retrieval and analysis. These challenges necessitated a more efficient and reliable solution to streamline employee attendance tracking and improve overall HR management.

2.4.3 Methodology

The Employee Attendance Management System (EMS) is a comprehensive software solution proposed and designed to streamline and automate various Human Resource (HR) functions within Go-Groups LTD. Key features include employee data management and attendance registration.

Accurate and efficient attendance registration is crucial for:

- **Improved Productivity:** Identifying and addressing absenteeism, tardiness, and overtime to enhance overall productivity and efficiency.
- **Compliance:** Ensuring adherence to labor laws and regulations.
- **Data-Driven Decisions:** Analyzing attendance data to inform workforce planning, resource allocation, and scheduling strategies.
- **Reduced Administrative Burden:** Automating processes to free up HR resources for strategic initiatives.

The EMS development followed an Agile Scrum methodology, emphasizing flexibility and continuous improvement, this methodology is also used within Go-Groups LTD to manage most of their software products.

- **Short Sprints:** Development was divided into short iterations (sprints) lasting two weeks.
- **Weekly Stand-up Meetings:** Weekly meetings fostered communication, progress tracking, and issue resolution within the development team.
- **Sprint Reviews:** Regular reviews demonstrated completed functionalities to stakeholders, gathering feedback and ensuring alignment with evolving needs.

This iterative approach fostered collaboration, continuous learning, and adaptability to changing requirements, resulting in a system that effectively addresses the specific needs and challenges of Go-Groups Ltd. I used a lot of knowledge learned from my coursework in school specifically from the courses SOFTWARE DEVELOPMENT TOOLS (AGL L5G) (CEF 345) and CLIENT AND OBJECT ORIENTED MODELING AND UML (CEF 331).

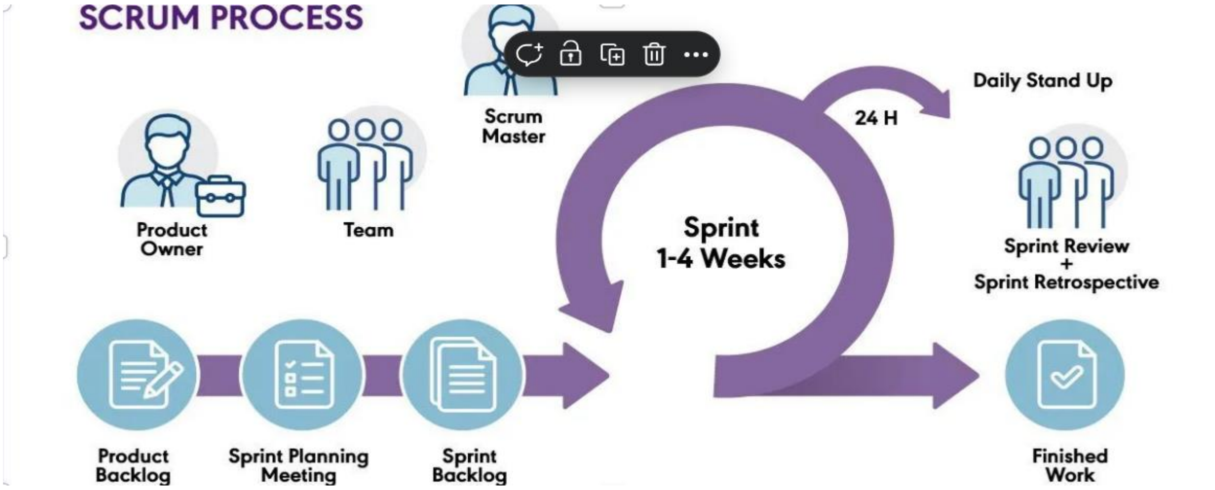


Figure 2: Agile Methodology (Scrum Framework) Structure

2.4.4 Analysis and Design Phase

The Employee Attendance Management System (EMS) was designed with a focus on modularity, scalability, and maintainability. The system employs a three-tier architecture, consisting of:

1. **Presentation Layer:** This layer handles the user interface and user interactions. It is built using Angular, providing a dynamic and responsive front-end using Angular for employees and administrators.
2. **Business Logic Layer:** This layer implements the core business logic of the system, such as employee registration and attendance tracking. It is implemented using Spring Boot, leveraging its robust framework for handling complex business rules and data processing.
3. **Data Access Layer:** This layer interacts with the database to store and retrieve employee data, attendance records, and other relevant information. MySQL was chosen as the database due to its flexibility and scalability, allowing efficient storage and retrieval of the system's data.

UML Diagrams

- **Use Case Diagram:** Visualizes user interactions with the system, identifying key functionalities and scenarios.
- **Sequence Diagram:** Illustrates the sequence of interactions between objects during a specific use case.
- **Entity-Relationship Diagram (ERD):** Represents the relationships between different entities in the system.
- **Class Diagram:** Represents the classes, attributes, and relationships between objects in the system.
- **Deployment Diagram:** Depicts the physical architecture of the system, including hardware and software components.

Data Modeling

The system's data model was designed to efficiently store and manage employee information, attendance records, and other relevant data. The following key entities were identified:

- **Employee:** Stores employee details such as name, roles, ID, department, and contact information.
- **Attendance:** Stores attendance records for each user, including date, time, and attendance status (present and absent).
- **Department:** Stores roles of different employees, including start date, end date, and approval status.
- **Roles:** Stores roles for each employee, specific to their department.

Analysis Techniques

- **Requirement Gathering:** Interviews with stakeholders to understand their needs and expectations within Go-Groups LTD.
- **Use Case Analysis:** Identifying user scenarios and interactions with the system.
- **Data Flow Diagrams (DFD):** Modeling the flow of data within the system.

System Design Diagrams (UML Diagrams)

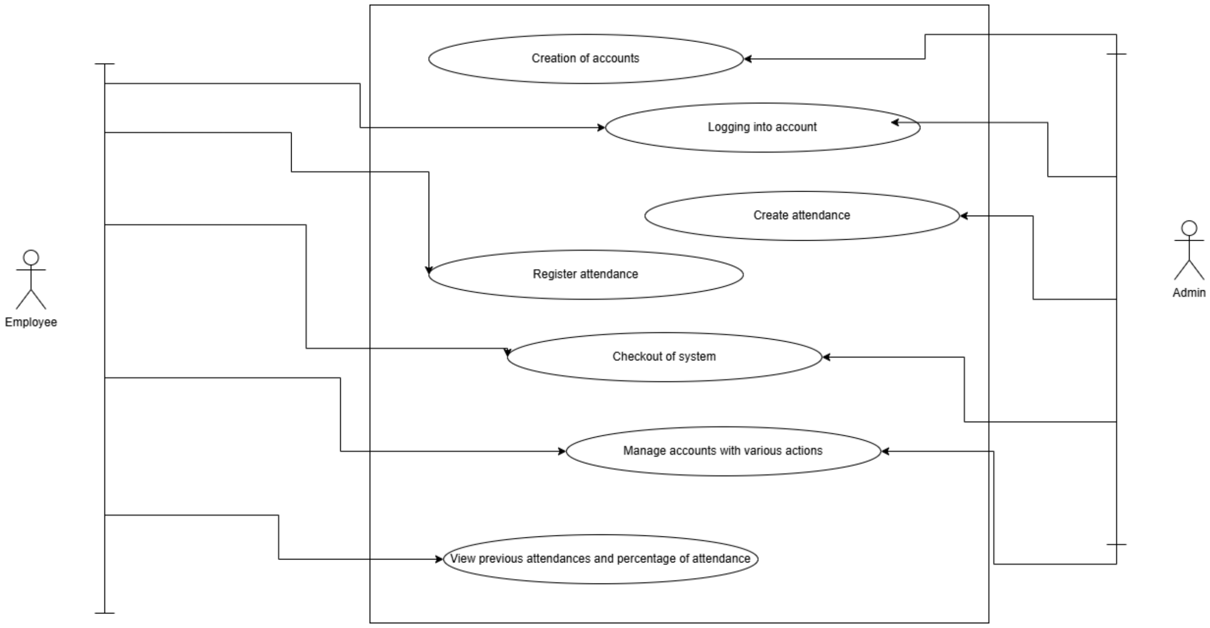


Figure 3: Use case Diagram of EMS

This diagram is a Use Case Diagram depicting the interactions between two actors, "Employee" and "Admin," and the functionalities of an attendance management system. The "Employee" actor can log in, register attendance, check out, and view their attendance history. The "Admin" actor has additional capabilities, including creating accounts, managing existing accounts, and creating attendance records. Both actors can log in and view attendance data. The diagram illustrates how different user roles interact with the system to perform specific tasks related to attendance tracking.

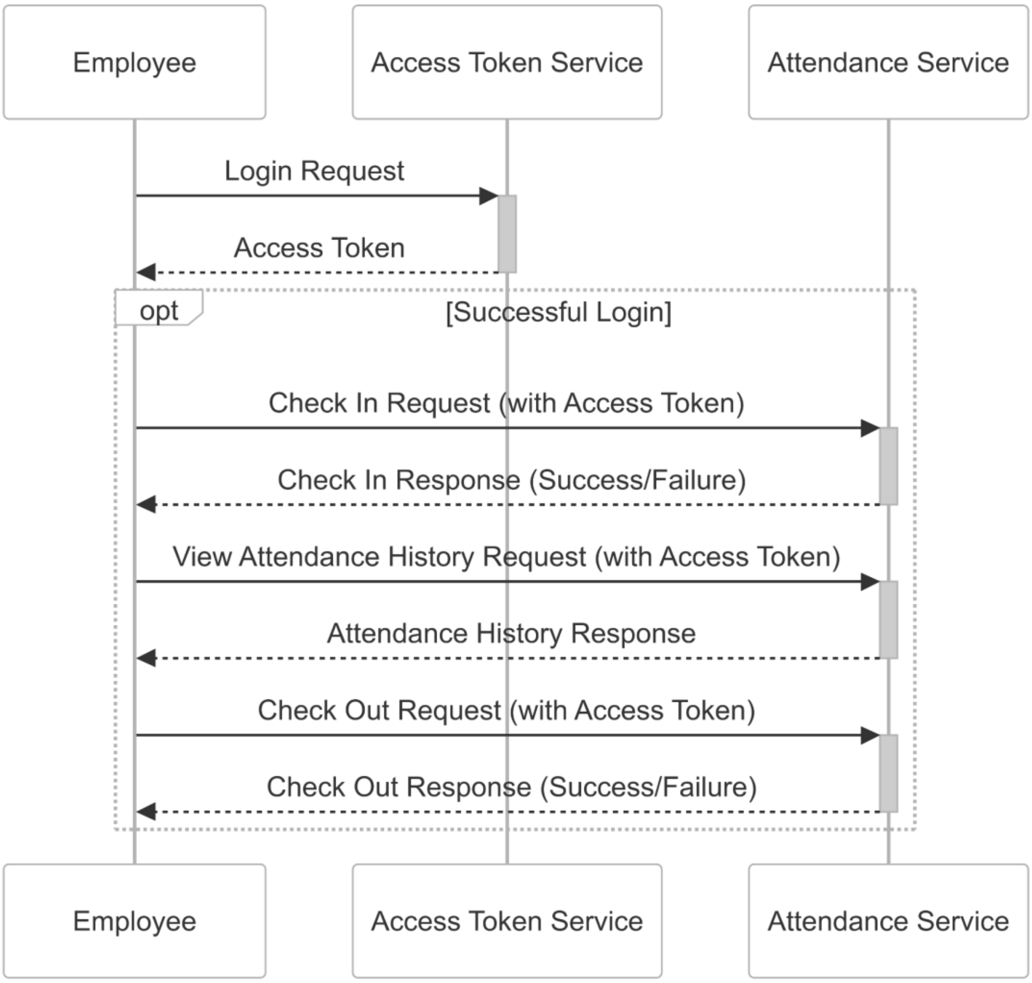

Employee
Access Token Service
Attendance Service
Login Request
Access Token
opt
[Successful Login]
Check In Request (with Access Token)
Check In Response (Success/Failure)
View Attendance History Request (with Access Token)
Attendance History Response
Check Out Request (with Access Token)
Check Out Response (Success/Failure)
Employee
Access Token Service
Attendance Service

Figure 4: Sequence Diagram of EMS

This diagram is a sequence diagram depicting the interaction flow between an Employee, an Access Token Service, and an Attendance Service. The Employee initiates the process by sending a Login Request to the Access Token Service, which responds with an Access Token upon successful authentication. Subsequently, the Employee uses this Access Token to interact with the Attendance Service. The "opt" (optional) fragment indicates that the following interactions (Check In, View Attendance History, and Check Out) only occur if the login was successful. For each of these actions, the Employee sends a request to the Attendance Service along with the Access Token, and the service responds with a success or failure message, or the requested attendance history. This diagram illustrates a typical authentication and authorization flow using access tokens in a microservices architecture.

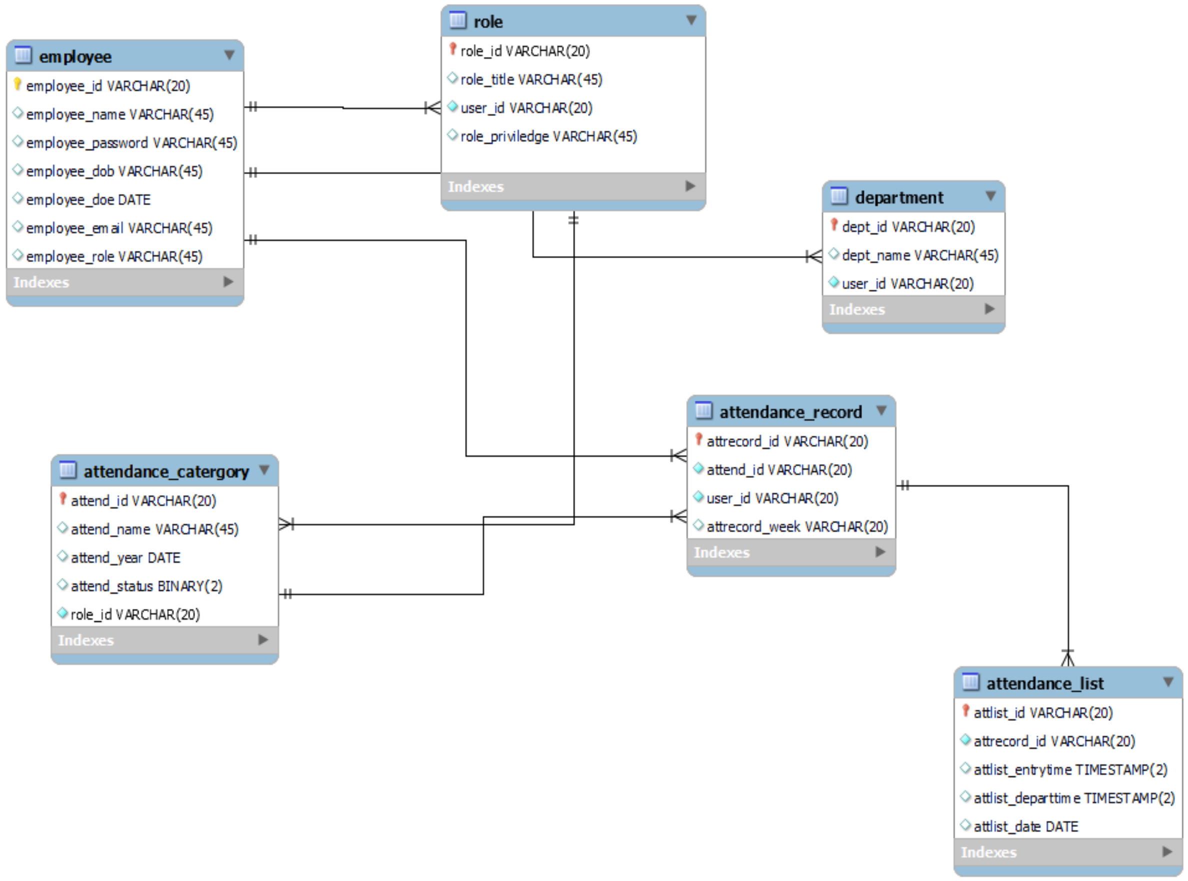


Figure 5: Entity Relation Diagram of EMS

This diagram represents an Entity-Relationship Diagram (ERD) illustrating the database structure for an attendance management system. It depicts six entities: "employee," "role," "department," "attendance_category," "attendance_record," and "attendance_list." The "user" entity stores the personal details of employees, while the "role" defines user roles and privileges. "Department" categorizes users based on their department. "Attendance_category" outlines different attendance types. "Attendance_record" stores overall attendance data, linked to users and categories. Finally, "attendance_list" records individual attendance entries with timestamps. The relationships between these entities are indicated by lines, showing how they are connected and how data flows between them. The diagram highlights the attributes (columns) within each entity and the primary and foreign keys used to establish relationships.

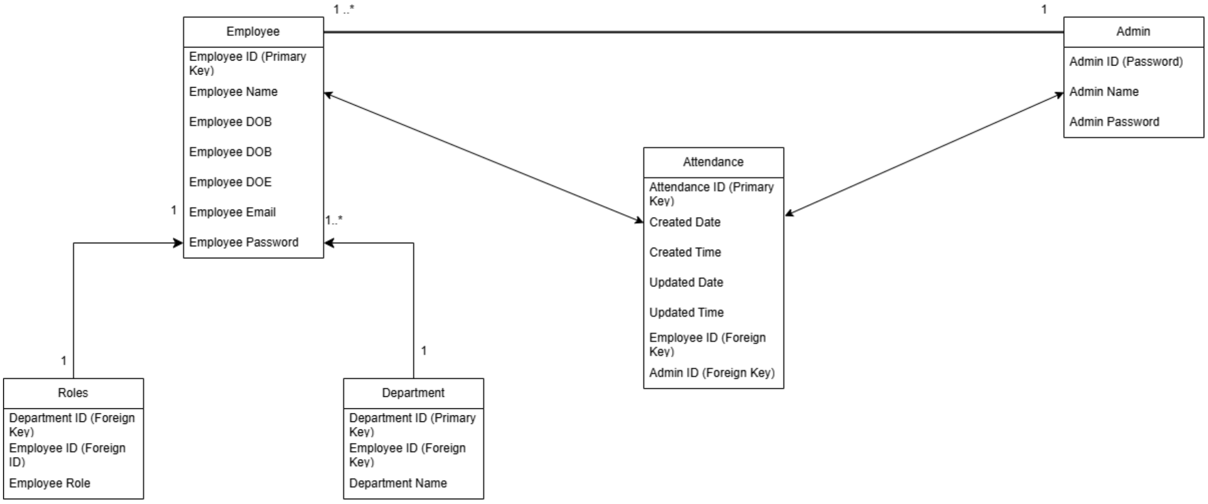


Figure 6: Class Diagram of EMS

This diagram is an Entity-Relationship Diagram (ERD) representing the database structure for an attendance management system. It illustrates the relationships between five entities: Employee, Admin, Attendance, Roles, and Department. The "Employee" entity stores employee information, uniquely identified by Employee ID (Primary Key). Each employee can have multiple roles (1..* relationship with "Roles") and belongs to one department (1 relationship with "Department"). The "Admin" entity stores administrator details, with Admin ID (Primary Key) and password. The "Attendance" entity records attendance data, linked to both employees and admins via foreign keys (Employee ID and Admin ID) and identified by Attendance ID (Primary Key). The "Roles" entity defines employee roles, with foreign keys linking back to Employee and Department. The "Department" entity stores department information, identified by Department ID (Primary Key) and linked to employees via a foreign key. The diagram visually represents how these entities relate to each other in the database, showing the primary and foreign keys used to establish connections and the cardinality of the relationships.

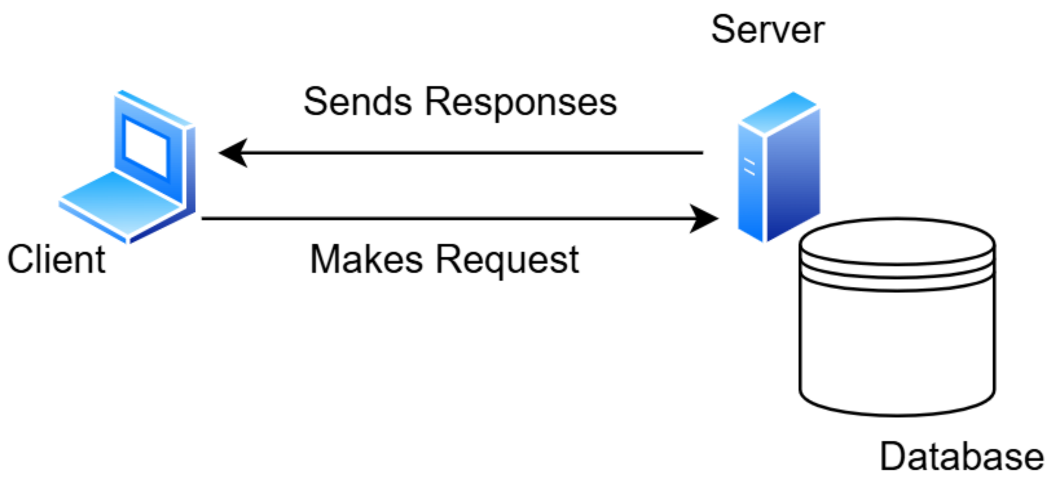


Figure 7: Deployment Diagram of EMS

This diagram illustrates a simplified client-server architecture with a database component. The Client, represented by a stylized computer, initiates the process by sending a "Makes Request" message to the Server, depicted as a rack-mounted server. The server then processes the request and interacts with the database, which is shown as a cylindrical storage unit, to retrieve or store data. Finally, the Server sends "Sends Responses" back to the Client, completing the interaction loop. This diagram highlights the fundamental flow of information in a typical client-server model, where the client requests services or data from the server, and the server processes those requests, potentially utilizing a database, before returning the results to the client.

2.4.5 Implementation

UI Design Implementation With Figma

Description: This project involves a detailed analysis of the provided UI design mockups, focusing on the implementation draft with Figma and the color psychology associated with the chosen color palette.

Implementation Details:

- **Color Palette:** Using a consistent color palette illustrated definitions for the styles, ensuring an accurate representation of dark blue and gold.
- **Typography:** The typography was established using Figma's text styles, considering font family, size, weight, and spacing.
- **Grid System:** The grid system was implemented to maintain consistency and visual hierarchy across the design.

Color Sensation:

Dark Blue: Evokes feelings of trust, reliability, and stability. It is often associated with authority, professionalism, and sophistication. In this design, dark blue conveys a sense of security and confidence in the application's functionality.

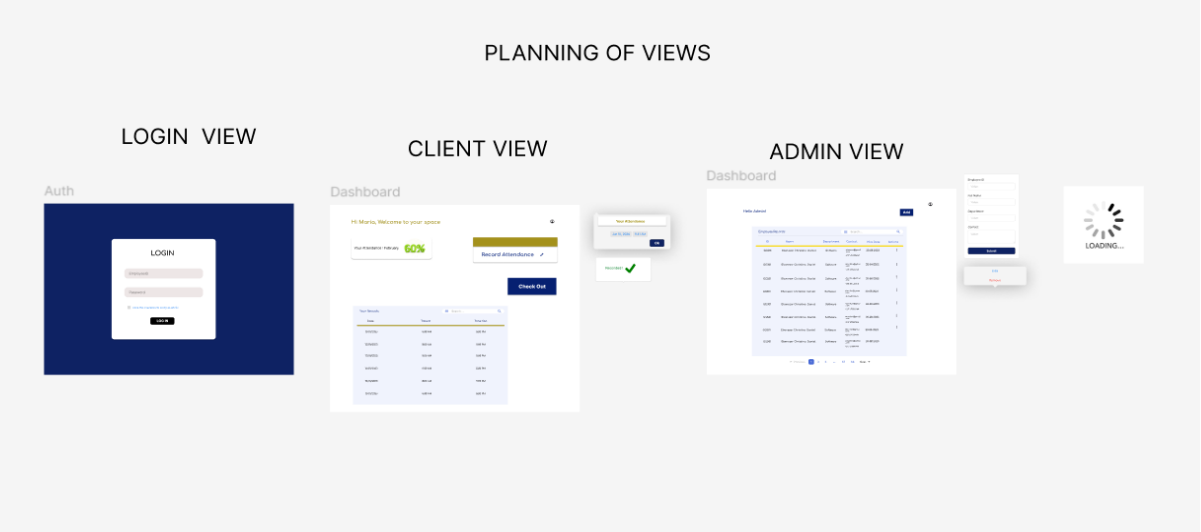


Figure 8: UI Draft with Figma

1. Front-End Development (Angular)

The front-end of the system was developed using Angular, a robust framework for building dynamic and responsive web applications. The following steps were taken to ensure a seamless user experience:

Component Development

Angular components were created to represent various elements of the user interface. These components include:

- **Login Module**: For employee and admin details such as employee ID and password on entry into the system.
- **Employee Attendance Module**: Allow employees to check in and out of their shifts at the current timestamp.
- **Admin Dashboard Module:** For administrators to view attendance records, and employee information and perform administrative actions like updates.

Each component was developed using **HTML** for structure, **CSS** for styling, and **TypeScript** for logic and interactivity.

Data Binding

Two-way data binding was implemented to synchronize data between the user interface and the underlying data model. This ensures that any changes made in the UI are immediately reflected in the data model and vice versa. For example:

- When an employee signs in for attendance, the data is instantly updated in the database.
- When an administrator views attendance records, the data is dynamically fetched and displayed in real-time.

Routing

Angular's routing module was utilized to enable seamless navigation between different sections of the application. Key routes include:

- **Employee Dashboard**: Displays employee attendance details and shows a log of attendance records for each employee
- **Admin Dashboard**: Provides administrators with an overview of system data and tools for managing employees.

Styling and User Interface Design

CSS and Angular Material were used to ensure a visually appealing and user-friendly interface. Key design considerations included:

- **Responsive Design**: Ensuring the application is accessible on various devices, including desktops, tablets, and smartphones.
- **Consistent Themes**: Using Angular Material components like buttons, cards, and tables to maintain a consistent look and feel.
- **Accessibility**: Implementing features like keyboard navigation and screen reader support to make the system inclusive.

2. Back-End Development (Spring Boot)

The backend of the Employee Management System (EMS), built with Spring Boot, employed a layered architecture to ensure a clean separation of concerns and maintainability. At the core were the entity files, which represented the database tables as Java objects, using JPA annotations to map them to the underlying relational database. These entities encapsulated the data structure for employees, departments, roles, and other relevant information. Repository interfaces, leveraging Spring Data JPA, provided a high-level abstraction for data access, minimizing boilerplate code for CRUD operations. These interfaces automatically generate database queries based on method naming conventions or custom JPQL queries. The service layers acted as intermediaries between the repositories and the controllers, implementing the business logic of the application. They handled data validation, and complex calculations, and orchestrated interactions between different repositories. By separating the data access and business logic into distinct layers, the system achieved better modularity, testability, and scalability. Spring Boot's dependency injection features facilitated the seamless integration of these components, enabling the efficient development of a robust and maintainable backend for the EMS.

3. Database Integration (MySQL)

The system uses MySQL as its relational database management system. The database was designed to store and manage all employee-related data efficiently.

Database Setup

A MySQL database was created with the following tables:

- **Employees Table**: Stores personal information, contact details, employee information, and role.
- **Attendance Tables**: Records employee sign-in and sign-out times and dates.
- **Admin Table**: Logs actions performed by administrators for auditing purposes.

Data Modeling and Migration

A robust data model was designed to ensure data integrity and efficiency. Key steps included:

- **Defining Relationships**: Establishing relationships between tables using primary and foreign keys. For example, the Attendance table is linked to the Employees table via an employee ID.
- **Normalization**: Applying normalization principles to eliminate redundancy and improve data consistency.
- **Data Migration**: Developing scripts to migrate existing employee data from legacy systems or spreadsheets into the new database.

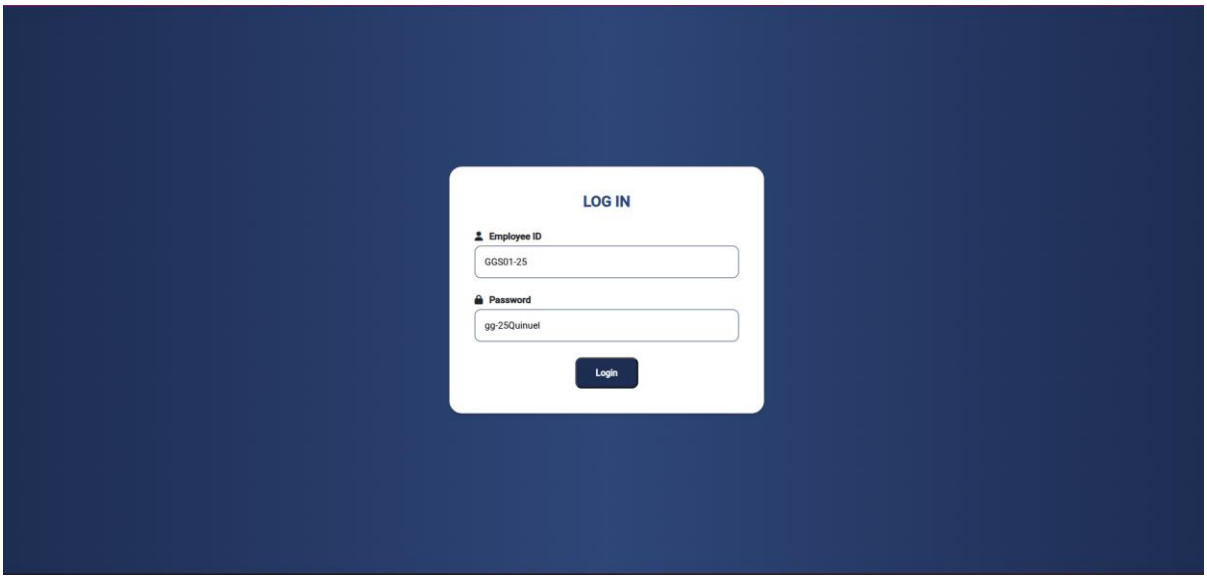
Database Roles and Permissions

To ensure data security, roles and permissions were defined:

- **Employee Role**: Allows access only to attendance modules.
- **Admin Role**: Grants full access to all system features, including the ability to view and edit employee data.

2.4.6 Outcome and Result

Highlights of the implementation can be seen in the figures below:

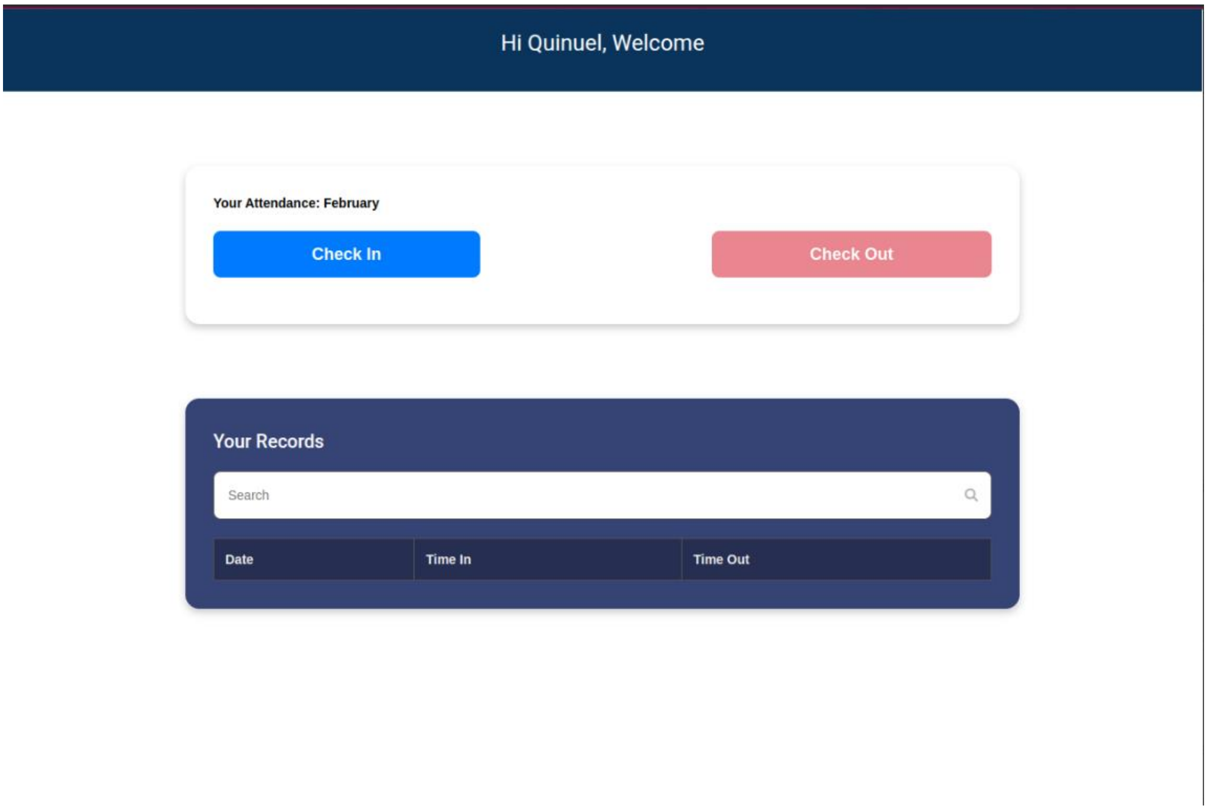


Employee ID: GGS01-25

Password: gg-25Quinuel

Figure 9: Employee Login View

This view depicts an employee logging into the system with their employee ID and password. Then an access token provided channeling the employee into the system after authenticating themselves



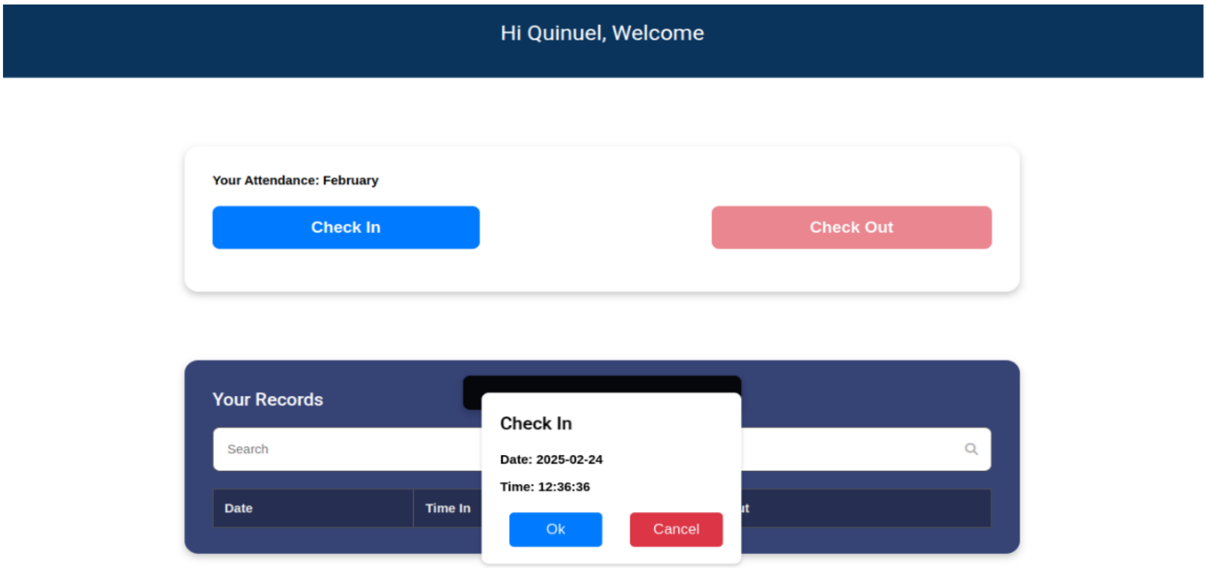
Hi Quinuel, Welcome

Your Attendance: February

Date	Time In	Time Out

Figure 10: Employee Login Attendance View

This view depicts the dashboard of the employee, showing a Check In button, Check Out button and a table of attendance records



Hi Quinuel, Welcome

Your Attendance: February

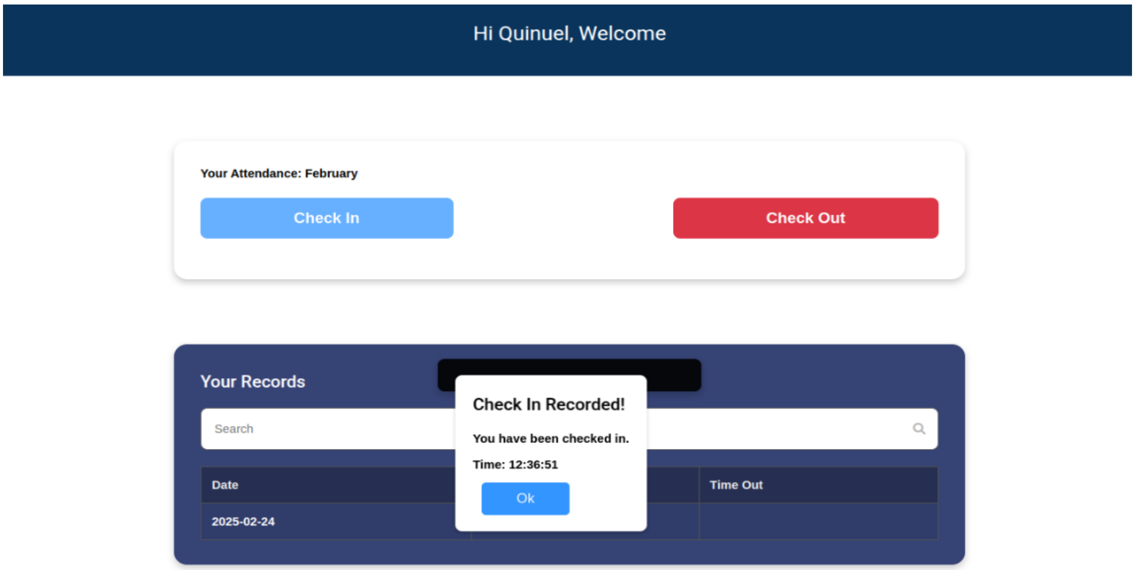
Check In

Date: 2025-02-24

Time: 12:36:36

Figure 11: Employee Checking in his/her attendance

This view depicts an employee registering their Check In time, and so on clicking the respective button, the other stays non-functional until the other is recorded. On clicking the Check In button, a pop-up is seen, to validate the date and time



Hi Quinuel, Welcome

Your Attendance: February

Date		Time Out
2025-02-24		

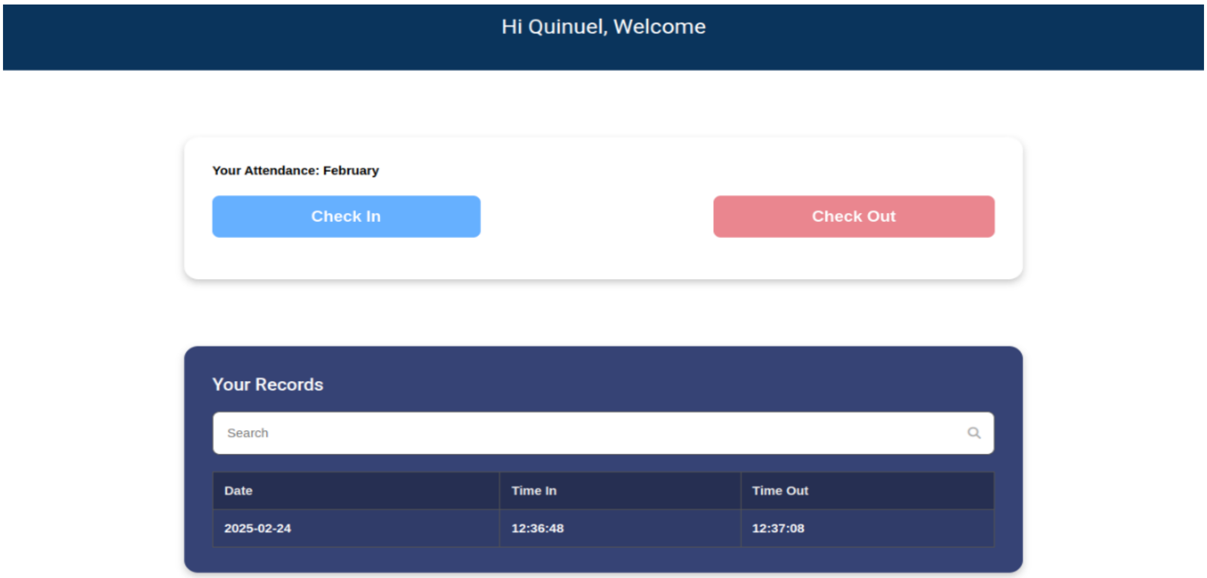
Check In Recorded!

You have been checked in.

Time: 12:36:51

Figure 12: Employee's attendance verified

This view depicts an employee who has registered Check In time. On clicking the Check In button, a pop-up is seen, to validate the date and time, followed by a successful confirmation message.



Hi Quinuel, Welcome

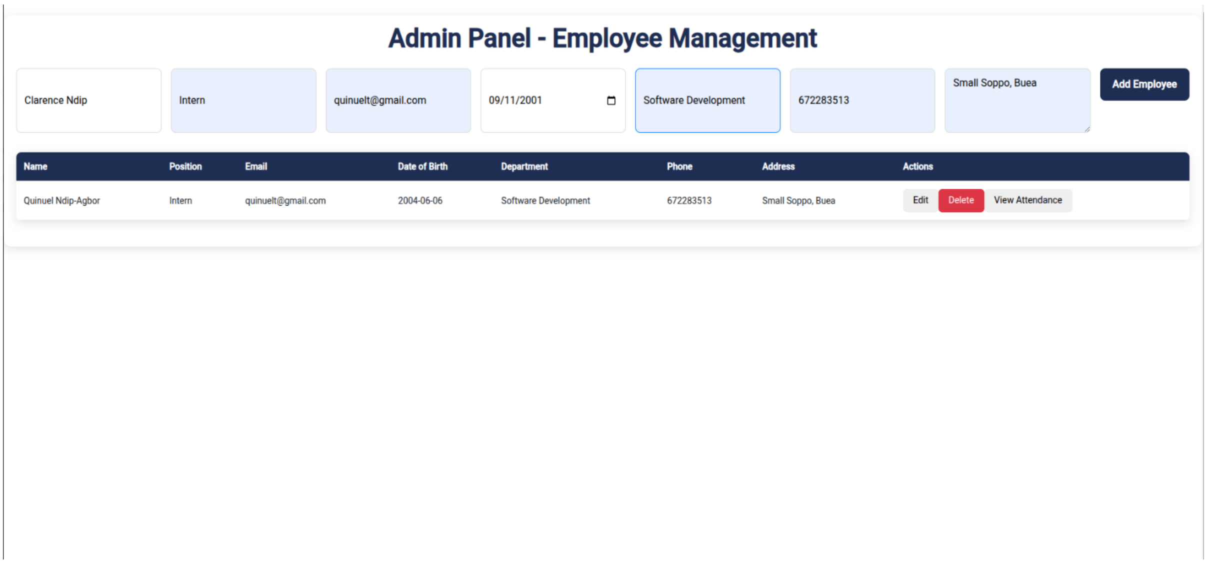
Your Attendance: February

Your Records

Date	Time In	Time Out
2025-02-24	12:36:48	12:37:08

Figure 13: Employee's attendance submitted for the day

This view depicts an employee's attendance recorded for a given day with the Check In and Check Out details.



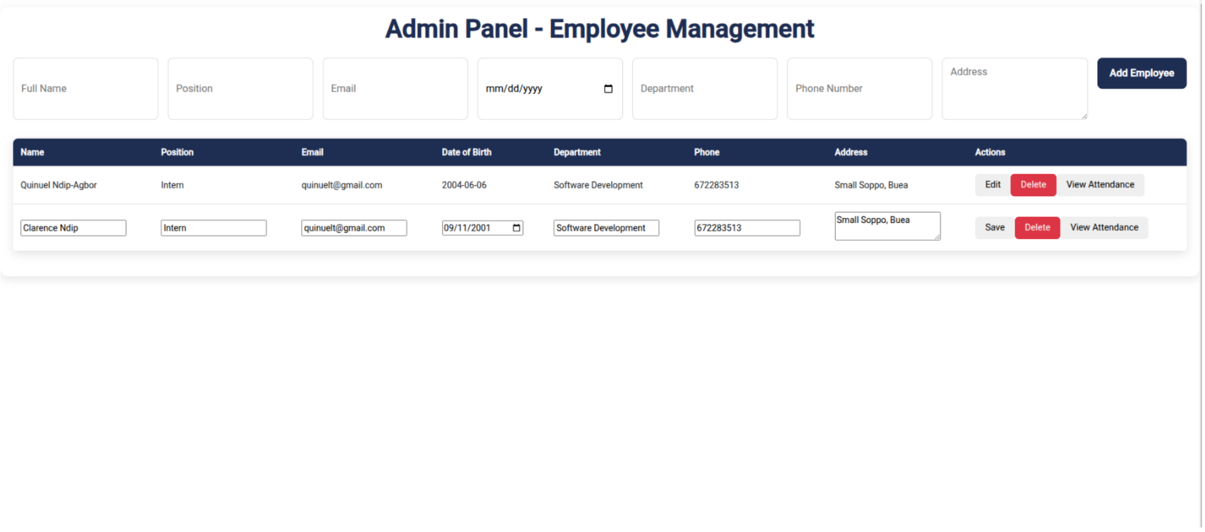
Admin Panel - Employee Management

Clarence Ndip | Intern | quinuelt@gmail.com | 09/11/2001 | Software Development | 672283513 | Small Soppo, Buea

Name	Position	Email	Date of Birth	Department	Phone	Address	Actions
Quinuel Ndip-Agbor	Intern	quinuelt@gmail.com	2004-06-06	Software Development	672283513	Small Soppo, Buea	Edit Delete View Attendance

Figure 14: Admin's Dashboard View (Employee added to the system)

This view depicts the admin's dashboard adding a new employee to the system

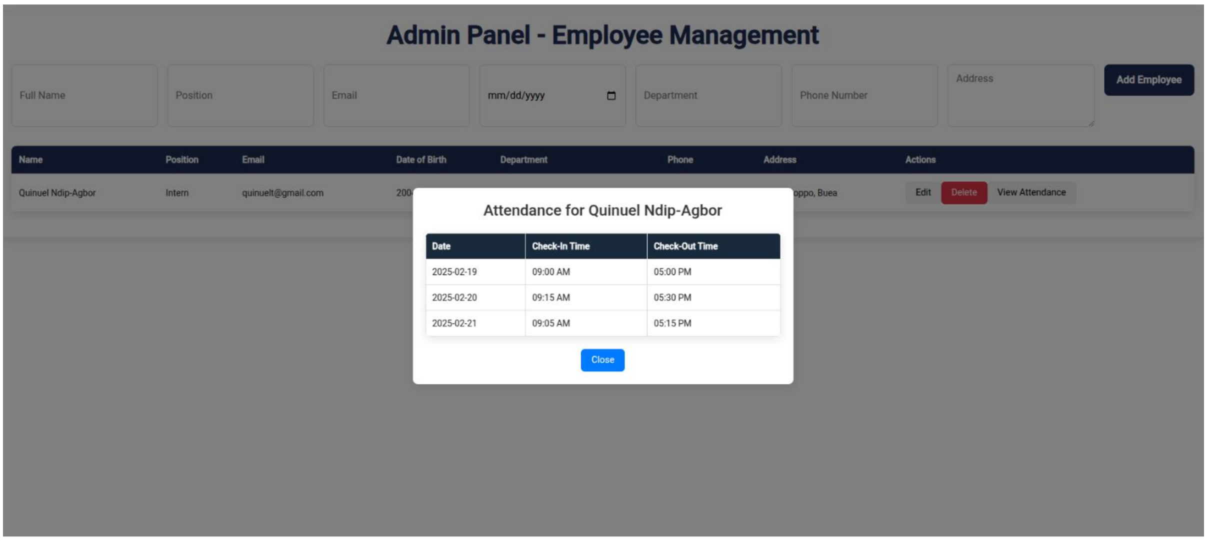


Admin Panel - Employee Management

Name	Position	Email	Date of Birth	Department	Phone	Address	Actions
Quinuel Ndip-Agbor	Intern	quinuelt@gmail.com	2004-06-06	Software Development	672283513	Small Soppo, Buea	Edit Delete View Attendance
Clarence Ndip	Intern	quinuelt@gmail.com	09/11/2001	Software Development	672283513	Small Soppo, Buea	Save Delete View Attendance

Figure 15: Admin editing Employee's Details

This view depicts the different actions that can be carried out by the admin which include: Edit, Delete, and View Attendance.



Admin Panel - Employee Management

Name	Position	Email	Date of Birth	Department	Phone	Address	Actions
Quinuel Ndip-Agbor	Intern	quinuelt@gmail.com	200			oppo, Buea	Edit Delete View Attendance

Attendance for Quinuel Ndip-Agbor

Date	Check-In Time	Check-Out Time
2025-02-19	09:00 AM	05:00 PM
2025-02-20	09:15 AM	05:30 PM
2025-02-21	09:05 AM	05:15 PM

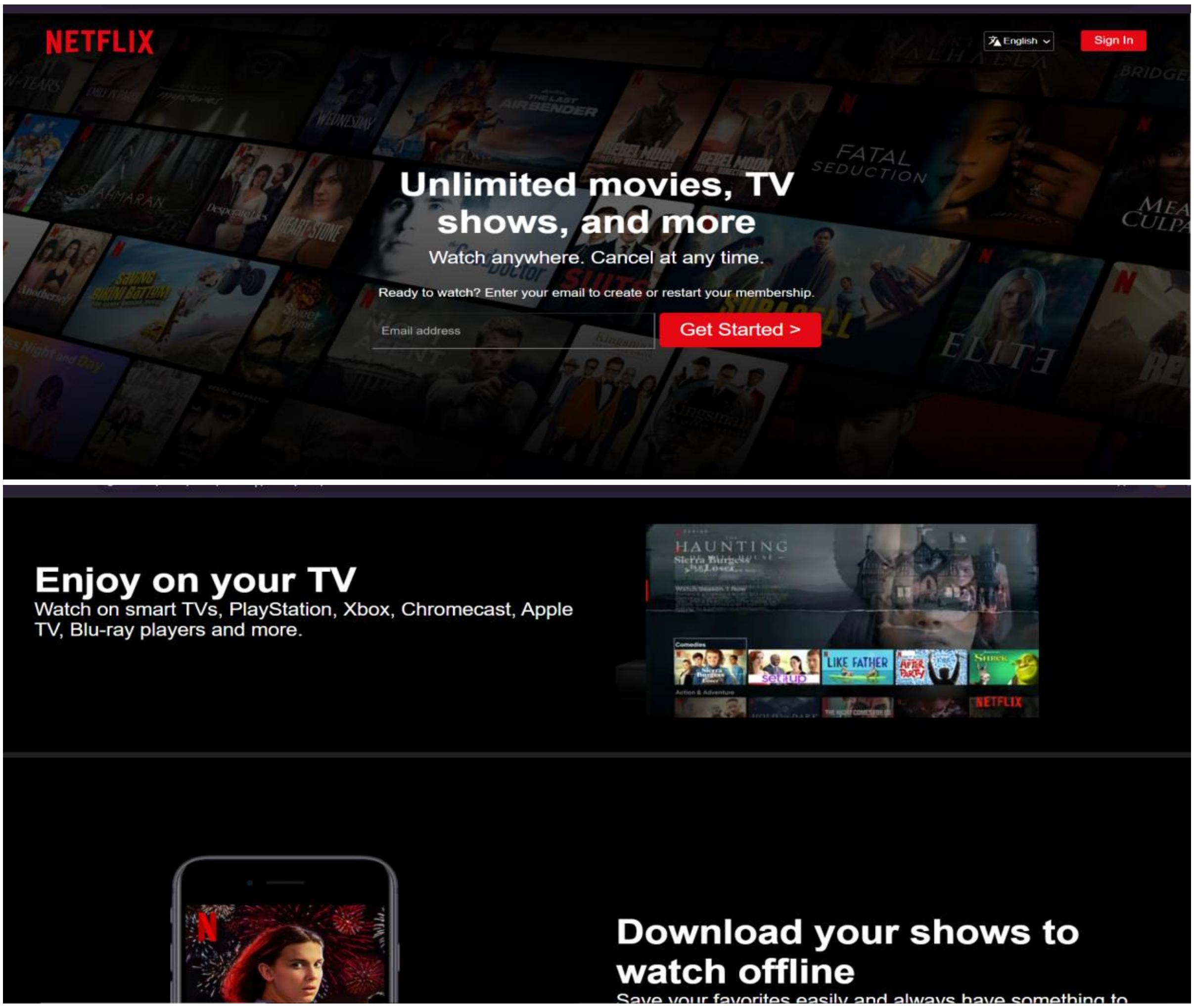
Figure 16: Admin Viewing Employee's Attendance

This view depicts the admin viewing an employee's attendance which he/she recorded.

2.4.7 Other Projects and Tasks

2.4.7.1 Netflix Home Page with HTML, CSS, and JavaScript:

The figure below shows a replication of the UI of the Netflix home screen using HTML, CSS, and JavaScript which is currently deployed on Vercel.



Unlimited movies, TV shows, and more

Watch anywhere. Cancel at any time.

Ready to watch? Enter your email to create or restart your membership.

Enjoy on your TV

Watch on smart TVs, PlayStation, Xbox, Chromecast, Apple TV, Blu-ray players and more.

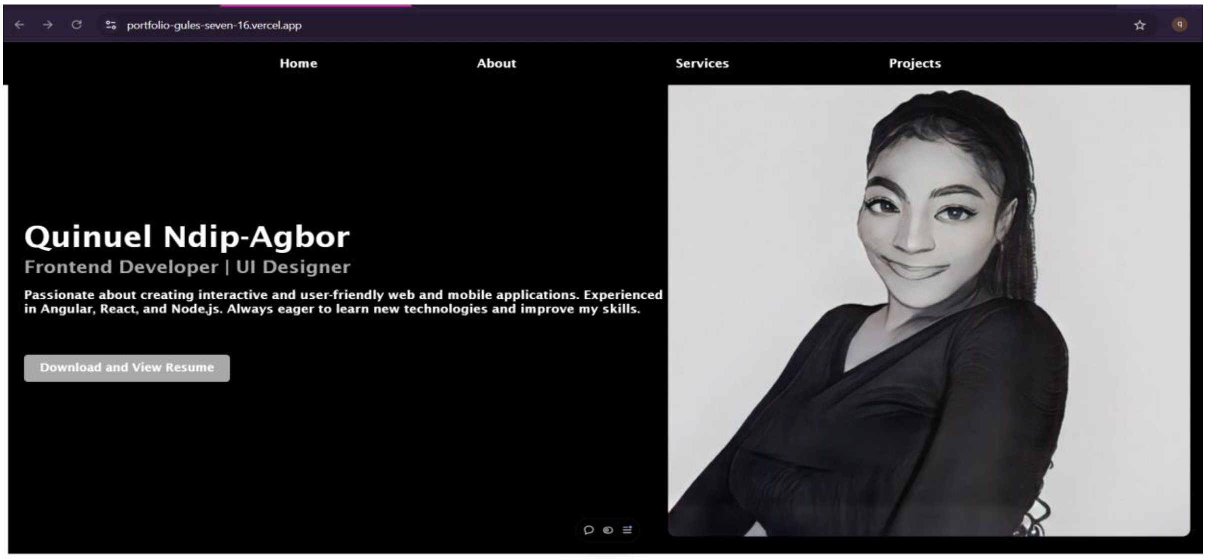
Download your shows to watch offline

Save your favorites easily and always have something to

Figure 17: Netflix Project with HTML, CSS, and JavaScript

2.4.7.2 Personal Portfolio with Angular and CSS:

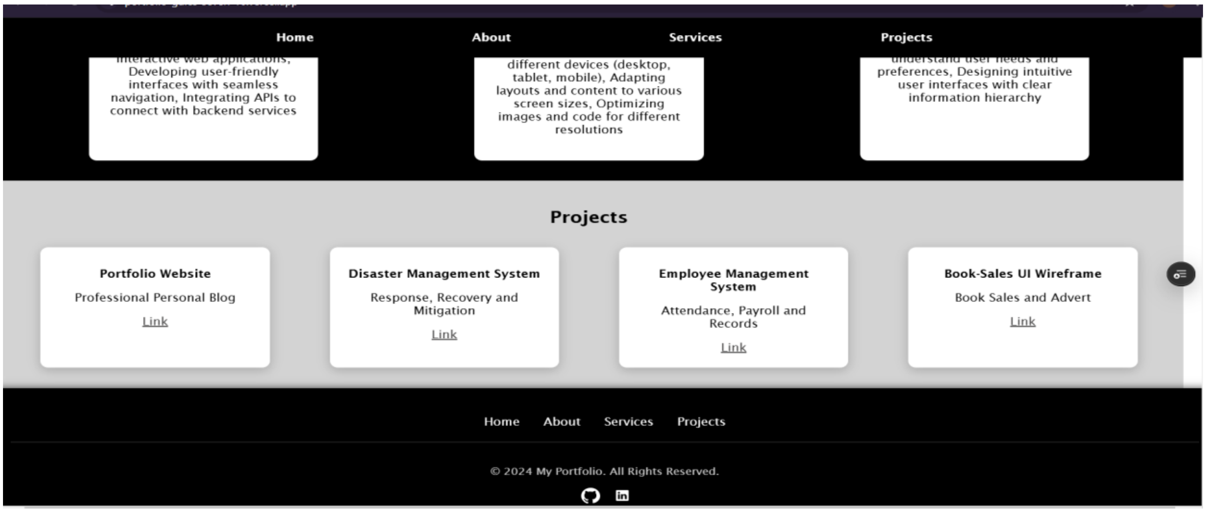
The figure below shows the UI of a mini-professional portfolio using Angular and CSS, which is currently deployed on Vercel.



Quinuel Ndip-Agbor

Frontend Developer | UI Designer

Passionate about creating interactive and user-friendly web and mobile applications. Experienced in Angular, React, and Node.js. Always eager to learn new technologies and improve my skills.



interactive web applications, Developing user-friendly interfaces with seamless navigation, Integrating APIs to connect with backend services

different devices (desktop, tablet, mobile), Adapting layouts and content to various screen sizes, Optimizing images and code for different resolutions

understand user needs and preferences, Designing intuitive user interfaces with clear information hierarchy

Projects

Portfolio Website
Professional Personal Blog
Link

Disaster Management System
Response, Recovery and Mitigation
Link

Employee Management System
Attendance, Payroll and Records
Link

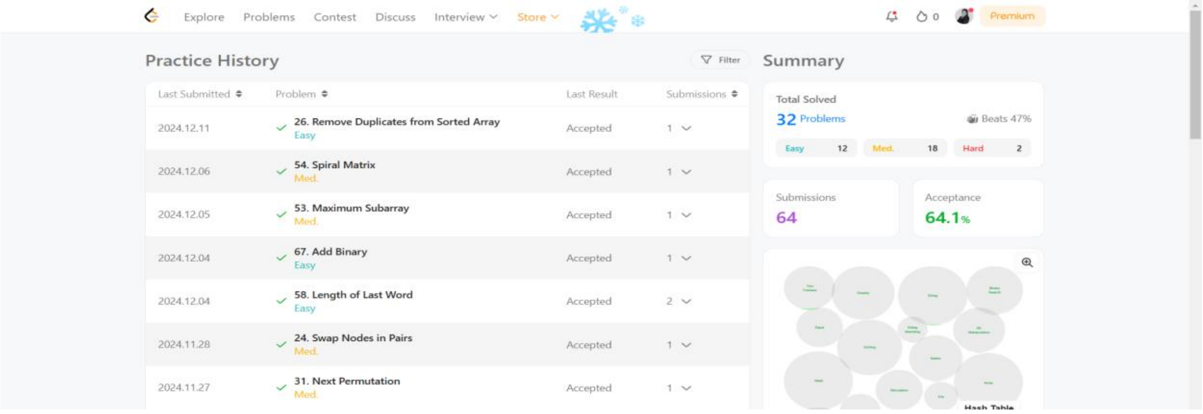
Book-Sales UI Wireframe
Book Sales and Advert
Link

© 2024 My Portfolio. All Rights Reserved.

Figure 18: Portfolio Project

2.4.7.3 Leetcode and Hackerrank

During my internship, I spent a significant amount of time solving problems ranging from easy to hard to strengthen my backend skills, using Typescript and Java summing a total of 100 problems together. It was amazing and I documented a good number of the problems I solved on GitHub.



Practice History

Last Submitted	Problem	Last Result	Submissions
2024.12.11	✓ 26. Remove Duplicates from Sorted Array Easy	Accepted	1
2024.12.06	✓ 54. Spiral Matrix Med.	Accepted	1
2024.12.05	✓ 53. Maximum Subarray Med.	Accepted	1
2024.12.04	✓ 67. Add Binary Easy	Accepted	1
2024.12.04	✓ 58. Length of Last Word Easy	Accepted	2
2024.11.28	✓ 24. Swap Nodes in Pairs Med.	Accepted	1
2024.11.27	✓ 31. Next Permutation Med.	Accepted	1

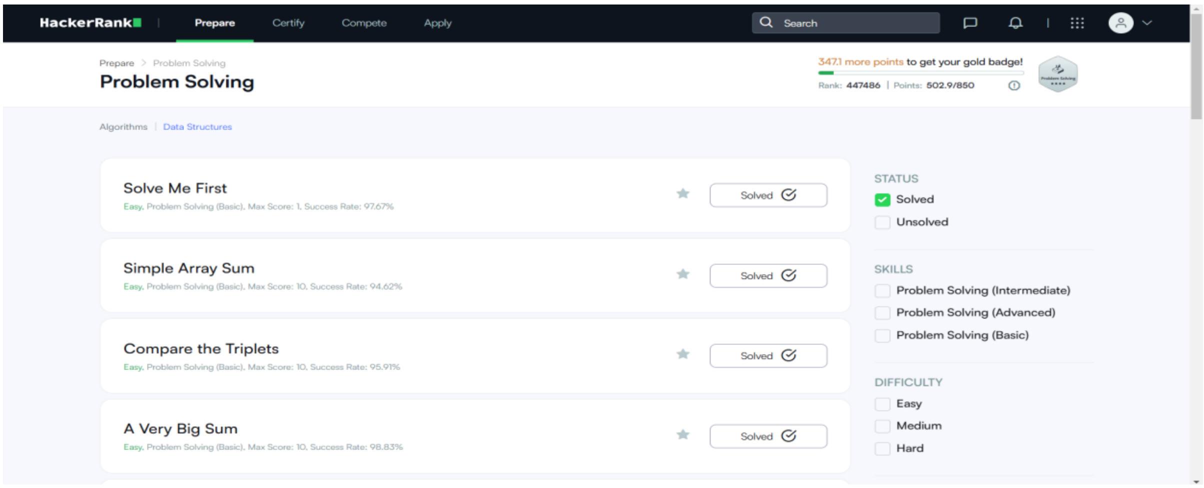
Summary

Total Solved
32 Problems
Beats 47%

Easy	12	Med.	18	Hard	2

Submissions: 64
Acceptance: 64.1%

Figure 19: Leetcode History



Prepare > Problem Solving

Problem Solving

347.1 more points to get your gold badge!
Rank: 447486 | Points: 502.9/850

Solve Me First
Easy, Problem Solving (Basic), Max Score: 1, Success Rate: 97.67%
Solved ✓

Simple Array Sum
Easy, Problem Solving (Basic), Max Score: 10, Success Rate: 94.62%
Solved ✓

Compare the Triplets
Easy, Problem Solving (Basic), Max Score: 10, Success Rate: 95.91%
Solved ✓

A Very Big Sum
Easy, Problem Solving (Basic), Max Score: 10, Success Rate: 98.83%
Solved ✓

STATUS
☑ Solved
☐ Unsolved

SKILLS
☐ Problem Solving (Intermediate)
☐ Problem Solving (Advanced)
☐ Problem Solving (Basic)

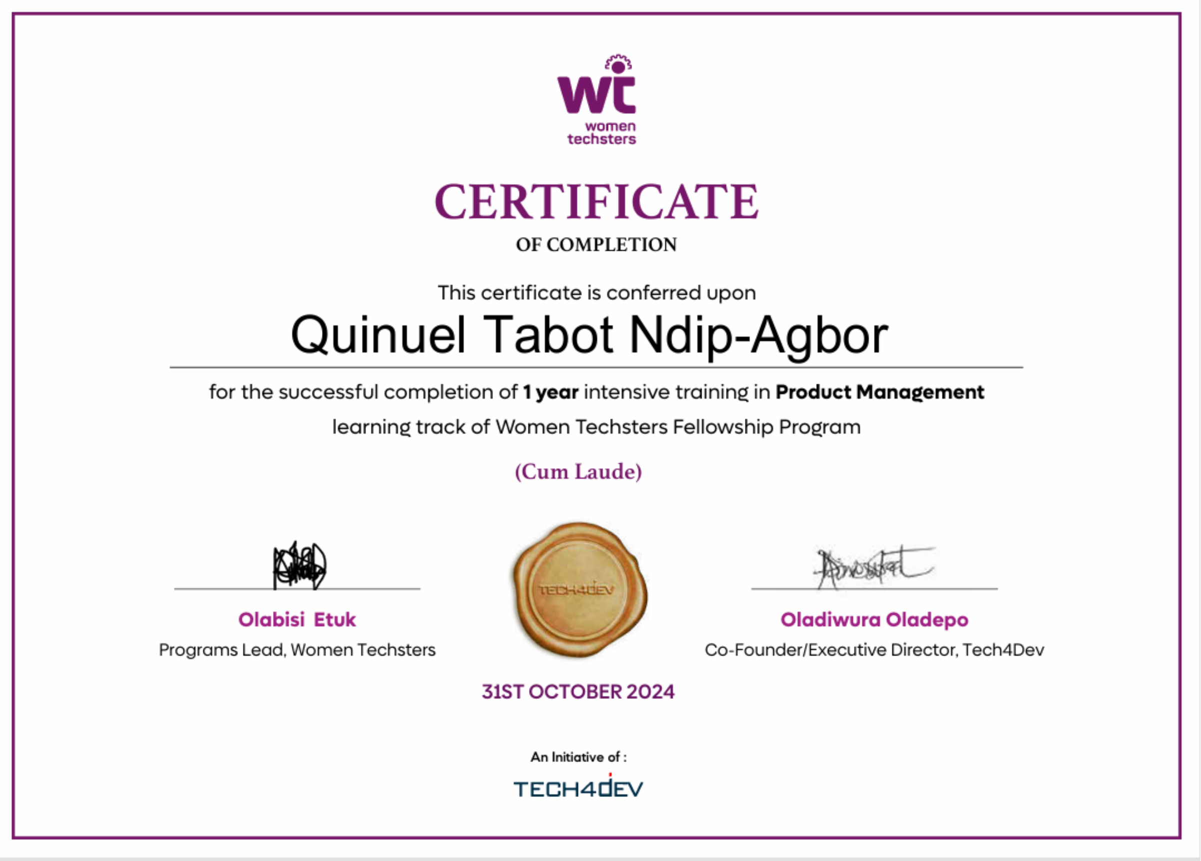
DIFFICULTY
☐ Easy
☐ Medium
☐ Hard

Figure 20: Hackerrank History

CHAPTER THREE: EXTRACURRICULAR ACTIVITIES

3.1 Participating in a Product Management Program organized by Women Techsters, Nigeria

I began my internship towards the last three months of the online course organized by Women Techsters, Nigeria for the Product Management track. It was a wonderful journey that involved practical work, online exams, volunteerism, online conferences, and group projects. This correlated with my software development training as it enabled me to understand in detail the concepts behind building valuable software. This helped me boost my portfolio, network with amazing tech enthusiasts, and gain experience.



wt women techsters

CERTIFICATE

OF COMPLETION

This certificate is conferred upon

Quinuel Tabot Ndip-Agbor

for the successful completion of **1 year** intensive training in **Product Management** learning track of Women Techsters Fellowship Program

(Cum Laude)

Olabisi Etuk
Programs Lead, Women Techsters

TECH4DEV

31ST OCTOBER 2024

Oladiwura Oladepo
Co-Founder/Executive Director, Tech4Dev

An Initiative of :
TECH4DEV

Figure 21: Certificate of Completion

3.2 Christmas Party at the Office

It was an amazing way to end the year and round up for the Christmas break with Go-Groups, with a little Christmas party that involved: greeting the team, exchanging gifts, and fun games.

Figure 22: Christmas Party at the office

3.3 Content Creation with the Marketing Team for the YouTube Channel

It was extremely a lot of fun playing stack of cups with the team organized by the marketing team for content, used on the social media platforms owned by Go-Groups LTD

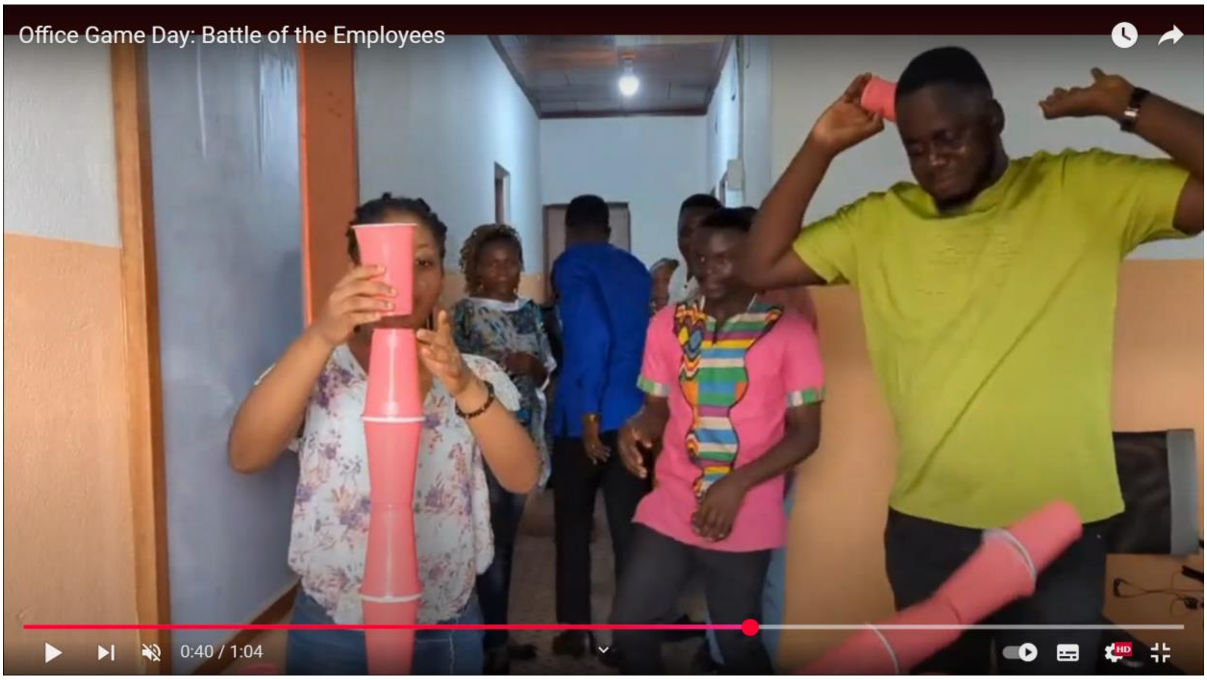


Figure 23: Creating YouTube Content with the marketing team

CHAPTER FOUR: DIFFICULTIES AND CHALLENGES

During my internship, I encountered minor challenges that affected my work experience temporarily. Let's go into detail about each of these challenges:

1) It was a requirement from the company that I use the Ubuntu OS for my projects, and being familiar with Windows and moving to a new operating system was quite a challenge. But it helped me learn a lot better, run source codes smoothly, and improve my overall experience.

2) I also encountered difficulties meeting deadlines accurately during the first months of the training due to my skill set level at the beginning but I was eventually able to get on track with the projects.

3) I faced difficulties managing errors in my codes, I faced errors that made me spend up to a week sorting out. However, I was able to thrive with the help of my supervisors and available resources.

CHAPTER FIVE: CONCLUSION AND RECOMMENDATION

5.1 Conclusion

In conclusion, my internship experience encompassed various aspects, including participating in the Women Techsters Online course, building projects with Angular and Spring Boot, and facing challenges, and adapting to a new environment. Despite these challenges, I was able to make the most of my internship, gaining valuable skills, expanding my professional network, and demonstrating resilience in overcoming obstacles.

5.2 Recommendations

1. Using Official Documentations: One major thing I learned from my challenges with source codes and learning new technologies is to use primary sources like documentation from software developers or companies online and understand them better before using secondary sources like YouTube and Stack Overflow.

2. Adaptability and Learning: Adapting to a new environment can be challenging, but it is also an opportunity for growth. Continue to cultivate a mindset of adaptability and embrace new technologies, tools, and frameworks. Actively seek out learning resources, online courses, and professional communities to further enhance my skills and stay updated with industry trends.

CHAPTER SIX: APPENDICES AND REFERENCES

Frameworks & Technologies Official Documentations:

Angular Team. (2023). *Angular documentation*. https://angular.io/

VMware. (2022). *Spring Boot documentation*. https://spring.io/projects/spring-boot/

Oracle Corporation. (n.d.). *Oracle Java documentation*. https://docs.oracle.com/en/java/

GitHub. (2022). *GitHub docs*. https://docs.github.com/

Mozilla. (2019). *CSS: Cascading Style Sheets*. MDN Web Docs. https://developer.mozilla.org/en-US/docs/Web/CSS

Software Engineering Principles:

Beck, K., Beedle, M., van Benedum, A., Cockburn, A., Cunningham, W., Fowler, M., & Thomas, D. (2001). *Manifesto for agile software development*. https://agilemanifesto.org/

Schwaber, K., & Sutherland, J. (2020). *The scrum guide*. https://scrumguides.org/

Other Aids:

Stack Overflow. (n.d.). *Stack Overflow*. https://stackoverflow.com/

W3Schools. (n.d.). *W3Schools*. https://www.w3schools.com/

Google. (2024). *Gemini*. https://gemini.google.com/app

Freepik. (2020). *Freepik*. https://www.freepik.com/

Coursework:

Dr. Djouela, I. (2022). *Object oriented modeling and UML coursework*. Faculty of Engineering and Technology, University of Buea.

Dr. Sop Deffo, L. (2023). *Client-server and web application development (PHP, JavaScript, XML, J2EE) coursework*. Faculty of Engineering and Technology, University of Buea.