

Shipment/delivery System

Senior Project

by

Antonio E. Al Makdissi,

Fawzy R. Ghazawy

Submitted to the School of Engineering of the

Lebanese International University

Tripoli, Lebanon

in partial fulfillment of the requirements for the degree of

BACHELOR OF SCIENCE IN COMPUTER ENGINEERING

Spring 2022-2023

Approved by:

Supervisor

Dr. Zaher Merhi

Committee Member

Dr. Milia Habib

DEDICATION

First and foremost, we would like to dedicate this accomplishment for ourselves, our past selves, to bring to light the progress we have made along the years starting from scratch till what we are today. It represents the culmination of our hard work and dedication thus far. However, this milestone does not serve as the endpoint of our aspirations, as we are going to strive in order to achieve even greater goals embracing further education and hard work through the many years to come. This dedication serves as a reminder of our unwavering determination to forge our path and make a lasting impact on the world.

Antonio E. Al Makdissi

& Fawzy R. Ghazawy

ACKNOWLEDGMENT

We would like to express our sincere gratitude to everyone who contributed to the success of this project. Firstly, we would like to thank our supervisor, Dr. Zaher Merhi, as well as the Dean and the rest of the individuals who spared no effort in guiding and instructing us throughout the different steps of the project.

We appreciate the time and effort that they invested in helping us develop our skills and achieve our goals. We would also like to acknowledge Dr. Milia Habib and other individuals who provided us with valuable insights and advice.

Finally, we extend our deepest appreciation to the team members who put forth tremendous dedication and hard work towards the success of this project. Their tireless efforts and collaboration were instrumental in achieving the project objectives. We are grateful for the teamwork and the valuable insights and perspectives each team member brought to the table. Their contributions and commitment to excellence have been remarkable and have paved the way for a successful outcome. It has been an enriching experience working with such a talented and motivated team. We would like to thank each team member for their remarkable work and commitment to this project. Without their contributions, this project would not have been possible. We are grateful for the opportunity to have worked on this project and for the knowledge and experience gained through this journey.

ABSTRACT

The cargo tracking app is a logistics solution designed to address the challenges faced by logistics providers and customers, such as lack of visibility, inefficient processes, limited communication, security concerns, competition, and increasing demand. The application provides real-time tracking, process automation, improved communication, enhanced security measures, a unique value proposition, and valuable data insights to help logistics providers optimize their operations, reduce costs, increase revenue, and improve customer satisfaction. The project involved utilizing various implementation tools, developing test cases and acceptance criteria, and overcoming several challenges to achieve the desired outcome. The cargo tracking app has the potential to revolutionize the logistics industry by streamlining processes and enhancing customer experience.

TABLE OF CONTENTS

DEDICATION	2
ACKNOWLEDGMENT	3
ABSTRACT	4
TABLE OF CONTENTS	5
LIST OF FIGURES	8
LIST OF TABLES	11
LIST OF SYMBOLS	12
CHAPTER 1 INTRODUCTION	1
1.1 Background	1
1.2 Problem Statement	2
1.3 General overview of the project	4
1.4 Project Outline	5
CHAPTER 2 SURVEY OF EXISTING METHODS AND SIMILAR SYSTEMS	7
2.1 Introduction	7
2.2 System 1: Aramex	7
2.3 System 2: Wakilni	9
2.4 System 3: GoShare	10
2.5 Methods/Systems Comparison	12
2.6 Conclusion and Motivation	15
CHAPTER 3 SYSTEM DESIGN	16

3.1	Introduction	16
3.2	Requirements and Specification Analysis	16
3.2.1	Functional Requirements	17
3.2.2	Use Case Diagrams	18
3.3	System Architecture	24
3.4	Class Diagrams	26
3.5	Sequence Diagrams	28
3.6	Activity Diagrams	38
3.7	Entity-Relationship (ER) Diagrams	45
3.8	Non-Technical Aspects	47
3.8.1	Financial Viability	47
3.8.2	Stakeholders	50
3.8.3	Scope	51
3.8.4	Risks	51
3.8.5	Schedule and Milestones	52
3.8.6	Ethical and Social Considerations	53
3.8.7	Environmental and Sustainability Considerations	54
3.8.8	Relevant Standards	54
3.9	Conclusion	56
CHAPTER 4	IMPLEMENTATION/SIMULATION AND TESTING	58
4.1	Introduction	58
4.2	Implementation Tools	58
4.3	Implementation Summary	60
4.4	Test Cases and Acceptance Criteria	76

4.5	Conclusion	79
CHAPTER 5 CONCLUSION AND FUTURE WORK		79
5.1	Conclusion	79
5.2	Future Work	81
APPENDIX A: IMPLEMENTATION DETAILS		83
APPENDIX B: USER MANUAL		85
APPENDIX C: DEPLOYMENT AND CONFIGURATION MANUAL		87
REFERENCES		90

LIST OF FIGURES

Figure 1:Shipyard full of cargo to be delivered [5]	2
Figure 2: Organized warehouse [5].....	3
Figure 3: web application preview [5]	5
Figure 4: confirmed delivery [5].....	6
Figure 5: Aramex's Website [9]	9
Figure 6: Wakilni Website [8]	10
Figure 7: GoShare [10]	12
Figure 8: client use case diagram.....	19
Figure 9: guest client use case	20
Figure 10: CEO use case diagram.....	21
Figure 11: Branch Manager use case diagram	22
Figure 12: worker use case diagram	23
Figure 13: IT use case diagram	24
Figure 14: System Architecture	26
Figure 15: class diagram	27
Figure 16: all users login sequence diagram.....	29
Figure 17: client sequence diagram	30
Figure 18: client(guest mode)	32
Figure 19: CEO sequence diagram	33
Figure 20: Branch Manager sequence diagram	34
Figure 21: worker sequence diagram.....	35
Figure 22: IT sequence diagram	36
Figure 23: login activity diagram.....	39
Figure 24: Client activity diagram	40

Figure 25: CEO activity diagram	41
Figure 26: Branch Manager activity diagram	42
Figure 27: Worker activity diagram.....	43
Figure 28: IT activity diagram	44
Figure 29: ER diagram.....	46
Figure 30: Scheduling Tasks and Milestones	52
Figure 31:Index page	60
Figure 32:services.html.....	61
Figure 33: about.html.....	61
Figure 34: place order(1)	62
Figure 35:Place order(2)	62
Figure 36:track order(outside)	63
Figure 37:Login page	63
Figure 38:Sign up page	64
Figure 39:client page.....	64
Figure 40:client's orders history.....	65
Figure 41:client track order(1)	66
Figure 42:client track order (2)	66
Figure 43: change password(common)	67
Figure 44:messages(common)	67
Figure 45: CEO landing page	68
Figure 46: view workers(CEO and IT)	69
Figure 47: CEO stats	70
Figure 48: Hire(CEO and IT).....	70
Figure 49:view branches(CEO and IT).....	71

Figure 50: Branch Orders(branch manager and worker)	71
Figure 51: Orders history manager	72
Figure 52:Mobile splash screen	73
Figure 53: Mobile Login Page	73
Figure 54: Mobile view branch orders	74
Figure 55:Mobile details and drop off	75

LIST OF TABLES

Table 2-1: Comparison Table Based on Graphical Interfaces	12
Table 2-2: Comparison Table Based on Content and Functionality.....	13
Table 2-3: Comparison Table Based on Features	14
Table 3-1: Cost Benefit Analysis.....	48
Table 4-1:Test cases.....	76

LIST OF SYMBOLS

AJAX: Asynchronous JavaScript And XML

AWS: Amazon Web Services

CMS: Content Management System

CEO: Chief Executive Officer

CSS: Cascading Style Sheets

ER: Entity-Relationship

Ex: example

GUI: Graphical User Interface

HTML : Hypertext Markup Language

ID : identification

ISO: International Standardization Organization

IT: Information technology

JS: JavaScript

LIU: Lebanese International University

PHP: Hypertext Preprocessor

SQL/MySQL: Structured Query Language

UI: User Interface

UML: Unified Modeling Language

UX: User Experience

WiFi: Wireless Fidelity

XML: Extensible Markup Language

CHAPTER 1

INTRODUCTION

1.1 Background

Our modern world is full of challenges. That's why we, as humans, try our best to invent new technologies in order to make our lives less tedious and easier to go by.

Logistics companies faced various challenges before the advent of the internet. One of the major issues was the lack of real-time visibility into the status and location of shipments. This made it difficult to track shipments, causing delays and higher costs. Pre-internet, logistics companies encountered numerous obstacles, including a lack of real-time visibility into shipment status and location, impeding the ability to track shipments and resulting in delays and escalated costs [1]. Moreover, reliance on manual paper-based processes led to a host of issues, including time inefficiencies and error-prone operations that resulted in increased costs. Additionally, limited access to information and communication channels posed further difficulties in coordinating with suppliers, customers, and carriers. Collectively, these issues resulted in reduced agility and responsiveness, hampering logistics companies' abilities to cater effectively to customer demands. However, with the advent of the internet, logistics companies have been able to surmount these challenges and enhance their operations significantly. The internet's vast capabilities, including real-time tracking and communication, have facilitated increased access to information and streamlined processes, resulting in improved efficiency, reduced costs, and heightened customer satisfaction. [2]

We saw the problems that shipment companies face, such as loss of cargo either through piracy or corruption, absence of trust between customer and company due to lack of communication, and insufficient data/intel on the companies' exchanges. [3]

Hence our attempt at developing our app: a cargo/delivery application that can address several challenges faced by logistics providers and customers, such as lack of visibility, inefficient processes, limited communication, security concerns, competition, and increasing demand. By providing real-time tracking, automating processes, improving communication and transparency, enhancing security, and offering a competitive advantage, a cargo/delivery application can improve customer satisfaction, reduce costs, optimize operations, and provide valuable data insights. [4]

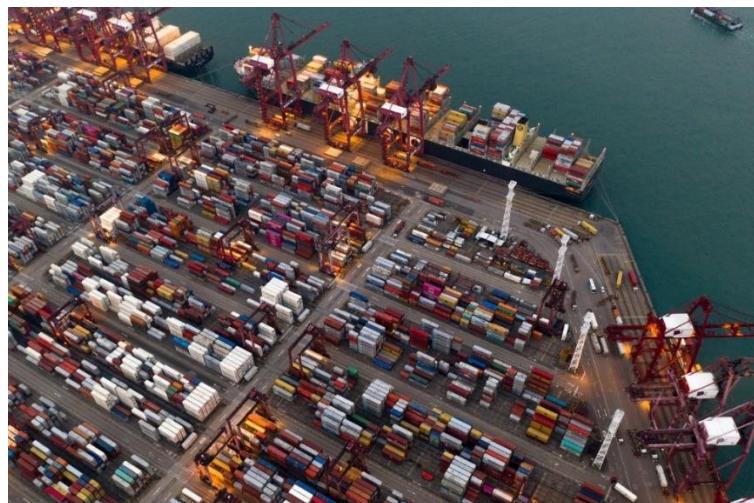


Figure 1:Shipyard full of cargo to be delivered [5]

1.2 Problem Statement

Since its foundation, the logistics and transportation industry encounter various difficulties such as inadequate visibility, ineffective procedures, restricted communication, security threats, intense competition, and growing demand for prompt and dependable shipping services. These challenges can cause delivery delays, higher costs, and lower customer satisfaction, which may have negative effects on the profitability and expansion of logistics providers. To solve these challenges, a cargo/delivery application is necessary to deliver instant

tracking, automate processes, enhance communication and transparency, boost security, and provide a competitive edge in the market [6].

- Optimize operations: The application should help logistics providers optimize their operations by reducing manual processes, increasing efficiency, and improving resource allocation.
- Reduce costs: By automating processes and improving efficiency, the application should help logistics providers reduce costs associated with labor, fuel, and other operational expenses.
- Increase revenue: By improving customer satisfaction and providing a competitive advantage, the application should help logistics providers attract new customers and increase revenue.



Figure 2: Organized warehouse [5]

1.3 General overview of the project

Our project aims to improve the efficiency, transparency, security, and customer satisfaction of logistics and transportation operations. This can be achieved through real-time tracking, process automation, improved communication, enhanced security measures, and a unique value proposition that sets the application apart from competitors. The cargo/delivery application aims to provide logistics providers and customers with a comprehensive, user-friendly platform for managing shipments and deliveries, using modern technologies and tools. The application will feature real-time tracking, automated processes, communication tools, enhanced security measures, and data insights to optimize operations and improve customer satisfaction. The real-time tracking feature will leverage GPS and other location-tracking technologies to provide customers with real-time updates on the status and location of their shipments. The process automation feature will automate manual processes involved in managing shipments, such as generating labels, managing inventory, scheduling deliveries, and handling customs documentation. The communication tools feature will provide notifications, alerts, messaging capabilities, and a customer service portal for addressing any issues that may arise. The enhanced security measures feature will include secure login credentials, real-time tracking, and integrated security systems that monitor shipments throughout the delivery process [7]. The data insights feature will provide valuable data on

shipment performance, customer behavior, and other key metrics, which can be used to improve operations and make better business decisions.



Figure 3: web application preview [5]

1.4 Project Outline

The project aims to develop a cargo/delivery application that will provide real-time tracking, automate processes, improve communication and transparency, enhance security, and offer a competitive advantage to logistics providers. The application will help logistics providers optimize their operations, reduce costs, increase revenue, and improve customer satisfaction. Please follow us throughout the steps of realizing the project, from research to development. As first we are going to conduct a thorough analysis of the requirements and needs of logistics providers and customers to determine the key features and functionalities of

the application. We will do this by surveying of existing methods and similar systems in CHAPTER 2.



Figure 4: confirmed delivery [5]

CHAPTER 2

SURVEY OF EXISTING METHODS AND SIMILAR SYSTEMS

2.1 Introduction

Also called "Requirements gathering and analysis", this phase is a critical step in the development of the cargo/delivery application project. In this phase, we will research closely already existing applications (like Wakilni [8], Aramex [9] and GoShare [10]) to understand the project's requirements and needs, identify pain points, and define the scope and goals of said project. This phase will involve conducting research, holding meetings, and gathering feedback from logistics providers and customers to develop a comprehensive set of requirements that will guide the development of the application. The project team will use this information to create use cases, user stories, and other documentation that will help ensure that the final product meets the needs of all stakeholders and delivers the desired outcomes. The requirements gathering and analysis phase will be followed by the design and development phases, where the project team will begin building the application based on the requirements identified in this phase.

2.2 System 1: Aramex

Aramex [9] is a global logistics and transportation company based in Dubai, United Arab Emirates. It provides a range of services including express courier delivery, freight forwarding, logistics and supply chain management. Aramex operates in more than 65 countries worldwide and is known for its innovative solutions and use of technology in the logistics industry. Their website offers:

- e-commerce solutions that assist start-ups and established businesses in supply chain management, technical support, and more [9]
- Logistics Services
- Bundling, Packaging and co-packing
- Aramex Application Programming Interface (APIs)
- Import address book
- Track shipments
- Calculate rates
- Receive shipment updates
- ClickToShip: a separate desktop application without constant link to the internet
- Aramex app: a mobile application that enables Aramex customers worldwide to perform a range of tasks, including tracking deliveries, monitoring shipment progress, managing their accounts, calculating the costs and transit times of upcoming shipments, scheduling pickups and deliveries at preferred locations, dates, and times, making secure payments, and receiving a clear breakdown of delivery dates and custom charges... [9]

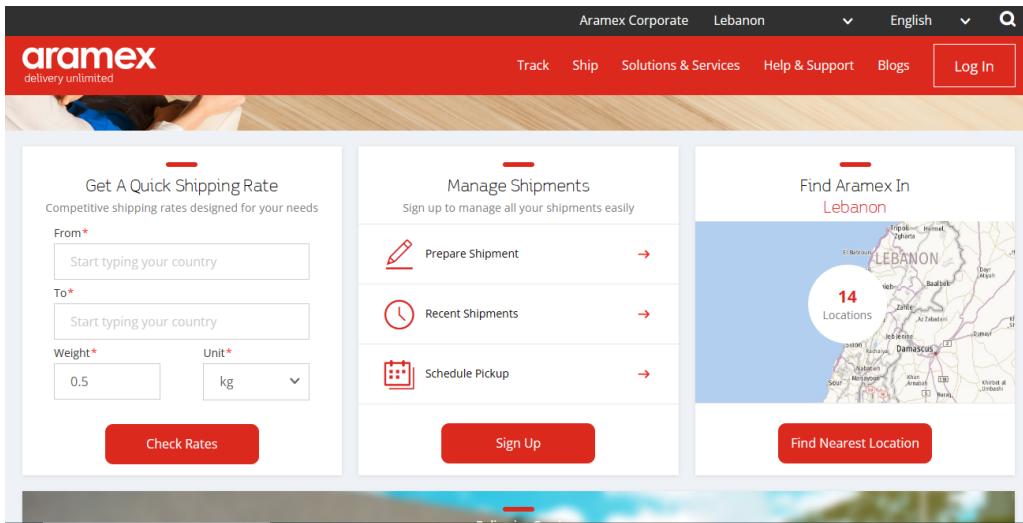


Figure 5: Aramex's Website [9]

2.3 System 2: Wakilni

Wakilni [8] is a delivery company that offers a range of innovative solutions to help individuals and businesses in the Middle East manage their daily tasks more efficiently. Founded in 2015, the company connects customers with trusted service providers who can take care of tasks like grocery delivery, laundry, home cleaning, and more. Wakilni's personalized approach and use of machine learning algorithms ensure that each customer receives customized service tailored to their unique preferences and behavior. Here's what their website offers:

- Donate to children feature.
- Promo code to take a discount.
- Shipment cost around 4\$-8\$ across Lebanon.
- Extra payment option for 24-72 hours' delivery.
- Join our team feature if they need some workers. Ex: driver.
- Live chat feature: they will ask for your name and the email

- A marketplace page that shows all the businesses in Lebanon that are registered in Wakelin and they provide the customer with categories to find what he is searching for; and provide a link for each one of them.
- Extra mile fund donations: As of October 2020, Wakilni will donate 750 liras from each commissioned delivery to its Extra Mile Fund. This money will go towards helping members of the Wakilni community reboot their businesses by accessing needed services, whether in terms of online exposure, marketing, legal counseling, etc. and Wakilni will be sponsoring this access to services. [8]

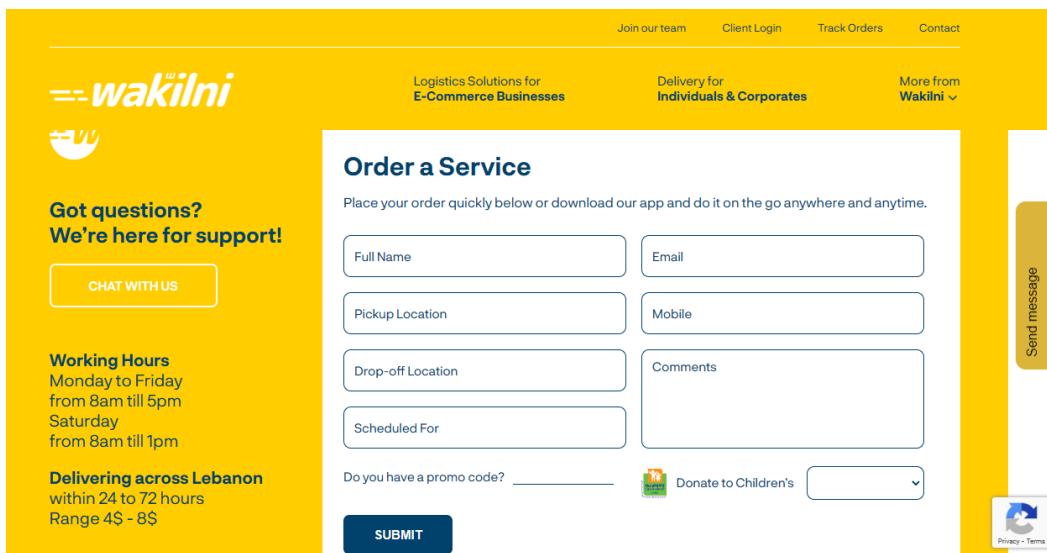


Figure 6: Wakilni Website [8]

2.4 System 3: GoShare

GoShare [10] is a rapidly growing logistics and transportation company based in San Diego, California. Founded in 2014, the company has quickly become a leading provider of on-demand delivery and moving services in the United States. We personally chose GoShare as it has received numerous accolades for its innovative approach to logistics and transportation. The company was named one of the "Most Promising

Startups" at the 2017 Forbes Under 30 Summit, and it has been featured in publications like the Wall Street Journal, TechCrunch, and CNBC.

Here are some key points about GoShare:

- GoShare offers a variety of services, including local delivery, hauling, and moving. Customers can use the company's app or website to book a driver and a truck for a range of jobs, from small-item delivery to full-scale residential moves.
- One of GoShare's key selling points is its on-demand availability. The company's large network of independent contractors allows it to offer fast service, with many jobs completed within an hour of booking.
- GoShare drivers are rigorously vetted and insured, so customers can trust that their items are in safe hands. Drivers are also equipped with a range of tools and materials to help them complete jobs efficiently and effectively.
- GoShare is committed to sustainability and reducing its carbon footprint. The company's fleet of trucks includes electric and hybrid vehicles, and it encourages its drivers to adopt eco-friendly driving habits.
- In addition to its core services, GoShare also offers business solutions for companies looking to streamline their logistics operations. The company's platform allows businesses to manage deliveries, track shipments, and access real-time data on their delivery performance.

Overall, GoShare is a dynamic and rapidly expanding company that is shaking up the traditional logistics industry. With its on-demand availability, focus on sustainability, and commitment to innovation, GoShare is poised for continued success in the years to come.

The screenshot shows the GoShare booking process at the 'Where' step. At the top, there's a blue header with the 'GoShare' logo and a 'SIGN IN' button. Below the header, a navigation bar shows five steps: 1 Where, 2 What, 3 Vehicle type, 4 Equipment optional, and 5 Review. The main form area has fields for 'Email', '1st Location' (with a placeholder '1'), '2nd Location' (with a placeholder '2'), and an 'ADD LOCATION' button. There are also fields for 'Pickup start date' and 'Pickup start time'. At the bottom of the form is a 'CONTINUE' button. Below the form, a footer contains links for 'TERMS OF SERVICE', 'PRIVACY POLICY', 'REVIEWS', 'TRUST', 'HELP', and 'CONTACT US', along with a copyright notice: '© 2023 GoShare Inc. All Rights Reserved'.

Figure 7: GoShare [10]

2.5 Methods/Systems Comparison

The purpose of this paragraph is to draw attention to the strengths and weaknesses of each of the methods mentioned earlier in relation to specific standards.

Table 2-1: Comparison Table Based on Graphical Interfaces

Criterion 1	Aramex	Wakilni	GoShare
Graphical Interface			
Good user interface	✓	✓	✓
Easy and effective navigation	✗	✓	✗
Simple and professional Design	✓	✓	✓
Responsive	✗	✓	✓

A well-designed graphical user interface (GUI) is essential for the success of an application. A good user interface (UI), easy navigation, professional design, and responsiveness are crucial components. In the case of the three systems we reviewed - Aramex, GoShare, and Wakilni - we found that each had their own strengths and weaknesses in terms of GUI design. When it comes to UI, all three systems were generally easy to use, with clear icons and labels for each

feature; a good and attractive GUI as it is eye catching and well laid out through the webpage. However, we found that Aramex and GoShare's navigation was slightly more confusing compared to Wakilni, which had a more intuitive layout that made it easier to find what we needed like the order part that was really accessible and available from the get-go. In terms of design, all three systems had a simple and professional look, with no obvious errors or unprofessional features. However, we noticed that Wakilni had a more modern and aesthetically pleasing design compared to Aramex and GoShare. When it came to responsiveness, we found that Aramex was slightly slower and less dynamic compared to GoShare and Wakilni. This could potentially impact user experience and satisfaction. A well-designed GUI with a good UI, easy navigation, professional design, and responsiveness improves the usability and appeal of an application.

Table 2-2: Comparison Table Based on Content and Functionality

Criterion 2	Aramex	Wakilni	GoShare
Content and Functionality			
Quality content structure	✓	✓	✓
Usability	✓	✓	✗
Dynamic content	✓	✓	✓
Content management system	✓	✓	✓

Quality content structure, usability, dynamic content, and content management systems are all critical components of a successful application. In all three systems, quality content structure ensures that information is presented in a clear, logical manner, with content divided into parts, categories, and subcategories, making it easy for users to find what they need. Each system clearly lays out content throughout the page to optimize user experience. Usability refers to the ease with which users can navigate an application, complete tasks, and achieve their goals.

While Aramex and Wakilni have strong usability, with an intuitive homepage that allows for easy ordering, GoShare's usability is lacking: their website requires users to navigate complex menus to place an order, making the user experience less seamless than its competitors. Dynamic content, such as personalized recommendations or real-time updates, is available in all three systems. The objects are kept up-to-date and relevant, though none of the systems have implemented truly personalized content. Finally, a content management system (CMS) [11] allows for efficient and effective management of an application's content, ensuring that it remains up-to-date and relevant. While it is not immediately clear what CMS each of the three systems is using, they appear to be using popular platforms such as WordPress, Drupal, or Joomla, or custom-built CMS systems. Together, these features help create an application that is intuitive, engaging, and valuable to its users.

Table 2-3: Comparison Table Based on Features

Criterion 3	Aramex	Wakilni	GoShare
Features			
Security measures	✓	✓	✓
Third party integration	✓	✓	✓
Accessible content and location	✗	✗	✗
Registration form	✓	✗	✓

All three systems we reviewed - Aramex, GoShare, and Wakilni - prioritize user data protection by implementing HTTPS and captchas throughout their webpages to prevent unauthorized access and display the padlock icon next to the URL for added security. In addition, Aramex, GoShare, and Wakilni offer visible third-party integrations with popular platforms such as Google, Facebook, and Instagram, making it easy for users to access other services that can enhance their experience like a high-contrast mode for example. However, none of them have

made explicit efforts to ensure that their content is accessible to all users, regardless of their physical abilities. Aramex and GoShare offer registration forms and Login pages, while Wakilni only offers a login page, which could be a potential drawback for new users who want to create an account. By taking these factors into consideration, developers can create a secure, functional, and personalized application that provides an optimal user experience.

2.6 Conclusion and Motivation

Through the study of other systems, we have identified the essential components that a successful application should possess. These components include a well-designed graphical user interface (GUI) [12], quality content structure, usability, dynamic content, content management systems, security measures, third-party integration, accessible content and location, and registration forms. A well-designed GUI with a good UI, easy navigation, professional design, and responsiveness enhances the user experience and increases usage and satisfaction. Quality content structure ensures that information is presented logically, while usability and dynamic content enhance user engagement. A content management system (CMS) [11] allows for efficient management of the application's content. Security measures, third-party integration, accessible content and location, and registration forms are also important considerations. By taking these factors into account, we can create an application that is secure, functional, accessible, and personalized, leading to increased user satisfaction and engagement.

CHAPTER 3

SYSTEM DESIGN

3.1 Introduction

The cargo/delivery application project involves developing a comprehensive platform for managing shipments and deliveries, using modern technologies and tools. To ensure that the final product meets the needs of all stakeholders and delivers the desired outcomes, several key phases will be undertaken, including requirements and specification analysis, functional requirements, use case diagrams, system architecture, class diagrams, sequence diagrams, activity diagrams, and financial viability [1]. These phases that normally involve working closely with stakeholders, gathering feedback, and developing a comprehensive set of requirements that will guide the development of the application, have been simulated via our own research. This will be followed by the design and development phases, where the project team will begin building the application based on the requirements identified in the earlier phases. Throughout the project, the team will conduct rigorous testing and quality assurance to ensure that the application is free of bugs, errors, and other issues that could affect the user experience or performance. The financial viability analysis will ensure that the project is feasible, economically viable, and sustainable in the long run...

3.2 Requirements and Specification Analysis

By allowing clients to request specific item deliveries, offering expedited delivery options, and providing real-time tracking, clients are more likely to trust your services and become repeat customers. Additionally, the four distinct user roles and companion mobile app for workers can help streamline the delivery process, improve communication, and increase

worker efficiency. The manager's ability to monitor and analyze delivery metrics can help identify areas for improvement and optimize the delivery process. Finally, features such as Captcha verification and chat communication can increase security and customer satisfaction, respectively.

3.2.1 Functional Requirements

Here the app's features:

- The platform allows clients (companies, stores, or individuals) to request delivery for specific items.
- Senders are required to provide shipment details, including package dimensions, weight, and whether the contents are fragile or urgent when placing an order. As well as contact info for sender and receiver.
- If the package is fragile, senders will incur an additional fee for specialized wrapping services.
- If the package is urgent, senders will incur an additional fee for faster delivery.
- A message feature is available for users to communicate with each other.
- To prevent spam accounts and safeguard against DDOS attacks, a Captcha verification process is implemented.
- The app uses md5 encryption for further protection of passwords
- Payment options include the sender or receiver paying for the delivery.
- The platform accommodates five distinct user roles; including a CEO, IT, branch manager, warehouse workers, and clients. Each role is granted access to specific templates and functionalities within the web platform/mobile application.
- Upon completion of each task, workers check off the corresponding item on the companion mobile app to signal readiness for the following step.

- Urgent tasks are communicated to workers via notifications within the app.
- Push notifications are sent to workers on a daily basis.
- The CEO is able to monitor and manage the hiring and firing of workers, track profits, and analyze delivery metrics, including successful and failed deliveries, punctuality, and worker efficiency (employee of the month).
- Clients are granted access to real-time delivery tracking, including the current location of the shipment, time it arrived there and its handler.
They can mark said delivery as failed.
- Failed deliveries can be checked by the manager to resend them to the customer.

3.2.2 Use Case Diagrams

A technique that simplifies information about a system and its users is called a use case diagram. It presents a visual representation of how different components of the system interact with one another. Use case diagrams describe the sequence of events in which the system performs its actions, but they do not provide details on how these events are executed.

A use case is an approach to identify, outline, and organize requirements for a system, such as a website for product sales and services. Use case diagrams are commonly created using UML, a standard language for representing real-world structures and systems. Compared to other types of diagrams, use case diagrams offer several advantages.

In the context of the website, the customer, CEO, branch manager and IT are the five actors depicted in the use case diagrams. The worker also has access to the companion mobile application. The following use cases illustrate the tasks and activities that each user can perform in the project.

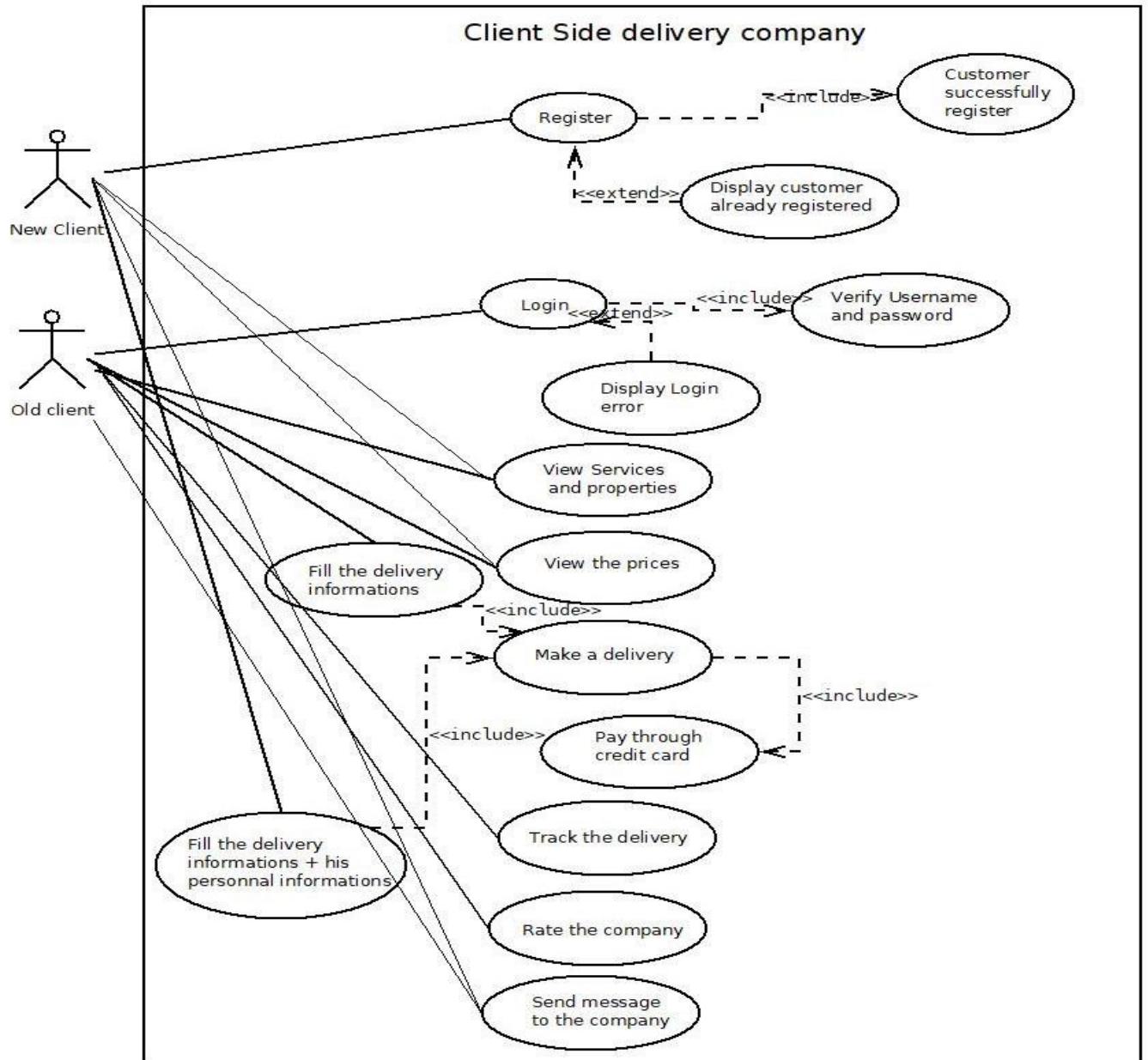


Figure 8: client use case diagram

The users are able to register and sign in to their accounts, which allows them to save their information for future use. Upon logging in, users can view the services offered by the company, along with their corresponding prices. The user interface includes three main tabs: Track Delivery, Order Delivery, and History. Order Delivery allows users to enter the necessary information to place an order, while Track Delivery enables them to view the current

location of a delivery by entering the order ID. The History tab provides users with access to their previous orders, including dates and costs. Additionally, users can provide feedback by rating the company or sending a message to the support team. As well as changing his own password and sending/reading messages.

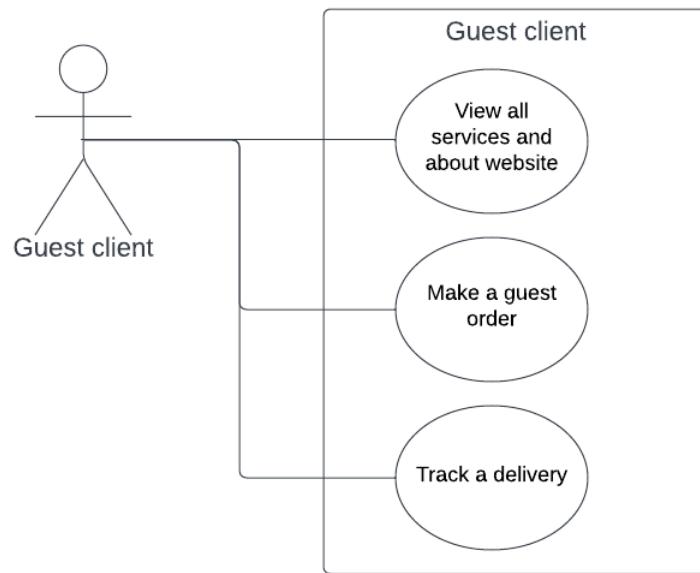


Figure 9: guest client use case

As a guest, not logged in, the client is able to, additionally to viewing the website's info, Track Delivery and Order Delivery.

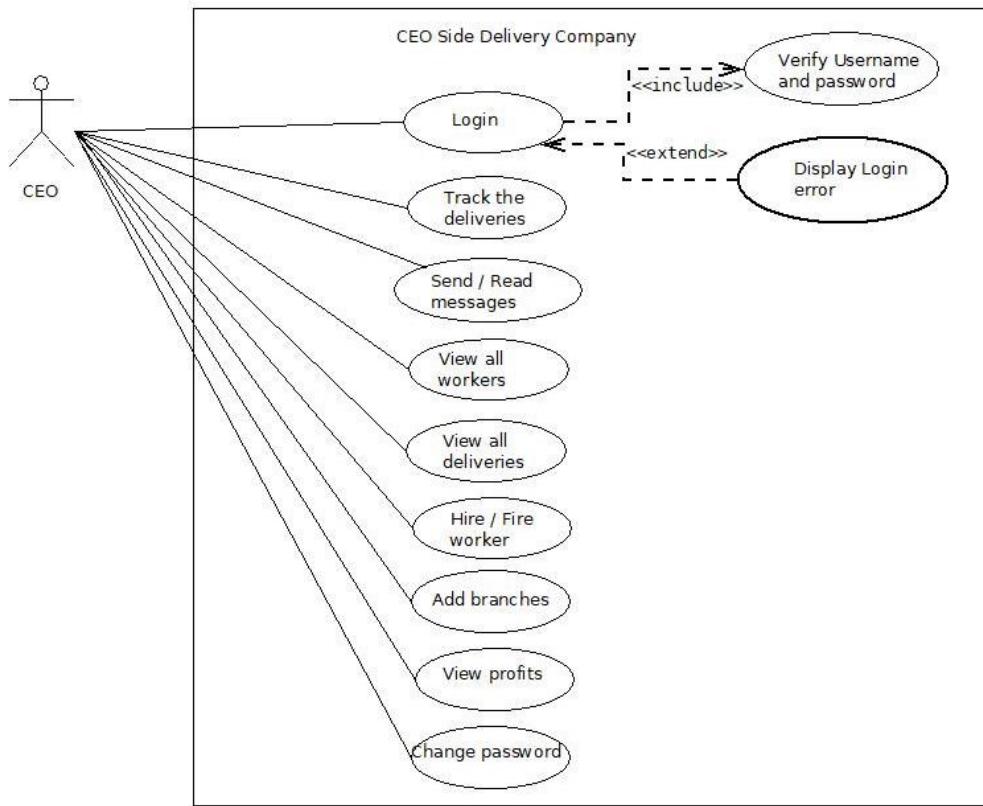


Figure 10: CEO use case diagram

As the executive leader of the company, the CEO holds authority over a broad range of features within the application. Upon logging in to the system, the CEO is granted access to various tools that enable him to track and manage deliveries, oversee branch operations, monitor worker activity, and hire/fire them, and view financial reports related to the company's profits. In essence, the CEO is afforded the ability to exercise a high degree of control over the various

aspects of the business via the application's user interface. As well as changing his own password and sending/reading messages.

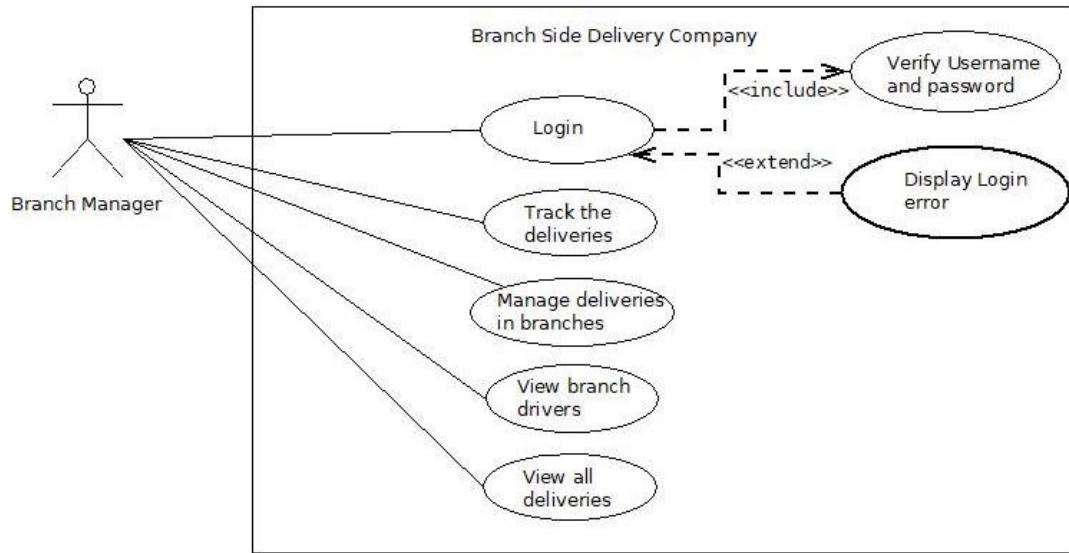


Figure 11: Branch Manager use case diagram

The branch manager is responsible for overseeing the delivery operations within his designated branch. Through the application's user interface, the branch manager is granted access to features that enable him to manage and view deliveries within his branch, as well as monitor the activity of workers assigned to those deliveries. By tracking the progress of deliveries and reviewing worker performance, the branch manager is equipped to detect and address any

issues that may arise and optimize the overall efficiency of delivery operations within his branch. As well as changing his own password and sending/reading messages.

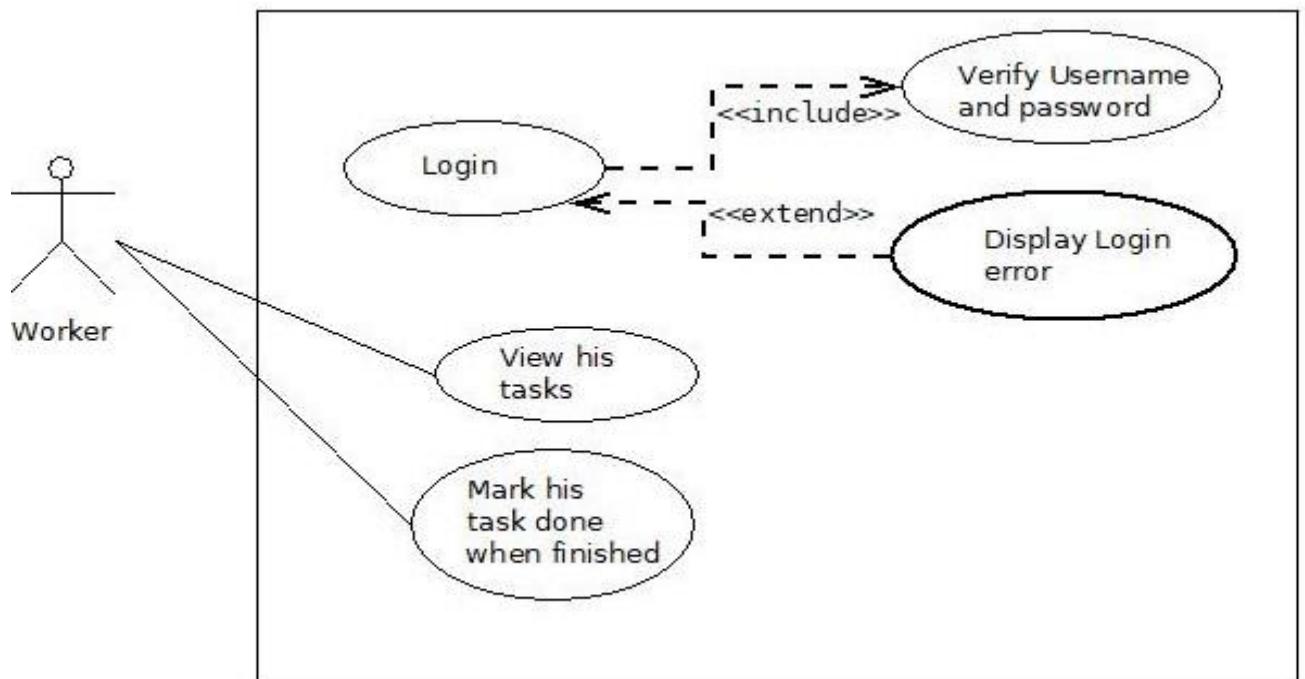


Figure 12: worker use case diagram

The worker's role within the application is crucial to the successful management and execution of delivery operations. While their tasks may seem minimal in comparison to other users, such as the branch manager or CEO, their responsibilities are nonetheless essential to the overall success of the business. In particular, the worker's primary function is to mark the current location of deliveries within the system, enabling accurate tracking and timely updates for clients. Additionally, workers may be required to perform other tasks related to delivery management, such as verifying order details and ensuring that packages are properly labeled and accounted for. As well as changing his own password and sending/reading messages.

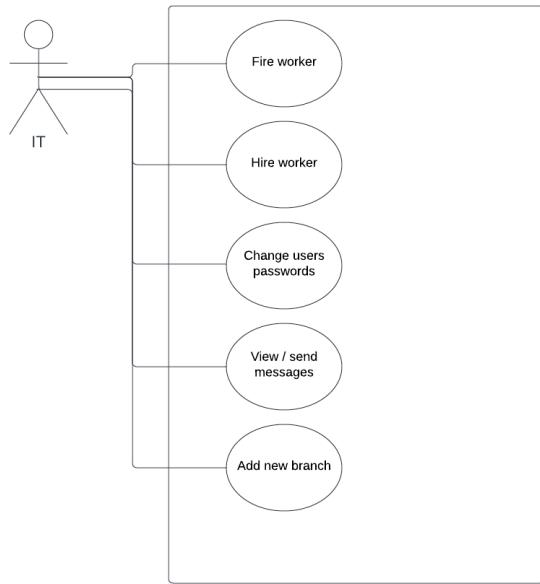


Figure 13: IT use case diagram

The IT department is primarily in charge of operations application's technical features, such as adding and removing employees from the system, changing their password, adding new branches, making sure the application is up-to-date and operating properly, and debugging technical issues. In addition to maintaining the application to meet the changing needs of the business, their contributions are essential in providing the required infrastructure and resources to assist workers in their jobs. Without their efforts, delivery operations could not run as smoothly and effectively as they do. As well as changing his own password and sending/reading messages.

3.3 System Architecture

System architecture is a critical aspect of software system or application design, and it involves defining the structure of the system, including its components, interactions, and relationships with other systems. The architecture serves as a foundation for system

development and maintenance, and it enables different teams to work together seamlessly. A well-designed system architecture can lead to a more efficient and effective system that provides a better user experience and allows for easier maintenance and future development.

Our project will adopt a multi-tier architecture, which consists of three main tiers: presentation, application, and database. The presentation tier will handle the user interface and interaction with the application, while the application tier will contain the business and processing logic. The database tier will be responsible for storing and managing the data used by the application. The system will utilize PHP, MySQL, HTML, CSS, JavaScript, and Bootstrap technologies, and RESTful API architecture to facilitate communication between the different tiers of the system. The mobile application will also use Java and XML. We will draw a figure to show the complete system architecture, which will provide a visual representation of the system and its components, and help to identify potential issues and bottlenecks, as well as to optimize the system for performance and scalability. Overall, our goal is to create a reliable, scalable, secure, and maintainable system that meets the needs of our users.

Below is a high-level diagram of the complete system architecture:

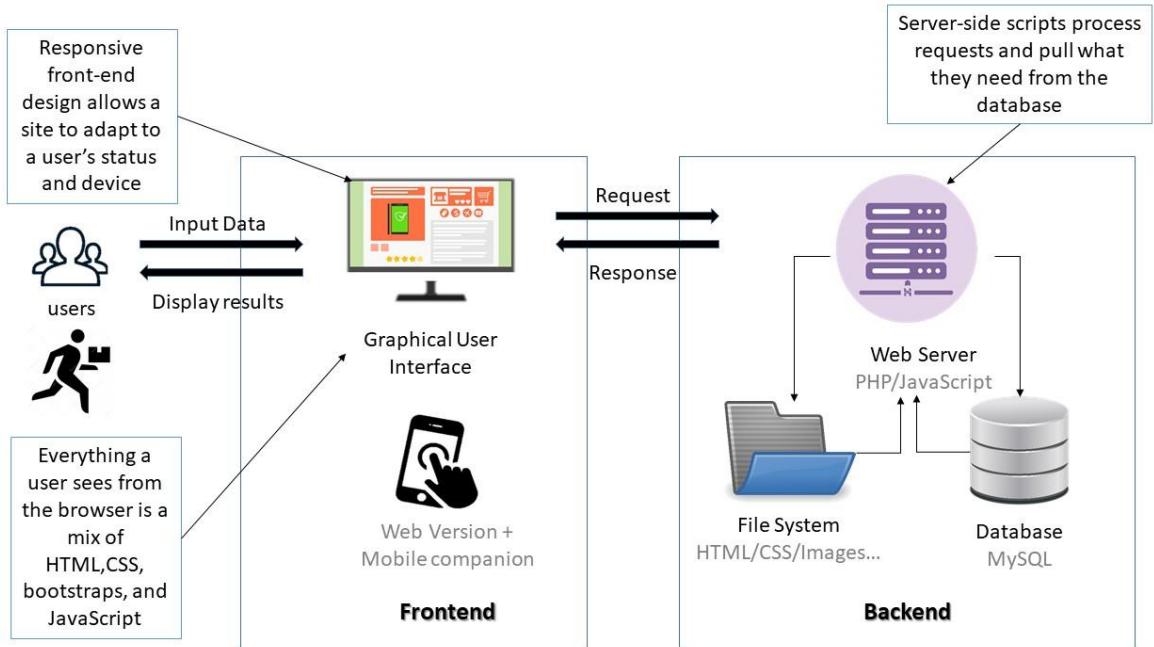


Figure 14: System Architecture

3.4 Class Diagrams

The class diagram is a static structural diagram in software engineering that depicts the classes, interfaces, and their relationships in a system. It provides a graphical representation of the system's object-oriented design, and helps to visualize the static view of the system. The class diagram is a part of the Unified Modeling Language (UML), which is widely used for modeling software systems. The class diagram shows the relationships among classes, such as inheritance, composition, aggregation, and association. It also shows the properties and methods of each class, which define the behavior and attributes of the objects of that class.

In our application, the class diagram might include classes such as users, orders, deliveries, payment, clients, and workers. Each class would have properties and methods that define its behavior and attributes, such as User class having properties like name, email, and password and methods like login(), signup(), and logout().

The class diagram would also show the relationships among these classes, such as the users class being composed of workers and clients' classes, and the users class having an association with the privilege class.

This diagram is an essential component of the system design process, as it helps to visualize and communicate the object-oriented design of the system to stakeholders and development teams:

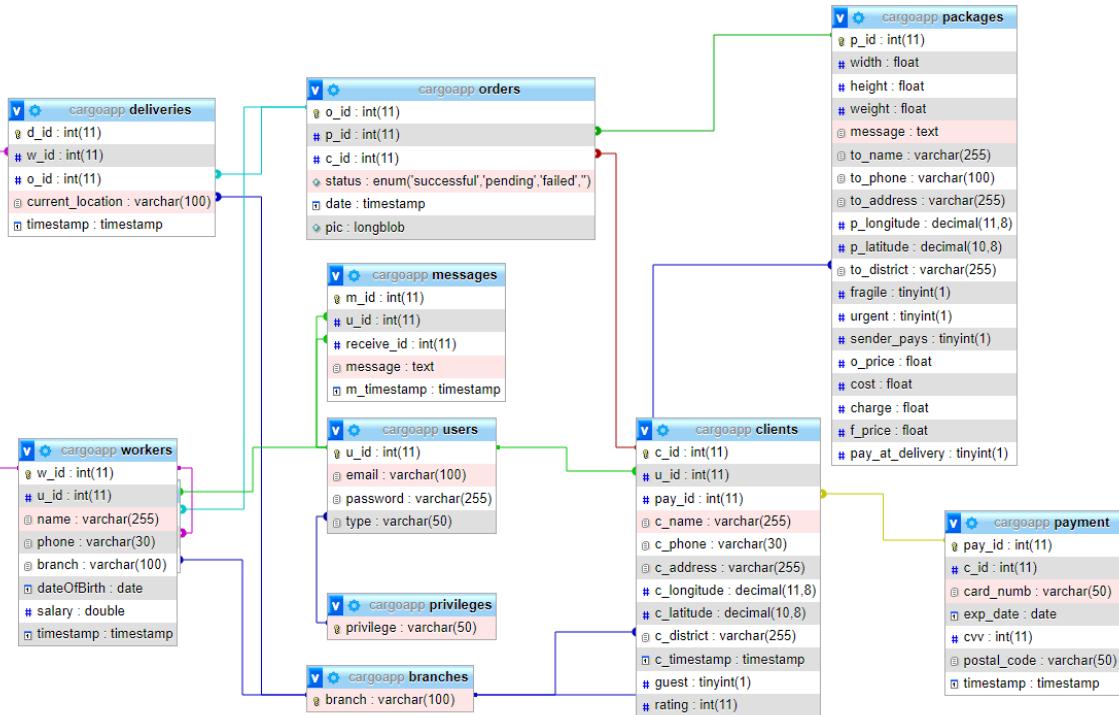


Figure 15: class diagram

3.5 Sequence Diagrams

A sequence diagram is a type of interaction diagram used to illustrate the interactions and relationships between different objects or components within an application or system. The diagram is time-focused and visually represents the order of interactions by using a vertical axis to depict time and horizontal arrows to indicate the flow of messages between objects or components. Sequence diagrams are useful for modeling complex processes and ensuring that different components within a system are communicating and interacting correctly. They can also help identify potential errors or areas of inefficiency in a system's design. By showing the interactions between objects or components, sequence diagrams can help developers better understand the overall flow of a system and make necessary adjustments to improve performance and functionality. To create a sequence diagram, we must first identify the objects or components within the system and the messages they send to each other. Each object is represented as a lifeline, and the messages are depicted as arrows connecting the lifelines. Additional information, such as the order and duration of messages, can be added to the diagram to provide more detailed insight into the system's operation. Overall, sequence diagrams are a valuable tool for designing, testing, and maintaining complex applications and systems.

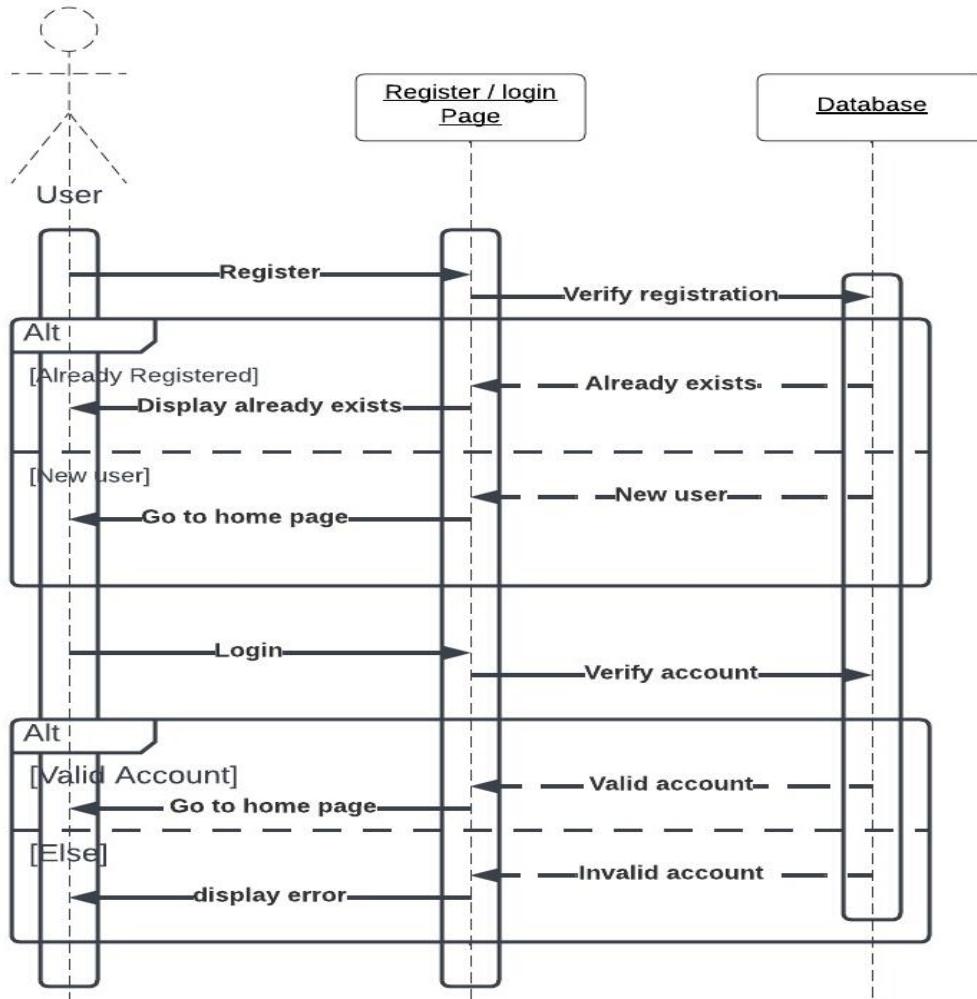


Figure 16: all users login sequence diagram

To access their accounts, all users have the option to navigate to the login page from the home screen. Once on the login page, the user is prompted to either sign up or log in to their existing account. For new users, signing up requires providing their personal information, such as their name, address, and contact information, which will be stored securely in the system's database. After successful registration, the user is directed to the login page where they enter their email address and password. Upon submission, the system verifies that the email and password are both correct and correspond to an existing user account in the database. If successful, the user

is redirected to their account dashboard. If unsuccessful, the user is prompted to try again or reset their password if needed.

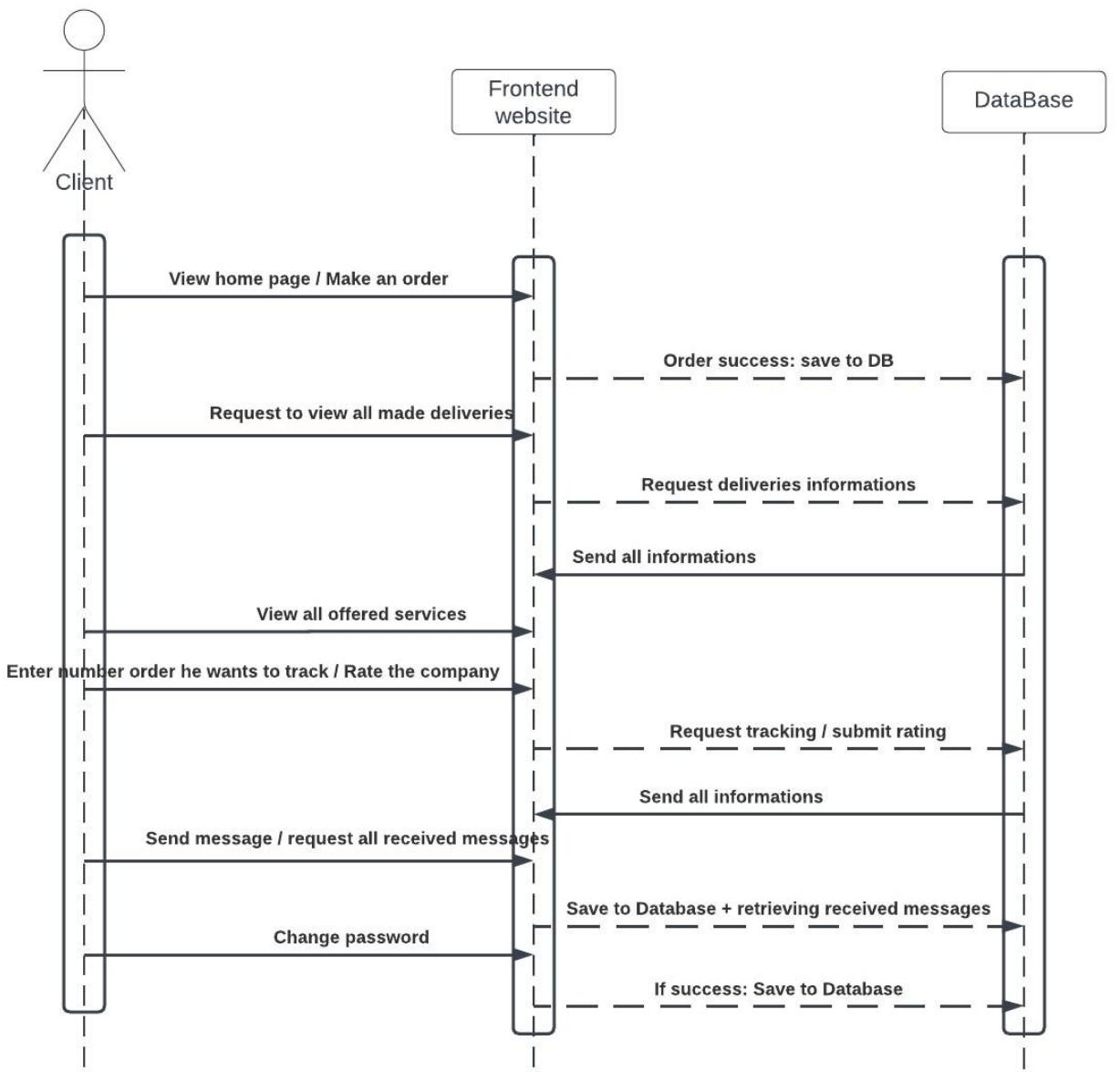


Figure 17: client sequence diagram

Within the customer dashboard, the client can explore various features and information. They have access to company details, services offered, and pricing information. To place an order, the client fills out a comprehensive form that includes shipper's details, receiver's details, and

shipment's details. Additional options like selecting fragile item or same-day delivery are available. To ensure security, a CAPTCHA verification is implemented to prevent spam and attacks. After submitting the order, the client receives a unique order number for tracking purposes.

To track an order, the client enters the order number into the system. The application retrieves the order details and provides a comprehensive history of locations and handlers involved in the delivery process. This real-time tracking feature enhances transparency and enables clients to stay informed about their package's progress.

User authentication and dashboard functionality are crucial for a seamless experience. The login and signup pages incorporate secure authentication measures to safeguard client information. Once successfully authenticated, clients are redirected to their personalized dashboard. Within the dashboard, they can view and manage their orders, including access to order history, options to add, edit, or cancel shipments, and real-time tracking capabilities. Additionally, the system provides an opportunity for clients to rate the company's services and offer valuable feedback to improve customer satisfaction.

Furthermore, clients have the flexibility to change their password through a dedicated password change page. This allows them to maintain account security and update their login credentials when needed. Additionally, the messages page enables clients to read received messages from other users and engage in communication. They can also send messages to other users using their email addresses.

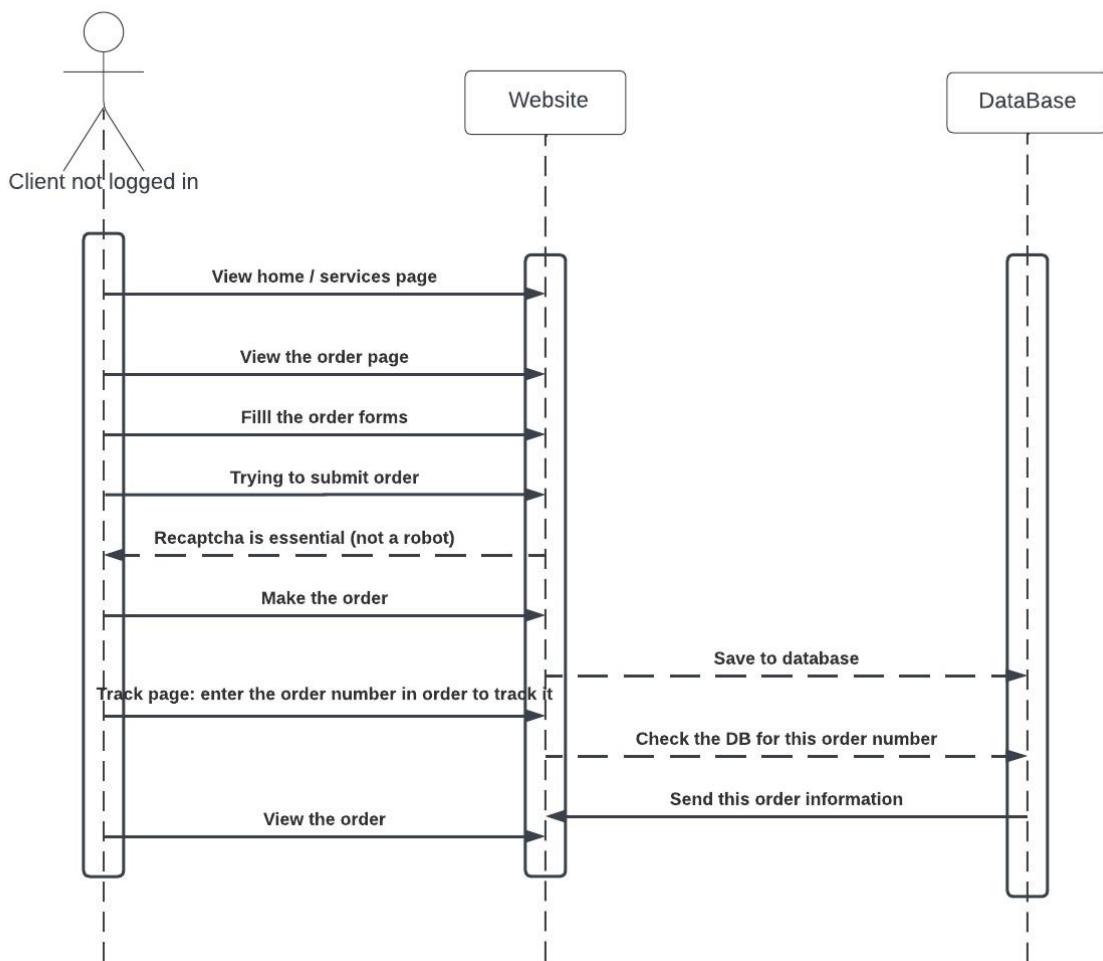


Figure 18: client(guest mode)

Alternatively, customers can track an order and place a new order without logging in.

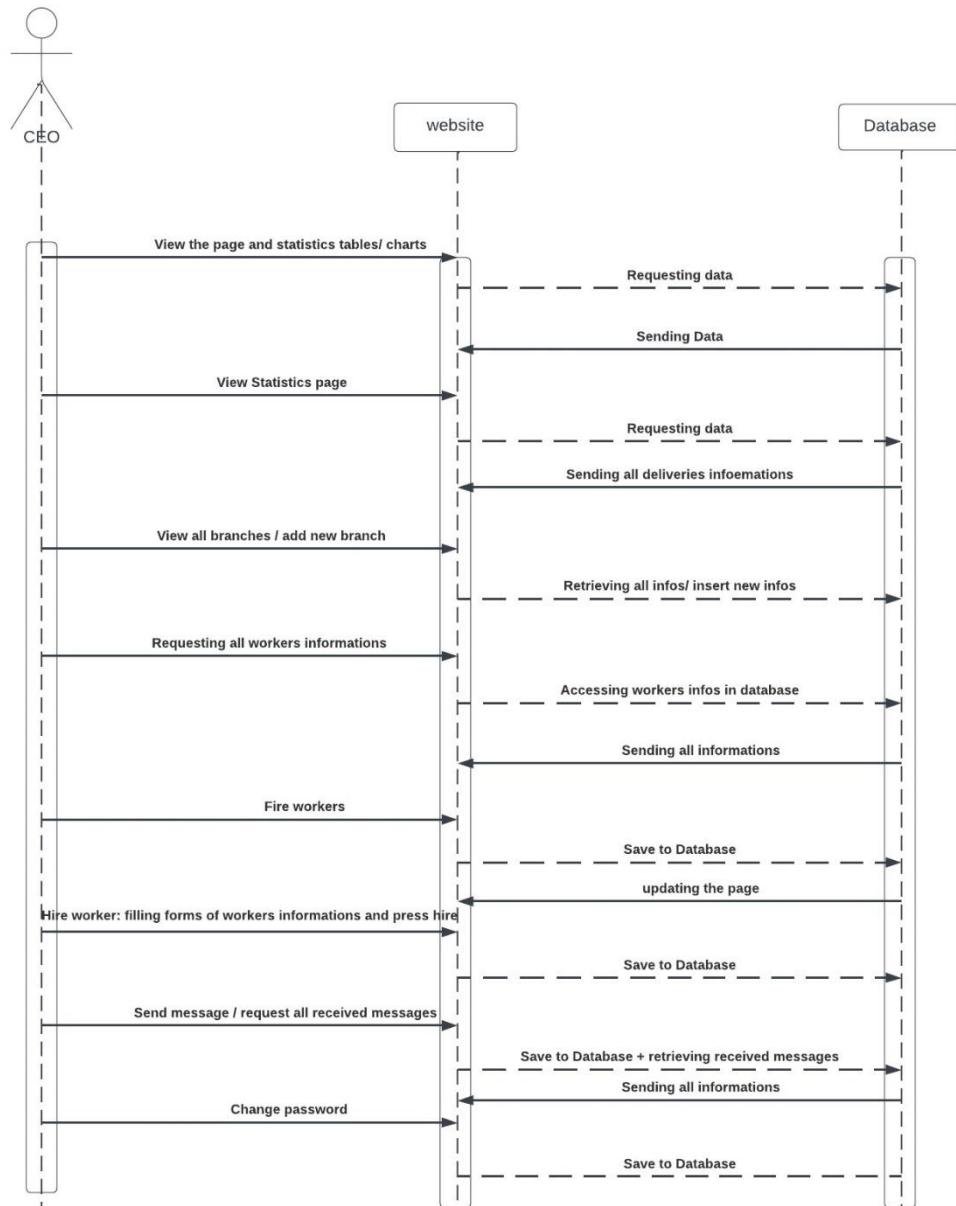


Figure 19: CEO sequence diagram

Upon login the CEO is presented with overall stats on the company like charts, stats...

The CEO can perform various actions. For example, the CEO can view the list of deliveries, which involves sending a request to the system to retrieve the relevant data. The system responds by returning the list of deliveries to the CEO. Another task that the CEO can perform is managing the branches of the company. This involves sending a request to the system to retrieve the list of branches. The system responds by returning the list of branches to the CEO.

The CEO can then add a specific branch by promoting one of the old workers to its branch manager, updating their information. The CEO can also request to view the workers, with the system responding by a list of them offering an option to fire each which will turn request that to the database.

(refer to client for change password and messages)

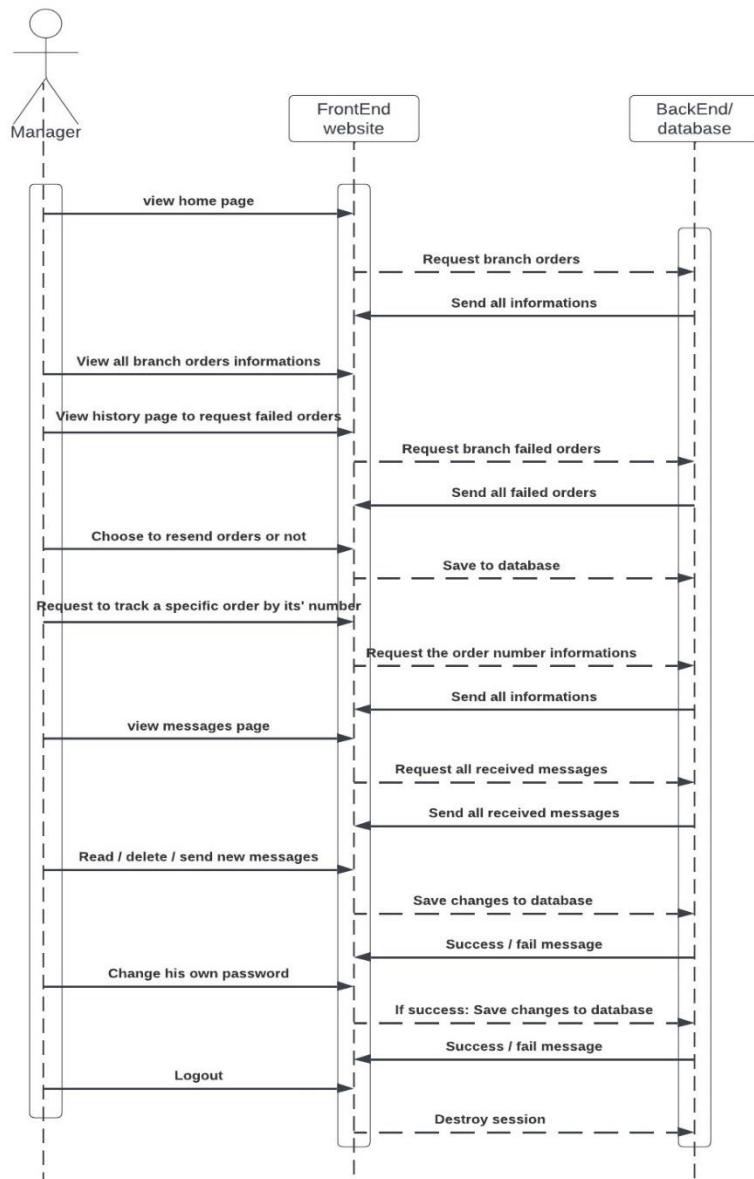


Figure 20: Branch Manager sequence diagram

The sequence diagram for the branch manager begins with the manager personal dashboard, where they can view their branch's delivery information and monitor worker activity. The manager can choose to view their branch's delivery list or worker list, where they can add or remove workers as needed. They can also update the status of each failed delivery and resend it to customer by marking it from "failed" to "pending". To do this, the manager sends a request to the system with the appropriate parameters. The system processes the request and updates the list of deliveries accordingly. If there are any issues or problems with a delivery, the manager can send a message to the worker or contact customer support for further assistance. Finally, when the manager is finished with their tasks, they can choose to log out of the system to ensure the security of their account.

(refer to client for change password and messages)

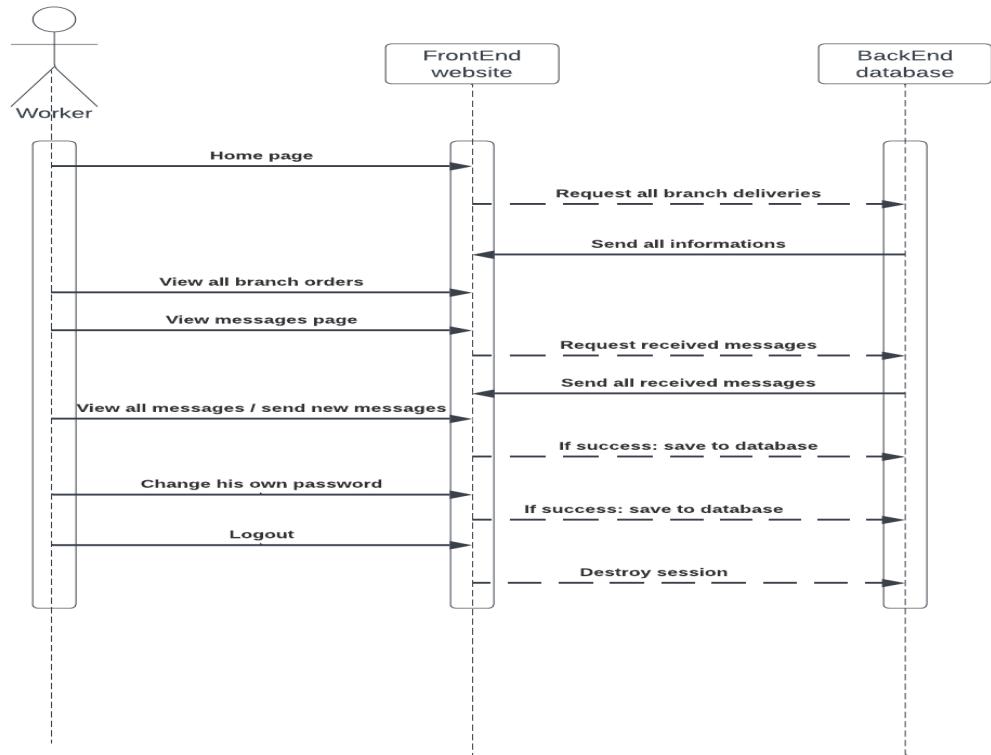


Figure 21: worker sequence diagram

The sequence diagram for the worker begins with the worker dashboard where they can view their assigned deliveries for the day. The worker can then select a specific delivery and mark its current location in the system. If the worker encounters any issues or problems during the delivery process, they can report it through the application by sending a message to their manager. The manager will receive the message and can take appropriate action to resolve the issue. Once the delivery has been successfully completed, the worker marks it as delivered in the system (mobile application). The delivery status is then updated in the database, allowing the client to track the package's progress. Finally, the worker can log out of the system to end their session. (refer to client for change password and messages)

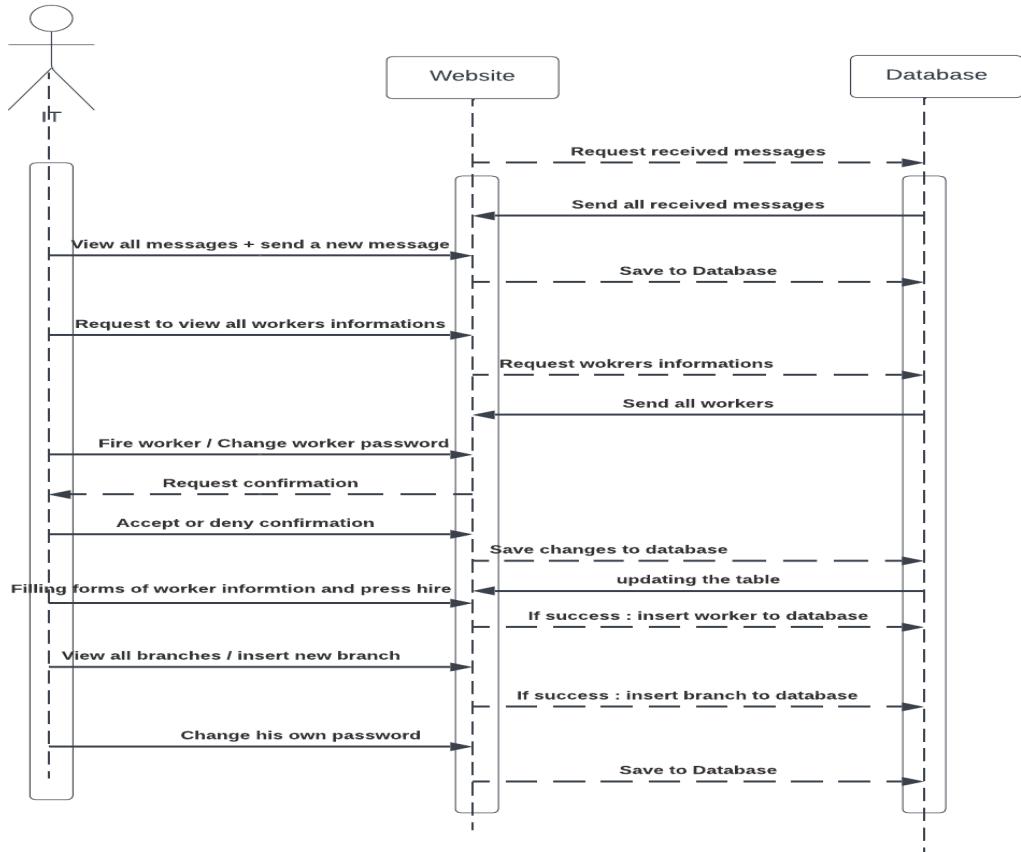


Figure 22: IT sequence diagram

The sequence diagram for the IT personnel's actions involves adding or deleting a worker from the system database. Additionally, when a worker is deleted, their saved information in the system is removed, preventing them from logging in to the website using their old account.

The IT personnel's sequence diagram can be described as follows:

1. The IT personnel follows the CEO's commands to add or delete a worker from the system.
2. If the IT personnel needs to add a new worker, they provide the necessary information, such as the worker's details and role.
3. The IT personnel submits the request to add the new worker to the system.
4. The system receives the request and updates the database by adding the new worker.
5. The system sends a success response to the IT personnel, indicating that the worker has been successfully added.
6. If the IT personnel needs to delete a worker, they select the worker from the list of existing workers.
7. The IT personnel confirms the decision to delete the worker from the system.
8. The system receives the request to delete the worker and removes their information from the database.
9. The system sends a success response to the IT personnel, confirming that the worker has been successfully deleted.
10. As a result of the worker's deletion, their saved information is removed from the system, rendering their old account inaccessible for logging into the website.
11. The IT personnel can then proceed with other tasks, such as managing messages or changing passwords, by referring to the respective sequences described in the client's section.

(refer to client for change password and messages)

3.6 Activity Diagrams

An activity diagram is a type of UML (Unified Modeling Language) diagram that shows the flow of activities within a system or process. It depicts the sequence of actions or steps involved in completing a task, and can be used to model a wide range of scenarios, including business workflows, software processes, and system interactions.

Activity diagrams are useful for visualizing complex processes, breaking them down into simpler steps, and identifying potential bottlenecks or inefficiencies. They can help stakeholders understand the flow of a process, and can also be used to communicate the design of a system to developers. An activity diagram consists of nodes and edges. The nodes represent activities, decisions, and initial or final states, while the edges represent the flow of control between them. Each activity node can contain a description of the action it represents, as well as other details such as inputs, outputs, and conditions.

Activity diagrams are particularly useful for modeling use cases, as they can provide a clear and concise overview of the steps involved in completing a particular task or process. They can also be used to document business processes, system interactions, and software workflows. Overall, activity diagrams are a valuable tool for system analysts, designers, and developers, helping them to better understand and communicate the flow of activities within a system or process.

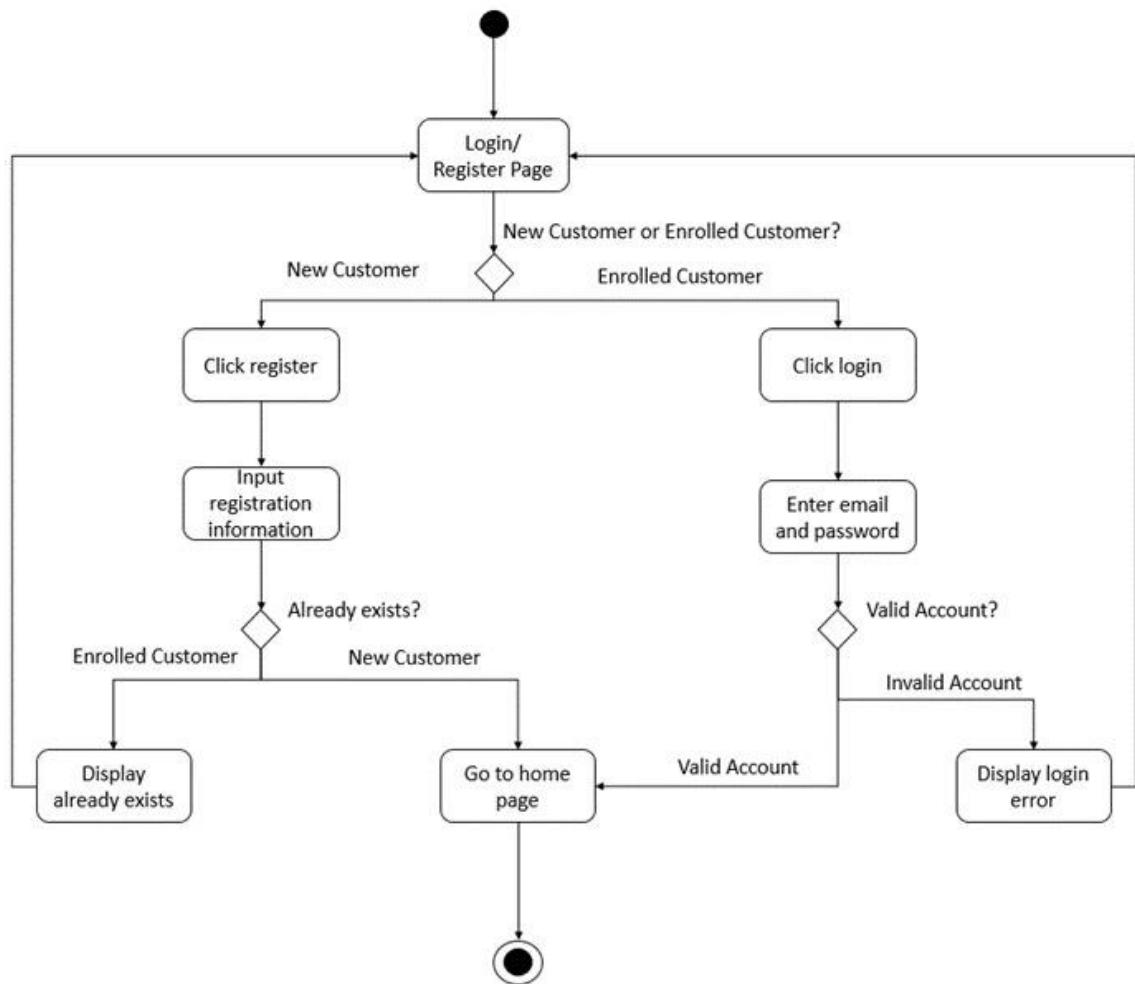


Figure 23: login activity diagram

When a user attempts to log in to the application, their email and password are verified against the database to ensure that they correspond to an existing account. If the login credentials are validated, the user is granted access to their personal page. However, if the verification fails, the user is unable to proceed beyond the login page until they take further action. Additionally, the application provides users with the option to register for an account. During the registration process, the system checks to see if the user's account already exists in the database. If the account exists, the registration process is terminated. On the other hand, if the account is not found, the user is successfully registered and automatically logged in.

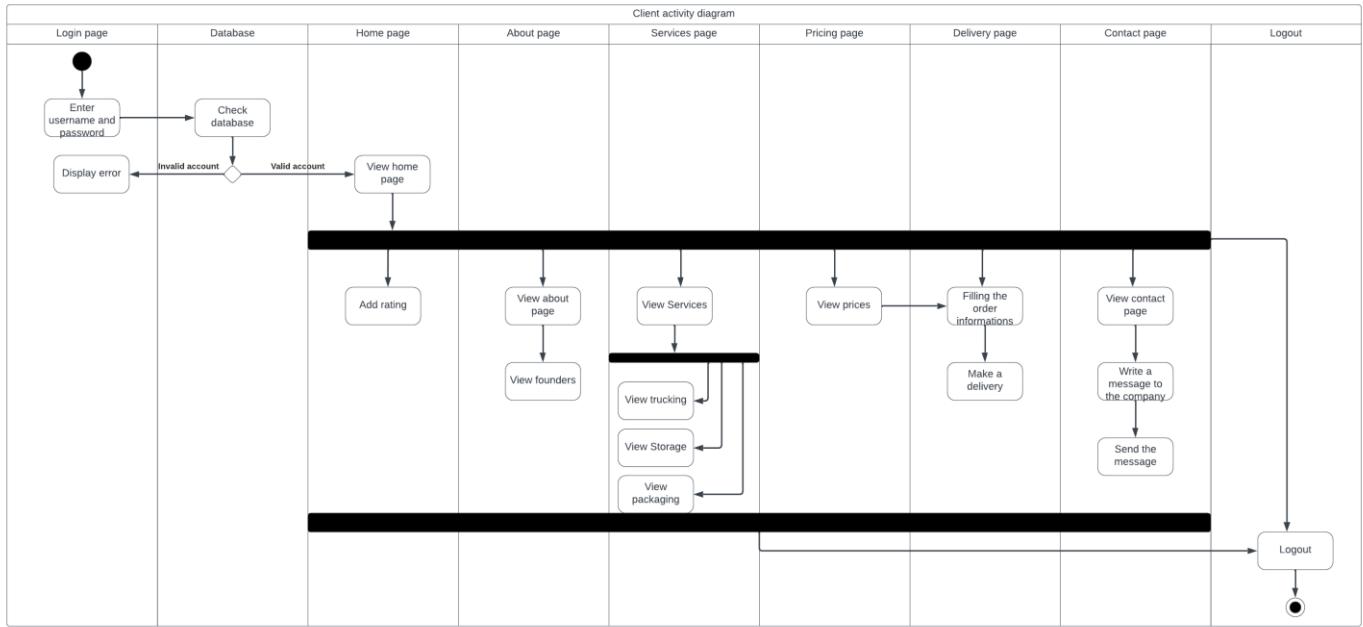


Figure 24: Client activity diagram

It starts with the login activity, where the client enters their credentials and is directed to their account page. From there, they can view their package history, search for packages, and place orders. When placing an order, the client must enter the package details and the desired delivery address. Once the order is placed, the client can track the status of their package and receive notifications on its progress. The client can also view their account details, including their contact information and order history. He is also able to rate our services. And logout once they finished their tasks.

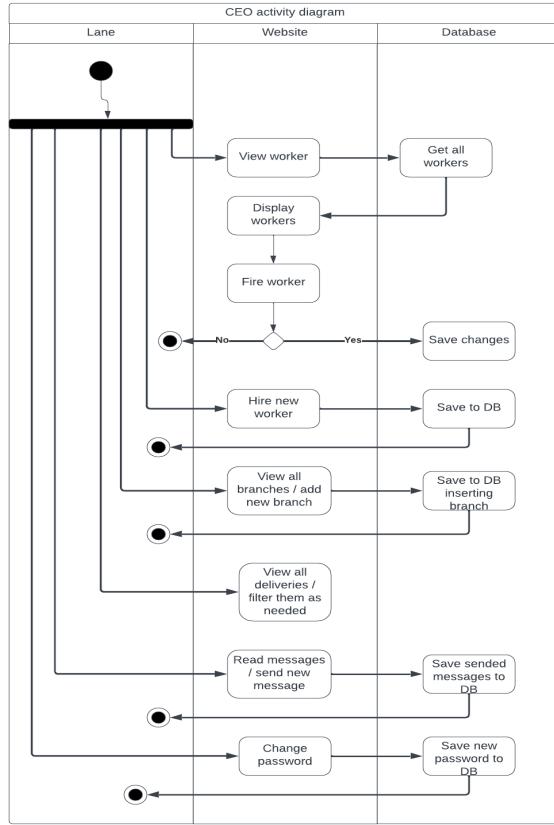


Figure 25: CEO activity diagram

The diagram begins with the CEO logging into the system and accessing the CEO dashboard, which allows them to view all deliveries made by the company and manage them efficiently. The CEO can also perform various tasks such as managing branch managers, IT staff, and monitoring the overall performance of the company. With the ability to add new branches to the system, the CEO can expand the company's reach and improve its operations. The diagram also includes a feature that enables the CEO to view reports and analytics, providing insights to make informed decisions. Finally, the CEO can log out of the system after completing the required tasks.

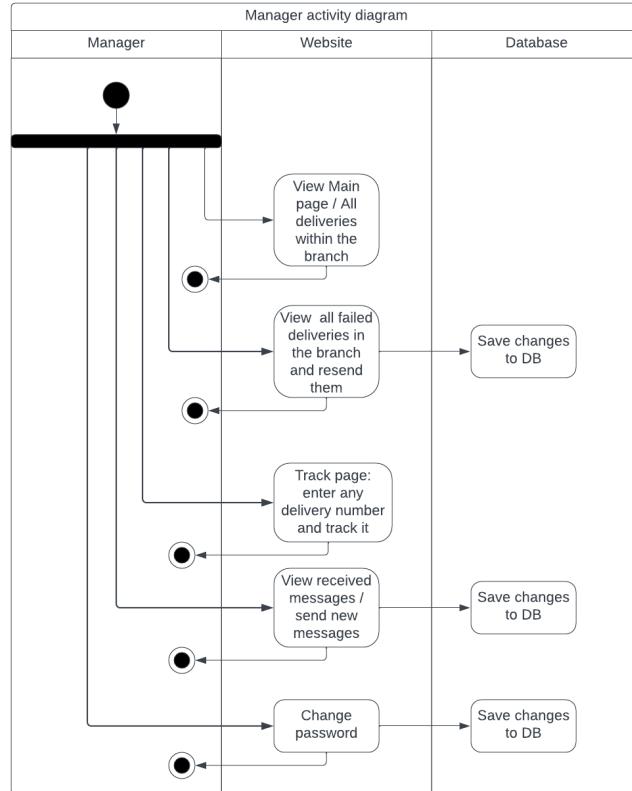


Figure 26: Branch Manager activity diagram

The branch manager activity diagram begins with the branch manager logging into the system and being directed to their dashboard. From here, the manager can view all deliveries made by their specific branch, track and manage them. The diagram includes the ability for the manager to assign tasks to workers and track their progress. The manager can also view the status of packages and update package information. In addition, the manager can communicate with other managers and the CEO via the messaging system. Finally, the manager can log out of the system when their tasks are complete.

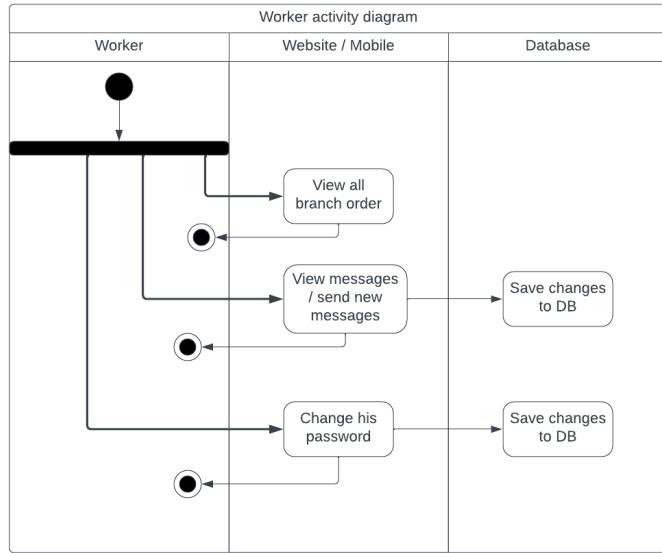


Figure 27: Worker activity diagram

The worker activity diagram starts with the worker logging into the system and being directed to their dashboard. From here, the worker can view the list of assigned deliveries and mark them as picked up or delivered once they have completed the delivery. If there are any issues with a delivery, such as damage or missing items, the worker can report it through the system. Additionally, the worker can view their personal profile. The worker can log out of the system once they have completed their tasks.

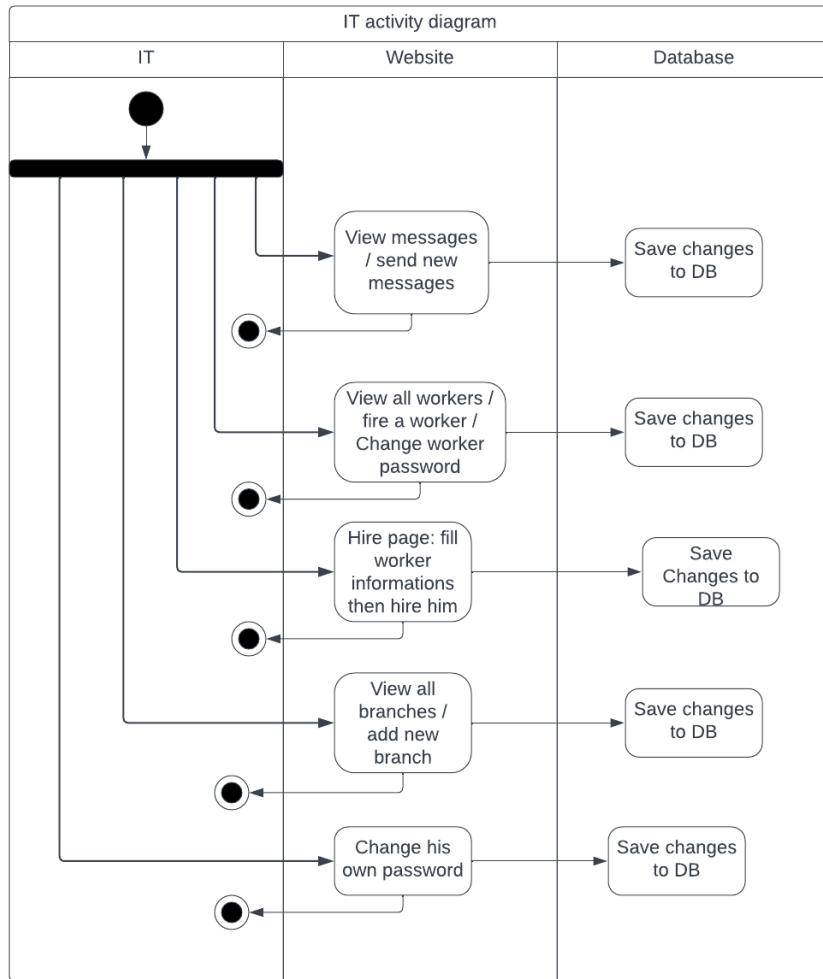


Figure 28: IT activity diagram

The diagram starts with the IT staff logging into the system and being directed to the IT dashboard. The IT staff can also view and manage user accounts, including resetting passwords and updating user information. Additionally, the IT staff can view messages from the CEO regarding specific tasks such as hiring or firing a worker, and take appropriate action based on those messages. Finally, the IT staff can log out of the system when finished with their tasks.

3.7 Entity-Relationship (ER) Diagrams

An ER diagram, also known as an entity relationship diagram, is a visual representation of the entities, attributes, and relationships within a database or information system. This diagram helps in the conceptual design of databases, providing a clear view of how data is organized and how different entities and attributes relate to each other. An ER diagram consists of various elements, including entities, attributes, relationships, and cardinalities. Entities are the objects or concepts within the system, such as customers or orders, while attributes are the properties or characteristics of these entities, such as a customer's name or address. Relationships represent how different entities are related to each other, such as how an order is related to a customer. Cardinalities specify the nature of these relationships, such as how many orders a customer can have. Drawing an ER diagram is an essential step in the database design process, allowing designers to create a blueprint for the data model and identify potential issues before implementation. ER diagrams are also useful in communicating database design concepts to stakeholders and developers, ensuring everyone has a shared understanding of the data model.

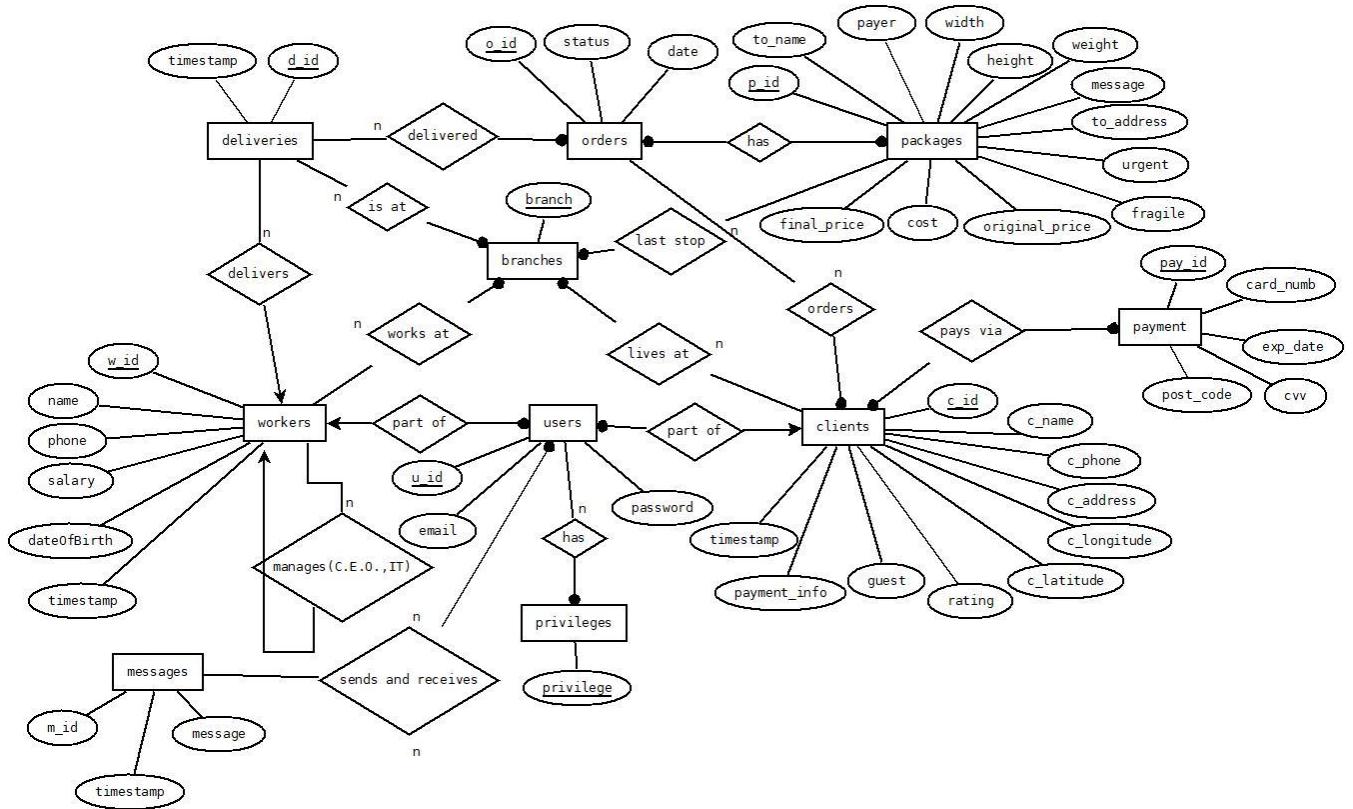


Figure 29: ER diagram

In the ER diagram provided, clients and workers are categorized as users and granted appropriate privileges based on their respective roles, which include Client, CEO, BranchManager, IT, and worker. Each client is associated with a specific district near a branch, while each worker is assigned to a particular branch within the company. The deliveries table is designed to track the location of deliveries and which worker delivered them. When a client places an order, the specific details are stored in the packages table. The order table serves as a link between the client, their package, and the delivery. The client and worker tables provide further information about these major user types. The CEO and IT have the ability to manage other workers, further highlighting their roles within the system. As well as everyone has the ability to send messages to each other knowing their email.

3.8 Non-Technical Aspects

3.8.1 Financial Viability

A cost-benefit analysis can help to determine the financial viability of the cargo tracking app. The analysis involves evaluating the costs of developing and implementing the app against the expected benefits or returns. Here is a simplified cost-benefit analysis for the cargo tracking app:

Benefits:

- Improved delivery management and tracking
- Increased customer satisfaction and loyalty
- Increased operational efficiency and productivity
- Improved decision-making based on real-time data
- Potential for increased revenue due to improved services
- Improved efficiency: The app can help streamline the package delivery process and reduce manual paperwork, leading to faster and more accurate deliveries.
- Enhanced customer experience: Clients can easily track their packages and receive real-time updates, improving their satisfaction with the delivery service.
- Increased transparency: The app can provide visibility into the delivery process, allowing for better monitoring and accountability.
- Better communication: The app can facilitate communication between clients and the delivery team, reducing misunderstandings and improving overall coordination.
- Competitive advantage: Implementing a modern and efficient delivery system can set the company apart from competitors and attract more clients.
- Reduces the number of lost packages, thereby increasing customer satisfaction and reducing costs associated with replacing lost items.

- Cost Savings: The app can help reduce manual labour and paperwork, potentially leading to cost savings in the long term.
- Scalability: The app can be scaled to handle a growing number of clients and packages, allowing for future growth and expansion.

Costs:

- Development and implementation costs (including hardware, software, and personnel)
- Maintenance and support costs (including ongoing updates and upgrades)
- Marketing and advertising costs to attract new clients

Table 3-1: Cost Benefit Analysis

Implementation	Amount (\$)
Costs	
Development team	-4000
Hardware and software	-3000
Web server hosting	-1000
Marketing and advertising	-500
Maintenance and updates	-500

Potential cybersecurity and data privacy costs	-500
Employee training and adoption costs	-100
Benefits	
Increased operational efficiency	15000
Improved customer satisfaction	10000
Reduced paperwork and manual labour	5000
Improved inventory management	3000
Increased revenue from new customers	2500
Total	25900.00

N.B.: Please note that these are estimated costs and may vary.

Based on the above analysis, the benefits of the cargo tracking app are expected to outweigh the costs, leading to a positive return on investment. However, it is important to note that the specific financial viability of the app will depend on a variety of factors, including the

size and competitiveness of the market, the effectiveness of the app's features, and the ability of the development team to stay within budget and timelines. Regular reviews and adjustments may be necessary to ensure the ongoing financial viability of the app.

3.8.2 Stakeholders

The stakeholders of this project can be divided into different groups, including:

- Customers: They are the primary users of the application, and will benefit from the convenience of placing orders, tracking deliveries, and managing their accounts. They may be harmed if the application fails to function properly, or if their personal information is compromised.
- Company management: This includes the CEO, branch managers, and IT personnel, who will benefit from the efficient management of deliveries and workers, as well as the ability to monitor the company's profits. They may be harmed if the application fails to meet their needs or if it causes more problems than it solves.
- Workers: They will benefit from the ability to easily track deliveries and manage their work schedules, but may also be harmed if the application adds unnecessary burdens to their work or if it fails to work properly.
- Third-party service providers: This may include payment processors, and other service providers that are integrated into the application. They will benefit from increased business, but may be harmed if the application fails to properly integrate with their services or if it causes problems for their own operations.
- Regulators and legal authorities: They may have a say in how the application works in order to ensure compliance with laws and regulations related to data privacy, consumer protection, and other relevant issues.

Ultimately, the stakeholders who should have a say in how the project works include all of the groups mentioned above, as well as any others who may be affected by the application's operations. This may include representatives from customer advocacy groups, worker organizations, and other relevant stakeholders who can provide valuable input on how the application can be improved and how potential harms can be mitigated.

3.8.3 Scope

The scope of this project includes the development of a web-based application for package delivery management. The application should allow clients to register and log in to their accounts, view available services, and place delivery orders. The application should also allow for the tracking of deliveries and the viewing of order history. The CEO, branch managers, and workers should have access to the application with varying levels of privileges depending on their roles. The application should be secure and scalable, able to handle a potentially large number of users and packages. The project should be completed within a set timeline and budget.

3.8.4 Risks

There are several risks that could potentially impact the success of this project. One major risk is technical challenges that may arise during the development process, such as compatibility issues or system malfunctions. Another risk is the possibility of security breaches or data leaks, which could lead to the loss of sensitive information or damage to the company's reputation. Additionally, the project may face financial risks if costs exceed the allocated budget or if there are unexpected expenses that arise.

Other factors that may impact the project's success include changes in regulations or market conditions, stakeholder disagreements or conflicts, and unforeseen external events such as natural disasters or global crises.

To mitigate these risks, the project team should conduct thorough testing and quality assurance procedures, implement robust security measures, and regularly review and adjust the project plan as needed to stay on track and within budget. Additionally, open communication and collaboration with stakeholders can help to identify and address potential issues before they become major problems.

3.8.5 Schedule and Milestones

Task name	Duration	28/2	6/3	13/3	20/3	27/3	3/4	11/4	18/4	25/4	2/5	8/5	15/5	22/5	29/5
Welcome and project brief	1 week		↔												
Conduct user research and gather requirements	1 week			↔											
Analysis of existing tools	1 week				↔										
Develop project plan and timeline	1 week					↔									
Diagrams	1 week						↔								
Overview of certain protocols	1 week							↔							
back-end functionality and database integration	5 week									↔					
user interface and front-end functionality	5 week									↔					
delivery tracking system	1 week											↔			
Conduct final testing and quality assurance	1 week												↔		

Figure 30: Scheduling Tasks and Milestones

Our project schedule can be divided into 4 major phases:

Phase 1: Planning and Design (4 weeks)

- Conduct project kick-off meeting with advisor and define project scope and goals
- Develop project plan and timeline
- Conduct user research and gather requirements

- Create wireframes and design prototypes
- Obtain necessary approvals from advisor

Phase 2: Development (10 weeks)

- Develop the user interface and front-end functionality
- Develop the back-end functionality and database integration
- Implement security measures and user authentication
- Develop delivery tracking system and mobile application
- Conduct thorough testing and quality assurance

Phase 3: Deployment and Launch (1 week)

- Deploy the application on a live server
- Train employees and users on how to use the application
- Conduct final testing and quality assurance
- Launch the application to the public

Phase 4: Post-Launch (Ongoing)

- Monitor application performance and user feedback
- Address bugs and issues as they arise
- Make regular updates and improvements to the application as needed
- Conduct regular security audits and updates

3.8.6 Ethical and Social Considerations

There are several ethical issues that should be taken into consideration when completing the design of the application [13]. One of the primary concerns is data privacy and security. As the application will store personal information of clients and workers, it is important to ensure that this data is securely stored and protected from unauthorized access [13].

Another ethical issue to consider is fairness and transparency in the application's decision-making processes. For example, if the application is used to assign work to individual workers, it is important to ensure that this process is fair and unbiased, and that workers are not discriminated against based on factors such as age, gender, or race.

Additionally, it is important to consider the potential impact of the application on the environment. For example, if the application is used to manage deliveries, it is important to consider how this may impact carbon emissions from transportation and explore ways to reduce the environmental impact.

Finally, it is important to ensure that the application does not perpetuate any harmful stereotypes or biases, particularly with regards to the representation of different groups of people such as clients and workers [14].

3.8.7 Environmental and Sustainability Considerations

As an application-based project, it is unlikely that there will be significant environmental impact associated with its development and use. Therefore, it can be stated that this project has no major environmental considerations or impact. However, if the project is to be implemented in such a way that involves the use of physical resources such as server infrastructure, energy consumption and disposal of hardware should be considered in order to minimize any potential environmental impact.

3.8.8 Relevant Standards

- Electronic Product Code (EPC): This standard provides a framework for identifying, tracking, and managing goods in the supply chain using RFID technology. [15]

- Electronic Data Interchange (EDI): This standard provides a format for exchanging business documents electronically, such as purchase orders, invoices, and shipping notices. [16]
- Application Programming Interface (API): This standard provides a way for different software systems to communicate with each other, enabling data sharing and integration. [17]
- Global Positioning System (GPS): This standard provides a way to determine the precise location of a vehicle or shipment using satellite technology. [18]
- Internet Protocol (IP): This standard provides a way to transmit data over the internet and other networks. [19]
- Transport Layer Security (TLS): This standard provides a way to secure data transmissions over the internet using encryption and authentication [20]
- PHP: a server-side scripting language commonly used for web development. There are several PHP standards organizations, such as the PHP Standards Group, that provide coding standards and best practices for PHP development. [21]
- JavaScript: a scripting language used primarily for client-side web development. The ECMAScript specification provides the standard for JavaScript syntax and behavior. [22]
- HTML/CSS: the markup and styling languages used to create web pages. The World Wide Web Consortium (W3C) provides standards for HTML and CSS development. [23]
- MySQL: the programming language used to manage relational databases. The ANSI SQL standard defines the syntax and behavior of SQL. [24]

In addition to these technical standards, there may be non-technical standards that are relevant to the project, such as organizational policies and procedures, industry regulations, or legal requirements.

Some standards that are beyond the scope of this project, due to time limitation:

- ISO 28000: This standard provides guidelines for implementing a security management system in the supply chain. [25]
- ISO 31000: This standard provides guidelines for implementing a risk management system in an organization. [26]
- Secure Sockets Layer (SSL): This standard provides a way to secure data transmissions over the internet using encryption. [27]

3.9 Conclusion

The cargo tracking application is designed to provide users with a seamless experience in tracking their packages. The application caters to multiple user types, including clients, workers, branch managers, IT, and CEO, each with different levels of access and privileges.

The ER diagram outlines the relationships and dependencies between different tables, while the sequence diagrams depict the step-by-step interactions between users and the system.

During the design process, various ethical and social considerations were taken into account, such as data privacy and security, as well as the potential impact of the application on stakeholders, including clients and workers.

To ensure the smooth functioning of the application, various technical standards, such as programming languages and database management systems, were used. However, there are also risks involved in the project, such as technical failures or unforeseen ethical issues.

Overall, the scope of the project was clearly defined, with a focus on delivering a user-friendly and efficient package tracking system.

CHAPTER 4

IMPLEMENTATION/SIMULATION AND TESTING

4.1 Introduction

In this chapter, we are going to take an in-depth look at the implementation of the cargo tracking app project. Divided into three main sections, namely Implementation Tools, Implementation Summary, and Test Cases and Acceptance Criteria, this chapter provides a comprehensive overview of the implementation process. The Implementation Tools section discusses the various tools and technologies used during the development of the project, while the Implementation Summary section provides a detailed overview of the development process, including challenges and solutions. The Test Cases and Acceptance Criteria section outlines the testing strategy and criteria used to ensure the functionality and quality of the application. Overall, this chapter serves as a comprehensive guide to the implementation and testing of the cargo tracking app project.

4.2 Implementation Tools

- ❖ Hardware:
 - Desktop/Laptop computers
 - Servers
 - Network devices (routers, switches, etc.)
 - Mobile devices (smartphones, tablets, etc.)
- ❖ CASE (Computer-Aided Software Engineering) tools
 - Dia and Lucidchart used for designing and creating software models, diagrams, and flowcharts
- ❖ Development software :

- Integrated Development Environment (IDE) such as Android Studio and Visual Studio Code (or Eclipse, or IntelliJ IDEA...)
- Text editor such as Notepad++ (or Sublime Text or Atom...)
- Version control software such as Git and GitHub (or SVN)
- Collaboration tools such as Microsoft Teams (or Slack)
- XAMPP for local web development environment
- ❖ Web development frameworks and libraries:
 - CSS frameworks such as Bootstrap (or Materialize)
 - JavaScript(AJAX)
 - Front-end HTML
 - Back-end PHP
- ❖ Mobile application development:
 - Java
 - XML
- ❖ Database and data management tools:
 - Relational database management systems (RDBMS) such as MySQL (or PostgreSQL, or SQL Server)
 - Data format: JSON
- ❖ Testing and debugging tools:
 - Debugging tools such as Chrome DevTools or Visual Studio Debugger
- ❖ Deployment and hosting tools:
 - Cloud computing platforms such as Render, Github (or AWS or Microsoft Azure)
 - Web server software such as Render(or Apache or Nginx)

Note: This list is not exhaustive and may vary depending on the specific implementation and requirements of the project.

4.3 Implementation Summary

In this section, we will provide a detailed summary of the implementation process for the cargo tracking app project. The project was implemented using PHP, CSS, Bootstrap, HTML, and JavaScript, following the Model-View-Controller (MVC) architecture pattern to ensure efficient development and maintenance of the application. The implementation focused on delivering a seamless user experience and efficient delivery management for the various user roles: CEO, IT, Branch Manager, worker, and customer.

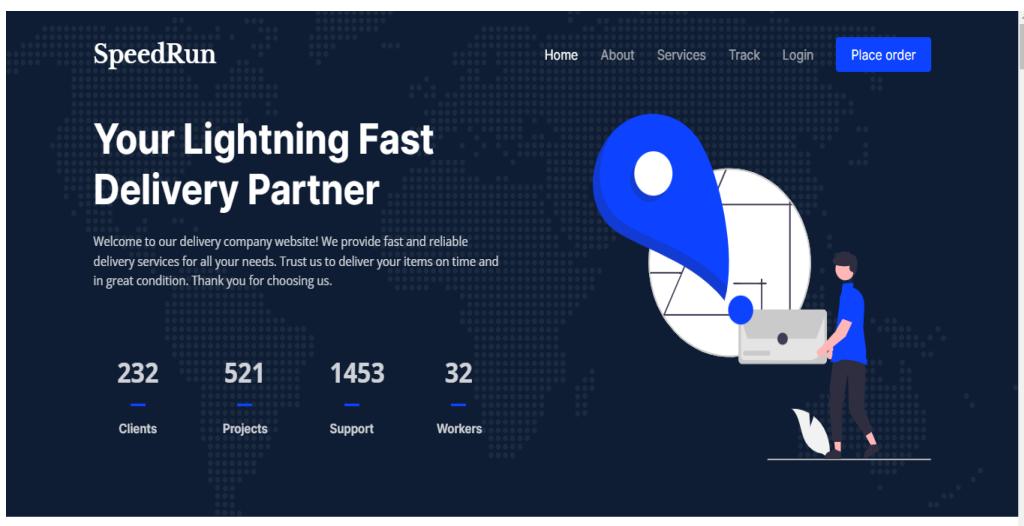


Figure 31:Index page

Starting with the index page (Figure 29), it serves as the homepage for the application, providing information about the company's services and pricing. The landing page features a visually appealing design and a user-friendly navigation menu, allowing users to easily access login, sign up, and contact information.

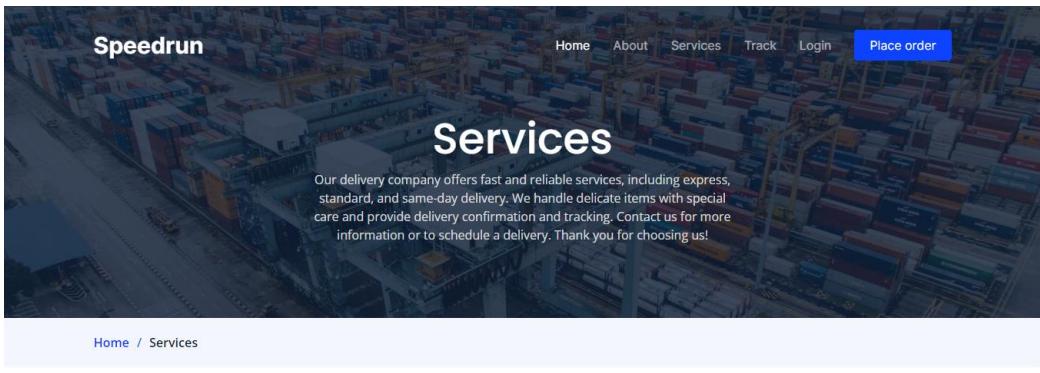


Figure 32:services.html

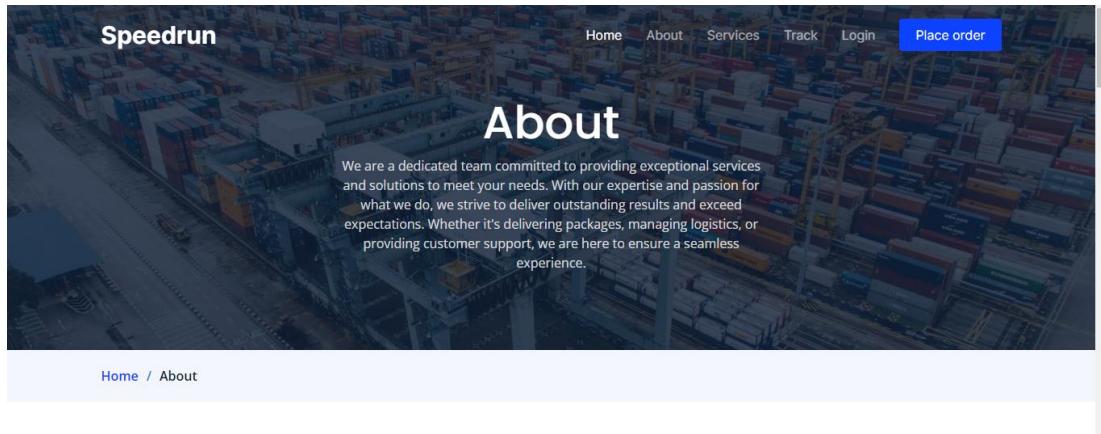


Figure 33: about.html

For users who haven't logged in, the app provides functionalities such as viewing the about page (Figure 31), services (Figure 30), tracking an order, and placing an order. These functionalities enable potential customers to gather information and interact with the app.

MAKE AN ORDER

To place an order [Home](#) [About](#) [Services](#) [Track](#) [Login](#) [Place order](#)

SHIPPER'S DETAILS

Shipper's name _____ Shipper Phone number _____
 Shipper's Address _____ Shipper district _____

RECEIVER'S DETAILS

Recipient's name _____ Recipient Phone number _____
 Recipient's Address _____ Delivery district _____

SHIPMENT DETAILS

Total width (cm) _____ Total height (cm) _____

[Place order](#)

Figure 34: place order(1)

Message

Is your item fragile?(extra cost)
 Same day delivery(extra cost)
 Receiver pays
 pay at delivery?

Your order request has been sent successfully. Thank you! Order ID:
 64

[Place order](#)

Figure 35:Place order(2)

To place an order (Figure 32 and 33), customers must fill in the required details in a form divided into Shipper's details, Receiver's details, and shipment's details. Additional options, such as fragile items and same-day delivery, come with extra costs. Pay at delivery means cash not credit. To ensure security and prevent spam, customers are required to complete a captcha. After placing an order, customers receive an order number that they can use to track their shipment.

The screenshot shows a web page with a dark header and footer. The main content area has a white background. At the top, there is a section titled "Order Details" containing the following information:

- Order number: 64
- Status: successful
- Cost: 10
- Cash on delivery: no
- Sender pays: no
- Fragile: no
- Same day delivery: yes

Below this is a table with three columns: "DATE AND TIME", "LOCATION", and "BY". The data is as follows:

DATE AND TIME	LOCATION	BY
2023-05-28 14:45:33	still at client	Antonio Al Makdissi(you)
2023-05-28 15:00:05	Al Koura	workerK
2023-05-28 15:00:21	Batroun	workerK
2023-05-28 15:00:56	Beirut	worker2
2023-05-28 15:01:17	delivered	worker

Figure 36:track order(outside)

Tracking an order is very simple and intuitive process, you input the order number and press track. The page will then proceed to show you the order details, and history of locations where you can track exactly when, where and who is handling your package.

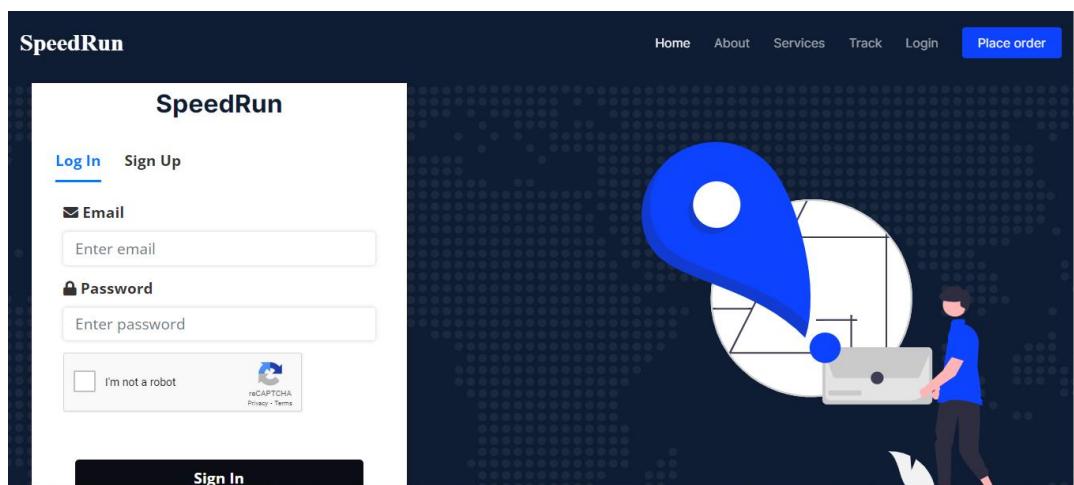


Figure 37:Login page

For the login page, any previous user can input his email and password, tick the captcha box and login successfully. Otherwise, they will have to create a new account.

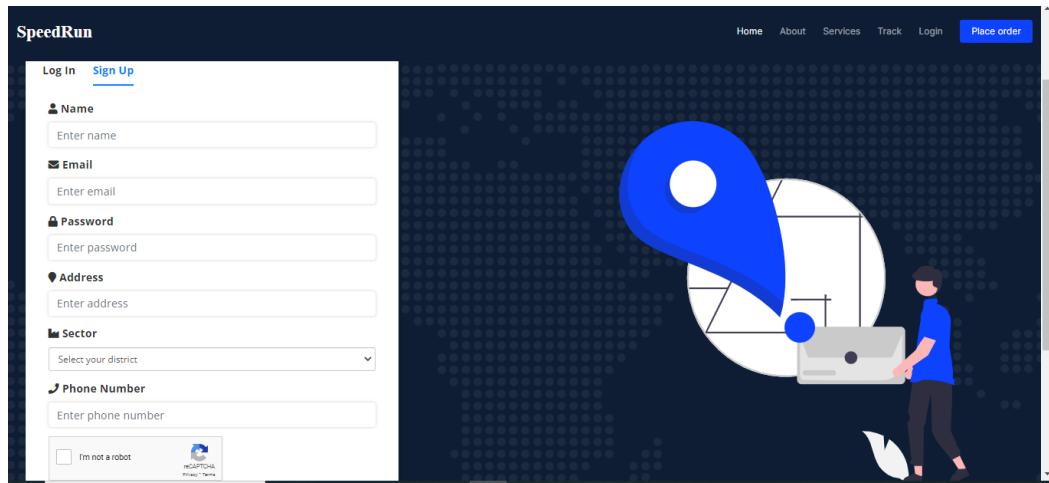


Figure 38:Sign up page

Creating an account by signing up, will only be a client account.

User authentication and the dashboard are essential parts of the cargo tracking app. The login and signup pages (Figures 35 and 36) are designed to provide a seamless user experience, with clear instructions and user-friendly input fields. Once authenticated, users are directed to their personalized dashboard, which acts as a centralized hub for managing various functionalities and accessing relevant information specific to their roles.

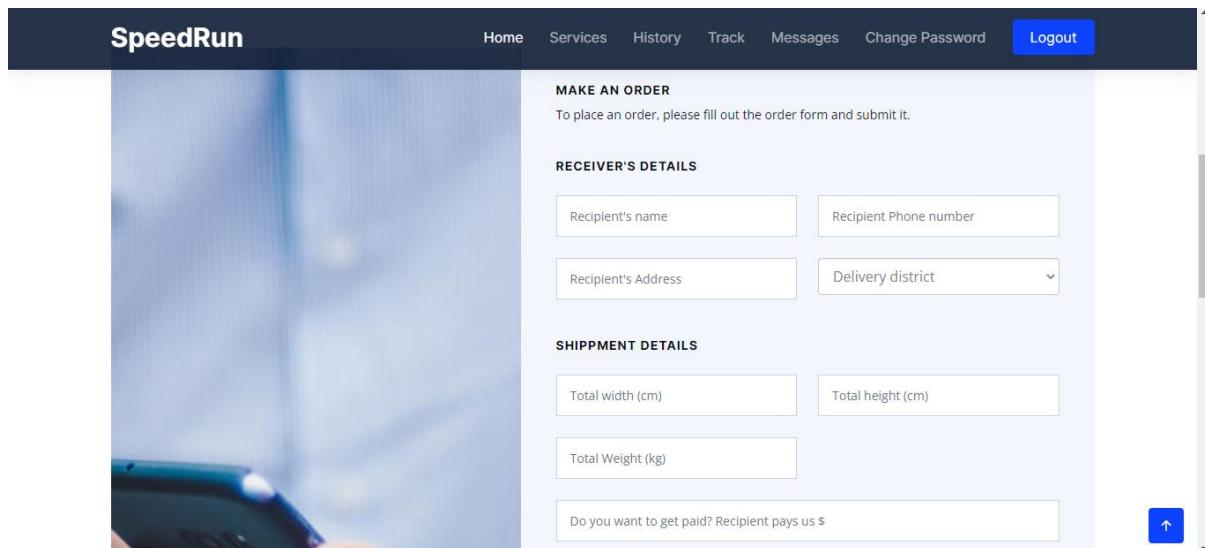
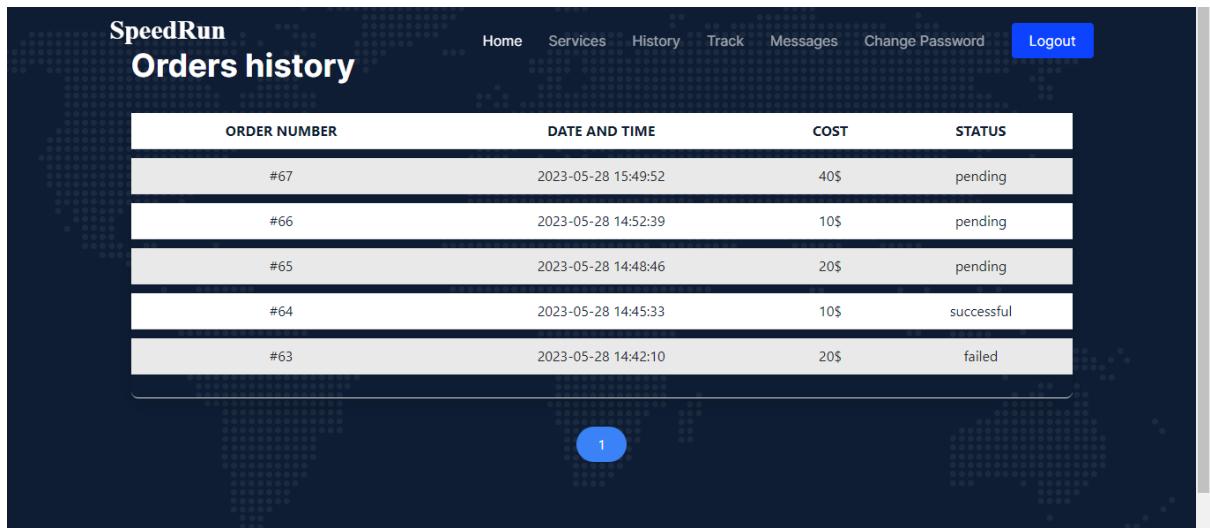


Figure 39:client page

Customers can place orders through the app by providing the necessary details, such as the pickup and delivery locations, package dimensions, and any additional instructions. The order page for a logged in client is slightly different. It only contains 2 sections receiver and shipment, as we already have the client's info and there's no need for them to fill them again each time. It also doesn't have a captcha as it was already authenticated.



ORDER NUMBER	DATE AND TIME	COST	STATUS
#67	2023-05-28 15:49:52	40\$	pending
#66	2023-05-28 14:52:39	10\$	pending
#65	2023-05-28 14:48:46	20\$	pending
#64	2023-05-28 14:45:33	10\$	successful
#63	2023-05-28 14:42:10	20\$	failed

Figure 40:client's orders history

Order History: Customers can access their order history within the app, allowing them to view details of previous orders. The history provides an overview of the customer's shipments and their status from newest to oldest. The manage shipments page allows the customer to view and manage their shipments, such as adding, editing, or cancelling shipments.

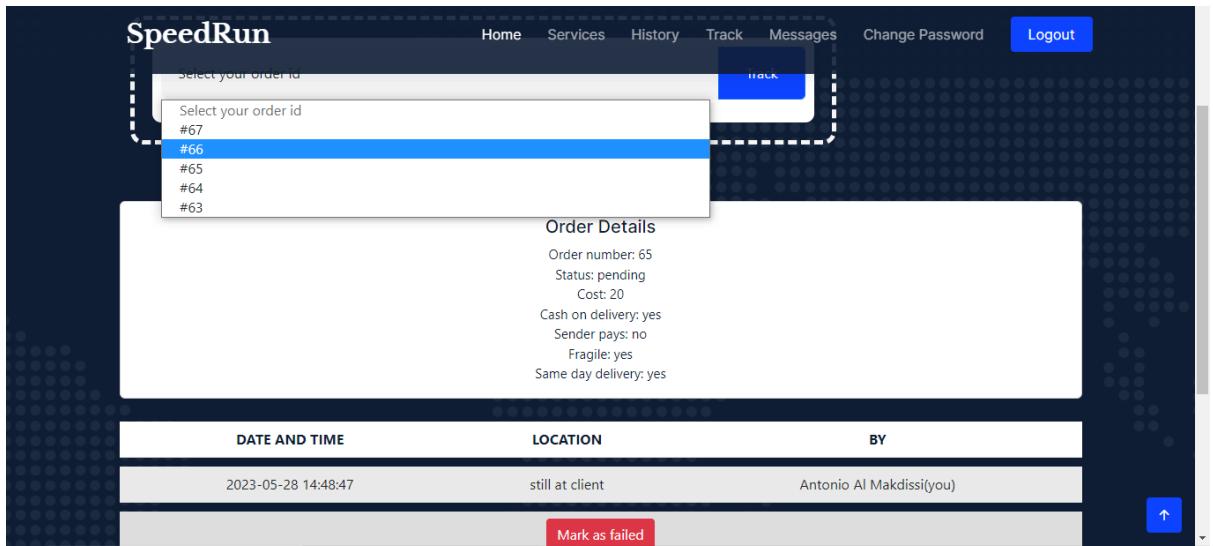


Figure 41:client track order(1)

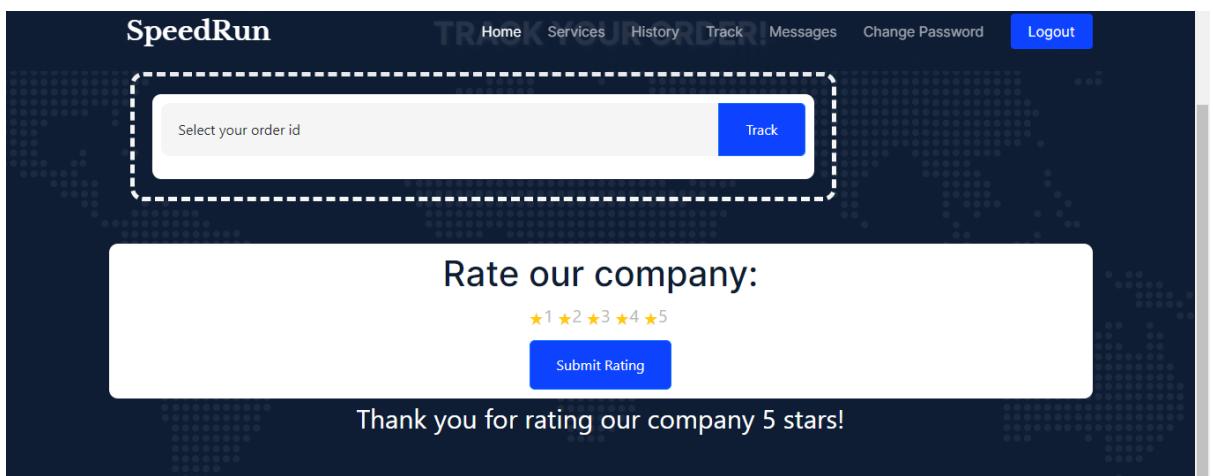


Figure 42:client track order (2)

Once an order is placed, customers can track its status in real-time. The app provides regular updates on the order's location, time and its handler, ensuring transparency and convenience for customers. Additionally, customers have the option to rate the company based on their experience. The app provides a rating system where customers can provide feedback and ratings, which helps the company improve its services and maintain customer satisfaction.

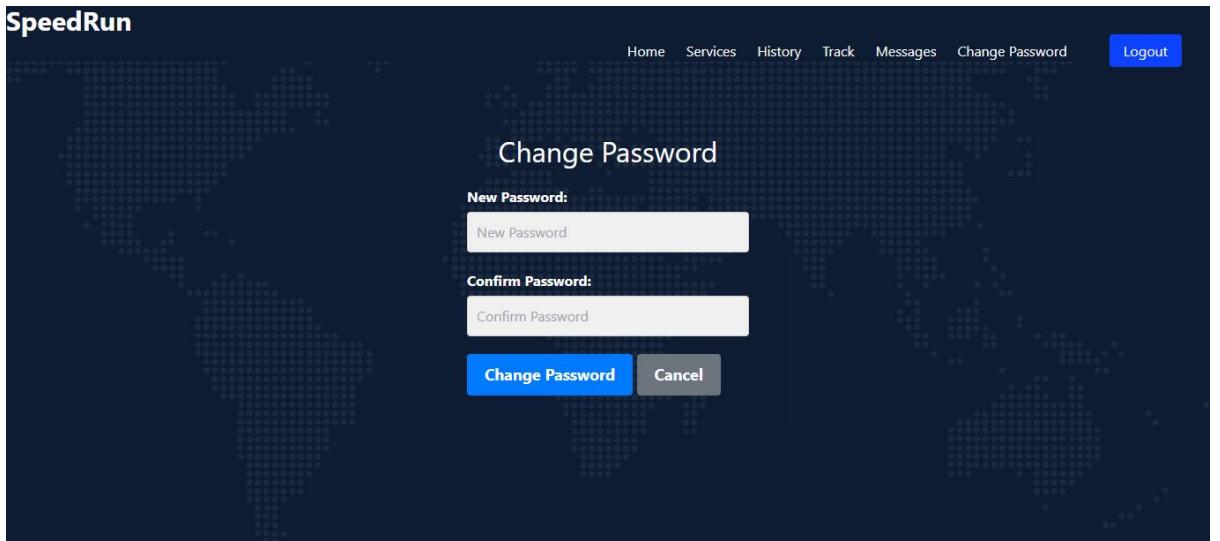


Figure 43: change password(common)

This page is common between all users, each user is able to change their own password by entering it twice confirming it and clicking on change password.

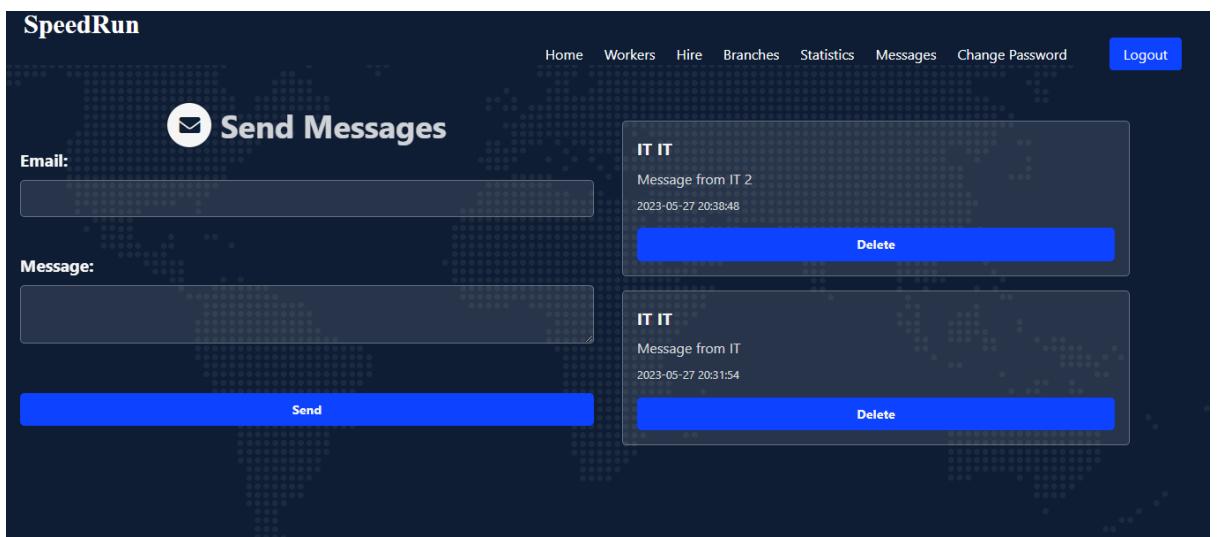


Figure 44:messages(common)

The messages page is also a common page between all users, allowing them to read their received messages from newest to oldest, with details: the sender and their position, the message, the time and the option to delete the message. They can also send

a message themselves, using the receiver's email. Clicking on the white mail icon will hide/show the messages.

Note: all common pages change nav-bars dynamically according to the user.

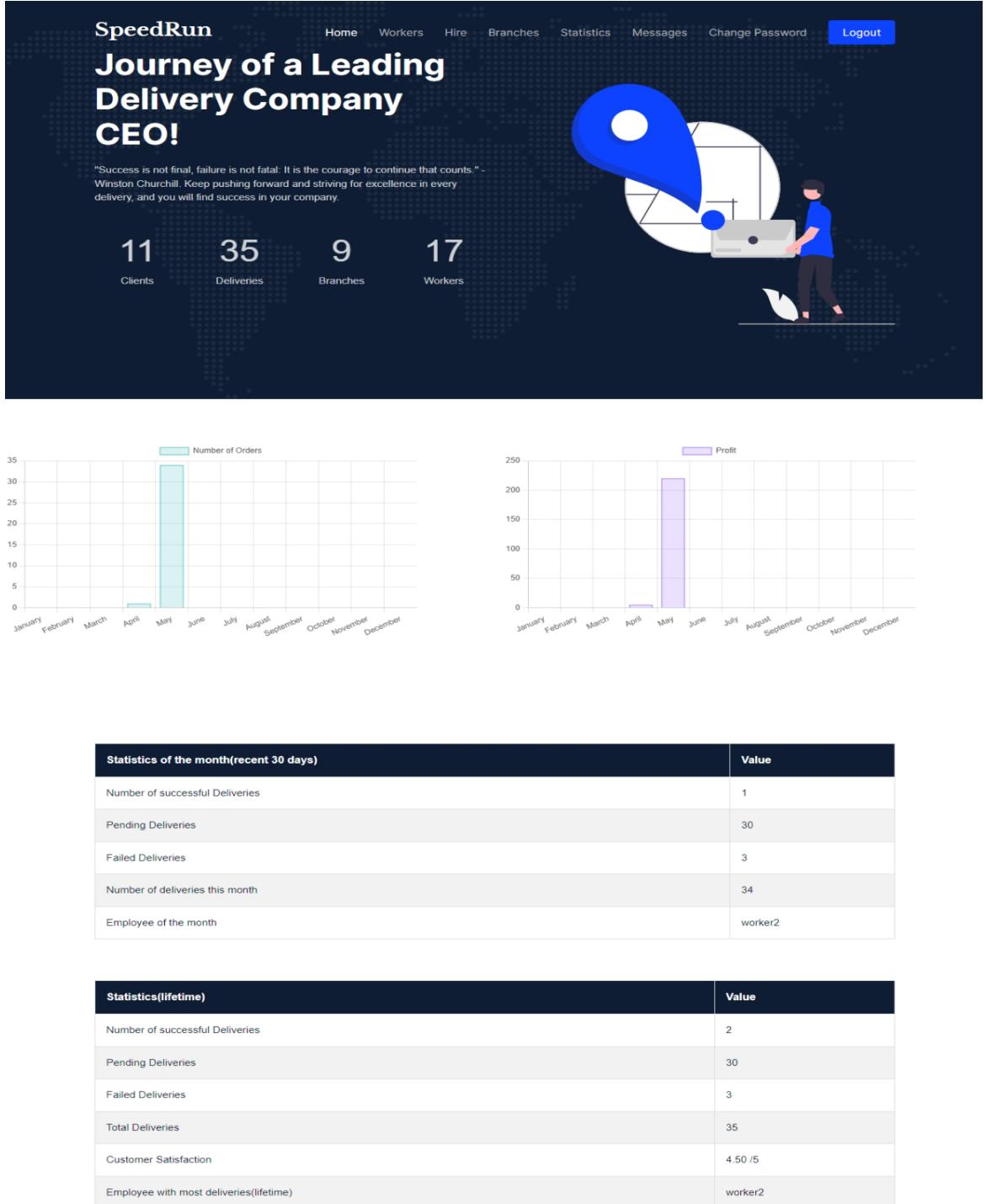


Figure 45: CEO landing page

CEO panel and IT personnel: The app includes an admin dashboard that provides access to privileged functionalities. The implementation involved the creation of several web pages for the CEO panel, including the landing page dashboard, view, fire and hire workers, view statistics and deliveries, and add branches. The dashboard provides an overview of the system's key performance indicators, such as the total number of shipments, users, and pending requests.

WORKER ID	NAME	BIRTHDATE	TYPE	PHONE	BRANCH	SALARY	DATE JOINED	PASSWORDS	ACTION
#1	CEO	1993-04-01	CEO	71787766	Beirut	10,000.00\$	2023-04-10 12:13:59	Change	
#2	IT	1993-04-01	IT	71787767	Beirut	1,000.00\$	2023-04-10 12:15:38	Change	
#7	me	2023-03-16	worker	8888	Akkar	3,000.00\$	2023-05-01 21:36:16	Change	<button>Fire</button>
#10	BranchManager	2000-01-01	BranchManager	89544848	Beirut	5,000.00\$	2023-05-03 13:39:08	Change	<button>Fire</button>
#11	worker	2023-05-03	worker	34454545	Beirut	1,000.00\$	2023-05-03 13:40:40	Change	<button>Fire</button>

Figure 46: view workers(CEO and IT)

The workers page allows the CEO and IT to view and manage all the registered users in the system, such as editing or deleting workers, except for themselves and the CEO. They can search for the worker by his ID, name or branch. **Note:** The IT has an additional functionality that the CEO doesn't have: the ability to change the password of others workers.

SpeedRun Orders history								
	DATE AND TIME	FRAGILE	URGENT	COST	PROFIT	PRICE FOR CUSTOMER	STATUS	
Show pending	2023-05-28 15:49:52	yes	yes	20\$	20\$	40\$	pending	
Show delivered	2023-05-28 14:52:39	no	no	5\$	5\$	10\$	pending	
Show failed	2023-05-28 14:48:46	yes	yes	10\$	10\$	20\$	pending	
Show All	2023-05-28 14:45:33	no	yes	5\$	5\$	10\$	successful	
#64	2023-05-28 14:42:10	yes	no	10\$	10\$	20\$	failed	
#63	2023-05-27 12:45:30	no	yes	15\$	15\$	30\$	pending	
#51	2023-05-26 22:05:29	no	yes	5\$	5\$	10\$	pending	
#50	2023-05-26 21:41:28	no	yes	5\$	5\$	10\$	pending	
#49	2023-05-26 21:40:24	no	yes	5\$	5\$	10\$	pending	
#48	2023-05-26 21:38:47	no	no	5\$	5\$	10\$	pending	
#47	2023-05-26 21:38:47	no	no	5\$	5\$	10\$	pending	

1 2 3 4

Figure 47: CEO stats

The stats page displays all previous orders from newest to oldest, with filtering option between the statuses of the orders with the important variables for the CEO.

SpeedRun								
Add Worker								
Name	workerT	Email	workerT@hotmail.com	Salary	1000	Enter date of birth:	11/10/1989	Logout
Password	Type	Worker	Sector	Beirut	Hire		
Phone Number	72626583							

Figure 48: Hire(CEO and IT)

SpeedRun			
Home Workers Hire Branches Statistics Messages Change Password Logout			
2023-05-28 14:13:03	Al Koura	BranchManagerK	BranchManagerK@hotmail.com
2023-05-28 14:13:22	Batroun	BranchManagerBa	BranchManagerBa@hotmail.com
2023-05-03 13:39:08	Beirut	BranchManager	BranchManager@hotmail.com
2023-05-17 19:09:36	Bekaa	BranchManager2	BranchManager2@hotmail.com
2023-05-28 14:12:12	Tripoli	BranchManagerT	BranchManagerT@hotmail.com
2023-05-28 14:12:21	Tripoli2	BranchManagerT2	BranchManagerT2@hotmail.com
Branch Location <input type="text" value="Enter branch location"/>		Branch Manager <input type="text" value="Select from workers"/>	
Add Branch			

Figure 49:view branches(CEO and IT)

The common pages between CEO and IT allowing them to add a new worker or a new branch to the company. Adding a new worker would be to already existing branch and we can choose between a worker or manager (to help existent manager). Adding a new branch creates a new branch with a choice of one of the old workers.

SpeedRun											
Home History Track Messages Change Password Logout											
Branch Orders											
ORDER NUMBER	URGENT?	FROM	PHONE SENDER	ADDRESS	CURRENT LOCATION	TO	PHONE RECEIVER	LAST STOP	DROP OFF	PRICE	Comments
#65	yes	Antonio Al Makdissi	88218282	kfaraaka	still at client	Gemma	71827262	Bekaa	example	20\$	
#67	yes	Antonio Al Makdissi	88218282	kfaraaka	still at client	Adib	89237564	Beirut	ashrafiyeh	40\$	
#66	no	Antonio Al Makdissi	88218282	kfaraaka	still at client	adel	89237564	Batroun	there	10\$	

Figure 50: Branch Orders(branch manager and worker)

This page allows the worker and manager to view all orders in their branch with their details, displayed by oldest to newest with the urgent orders taking priority over unurgent ones.

The screenshot shows a web application titled "SpeedRun" with a dark blue header bar. The header includes links for "Home", "History", "Track", "Messages", "Change Password", and "Logout". Below the header, the title "Orders history" is displayed in bold. A table with a light gray header row lists order details. The columns are labeled: RENT LOCATION, TO, PHONE RECEIVER, LAST STOP, DROP OFF, PRICE, PAYER, CASH?, FRAGILE?, URGENT?, STATUS, DATE AND TIME, and ACTION. The first row of data in the table includes: still at client, Aafif, 71827262, Akkar, Menyara, 20\$, sender, yes, yes, no, failed, 2023-05-28 14:42:10, and a blue button labeled "Resend order". At the bottom center of the page, there is a small circular icon containing the number "1", indicating the current page of the table.

Figure 51: Orders history manager

The manage shipments page allows the Branch Manager to view previous shipments, successful or failed. With the option to resend failed orders to the customer. The manager can also track orders(refer to previous sections).

Note: all pages with long tables are using pagination to divide the table into multiple pages for easier reading.



Figure 52:Mobile splash screen

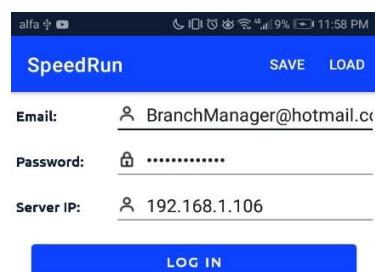


Figure 53: Mobile Login Page

The worker and branch manager have exclusive access to the mobile application. The app enables efficient delivery management for the company. Workers, and managers alike, assigned to specific branches can access the app's mobile version to view and manage the orders assigned to their branch. The application starts with a splash screen with the company's logo, then onto the login activity. The activity has the option to save/load credentials via button press. But it also does it automatically: the application loads any last saved credentials upon start up, and saves the credentials upon any successful login. It has an info button, to indicate that it's for authorized personnel only.

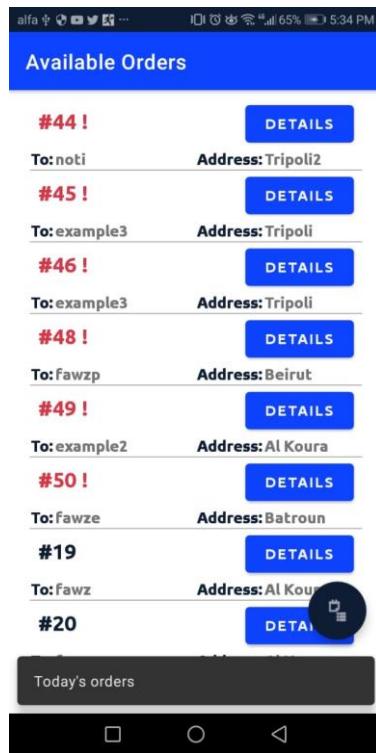


Figure 54: Mobile view branch orders

Once logged in, the application shows all the available orders for the user listing them from oldest to newest, with urgent orders taking priority and marked in red. Tapping the order will display its destination address in a toast. Pressing on details brings us to the details activity.



Figure 55:Mobile details and drop off

Here they can see all the details of the order and update the status of the order, such as marking it as delivered or dropped off at another branch. This functionality ensures smooth coordination and effective tracking of deliveries.

Note: the mobile application also sends notification to the worker upon receiving an urgent order. And a push notification every day at 12pm to check their orders.

The implementation process involved thorough requirement gathering, design, development, testing, and deployment. Libraries and tools such as Bootstrap, HTML, PHP, CSS, and MySQL database were utilized to enhance the development process and ensure robust data storage. The mobile application (Figures 50-53) provided exclusive access for workers and branch managers, enabling efficient delivery management and order tracking on the go.

The implementation of the "SpeedRun" cargo tracking app aimed to deliver a seamless user experience, efficient delivery management, comprehensive reporting, and enhanced features to meet the needs of both customers and the company

4.4 Test Cases and Acceptance Criteria

To ensure that the cargo tracking app is functioning properly, we will create test cases for each web page of the app. The test cases for the cargo tracking app have been designed to thoroughly assess its functionality and ensure a seamless user experience. Each test case represents a specific action and its expected outcomes, categorized into success and fail scenarios.

Table 4-1:Test cases

Test #	Action	Outcome	
		Success	Fail
1	Login	Redirects you to your correct account	No redirection: Wrong email or password Failed captcha OR: suggest to sign up
2	Sign up	Should always work as we denied incorrect user input	
3	Logout		Always logs out
4	Enter a page through URL		Always redirected to login
5	Message(read)	Display messages from new to old	No messages
6	Message(send)	Message Sent	Check email/try again later
7	Delete Message	Are you sure?-> Message gets deleted	Cancel->nothing Please try again later
8	Change Password	Password Changed	Passwords do not match. Please try again. error changing the password
9	Place Order(client)	Your order request has been sent successfully. Thank you!+Order ID	Failed to insert into packages/orders
10	Track Order	Displays Order route	Wrong order id
11	Mark as failed	Marked as failed	Error try again later
12	Rate	Thank you for rating or company x stars	Error try again later

13	History	Displays all previous orders from most recent to oldest	No orders yet!
14	CEO(landing)	Displays some statistics	Error try again later
15	View workers	Displays worker in order by id	Error try again later
16	Fire worker	Are you sure?->removes worker	Cancel->nothing Error try again later
17	Hire worker	Worker Hired Successfully	Error try again later
18	Add Branch	Branch Added Successfully	Error try again later
19	Statistics	Displays stats of deliveries from newest to oldest with filtering	Error try again later
20	View branch order	Displays all pending deliveries in the branch	Error try again later Error try again later
21	Resend Order	Marks order as pending	Error try again later
22	Deliver order to other branch	OK	Error try again later

The first set of test cases focuses on user authentication and account management. Users are required to log in with their credentials, and successful login redirects them to their personalized dashboard. In case of incorrect login details, users are not redirected and are prompted to sign up if they don't have an account. The signup process is expected to always work, as the system denies incorrect user input. Additionally, the logout feature should reliably log out users from their accounts.

The test cases also cover scenarios related to messaging functionality. Users can read, send, and delete messages. When reading messages, they should be displayed in the correct order, from newest to oldest. Sending a message should result in successful delivery, while deleting a message should remove it from the system. In case of any errors during the message-related actions, users are advised to try again later.

Changing passwords is another important aspect, and the test cases evaluate this functionality. Successful password changes should be reflected in the system, while any

errors or mismatches in password entry should be appropriately handled with clear instructions for the user.

Order placement and management are thoroughly tested. Users should be able to place orders successfully, with the system providing an order ID upon submission. Tracking orders should display the correct route, and marking an order as failed should update its status accordingly. Users are also given the ability to rate the company and access their order history. Any errors encountered during these processes should be handled gracefully, with appropriate error messages guiding the user.

The test cases extend to the administrative features of the app. The CEO and IT personnel have access to privileged functionalities. The CEO can generate reports summarizing profitable and unprofitable deliveries, view statistics, and manage admin accounts. IT personnel can change passwords, hire/fire workers, and add new branches to the system. Each of these actions should execute successfully, providing the intended outcomes.

Finally, the worker mobile app is tested for its ability to access order information within the assigned branch and update the status accordingly. This real-time tracking feature ensures efficient record-keeping and smooth order management.

By conducting these comprehensive test cases, the cargo tracking app can be thoroughly assessed for its functionality, reliability, and user-friendliness. Any issues encountered can be addressed, ensuring a high-quality and professional application that meets the needs of its users.

4.5 Conclusion

In this chapter, we discussed the implementation details of the cargo tracking app. We reviewed the tools used in the implementation process, including the programming languages, frameworks, and databases utilized to build the application. We also provided a summary of the implementation process, detailing the various web pages and panels of the app, and how each one functions.

Additionally, we outlined the test cases and acceptance criteria for the cargo tracking app, ensuring that every operation is functioning correctly, and that the app meets the requirements and expectations of its users.

Overall, through the careful planning, execution, and testing of the cargo tracking app, we can ensure that it is a reliable and effective solution for tracking and managing cargo shipments. By implementing the appropriate tools and testing thoroughly, we can ensure that the app meets the needs of its users and provides a seamless experience.

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 Conclusion

The cargo tracking application project was initiated to address the complex challenges encountered by logistics providers and customers, including lack of transparency, inefficient processes, limited communication channels, security risks, stiff competition,

and the ever-increasing demand for services. The project's primary objective was to provide an innovative solution that leverages technology to offer real-time tracking, process automation, improved communication, enhanced security measures, a unique value proposition, and valuable data insights.

To achieve this goal, the project team employed various implementation tools, including state-of-the-art software development methodologies, programming languages such as JavaScript, PHP, HTML, CSS, Java, XML. Furthermore, the team developed comprehensive test cases and acceptance criteria to ensure the application meets the required standards.

Throughout the project's lifecycle, the team faced numerous challenges that required diligent collaboration, proper task management, and persistence to overcome. However, the team's unwavering dedication and commitment to excellence enabled them to deliver a top-of-the-line cargo tracking application that has the potential to revolutionize the logistics industry.

With its real-time tracking, process automation, improved communication channels, and enhanced security measures, the cargo tracking application is poised to optimize logistics providers' operations, reduce costs, increase revenue, and improve customer satisfaction. Furthermore, the application's valuable data insights can provide logistics providers with actionable insights that can inform decision-making and drive business growth.

In conclusion, the cargo tracking application project was a significant undertaking that required the team's full dedication and expertise to deliver a cutting-edge solution that can transform the logistics industry. The project's success is a testament to the team's proficiency in software development, project management, and problem-solving skills,

and it underscores the power of technology to address complex challenges faced by businesses and society.

5.2 Future Work

Expanding the work done in this project presents several exciting opportunities to enhance the functionality and capabilities of the cargo tracking application. One area that can be further developed is the strengthening of cybersecurity measures. Implementing standards such as ISO 28000, which provides guidelines for supply chain security management, and ISO 31000, which focuses on risk management, can bolster the application's security framework. By adhering to these standards, the application can ensure the confidentiality, integrity, and availability of sensitive information, safeguarding it from potential threats.

To improve the tracking capabilities of the application, integrating a GPS tracking system can be a valuable addition. This would enable real-time monitoring and precise location tracking of shipments, enhancing overall visibility and enabling more accurate delivery estimations. Additionally, incorporating a pricing system within the application can streamline the process of calculating costs for various services. Introducing pricing bundles and the ability to dynamically adjust prices based on different locations or factors can provide more flexibility and customization options for customers.

Furthermore, expanding the application's accessibility features can contribute to a broader user base. Adding accessibility options such as dark mode and translation capabilities can enhance usability for users with diverse preferences and language requirements. By making the application more dynamic, it can adapt to users' needs and preferences, further improving user satisfaction.

Another aspect of future work involves the implementation of additional features such as workers taking pictures of the order during various stages of the delivery process. This would provide visual updates to customers and increase transparency regarding the condition and location of their shipments. Additionally, incorporating a feature that allows IT personnel to modify user information, such as name and branch, can streamline administrative tasks and ensure accurate user profiles.

Scalability is another crucial aspect to consider for future development. Implementing a scalable architecture allows the application to handle increased user traffic and data volume efficiently. Separating individual customers from other companies and providing segregation based on different service types, such as cars and pickups, can enable better management and organization of data, ensuring a smoother user experience.

To facilitate communication and feedback, incorporating an email sending form within the application can provide a convenient means for customers to reach out to the support team or provide inquiries and suggestions. Additionally, integrating a built-in warning system for worker performance evaluation can help in maintaining high service standards. Automatically issuing warnings to workers based on predefined criteria, such as after three warnings, can streamline the process of identifying and addressing performance issues.

In conclusion, the future work for this cargo tracking application holds great potential for improvement and expansion. By focusing on areas such as cybersecurity, GPS tracking, pricing systems, accessibility features, scalability, data segregation, and communication enhancements, the application can evolve into a more robust, user-friendly, and efficient tool for managing and tracking shipments.

APPENDIX A: **IMPLEMENTATION DETAILS**

User Login and Dashboard:

Users can securely log in to their accounts using their unique username and password.

Upon successful login, users are directed to their personalized dashboard, which contains relevant information and actions specific to their role.

CEO and IT Access:

The CEO and IT personnel have privileged access to the system.

They can manage website admin accounts, including adding and removing admins as required.

The IT personnel have the authority to change user passwords.

The CEO can access comprehensive company statistics, such as the number of deliveries, success rates, pending and failed deliveries, worker effectiveness, and customer satisfaction.

The CEO can check the most profitable and least profitable deliveries, providing valuable insights for decision-making.

Manager Role:

Branch managers have access to an exclusive table of orders specific to their branch.

They can add or remove workers within their branch and view messages sent by website admins.

The manager can access an admin report presenting the best-selling and least-selling orders in a structured table format, aiding in strategic planning.

The branch manager updates the status of orders as they progress through various stages, ensuring accurate tracking and management.

A comprehensive list of orders, including those placed by the same customer, is available to the branch manager for efficient order handling.

Customer Panel:

Customers can explore the website's services and browse customer ratings and reviews of products and the overall website experience without logging in.

To perform specific operations, such as placing an order, tracking orders, viewing order history, marking an order as failed, and providing feedback on the company, customers are required to log in.

New customers can create an account through a streamlined registration process to enjoy personalized services.

Logged-in customers can also send messages to website admins and submit their own ratings and reviews.

Worker Mobile App:

Workers, using the dedicated mobile app, can access order information specific to their branch. They can update the order status, marking it as delivered or dropped off at another branch, ensuring real-time tracking and accurate record-keeping.

APPENDIX B: USER MANUAL

The cargo tracking app provides a secure and personalized experience for each user, ensuring that only authorized individuals can access their respective dashboard pages. To log in, users must enter their unique username and password, guaranteeing the protection of their account information. Once logged in, users have various options at their disposal, including the ability to change their password, send, read, and delete messages, and log out whenever necessary.

For high-level personnel such as the CEO and IT staff, the app offers privileged access and advanced functionalities. The CEO can effectively manage website admin accounts, overseeing their creation and removal as required. They also have the authority to review comprehensive company statistics, empowering them to make informed decisions. The CEO can generate reports that provide valuable insights into the most and least profitable deliveries, enabling them to identify areas for improvement. Additionally, the IT staff possesses the capability to handle password changes, ensuring the security of user accounts. They are also responsible for managing the workforce, including hiring or terminating workers, and can add new branches to the system, expanding the app's reach and capabilities.

Branch managers, on the other hand, enjoy exclusive access to an entire table of orders specific to their branch. This comprehensive view allows them to efficiently manage the workflow, assign tasks to workers, and monitor progress. They can also stay updated on important messages from website administrators, ensuring effective communication within the organization. Furthermore, branch managers have access to reports that highlight the best-selling and least-selling products, enabling them to strategize and make data-driven decisions. With the ability to update order statuses, they play a crucial role in maintaining accurate and up-to-date information.

Customers have the flexibility to explore services, view product ratings, and read website reviews without the need to log in. However, by logging in, they unlock additional functionalities that enhance their overall experience. Logged-in customers can easily place orders, track the status of their shipments in real-time, and access their complete order history. They also have the option to mark orders as failed if necessary, allowing for efficient problem resolution. Furthermore, customers can provide feedback by rating the company, contributing to the continuous improvement of services. Creating an account is a seamless process, enabling new users to quickly register and benefit from the app's full range of features.

The worker mobile app caters specifically to employees involved in the delivery process. Through the app, workers can conveniently access order information within their assigned branch, ensuring seamless coordination and streamlined operations. They can view detailed order instructions, track progress, and update the status of each order in real-time. This real-time tracking capability allows for efficient communication with other team members and minimizes delays or errors. By utilizing the worker mobile app, employees contribute to accurate record-keeping and provide customers with up-to-date information on their deliveries.

In summary, the cargo tracking app offers a comprehensive and user-friendly solution for various stakeholders involved in the logistics and delivery process. It ensures secure access to

personalized dashboards, empowering CEOs, IT personnel, branch managers, customers, and workers with role-based functionalities tailored to their specific needs. The app promotes efficient communication, data-driven decision-making, and seamless order management, ultimately enhancing the overall user experience and facilitating effective cargo tracking and management.

APPENDIX C: **DEPLOYMENT AND CONFIGURATION MANUAL**

Deployment and Configuration Details:

1. Hosting Platform: Render

- Choose Render as the hosting platform for the cargo tracking application.
- Sign up for a Render account and create a new project.

2. Server Configuration:

- Select the appropriate server configuration based on the application's requirements (e.g., CPU, memory, disk space).
- Specify the desired region for hosting the application servers.

3. Version Control and Deployment:

- Connect the application's source code repository (e.g., Git) to Render.
- Configure the deployment settings to automatically build and deploy the application whenever changes are pushed to the repository.

4. Environment Variables:

- Define and configure the necessary environment variables for the application.
- These variables may include database credentials, API keys, and any other sensitive information required for the application to function.

5. Database Setup:

- Choose a suitable database solution for the cargo tracking application (e.g., MySQL, PostgreSQL).
- Configure the database connection settings, including the host, port, username, and password.
- Import the database schema and initial data if needed.

6. SSL Certificate:

- Enable SSL certificate provisioning to ensure secure communication between the application and users.
- Configure the SSL settings in Render to enable HTTPS access.

7. Domain Setup:

- Map a custom domain to the deployed application.
- Configure DNS settings to point the domain to the Render servers.

Troubleshooting Techniques:

1. Application Logs:

- Access the application logs provided by Render to diagnose any errors or issues.
- Analyse the logs for any error messages or exceptions that may indicate the source of the problem.

2. Environment Variable Validation:

- Double-check the correctness and completeness of the environment variables.
- Ensure that the required variables are properly set and contain the correct values.

3. Database Connection:

- Verify the database connection settings, including the host, port, username, and password.
- Confirm that the database server is running and accessible from the application servers.

4. SSL Certificate Issues:

- If encountering SSL certificate-related problems, ensure that the certificate is correctly provisioned and configured.
- Check the certificate expiration date and validity.

5. Dependency Management:

- If the application relies on external dependencies or libraries, verify that they are correctly installed and configured.
- Update the dependencies to their latest versions if needed.

6. Network and Firewall:

- Check for any network or firewall configurations that might block or restrict access to the application.
- Ensure that the necessary ports are open and accessible.

7. Render Documentation and Support:

- Consult the Render documentation and knowledge base for specific troubleshooting guides and resources.
- Reach out to Render's support team for assistance with any complex or persistent issues.

By following the outlined deployment steps and utilizing effective troubleshooting techniques, we can ensure a smooth and successful deployment of the application on Render.

REFERENCES

- [1] X. Y. a. X. C. J. Huang, "E-commerce logistics in China: current status, issues and future prospects," in *Journal of Industrial Engineering and Management Science*, 2019.
- [2] L. Wang, "The Impact of the Internet on Logistics Management," *Journal of Applied Sciences*, [Online]. Available: <https://thescipub.com/pdf/10.3844/jassp.2014.1507.1514>. [Accessed 2023].
- [3] M. Nuce, "The Evolution of Logistics: What's Changed Since the 90s?," *Inbound Logistics*, [Online]. Available: <https://www.inboundlogistics.com/cms/article/the-evolution-of-logistics-whats-changed-since-the-90s/>. [Accessed 2023].
- [4] K. Page, "The Impact of Technology on Logistics and Supply Chain Management," *Google Books*, [Online]. Available: <https://books.google.com/books?id=J0fGDwAAQBAJ&printsec=frontcover#v=onepage&q&f=false>. [Accessed 2023].
- [5] [Online]. Available: <https://pixabay.com/images/search/shipment/>.
- [6] S. T. a. H. Huang, "The Role of Internet in Logistics: A Case Study," in *Journal of Industrial Engineering and Management Science*, 2019.
- [7] Y. Z. a. Z. Yang, "The Impact of Internet on Logistics Management: A Literature Review," in *International Journal of Trade, Economics and Finance*, 2017.
- [8] "Wakilni," Wakilni, [Online]. Available: <https://wakilni.com/>. [Accessed 2023].
- [9] "aramex solutions and services," Aramex, [Online]. Available: https://www.aramex.com/lb/en/aramex-home?utm_source=google&utm_medium=cpc&utm_campaign=AMX-CONV-WW-SC

IP-

20210506&utm_term=general&gclid=CjwKCAiA3pugBhAwEiwAWFzwdTlc3N3GRc
FIX8FkntH0epGAOly105NUqFkEGTn-Opi57y01-2TpjRoC5mIQAvD_BwE. [Accessed
2023].

[10] "Go share," GoShare, [Online]. Available: <https://goshare.co/>. [Accessed 2023].

[11] "techttarget," techttarget, [Online]. Available:

<https://www.techttarget.com/searchcontentmanagement/definition/content-management-system-CMS>. [Accessed 2023].

[12] "TechnoPedia," TechnoPedia, [Online]. Available:

<https://www.techopedia.com/definition/5435/graphical-user-interface-gui>. [Accessed 2023].

[13] V. d. H. J., "Value sensitive design and responsible innovation," in *Information and Computer Ethics*, Springer, Dordrecht., Delft University of Technology, 2013, pp. 67-83.

[14] R. A. P. D. & M. Y. Calvo, "Ethical design in emerging technologies," in *Frontiers in Digital Humanities*, 6, 2019, pp. 1-7.

[15] EPCglobal Inc., "EPC Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860 MHz - 960 MHz," EPCglobal Inc., 2015. [Online]. Available: https://www.gs1.org/docs/epc/Gen2_Protocol_Standard.pdf. [Accessed 2023].

[16] EDI Alliance, "EDI Standards Overview," EDI Alliance, 2023. [Online]. Available: <https://www.edibasics.com/what-is-edi/>. [Accessed 2023].

- [17] ProgrammableWeb, "API Basics: What is an API and How Does it Work?," ProgrammableWeb, 2013. [Online]. Available: <https://www.mulesoft.com/programmableweb>. [Accessed 2023].
- [18] U.S. Department of Defense, "GPS Standard Positioning Service Signal Specification," U.S. Department of Defense, 2014. [Online]. Available: <https://www.gps.gov/technical/ps/SPS-performance-standard.pdf>. [Accessed 2023].
- [19] IETF, "Internet Protocol, Version 4 (IPv4) Specification," IETF, September 1981. [Online]. Available: <https://tools.ietf.org/html/rfc791>. [Accessed 2023].
- [20] T. & R. E. Dierks, "The Transport Layer Security (TLS) Protocol Version 1.2. IETF RFC 5246," 2008. [Online]. Available: <https://tools.ietf.org/html/rfc5246>. [Accessed 2023].
- [21] PHP-FIG, "PHP: Hypertext Preprocessor. (n.d.). PHP Standards Recommendations.," PHP-FIG, 2023. [Online]. Available: <https://www.php-fig.org/psr/>. [Accessed 2023].
- [22] Ecma International, " ECMAScript Language Specification," Ecma International, 2020. [Online]. Available: <https://www.w3.org/TR/html51/>. [Accessed 2023].
- [23] World Wide Web Consortium, "HTML5.1, Second Edition.," World Wide Web Consortium, 2017. [Online]. Available: <https://www.w3.org/TR/html51/>. [Accessed 2023].
- [24] ISO/IEC, "Information Technology - Database Languages - SQL.," ISO/IEC, 2016. [Online]. Available: <https://www.iso.org/standard/63555.html>. [Accessed 2023].
- [25] International Organization for Standardization., "ISO 28000:2007 - Specification for security management systems for the supply chain," International Organization for

Standardization., 2007 . [Online]. Available: <https://www.iso.org/standard/44706.html>. [Accessed 2023].

[26] International Organization for Standardization, "ISO 31000:2018 - Risk management - Guidelines," International Organization for Standardization, 2018. [Online]. Available: <https://www.iso.org/standard/65694.html>. [Accessed 2023].

[27] Mozilla Developer Network, "Transport Layer Security (TLS) and its predecessor, Secure Sockets Layer (SSL)," Mozilla Developer Network, 1999. [Online]. Available: https://developer.mozilla.org/en-US/docs/Web/Security/Transport_Layer_Security. [Accessed 2023].

[28] C. Sandvig, "The internet as infrastructure," in *Communities and technologies* , Springer, Cham., 2014, pp. 245-254.