AN ENHANCED LOGISTICS SYSTEM FOR THE ELECTRICAL APPLIANCE INDUSTRY

WEB-BASED PLATFORM FOR INTELLIGENT DELIVERY SCHEDULING WITH INSTALLATION TIME ESTIMATION, RESIDENTIAL ZONE COMPLIANCE, ROUTE AND TRUCK SPACE OPTIMIZATION

CHEW JIA HUI

FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY
UNIVERSITY OF MALAYA
KUALA LUMPUR

2025

AN ENHANCED LOGISTICS SYSTEM FOR THE ELECTRICAL APPLIANCE INDUSTRY

WEB-BASED PLATFORM FOR INTELLIGENT DELIVERY SCHEDULING WITH INSTALLATION TIME ESTIMATION, RESIDENTIAL ZONE COMPLIANCE, ROUTE AND TRUCK SPACE OPTIMIZATION

CHEW JIA HUI

PROJECT REPORT SUBMITTED IN FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF BACHELOR OF COMPUTER SCIENCE (SOFTWARE ENGINEERING)

FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY
UNIVERSITY OF MALAYA
KUALA LUMPUR

2025

UNIVERSITY OF MALAYA

ORIGINAL LITERARY WORK DECLARATION

Name of Candidate: Chew Jia Hui (I.C/Passport No: 030707-01-0032)

Matric No: 22004739/1

Name of Degree: Bachelor of Computer Science (Software Engineering)

Title of Project Paper/Research Report/Dissertation/Thesis ("this Work"):

An Enhanced Logistics System for the Electrical Appliance Industry

Field of Study:

Web-Based Platform for Intelligent Delivery Scheduling with Installation Time Estimation, Residential Zone Compliance, Route and Truck Space Optimization

I do solemnly and sincerely declare that:

(1) I am the sole author/writer of this Work;
(2) This Work is original;
(3) Any use of any work in which copyright exists was done by way of fair dealing and for permitted purposes and any excerpt or extract from, or reference to or reproduction of any copyright work has been disclosed expressly and sufficiently and the title of the Work and its authorship have been acknowledged in this Work;
(4) I do not have any actual knowledge nor do I ought reasonably to know that the making of this work constitutes an infringement of any copyright work;
(5) I hereby assign all and every rights in the copyright to this Work to the University of Malaya ("UM"), who henceforth shall be owner of the copyright in this Work and that any reproduction or use in any form or by any means whatsoever is prohibited without the written consent of UM having been first had and obtained;
(6) I am fully aware that if in the course of making this Work I have infringed any copyright whether intentionally or otherwise, I may be subject to legal action or any other action as may be determined by UM.

Candidate's Signature  Date: 3rd July 2025

Subscribed and solemnly declared before,

Witness's Signature Date:

Name:

Designation:

AN ENHANCED LOGISTICS SYSTEM FOR THE ELECTRICAL APPLIANCE INDUSTRY

ABSTRACT

This project presents the development of an enhanced logistics system designed specifically for the electrical appliance industry. The system addresses operational inefficiencies in the current delivery and installation workflow by introducing a web-based platform capable of intelligent scheduling and dynamic rescheduling. Key innovations include installation time estimation based on product types, residential zone compliance filtering, delivery route optimization and truck space utilization planning. The system also features role-specific modules for admins, delivery teams, outsources installers and warehouse loading staff to streamline communication and improve coordination. Data was gathered through stakeholder meetings, existing system analysis and literature reviews, which informed the development of five modules. The system was implemented using ReactJS and Firebase and validated through stakeholder testing sessions. Results demonstrate improved scheduling precision, reduced delivery disruptions and better adaptability to real-world challenges in appliance logistics.

Keywords: intelligent scheduling, route optimization, installation estimation, logistics system, web-based platform

SISTEM LOGISTIK YANG DIPERTINGKAT UNTUK INDUSTRI PERALATAN ELEKTRIK

ABSTRAK

Projek ini memperkenalkan pembangunan sistem logistik yang dipertingkatkan untuk industri peralatan elektrik. Sistem ini menangani ketidakcekapan operasi dalam proses penghantaran dan pemasangan semasa dengan memperkenalkan platform berasaskan web yang mampu menjadual secara pintar, menjejak masa nyata, dan menjadual semula secara dinamik. Inovasi utama termasuk anggaran masa pemasangan berdasarkan jenis produk, penapisan pematuhan zon kediaman, pengoptimuman laluan penghantaran, dan perancangan penggunaan ruang lori. Sistem ini turut menyediakan modul khusus mengikut peranan untuk pentadbir, pasukan penghantaran, juruteknik pemasangan dan staf gudang bagi memudahkan komunikasi dan penyelarasan. Data diperoleh melalui mesyuarat pihak berkepentingan, analisis platform, dan kajian empirikal yang membentuk asas pembangunan empat modul pintar. Sistem ini dibangunkan menggunakan ReactJS dan Firebase serta disahkan melalui sesi ujian dengan pihak berkepentingan. Hasilnya menunjukkan peningkatan dalam ketepatan penjadualan, pengurangan gangguan penghantaran, dan keupayaan sistem menyesuaikan diri dengan cabaran dunia sebenar dalam logistik peralatan elektrik.

Kata kunci: penjadualan pintar, pengoptimuman laluan, anggaran pemasangan, sistem logistik, platform berasaskan web

ACKNOWLEDGEMENTS

First, I would like to sincerely thank Assoc. Prof. Dr. Chiew Thiam Kian, my project supervisor, for his continuous support and guidance throughout this first phase of the Final Year Project. His advice and feedback have really helped me stay on track and better understand how to manage each stage of the project. I am truly grateful for the time and effort he has put into mentoring us.

I would also like to thank Mr. Tan Wai Kiat from TBM Electrical for his useful feedback and suggestions during our discussions. His input gave us a clearer understanding of the real challenges faced in the industry and that helped us improve our system planning and development.

A big thank you as well to my teammate, Wong Wen Hao, for being a reliable and hardworking partner throughout this phase. His effort and dedication played a huge role in making sure we met our development goals, and I have really enjoyed working with him on this project.

I also appreciate the helpful comments and suggestions given by our panels, Dr. Chiam Yin Kia and Dr. Adeleh Asemi Zavareh during the viva and monitoring sessions. Their feedback has been valuable and we will apply it to improve our work in the second phase of the project.

Lastly, thank my friends and family for all the encouragement and support throughout this semester. It has been a tough one, but their presence made it more manageable.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS AND ABBREVIATIONS

RUP	:	Rational Unified Process
UI/UX	:	User Interface / User Experience
ETA	:	Estimated Time of Arrival
UAT	:	User Acceptance Testing
TBM	:	Tan Boon Ming Electrical Sdn Bhd (project stakeholder)
VRP	:	Vehicle Routing Problem
3L-CVRP	:	Three-Dimensional Loading Capacitated Vehicle Routing Problem
CVRP	:	Capacitated Vehicle Routing Problem
DVRP	:	Dynamic Vehicle Routing Problem
HVRP	:	Heterogeneous Vehicle Routing Problem
AI	:	Artificial Intelligence
GA	:	Genetic Algorithm
IWOA-HMOHA	:	Improved Whale Optimization Algorithm with Hybrid Multi-Stage Heuristics Approach
IACO	:	Improved Ant Colony Optimization
GELS	:	Gravitational Emulation Local Search
MLP	:	Multi-Layer Perceptron
CRM	:	Customer Relationship Management
ERP	:	Enterprise Resource Planning
GPS	:	Global Positioning System
IoT	:	Internet of Things

LIST OF APPENDICES

CHAPTER 1: INTRODUCTION

1.1 Overview

This project aims to develop an integrated web-based platform and mobile app system that enhances the delivery and installation of electrical appliances. It replaces the current static, manual scheduling process with an intelligent platform. Key features include automated estimation of installation time, filtering of time slots based on residential zone compliance, delivery route and truck space optimization, and dynamic rescheduling to handle real-world disruptions. These capabilities ensure better planning accuracy, reduce delays and improve overall operational efficiency.

In addition to intelligent scheduling, the system supports real-time communication between customers, delivery teams and installation teams. Customers receive timely updates and can track the live location of assigned teams, improving transparency and trust. Each user role, including admins, delivery and installation teams, outsourced installers, and warehouse staff, has access to role-specific interfaces to manage tasks efficiently. Together, these features streamline coordination and significantly improve the end-to-end service experience.

1.2 Problem Statement

Currently, the scheduling process for delivering and installing electrical appliances relies on static time ranges, without intelligent features to estimate the actual time required for each task. **Different types of installations require varying amounts of time**; for example, simple unboxing is quick, while a more complex installation, such as drilling or plumbing, takes significantly longer. However, both are often assigned the same time slots, which leads to poor planning and wasted time. On top of that, many deliveries are affected by **residential zone constraints**, which vary based on building types, such as gated communities, high-rise condominiums and landed housing. These constraints include restricted access hours, noise regulations and the requirement for entry permits or security clearance. These rules are usually not considered when scheduling, which can cause delays when teams are denied entry or need to wait for approval. **Delivery teams are also left to plan their own routes**, which often results in unnecessary travel and longer delivery times. Another issue is **the lack of a system to help plan how appliances should be arranged in the delivery truck**. Without proper information on product sizes and dimensions, the truck space is not used efficiently, which sometimes leads to more trips than necessary.

The situation becomes even more complicated when unexpected issues arise, such as a **last-minute change in team availability or a sudden emergency**. At the moment, there is no system in place that can quickly adjust the schedule or automatically reassign tasks. Everything has to be done manually by the planners, which takes time and increases the chance of mistakes. When rescheduling is needed, it also does not include re-evaluating the delivery route, considering housing access requirements, or reorganizing the truck load. These disruptions not only affect delivery performance but also reduce customer satisfaction. A smarter, more automated scheduling and rescheduling system is needed, one that can efficiently handle real-world challenges and help the company provide better and more reliable service.

1.3 Project Objectives

The main objective of this project is **to optimize the end-to-end delivery and installation process for electrical appliances, aiming to enhance communication and coordination across teams and customers**. The objectives listed below are specifically focused on the system components under my responsibility, which contribute to achieving the overall project goal.

1.3.1 To develop an intelligent scheduling system that considers installation time estimation, residential zone compliance, route and truck space optimization

This objective focuses on improving the current static scheduling process by creating a smarter system that understands the real demands of each delivery. It will take into account the actual time needed for different types of installations, from simple unboxing to complex setups involving plumbing or drilling. The system will also take into account residential zone constraints, such as access rules for condominiums, noise restrictions and entry permit requirements based on building types when assigning suitable delivery and installation slots. In addition, it will help delivery teams plan efficient routes and make better use of lorry space by considering the size and arrangement of appliances. The goal is to reduce wasted time, minimize unnecessary travel and make each delivery smoother and more predictable.

1.3.2 To implement a dynamic rescheduling function that can respond quickly to last-minute changes and disruptions

In the event of emergencies or sudden changes in team availability, admins currently must manually adjust schedules, which is time-consuming and increases the risk of errors. This objective is to develop a system that can automatically reassign delivery time slots, update delivery routes and reorganize truck loads in real time. During rescheduling, the system will continue to factor in residential zone constraints, such as building type-specific access rules, noise restrictions and permit requirements to ensure minimal service disruption and maintain a high level of customer satisfaction.

1.3.3 To build a web-based platform that integrates scheduling, routing, truck loading and residential zone compliance into a centralized system

This objective involves building an all-in-one online platform where admins, delivery and installation teams, outsourced installers and warehouse loading teams can view, manage and update all aspects of their work. The web-based system will provide a clear overview of the schedule, delivery routes, product dimensions and access requirements. It will help different teams stay coordinated, improve communication and allow faster adjustments when needed.

1.3.4 To conduct user acceptance testing (UAT) with admins and field teams to validate system usability, practicality and its ability to handle real-world delivery challenges

This objective is to make sure the system truly works for the people who use it every day. By involving actual admins and field teams in testing, we can see how the system performs in real delivery situations. They will use the system to schedule, reschedule, plan routes and organize truck loads while dealing with real constraints such as access restrictions, building types and emergency scenarios. Their feedback will help us improve the system so that it is not only functional, but also practical, easy to use and genuinely helpful in making daily operations smoother and more efficient.

1.4 Expected Outcomes

This project will deliver a web-based intelligent system designed to improve how electrical appliance deliveries and installations are scheduled, managed and adjusted in real time. The system will be made up of several interconnected modules, each addressing specific needs based on real industry challenges. The overall system architecture and its core modules are illustrated in Figure 1.1.

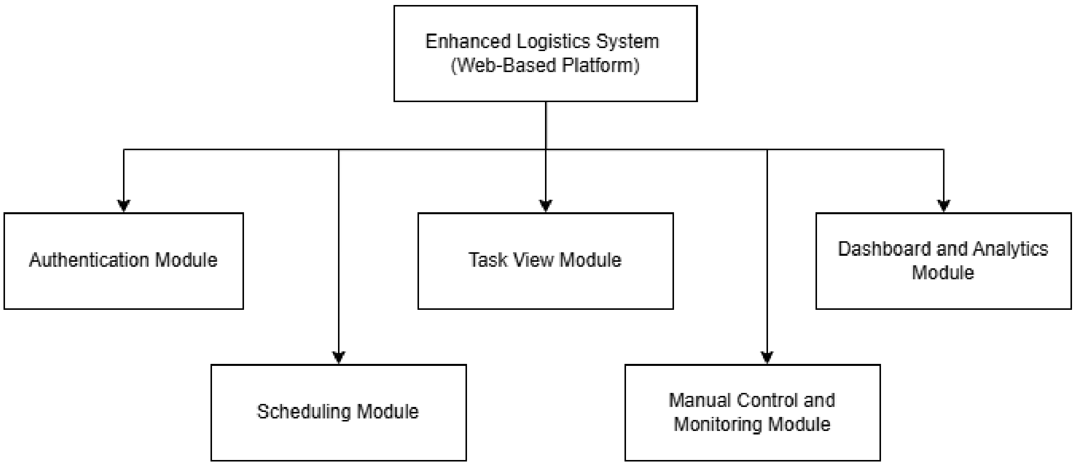

Figure 1.1: Web-Based Platform Modules

1.5 Project Scope

1.5.1 Target Users

1.5.1.1 Admins

The admins are responsible for overseeing the overall scheduling operations, resolving user-submitted reports and monitoring system performance. They rely on the system to manage delivery and installation schedules, reassign teams when necessary and ensure that services run smoothly. Admins are also in charge of evaluating operational performance, using analytics dashboards to track both employee task completion and order delivery metrics. Their role is critical to maintaining service quality, operational visibility and timely intervention when issues arise.

1.5.1.2 Delivery and Installation Teams

The delivery teams use the system to view their assigned delivery schedules and the recommended delivery routes. In addition to delivering electrical appliances, they are also responsible for performing simple installations, such as basic setup or unboxing tasks. Their ability to access updated plans ensures that deliveries and installations are carried out efficiently and on time. With clear schedules and route suggestions, delivery teams can reduce unnecessary travel, stay on track with their tasks and improve service punctuality, all of which contribute directly to higher customer satisfaction.

1.5.1.3 Outsourced Installers

The outsourced installers are provided with access to their assigned installation schedules. This ensures that they are informed in advance about their tasks and can coordinate effectively with the delivery team and customers. Timely and accurate scheduling supports better planning and minimizes delays in completing appliance installations.

1.5.1.4 Warehouse Loading Team

The warehouse loading team depends on the system to view the loading schedule for each truck. This allows them to prepare and arrange appliances efficiently, ensuring that trucks are loaded based on space availability and delivery sequence. Clear loading instructions help avoid mistakes and reduce the risk of overloading, which supports smoother delivery execution.

1.5.2 System Scope

1.5.2.1 Authentication Module

This module provides essential access control features for the system. It supports user registration, account login, password reset and secure logout functionalities. These core features ensure that only authorized users can access the system, maintain data confidentiality and enable a personalized experience for each user role.

1.5.2.2 Scheduling Module

The Scheduling Module is the backbone of the system's intelligent planning capabilities. It handles the estimation of installation durations based on item complexity and quantity, filters delivery time slots by residential zone access rules and optimizes delivery sequences and truck capacity. Additionally, it assigns the most suitable delivery time windows based on operational constraints and automatically reschedules tasks when the original time slot is rejected or becomes unavailable. This module ensures that delivery and installation plans are both efficient and compliant with real-world constraints.

1.5.2.3 Task View Module

The Task View Module enables operational teams, specifically the warehouse loading team, delivery teams and outsourced installers to access task-specific schedules. Delivery teams can view their assignments and optimized route suggestions, while outsourced installers receive updated installation tasks. Warehouse loading staff can review truck loading plans. This module plays a crucial role in keeping all field teams aligned and well-informed about their responsibilities.

1.5.2.4 Manual Control and Monitoring Module

The Manual Control and Monitoring Module provides admins with the ability to intervene when necessary. They can view and manually edit delivery schedules and team assignments to address changes or unforeseen issues. The module also allows admins to resolve reports submitted by users, ensuring that problems encountered during the process are addressed promptly and appropriately.

1.5.2.5 Dashboard and Analytics Module

This module equips admins with insights into system performance. It displays employee performance metrics, such as task completion counts and punctuality, and also monitors order delivery performance, including delivery timeliness and completion status. This module supports data-driven decision-making and continuous improvement across operations.

1.5.3 Job Distribution

Figure 1.2 illustrates the job distribution within the project system, showing how development tasks are evenly divided between team members. Each module is colorcoded to indicate responsibility, with Wong Wen Hao primarily handling the mobile application system (in green) and Chew Jia Hui focusing on the web-based platform (in purple). This visual clearly outlines the allocation of work between both members.

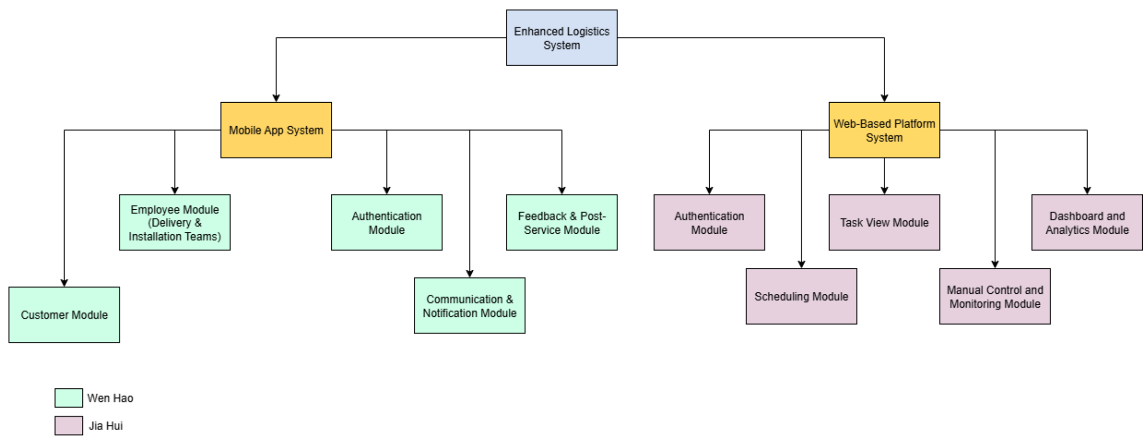

Figure 1.2: Job Distribution

1.6 Project Schedule

Table 1.1 and Table 1.2 outline the planned activities and estimated timeline for each phase. Phase I focuses on background study, problem analysis, requirements gathering and early-stage prototyping. Phase II covers detailed design, full system development, testing and final deployment.

A more detailed month-by-month Gantt-style project timeline, including task breakdowns and progress tracking, is provided in Appendix B for reference.

Table 1.1: Phase I Schedule

ID	Task	Mar 2025	Apr 2025	May 2025	June 2025
1	Background Study				
2	Problem Analysis and Literature Review				
3	Requirements Elicitation and Analysis				
4	Tools Selecting and Mastering				
5	Preliminary System Design				
6	Initial Prototype Development				
7	Documentation				

Table 1.2: Phase II Schedule

ID	Task	Jul 2025	Aug 2025	Sep 2025	Oct 2025	Nov 2025	Dec 2025
1	Detailed system design						
2	System development						
3	Internal system testing and evolution						
4	User testing						
5	Deployment and demonstration						
6	Documentation						

1.7 Project Repository

We are using GitHub as a version control software to keep track of changes and manage our source code.

1.8 Report Structure

Chapter 1: Introduction provides the project overview, problem statement, project objectives, expected outcomes, project scope, project schedule and project repository.

Chapter 2: Literature Review presents relevant literature sources as well as a comparison of similar systems which currently exist.

Chapter 3: Research Methodology discusses the software development methodology used as well as the technique used for requirements elicitation.

Chapter 4: Requirement Analysis discusses in detail about the functional and non-functional requirements elicited.

Chapter 5: System Design discusses the design of the enhanced logistics system with the aid of various UML diagrams.

Chapter 6: Conclusion consists of a short summary about the whole project.

CHAPTER 2: LITERATURE REVIEW

2.1 Existing Literature

2.1.1 Introduction

The logistics and transportation sectors are undergoing a profound transformation, largely driven by the twin forces of rapid urbanization and the explosive growth of e-commerce. As the global urban population is expected to increase by 2.5 billion by 2050, urban infrastructure, particularly road networks, will face significant challenges such as traffic congestion and delivery inefficiencies (Engesser et al., 2023). This is compounded by rising consumer expectations for faster delivery options, including next-day and even same-day shipping services. These evolving consumer demands place additional pressure on logistics providers to refine delivery scheduling, routing and overall operational efficiency.

The emergence of Industry 4.0 has accelerated the integration of advanced digital technologies, such as cyber-physical systems, into logistics operations. These technologies have enabled more accurate urban modelling and enhanced delivery prediction capabilities. However, despite technological advancements, urban logistics remains fraught with challenges, especially in the area of last-mile delivery. For instance, in densely populated urban centers like Paris, delivery trucks contribute to as much as 60% of vehicle-related emissions and play a major role in traffic congestion (Mohammad et al., 2023).

2.1.2 Challenges in Logistics Scheduling

2.1.2.1 Increasing Volume of Shipments

The rapid ascent of e-commerce platforms has resulted in a dramatic increase in parcel volumes worldwide. A clear example can be seen in Amazon’s delivery statistics in the U.S., where the number of deliveries nearly doubled from 3.5 billion in 2019 to an estimated 6.5 billion by 2022. This exponential growth significantly strains existing logistics infrastructure, including delivery fleets, personnel and technological systems, and highlights the need for scalable and resilient logistics scheduling frameworks (Mohammad et al., 2023).

2.1.2.2 Sustainability

As parcel volumes increase, so does the environmental impact of transportation operations. The expansion of delivery fleets needed to handle growing shipment volumes contributes to air pollution, increased carbon emissions, noise and road wear. These factors negatively affect urban living standards and infrastructure longevity. Societal awareness of environmental sustainability, coupled with government regulations, has increased the demand for greener logistics systems. One promising solution involves self-collection lockers, where the installation of 60 units can potentially reduce up to 193 tons of carbon emissions annually, offering both environmental and operational benefits (Mohammad et al., 2023).

2.1.2.3 Complexity of Scheduling Operations

Modern logistics operations are increasingly complex due to unstable customer demand, fluctuating fuel costs, globalized supply chains and mounting expectations for speed and accuracy. Traditional logistics systems, which often rely on static planning templates, lack the agility and adaptability required in today's fast-paced environment. This necessitates the adoption of intelligent, data-driven scheduling systems that can adapt to real-time variables and uncertainties. The challenge lies in transitioning from reactive systems to predictive, proactive and optimized frameworks (Royappa et al., 2024).

2.1.2.4 Delivery Costs and Operational Inefficiencies

Delivery operations in urban areas are severely affected by factors such as traffic congestion, limited parking, inaccurate or incomplete customer addresses and uncertain customer availability. These issues result in increased operational costs and reduced delivery reliability. Notably, first-attempt delivery failure rates vary widely—from 12% to as high as 60%—due to lifestyle factors, such as the growing number of single-occupant households and the increase in dual-income families, where no one is home to receive deliveries (Mohammad et al., 2023).

2.1.2.5 Time Pressure

The rise in e-commerce has resulted in time-sensitive delivery systems where logistics providers must meet next-day or even same-day delivery commitments. These demands increase the stress on last-mile operations, which already deal with fluctuating parcel volumes. Volume typically peaks on Mondays and during seasonal events such as holiday sales. This time pressure introduces stochastic workloads and makes scheduling far more difficult, requiring highly dynamic and responsive systems (Mohammad et al., 2023).

2.1.2.6 Labour Shortages and Ageing Workforce

Another significant concern is the shortage of skilled delivery personnel, especially in urban areas. Parcel delivery is physically demanding and an ageing workforce further complicates recruitment and retention. While the integration of new technologies, such as parcel lockers, autonomous vehicles and robotic delivery systems, shows promise, human interaction still plays a key role in last-mile delivery, particularly in maintaining customer satisfaction and trust (Mohammad et al., 2023).

2.1.3 Vehicle Routing Problems (VRP) and Their Variants

The Vehicle Routing Problem (VRP) is a foundational topic in logistics and operations research. It involves determining the most efficient set of routes for a fleet of vehicles to deliver goods to a number of geographically dispersed customers. The problem is NP-hard and its solution becomes exponentially more complex as the number of customers increases. The objective typically includes minimizing total travel distance, cost, or delivery time while satisfying constraints such as vehicle capacity, customer demands and service time windows (Mageswari, 2024).

2.1.3.1 Classic Vehicle Routing Problem (VRP)

The classical VRP assumes a fleet of identical vehicles that start and end at a single depot, serving all customers exactly once. The key constraints include vehicle capacity and route feasibility. It forms the basis for many extensions and specialized models. Given its NP-hard nature, exact algorithms are rarely used in large-scale settings, which has led to the development of heuristic and metaheuristic approaches for obtaining near-optimal solutions efficiently (Mageswari, 2024).

2.1.3.2 Capacitated Vehicle Routing Problem (CVRP)

The Capacitated Vehicle Routing Problem (CVRP) introduces vehicle load limits to the standard VRP model. Each customer has a specific demand and each vehicle must not exceed its maximum carrying capacity. The goal is to construct efficient delivery routes that ensure all customers are served without violating vehicle constraints (Ibrahim et al., 2020).

However, capacity constraints pose significant challenges during optimization. As the number and complexity of constraints increase, the search space narrows, raising the likelihood of converging on local optima and reducing the performance of standard solution techniques (Zhang et al., 2021).

2.1.3.3 Heterogeneous Vehicle Routing Problem (HVRP)

The Heterogeneous Vehicle Routing Problem (HVRP) considers a fleet of vehicles with varying capacities, fixed costs and operating expenses. This variation better reflects real-world logistics, especially in last-mile delivery, where a mix of vehicle types is used based on accessibility, load requirements, or cost considerations (Ibrahim et al., 2020).

Additional variants under the HVRP umbrella include the Load-Specific Capacity VRP, where only certain vehicle types can carry specific kinds of goods, such as temperature-sensitive or fragile products. Multi-compartment vehicles that segregate loads by type are a common example (Simeonova et al., 2018).

An advanced variant, the Heterogeneous Vehicle Routing Problem with Time Windows and Limited Resources (HVRPTW-LR), introduces additional operational constraints that further align the model with real-world delivery scenarios (Molina et al., 2020). In this formulation, each vehicle is assigned to exactly one route, which must begin and end at a central depot. The total demand along a route must not exceed the assigned vehicle's capacity, and each customer if served must be visited only once, with their full demand fulfilled by a single vehicle. Importantly, not all customers need to be served in a given planning cycle, allowing for flexibility under resource constraints.

2.1.3.4 Dynamic Vehicle Routing Problems (DVRPs)

The Dynamic Vehicle Routing Problem (DVRP) handles scenarios in which crucial information—such as customer demand, travel times, or traffic conditions—changes in real time. DVRP systems must adapt routing decisions dynamically as new data becomes available (Zhang et al., 2021).

Modern technologies such as IoT, GPS and AI allow for real-time traffic and customer updates, enabling logistics systems to revise routes dynamically for improved efficiency and responsiveness. These features are critical in urban logistics, where disruptions such as congestion or last-minute customer requests are common (Zhang et al., 2021).

Additionally, DVRP accommodates planning scenarios where customer requests are received during the route execution phase. The system must repeatedly solve the routing problem, incorporate new constraints and adjust existing plans in real time (Lahyani et al., 2015).

2.1.3.5 Vehicle Routing Problems Over Time

In VRP models that span over time, vehicles are expected to perform multiple routes either within the same day or across several days. Here, the planning horizon is discretized into defined periods (e.g., daily) and the timing of each route becomes a critical decision variable.

These models determine not only which customers are visited and in what sequence, but also when each visit should occur. If a vehicle performs multiple routes in a single day, the start time of a new route must consider the return time from the previous one. Such formulations are especially useful for long-term planning and multi-shift logistics operations (Mor & Speranza, 2022).

2.1.3.6 Vehicle Routing Problem with Time Windows (VRPTW)

The Vehicle Routing Problem with Time Windows (VRPTW) adds an additional temporal constraint to the classical VRP. Each customer must be served within a designated time window. Introduced by Solomon in 1987, VRPTW has been applied in contexts such as ride-sharing, grocery delivery, home healthcare and Just-In-Time manufacturing (Liu et al., 2023). The problem often aims to minimize the number of vehicles used and total travel distance while ensuring deliveries occur within the specified time frames (Arora et al., 2025).

2.1.3.7 Three-Dimensional Loading Capacitated Vehicle Routing Problem (3L-CVRP)

The 3L-CVRP consists of two interrelated subproblems: the Urban VRP—accounting for city traffic, congestion and delivery constraints—and the Three-Dimensional Bin Packing Problem (3DBPP), which ensures safe and efficient cargo placement (He et al., 2024).

The 3L-CVRP combines routed decisions with three-dimensional packing constraints. This variant is particularly relevant in industries like electrical meter distribution, where goods are fragile, high-value and irregularly shaped. The model integrates vehicle routing with 3D bin packing to ensure optimal space utilization and safe loading during transportation (He et al., 2024).

2.1.3.8 Multi-Depot Green Vehicle Routing Problem (MDGVRP)

The Multi-Depot Green Vehicle Routing Problem (MDGVRP) addresses both environmental and logistical concerns. Vehicles depart from multiple depots to serve customer locations, optimizing fuel consumption, emissions, travel distance and cost. This is particularly useful in multi-tier distribution networks where sustainability is a priority (Li et al., 2019).

2.1.3.9 Rich Vehicle Routing Problem (RVRP)

The Rich VRP extends the classical VRP by integrating multiple real-world constraints and decision-making layers. A notable example includes a routing model that handles heterogeneous fleets, multiple depots, long-haul transportation and outsourced last-mile delivery. Vehicles may originate from the company's own fleet or third-party providers, each governed by unique costs, capacities and driver work-hour regulations (Alcaraz et al., 2019).

One key constraint in this model is that each vehicle may visit only one subcontracted depot during the trip. The model also includes complex decisions such as determining who performs the last-mile delivery and how to allocate tasks between internal and outsourced resources (Alcaraz et al., 2019).

2.1.3.10 Stochastic VRP (SVRP)

The Stochastic VRP (SVRP) incorporates randomness into one or more problem parameters, such as customer demand, travel time, or service duration. These uncertainties are modelled using probability distributions, making it impossible to strictly follow a fixed route plan. Different types of SVRP include those with stochastic demands, stochastic customer arrivals and uncertain service times. The main objective is to minimize the expected total cost or distance while accounting for the probabilistic nature of key inputs (Berhan et al., 2014).

2.1.4 Solution Approaches to Logistics Scheduling

2.1.4.1 Traditional AI and Machine Learning Techniques

Artificial Intelligence (AI) has increasingly become integral in enhancing logistics scheduling. It offers predictive capabilities and decision-making support that traditional systems often lack. AI models such as Random Forests, Gradient Boosting and Multi-Layer Perceptrons (MLP) have been applied using historical logistics data to anticipate delivery durations, detect disruptions and forecast demand. Among these, MLP has shown the highest predictive accuracy, making it particularly effective for dynamic logistics environments (Royappa et al., 2024).

Beyond predictive modelling, AI has improved routing decisions by shifting from static to dynamic algorithms. Traditional static methods fail to accommodate live changes such as traffic congestion, weather, or customer schedule shifts. In contrast, AI-enabled systems process real-time data inputs and continuously optimize routes accordingly. This not only enhances efficiency but also strengthens the network's resilience to disruptions, significantly improving adaptability in logistics operations (Royappa et al., 2024).

2.1.4.2 Exact algorithms

Exact algorithms aim to generate optimal solutions by exhaustively searching the solution space, though they are typically only feasible for small to medium-sized instances due to computational limitations. One notable technique is the **Branch-and-Cut algorithm**, widely applied in solving the Travelling Salesman Problem (TSP) and its VRP extensions. This method iteratively solves subproblems and eliminates non-promising solutions through cutting planes. Ibrahim et al. (2020) implemented a full version of this algorithm, noting the extensive use of linear programming during the process.

Another powerful method is **Column Generation (CG)**, which is especially suitable for VRP variants with added constraints, such as the Vehicle Routing Problem with Time Windows (VRPTW). It works by solving a relaxed master problem and generating new columns (routes) iteratively until an optimal solution is found. Ibrahim et al. (2020) also applied CG to the Heterogeneous Fleet Vehicle Routing Problem (HVRP), highlighting its versatility.

Other notable exact approaches include Branch-and-Bound, Branch-and-Price, Constraint Programming and Dynamic Programming. While these methods ensure optimality, their high computational costs limit their application to smaller problem sizes (Tan & Yeh, 2021).

2.1.4.3 Heuristics algorithms

Heuristic algorithms provide approximate solutions quickly, making them valuable for larger and more complex routing problems. Constructive heuristics such as Nearest

Neighbour, Savings, Insertion and Sweep methods build feasible routes by following simplified, rule-based logic. For example, while Nearest Neighbour is intuitive and fast, it often yields suboptimal results under multiple constraints. In contrast, the Savings method balances simplicity and flexibility, proving useful in multi-depot and dynamic contexts (Liu et al., 2023).

Improvement heuristics refine initial solutions through iterative adjustments, using intra-route (within a single route) and inter-route (across routes) strategies. Operators such as Relocate, Exchange and λ-opt (e.g., 2-opt, 3-opt), or more advanced ones like CROSS and λ-interchange, are applied to improve route efficiency. Recently, the integration of machine learning to guide these local searches has further improved both solution quality and computational efficiency (Liu et al., 2023).

Although heuristics do not guarantee optimality, they offer significant speed advantages and are often used to generate strong initial solutions for more complex metaheuristic methods (Tan & Yeh, 2021).

2.1.4.4 Metaheuristics algorithms

Metaheuristics extend heuristic methods by employing more advanced search strategies to explore the solution space and escape local optima. **Genetic Algorithms** (GAs) are among the most widely applied, simulating biological evolution to improve routing solutions across generations. GAs encode routes as chromosomes and use operators like selection, crossover (e.g., Order Crossover, Partially Mapped Crossover), and mutation (e.g., swap, inversion) to iteratively generate better results. Mageswari (2024) reports that GAs have effectively solved VRP variants involving time windows, heterogeneous fleets and multiple depots.

Recent studies have emphasized the importance of meta-learning to select the most suitable meta-heuristic for solving specific VRP instances. Gutierrez-Rodríguez et al. (2019) proposed a meta-learning framework that characterizes each VRPTW instance using two sets of meta-features: basic meta-features (e.g., vehicle capacity, time windows, customer count) and landmarking meta-features derived from the performance of simple heuristics. A multilayer perceptron (MLP) classifier is then used to predict the best-performing algorithm for each instance. Their approach significantly improves solver selection accuracy and provides insights into why specific meta-heuristics outperform others based on problem instance characteristics.

For solving the complex 3L-CVRP, He et al. (2024) introduced a two-stage optimization approach combining mathematical modelling and metaheuristic algorithms. Their enhanced algorithm, IWOA-HMOHA (Improved Whale Optimization Algorithm with Hybrid Multi-Stage Heuristic Approach), incorporates:

- Adaptive inertia weight to balance exploration and exploitation
- Differential evolution to avoid premature convergence
- Population classification to improve convergence speed

In addressing the Multi-Depot Green VRP (MDGVRP), Li et al. (2019) proposed an Improved Ant Colony Optimization (IACO) algorithm. Inspired by the behaviour of ants, IACO uses pheromone trails and heuristic rules to construct efficient delivery routes. Unlike standard ACO, IACO introduces a modified pheromone update mechanism that improves convergence and avoids local minima. The algorithm demonstrated superior performance in both small and large-scale tests, even outperforming LINGO (which failed on large instances due to high complexity) by yielding better results with minimal additional computation time.

For the VRPTW, Arora et al. (2025) evaluated multiple metaheuristics including Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Tabu Search (TS) and Iterated Local Search (ILS) using Solomon benchmark datasets. Their study found that GA consistently produced the best results in terms of minimizing travel distance and maintaining constraint satisfaction. PSO ranked second, while TS and ILS delivered more inconsistent performance, especially in larger problem sets.

In solving the Capacitated VRP (CVRP), Rahmani Hosseinabadi et al. (2019) proposed a novel hybrid approach called GELSGA, which combines Genetic Algorithm with Gravitational Emulation Local Search (GELS). While GA provides strong exploratory capabilities, GELS offers local refinement to enhance convergence and solution quality. This hybrid algorithm demonstrated competitive performance on benchmark datasets and effectively balanced exploration and exploitation within the search process.

2.1.5 Results and Effectiveness

The integration of AI and advanced optimization methods into logistics operations has produced tangible performance gains. According to Royappa et al. (2024), AI-enabled route planning has led to a 20% reduction in delivery times and a 15% drop in fuel consumption, which are benefits that are especially significant in congested urban environments.

Moreover, AI contributes to improved vehicle load efficiency. By analysing attributes like volume, weight and destination, AI systems can raise vehicle utilization rates from 72% to 90%. This efficiency translates into fewer delivery trips and approximately 12% fuel savings, simultaneously enhancing cost-efficiency and sustainability (Royappa et al., 2024).

In the context of 3L-CVRP, He et al. (2024) demonstrated that the IWOA-HMOHA algorithm not only increased the average vehicle loading rate but also optimized the spatial placement of delicate items like electric meters. The model's three-dimensional visualization confirmed compact, secure stacking, supporting both operational and safety standards.

Li et al. (2019) validated the IACO algorithm through a series of simulations. While global optima from LINGO served as the benchmark in small-scale problems (with only a 0.065 difference), IACO significantly outperformed traditional ACO in large-scale setups. In one scenario, a 2.34% increase in runtime led to a 198% improvement in the solution's objective value. IACO also demonstrated superior cost minimization and emission reduction, reinforcing its applicability in real-world green logistics.

Rahmani Hosseinabadi et al. (2019) showed that their hybrid GELSGA algorithm could efficiently tackle the CVRP by combining the generative strength of Genetic Algorithms with the precision of GELS. The algorithm maintained high solution diversity while avoiding premature convergence, proving to be a robust method for large-scale distribution challenges.

2.1.6 Summary

2.1.6.1 Summary of Findings

The rapid rise of e-commerce and urbanization has significantly reshaped the logistics landscape, particularly in last-mile delivery. A major theme across the literature is the growing complexity and demand for high-performance logistics systems capable of meeting both operational and environmental targets. As urban populations grow and consumer expectations evolve, delivery systems are under increasing pressure to be faster, smarter and more sustainable (Engesser et al., 2023; Mohammad et al., 2023).

A variety of challenges have been identified, including increased parcel volumes, higher delivery frequency, constrained road infrastructures, growing demand for green logistics, time pressure and labour shortages. These challenges are especially pronounced in dense urban areas where traditional logistics models struggle to maintain efficiency (Mohammad et al., 2023; Royappa et al., 2024).

In response, researchers have developed various models and methods to optimize logistics performance. The Vehicle Routing Problem (VRP) and its numerous variants—such as CVRP, DVRP, VRPTW and 3L-CVRP—have served as foundational models to improve routing efficiency under various real-world constraints (Mageswari, 2024; Zhang et al., 2021; He et al., 2024). These models are further supported by advanced solution techniques, ranging from traditional AI and machine learning models to heuristics and metaheuristics like Genetic Algorithms, Whale Optimization Algorithms and Ant Colony Optimization.

Moreover, specific advancements in intelligent scheduling, such as IWOA-HMOHA and IACO, demonstrate the potential of combining real-time data, space planning and adaptive algorithms to improve delivery operations, especially in scenarios involving fragile or high-value goods (He et al., 2024; Li et al., 2019).

2.1.6.2 Gaps in Current Literature

Despite extensive research on vehicle routing problems (VRP) and various optimization strategies, notable gaps remain—particularly when addressing the complex logistics requirements of the electrical appliance industry. Unlike standard parcel delivery systems, this sector involves bulky, fragile and high-value items that often require on-site installation, specialized handling and precise coordination among delivery and installation teams.

Firstly, most existing VRP literature emphasizes route efficiency or truck space utilization in isolation, with minimal attention to installation-related factors. In the context of electrical appliances, a delivery is often not complete upon arrival, it must be followed by installation tasks that vary in duration based on appliance type, installation complexity and site conditions. These tasks often require trained technicians, further complicating scheduling. Yet, no current routing model explicitly integrates installation time estimation or technician-team scheduling into route planning.

Secondly, while the 3L-CVRP model addresses spatial constraints like three-dimensional truck loading, it falls short in capturing industry-specific considerations such as upright-only placement for refrigerators, shock-sensitive loading for washing machines, or directional stacking for smart devices. Moreover, these models typically assume that all goods can be handled interchangeably, overlooking item-specific handling protocols mandated by appliance warranties and safety regulations.

Thirdly, residential zone compliance, such as restricted delivery hours, building access limitations (e.g., no-lift apartments), or coordination with building management, are rarely addressed in mainstream VRP or DVRP formulations. These constraints are highly relevant in urban appliance delivery, where crews may face physical and procedural access barriers that can delay or invalidate schedules if unaccounted for.

Lastly, although metaheuristic models like IWOA-HMOHA (He et al., 2024) and IACO (Li et al., 2019) offer promising results in solving routing or loading problems, they operate on generalized logistics assumptions. None of the reviewed algorithms provide an integrated, constraint-aware scheduling framework that dynamically balances delivery routing, truck space loading, installation time estimation and technician availability, while also adapting to residential zone constraints. This lack of holistic integration limits their direct applicability to logistics systems in the electrical appliance sector.

In summary, the current literature lacks a comprehensive intelligent scheduling model tailored to the electrical appliance industry, where delivery success depends not only on reaching the customer on time, but also on aligning installation time estimation, residential zone compliance, route and truck space optimization into a single, adaptive system. Bridging this gap represents a significant research opportunity with the potential to substantially improve service quality, resource efficiency and customer satisfaction in installation-based delivery ecosystems.

2.2 Existing Systems Analysis

2.2.1 Tookan

Tookan is a delivery management platform developed by Jungleworks that enables businesses to manage on-demand delivery operations efficiently (see Figure 2.1)[1]. It provides features like real-time fleet tracking, route optimization, geofencing, automated task assignment, customer notifications and performance analytics. It is suitable for various sectors including food delivery, e-commerce, logistics and field service. Tookan focuses on improving last-mile delivery efficiency but does not offer integrated tools for installation time estimation or workforce scheduling for installation teams, which are critical in industries like electrical appliance delivery and setup.

[1] https://jungleworks.com/tookan/

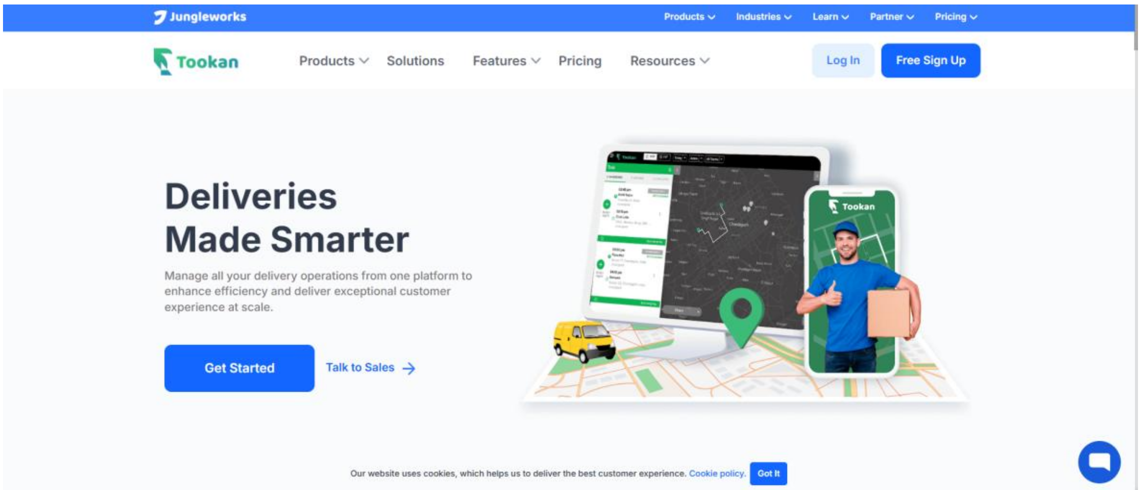

Figure 2.1: Screenshot of Tookan's official website

2.2.2 LogiNext Mile

LogiNext Mile is a real-time delivery tracking and route optimization software designed for enterprise-level logistics (see Figure 2.2)[2]. It supports delivery route planning, real-time ETA tracking, predictive alerts and customer communication. The platform integrates with CRM and ERP systems, supports multi-depot operations, and helps reduce delivery costs. It is widely used in retail, courier and field service industries. While it has strong logistics features, it does not support installation-related workflows like technician scheduling or installation duration estimation, which limits its suitability for the electrical appliance industry.

[2] https://www.loginextsolutions.com/platform/mile/delivery-schedule-planning-software

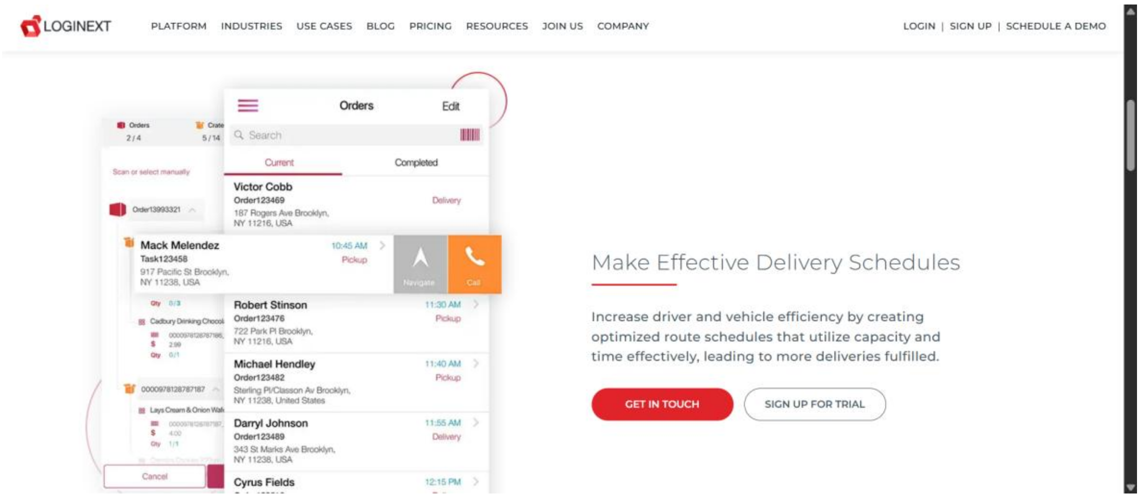

Figure 2.2: Screenshot of LogiNext Mile's official website

2.2.3 Onfleet

Onfleet is a modern delivery management platform known for its intuitive UI and powerful real-time tracking features. It provides route optimization, proof of delivery, recipient notifications, delivery analytics and driver performance monitoring (see Figure 2.3)[3]. It also supports SMS communication and customer feedback collection. Onfleet is commonly used in food, grocery, pharmacy and parcel delivery sectors. However, it lacks capabilities like truck loading optimization and installation task planning, which are essential for appliance deliveries involving both transport and technical setup.

---

[3] https://onfleet.com/

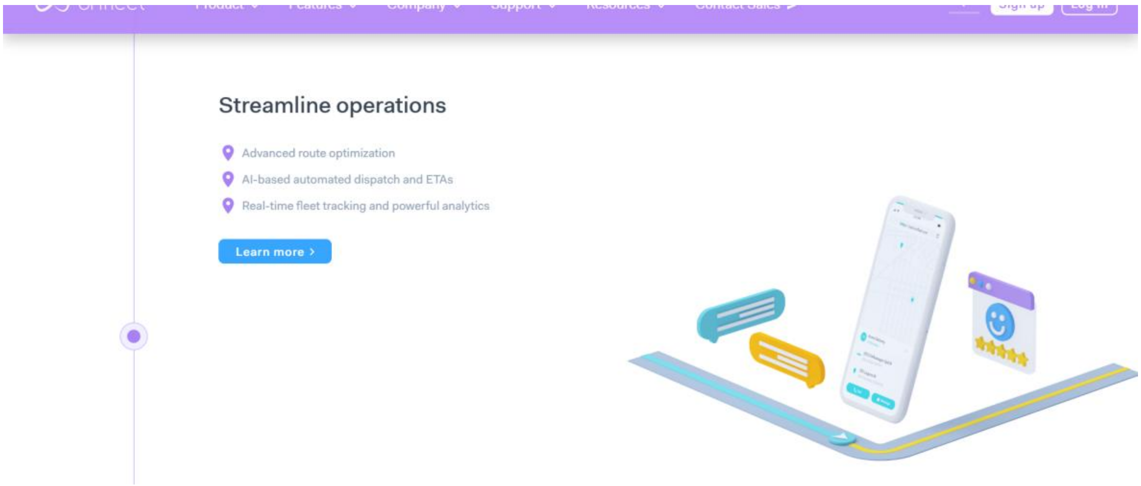

Figure 2.3: Screenshot of Onfleet's official website

2.2.4 DispatchTrack

DispatchTrack is an end-to-end delivery optimization platform that handles scheduling, routing, tracking, proof of delivery and customer communications (see Figure 2.4)[4]. It is designed for large-scale logistics operations, including furniture and appliance delivery. It supports configurable time windows and customer preferences, and includes robust analytics dashboards. DispatchTrack does offer some truck space optimization and load planning, making it slightly more relevant for appliance delivery than others. However, it still lacks specific features for estimating installation time and allocating technical teams accordingly.

---

[4] https://www.dispatchtrack.com/

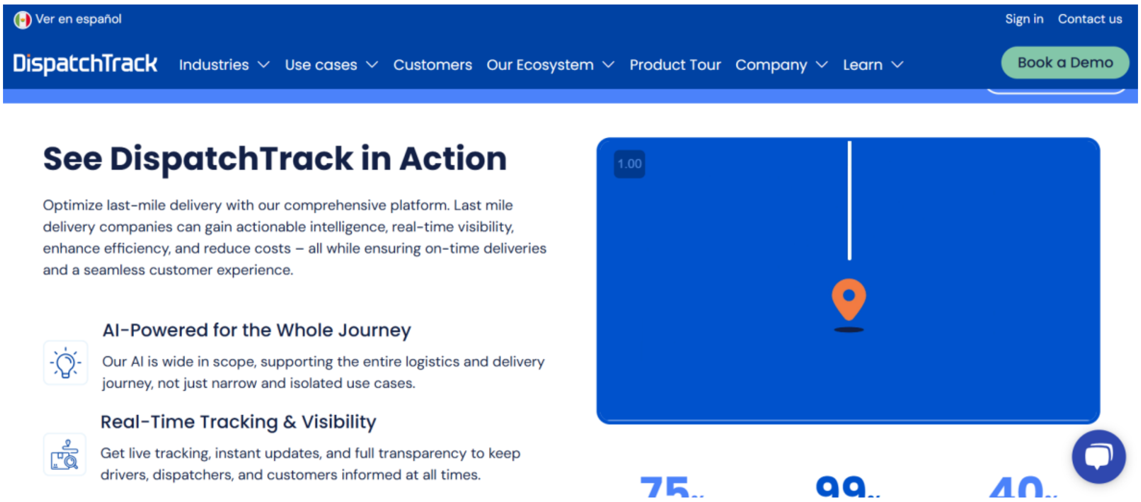

Figure 2.4: Screenshot of DispatchTrack's official website

2.2.5 Bringg

Bringg is an advanced delivery and fulfillment orchestration platform used by major retailers and logistics providers (see Figure 2.5)[5]. It enables businesses to streamline last-mile logistics through features like delivery scheduling, real-time tracking, curbside pickup, route optimization and seamless third-party delivery integrations. Bringg also supports multi-modal deliveries and operational analytics. It includes truck capacity utilization features, but like the others, it does not have built-in modules for estimating installation time or managing technician team schedules, making it less ideal for logistics systems that need to coordinate delivery and installation together.

---

[5] https://www.bringg.com/

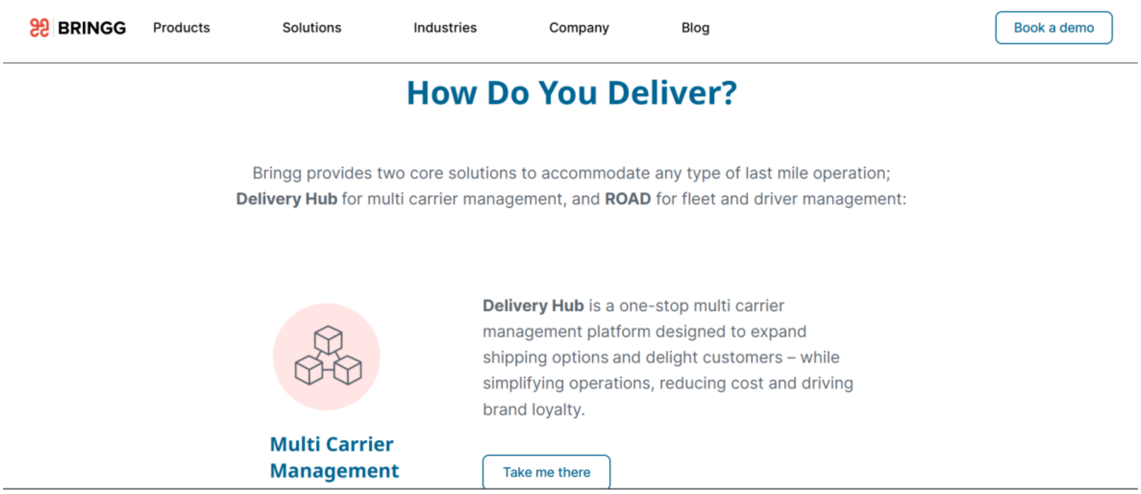

Figure 2.5: Screenshot of Bringg's official website

2.3 Comparison Between Existing Systems

Table 2.1 provides a comparative overview of key features offered by several existing logistics scheduling systems. It highlights functional gaps in current solutions, particularly in installation-specific aspects such as installation time estimation and installation team schedule access which this project aims to address.

Table 2.1: Feature Comparison of Existing Logistics Scheduling Systems

Feature	Tookan	LogiNext Mile	Onfleet	Dispatch Track	Bringg
Installation Time Estimation	No	No	No	No	No
Residential Zone Compliance	Yes	Yes	Yes	Yes	Yes
Delivery Route Optimization	Yes	Yes	Yes	Yes	Yes
Truck Space Optimization	No	No	No	Yes	Yes
Installation Team Schedule Access	No	No	No	No	No
Admin Manual Control (Edit Schedule, Reassign Teams)	Yes	Yes	Yes	Yes	Yes
Admin Dashboard for Performance Monitoring	Yes	Yes	Yes	Yes	Yes

2.4 Summary of Literature Review and Existing Systems Analysis

The literature highlights growing challenges in urban logistics, including rising parcel volumes, sustainability concerns and last-mile delivery inefficiencies. Although various VRP models and solution methods (e.g., AI, heuristics, metaheuristics) exist, most focus on routing and truck space alone, with limited attention to installation time estimation, technician scheduling, or residential zone constraints, all critical in appliance delivery operations.

Our proposed system directly addresses these gaps by integrating delivery and installation planning into a unified platform. It introduces installation time estimation, technician schedule coordination and zone-aware delivery logic, allowing for precise, constraint-aware time slot proposals. These features are absent in existing commercial platforms.

Current systems like Tookan, LogiNext Mile, Onfleet, DispatchTrack and Bringg support route optimization and admin control, but none provide modules for installation workflows. Only DispatchTrack and Bringg offer limited truck loading support and none estimate setup duration or adapt to residential delivery restrictions.

In short, our system is designed to fill a critical gap in both academic research and current logistics tools, offering a comprehensive solution for coordinated delivery and installation of electrical appliances.

CHAPTER 3: METHODOLOGY

3.1 Software Development Methodology

This project adopts the Rational Unified Process (RUP) as the software development methodology. RUP is a structured, iterative and incremental software development process framework that emphasizes well-defined phases and continuous stakeholder involvement. The choice of RUP aligns well with the complexity and scope of this project, which involves multiple modules such as scheduling module, task view module tailored for the electrical appliance industry. RUP's emphasis on iterative refinement, stakeholder validation and modular development makes it suitable for managing the evolving requirements and constraints identified during stakeholder consultations with TBM Electrical.

3.1.1 Overview of Unified Process Model

The Rational Unified Process (RUP) is a comprehensive and adaptable software development framework that divides the software development lifecycle into four distinct phases: Inception, Elaboration, Construction and Transition. These phases are carried out iteratively, with each iteration producing an incrementally refined and enhanced version of the software system. An overview of the RUP lifecycle is illustrated in Figure 3.1.

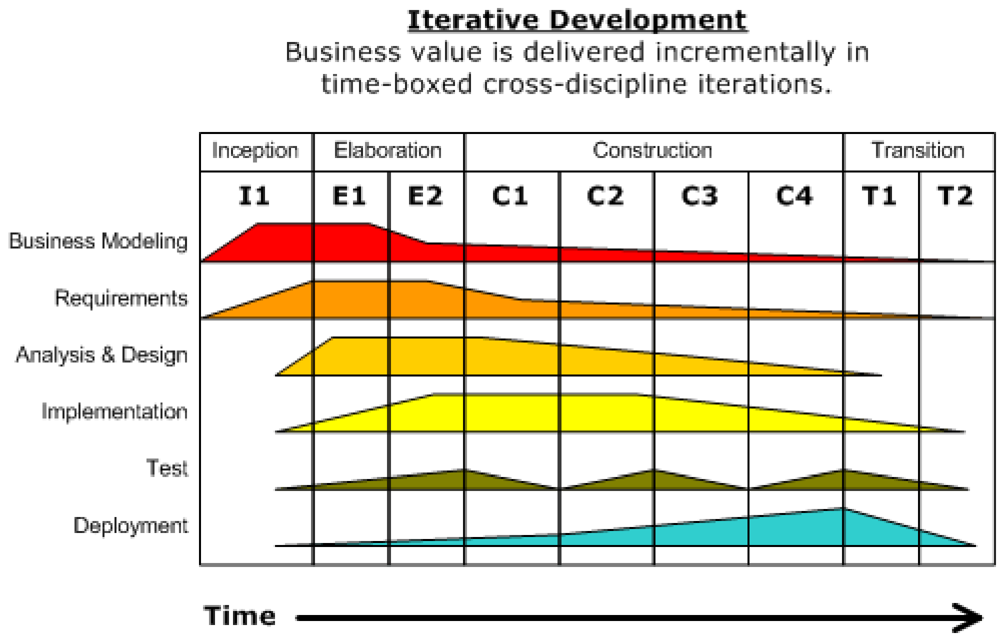

Figure 3.1: Phases of the Rational Unified Process (RUP)

3.1.1.1 Inception Phase

This phase focuses on defining the business case, identifying key system objectives, stakeholders, and major features. For this project, the inception phase was used to conduct stakeholder meetings with TBM Electrical, finalize objectives such as intelligent delivery scheduling and real-time tracking, and determine the feasibility of the proposed solution.

3.1.1.2 Elaboration Phase

During this phase, the project's requirements are further detailed and validated. High-level architectural and design models are created. This project’s elaboration phase included requirement gathering, use case modelling, ER diagram creation, system architecture design and UI/UX prototyping using Figma.

3.1.1.3 Construction Phase

This is the core development phase where the actual system components are implemented, tested and integrated. In this project, the construction phase involved developing frontend interfaces in ReactJS and backend services using Firebase, including unit and integration testing.

3.1.1.4 Transition Phase

The system is deployed to the end users and any final bugs are resolved based on feedback. This phase included UAT testing with TBM Electrical staff, refining the UI/UX, resolving issues and preparing the system for final deployment.

3.1.2 Characteristics of Unified Process

The Rational Unified Process is defined by several core characteristics, all of which were effectively leveraged in this project.

3.1.2.1 Iterative and Incremental Development

RUP emphasizes developing the system in small iterations, allowing for continuous feedback and improvements. In this project, features were developed and tested in successive sprints, allowing the team to refine modules such as route optimization and delivery constraints filtering progressively. For example, the delivery scheduling logic was prototyped early and improved iteratively based on feedback from TBM staff.

3.1.2.2 Use Case Driven

RUP places use cases at the center of requirement analysis and design. Each functional requirement in this project was supported by a detailed use case description. This approach ensured that system design closely aligned with actual user workflows, such as the admin reassigning installation teams. Use cases also guided the creation of UI mockups and the planning of backend logic.

3.1.2.3 Architecture-Centric

RUP emphasizes establishing a solid architecture early in development. In this project, system architecture diagrams were prepared during the elaboration phase to define the interaction between frontend modules, Firebase backend and supporting services like maps. This early focus on architecture ensured scalability, maintainability and consistent data flow between modules.

3.1.2.4 Risk-Driven

RUP promotes identifying and addressing high-risk elements early. This project addressed several potential risks during early phases, including route-planning complexity and Firebase data structuring. These areas were prototyped and tested early to reduce integration issues later.

3.1.3 Mapping RUP Phases to Project Timeline

Table 3.1 maps the four RUP phases to the actual FYP timeline and work conducted. This mapping demonstrates how RUP supported structured progress through each FYP milestone while allowing for iterative feedback and refinements. A more detailed breakdown of weekly activities is provided in the project timeline spreadsheet (see Appendix B). This detailed timeline captures progress tracking in finer granularity and offers additional insight into how the development evolved through continuous iterations.

Table 3.1: Mapping of RUP Phases to Final Year Project Timeline and Activities

RUP Phase	Timeline	Key Activities
Inception	FYP1 (Mac–May)	Literature review, stakeholder interviews, objective finalization, system scope definition
Elaboration	FYP1 (May–June)	Requirement analysis, use case modeling, ER and class diagrams, UI/UX mockups
Construction	FYP1&2 (June–Oct)	System development using ReactJS & Firebase, iterative testing, GitHub version control
Transition	FYP2 (Oct–Dec)	UAT testing, bug fixing, UI refinement, system deployment and documentation

3.2 Data Gathering Methodologies

To design an effective logistics scheduling system tailored to the electrical appliance industry, multiple data collection methods were employed. These methods provided essential insights into real-world challenges, operational workflows and stakeholder expectations.

3.2.1 Stakeholder Resource

The official website of TBM Electrical[6] served as a key resource for understanding TBM Electrical's operations, product offerings and service delivery scope. The platform provided specifications and dimensions for a wide range of appliances, including TVs, refrigerators, air-conditioners and washing machines. This information was later leveraged for truck space optimization and installation time estimation modules.

3.2.2 Stakeholder Meetings

A series of structured meetings and continuous communication were held with the key stakeholder, Mr. Tan Wai Kiat and his HR staff, Mr. Hoo June Yee at TBM Electrical. These sessions were vital for uncovering operational bottlenecks, validating requirements and co-defining features for the enhanced scheduling system. Prior to conducting these meetings, a formal collaboration agreement was established with TBM Electrical (see Appendix A).

3.2.2.1 Stakeholder Meeting 1

The first stakeholder meeting was held on 15 April 2025, from 4:00 p.m. to 5:30 p.m., via Microsoft Teams. Attendees included Mr. Tan Wai Kiat, Mr. Hoo June Yee, project

[6] https://shop.tbm.com.my

supervisor Dr. Chiew Thiam Kian and student Wong Wen Hao and Chew Jia Hui. A screenshot of the meeting session is provided in Figure 3.2.

During the discussion, several key operational insights were revealed. One major challenge involved access and registration procedures at high-rise buildings and gated communities. These often require pre-authorization of delivery vehicles and personnel, and impose time-based restrictions on installation activities that may generate noise. Additionally, it was highlighted that communication with customers during delays, such as those caused by traffic congestion or adverse weather was often insufficient, leading to missed appointments and disruptions to the daily delivery schedule.

To address these concerns, Mr. Tan proposed the inclusion of several key system features. These included real-time vehicle tracking, predictive estimated time of arrival (ETA) calculations, intelligent time slot recommendations tailored to each building's constraints, and automated delay notifications to customers, aimed at improving transparency and reducing service disruption.

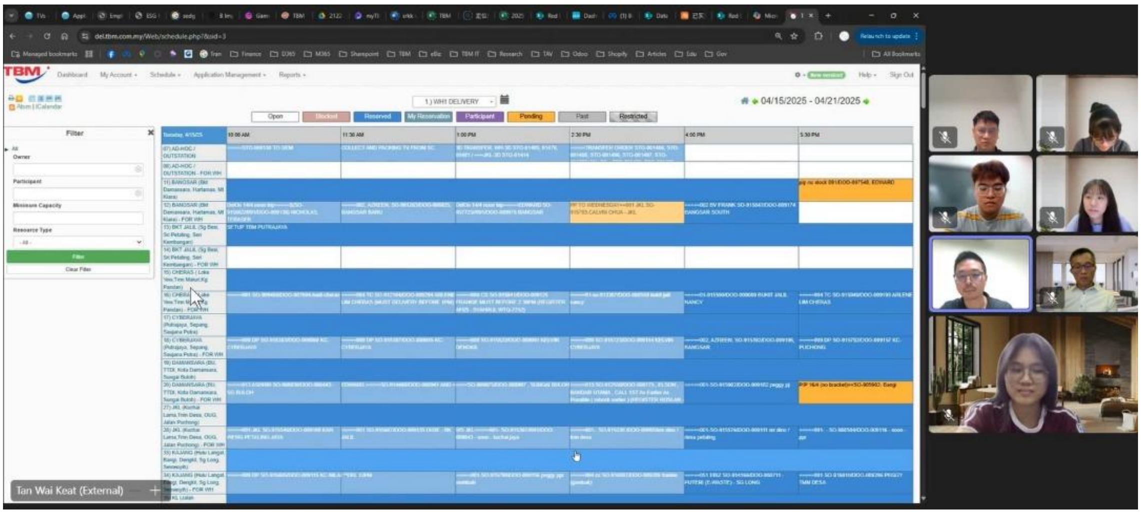

Figure 3.2: Screenshot of Stakeholder Meeting 1

3.2.2.2 Stakeholder Meeting 2

The second stakeholder meeting took place on 16 May 2025 via WhatsApp. The informal nature of the platform facilitated quick exchanges of information and clarification of operational challenges. A screenshot of the discussion thread is presented in Figure 3.3.

During this session, we gathered average installation time data for various appliance categories. These estimates were critical for the development of an installation-aware scheduling mechanism that accounts for setup duration when allocating delivery slots. Another key insight involved site access variability across different customer properties. These variations include parking limitations, availability of lifts or staircases and differing registration protocols, all of which supported the need for a database to record location-specific access requirements.

Further discussion addressed physical constraints of the delivery trucks, especially in relation to product stacking and fragility. TBM's operations utilize two primary truck types, a 1-ton open truck and a 3-ton covered truck with differing constraints in terms of volume, protection and route suitability. Lastly, the team highlighted the importance of dynamic dispatching, sharing examples of real-life disruptions such as driver absenteeism, vehicle breakdowns and unscheduled jobs. These challenges demonstrated the necessity of implementing flexible route reallocation and driver-team reassignment capabilities in the proposed system.

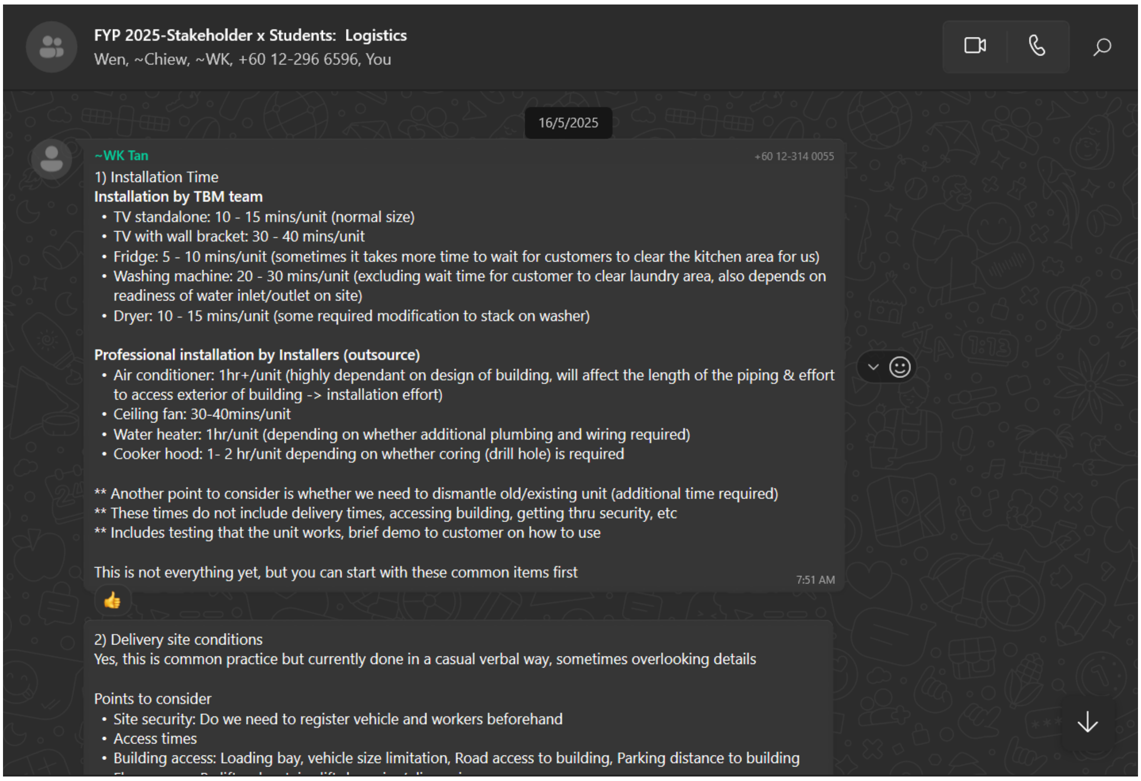

Figure 3.3: Screenshot of Stakeholder Meeting 2

3.2.2.3 Stakeholder Meeting 3

The third stakeholder meeting was held on 6 June 2025 through WhatsApp. It focused on refining logistics planning components and validating data regarding delivery zones and fleet composition. A screenshot of the WhatsApp conversation is provided in Figure 3.4.

The stakeholder confirmed that TBM operates a fleet of 20 trucks, which are deployed according to strategic considerations. The 1-ton trucks are typically assigned to short-distance or narrow-access areas and are scheduled for two trips per day with a mid-day reload. In contrast, 3-ton trucks usually perform a single, full-day trip due to their larger capacity and deployment to wider-access locations.

Another important clarification involved TBM's delivery zone structure. The company operates across 14 fixed zones within the Greater Kuala Lumpur area and maintains one flexible zone designated for ad-hoc outstation deliveries. Weekend and public holiday operations are constrained by residential property rules, particularly for high-rise condominiums that prohibit deliveries on non-working days. This limits system scheduling flexibility and reinforces the need for access-aware planning.

Lastly, current practices in truck and team assignment were discussed. The stakeholder revealed that assignments are manually determined by experienced dispatchers, relying on intuition rather than data. This insight directly influenced the development of a rule-based optimization layer in the proposed system to support data-driven decision-making.

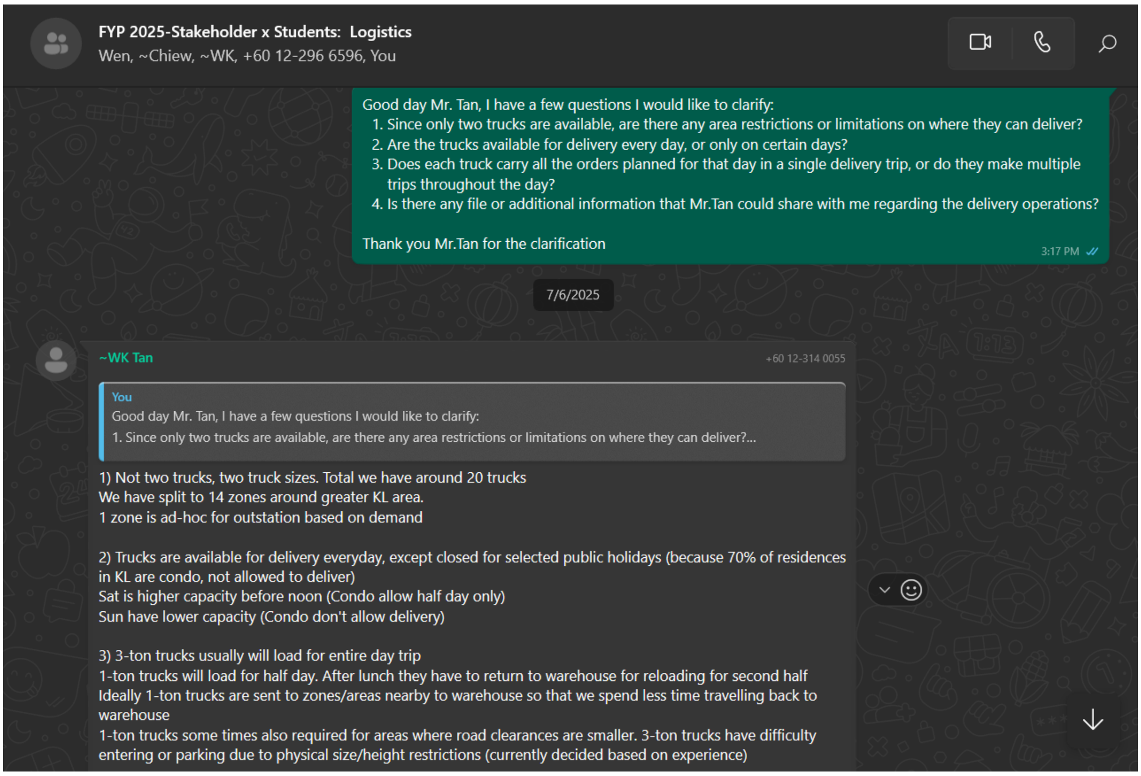

Figure 3.4: Screenshot of Stakeholder Meeting 3

3.2.3 Research Paper (Article)

To ground the project in proven concepts and identify gaps in existing approaches, a total of 20 academic and industry articles were reviewed. These resources were selected based on their relevance to structured logistics systems.

Through this literature review, key elements of an effective logistics system were identified, including the ability to handle varying delivery constraints and automate resource allocation. Several works also emphasized the benefits of using intelligent algorithms for improving delivery efficiency, reducing idle time and minimizing fuel consumption. These findings played a crucial role in shaping the direction of the proposed scheduling system.

3.2.4 Existing System

To understand the practical constraints of the current logistics operations, screenshots of TBM's existing delivery scheduling interface were collected and reviewed (see Figure 3.5). The current system operates on a manual and static time slot assignment model, where scheduling decisions are made based on human experience and fixed calendar views. There is no centralized platform that dynamically accounts for product installation time, truck space availability, or real-time delays.

This static approach creates significant bottlenecks, especially when dealing with last-minute changes such as traffic disruptions, customer no-shows, or complex installation requirements. Moreover, customer updates are not automated, and drivers must manually inform the operations team of their progress, resulting in potential miscommunication and reduced workflow transparency. These shortcomings informed the design rationale for an intelligent, real-time scheduling engine in the proposed system that supports constraint-aware slot generation, dynamic rescheduling and real-time visibility for all involved stakeholders.

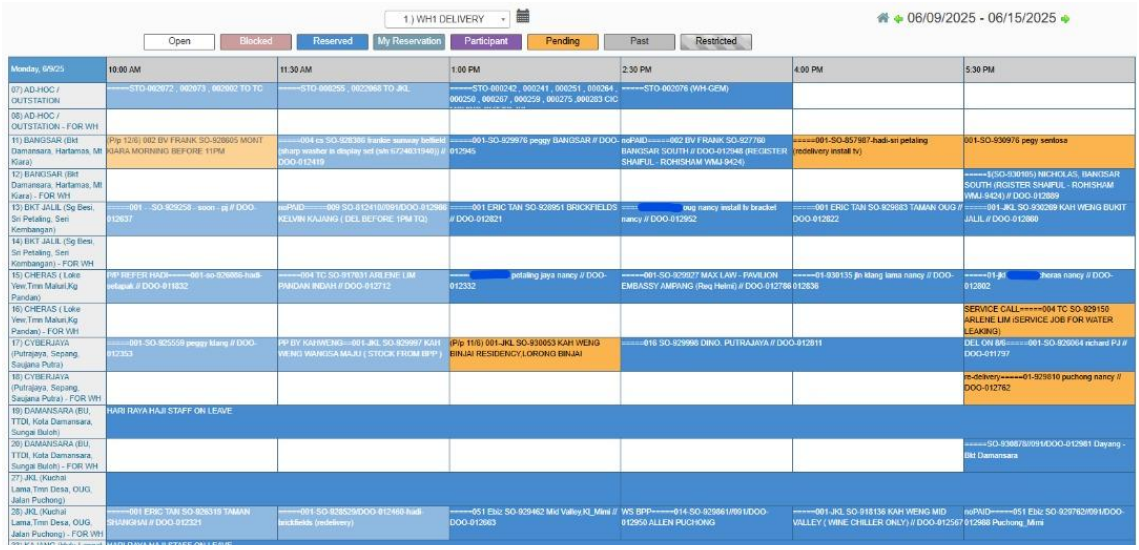

Figure 3.5: TBM's current delivery scheduling interface

3.3 Collaboration Methodology

Effective collaboration tools were essential to ensure smooth coordination among team members and to manage the development process throughout the project. Two main platforms were used: GitHub for source code management and Google Spreadsheet for project tracking and timeline planning.

3.3.1 GitHub

GitHub was used as the primary platform for managing the system's source code. It served both as a collaborative workspace and as a version control system, allowing the team to work on different modules concurrently without conflicts. Through GitHub, we could track changes, manage branches for different features or bug fixes and review code contributions effectively. This ensured transparency in development progress and helped maintain a clean, structured codebase.

3.3.2 Google Spreadsheet

Google Spreadsheet was used as the main project management and task planning tool. It allowed us to create a detailed project timeline, broken down into specific tasks, deliverables and responsible team members (PICs). Unlike a general Gantt chart or monthly overview, this spreadsheet provided a granular breakdown of work items along with clear milestones and deadlines. It was updated iteratively to reflect real-time progress and planning adjustments. The detailed project timeline is included in the Appendix B for reference.

CHAPTER 4: REQUIREMENT ANALYSIS

4.1 Overview

The chapter provides a comprehensive overview of the system's requirements and design through detailed specifications and modelling techniques. It begins by outlining both the functional and quality requirements, ensuring that the system meets user needs and maintains high standards for security and usability. The chapter then presents use case modelling, including both diagrams and descriptive scenarios, to clearly illustrate the interactions between users and the system across various. Furthermore, the inclusion of UML activity diagrams for key modules offers valuable insights into the underlying workflows and decision points, highlighting how different actors and system components collaborate.

4.2 System Requirements

4.2.1 Functional Requirements (Web-Based Platform)

4.2.1.1 Authentication Module

Table 4.1 presents the functional requirements for the Authentication Module, outlining core access features such as registration, login, password reset and logout.

Table 4.1: Functional Requirements for Authentication Module

Req ID	Functional Requirement	Use Case ID	Priority
FR-01-001	The system shall allow users to register a new account.	UC-01	High
FR-01-002	The system shall allow users to sign in to an account.	UC-02	High
FR-01-003	The system shall allow users to reset passwords when they forget their password.	UC-03	Medium
FR-01-004	The system shall allow users to log out of the account.	UC-04	Medium

4.2.1.2 Scheduling Module

Table 4.2 lists the functional requirements for the Scheduling Module, covering delivery and installation time estimation, constraint filtering, route and truck space optimization, and dynamic time slot assignment.

Table 4.2: Functional Requirements for Scheduling Module

Req ID	Functional Requirement	Use Case ID	Priority
FR-02-001	The system shall calculate the total estimated installation time based on the estimated installation complexity and quantity of purchased items.	UC-17	High
FR-02-002	The system shall filter out time slots that conflict with residential zone access constraints, including building-type-specific rules such as permitted delivery hours, noise restrictions and entry permits.	UC-17	High
FR-02-003	The system shall optimize the sequence of confirmed deliveries within each time window to minimize travel time and improve delivery efficiency.	UC-17	Medium
FR-02-004	The system shall assess truck space availability during scheduling and exclude overloaded combinations during route optimization.	UC-17	Medium
FR-02-005	The system shall evaluate delivery and installation combinations using estimated installation duration, residential zone access constraints, truck capacity and route proximity to select the most suitable time window for the current order. The selected time window should allow the highest number of orders to be delivered and installed efficiently while meeting all planning constraints.	UC-17	High
FR-02-006	The system shall automatically assign a new delivery and installation time slot when the initial slot is rejected or unavailable.	UC-17	High

4.2.1.3 Task View Module

Table 4.3 details the functional requirements for the Task View Module, which provides different operational teams with access to their relevant delivery, installation and loading schedules.

Table 4.3: Functional Requirements for Task View Module

Req ID	Functional Requirement	Use Case ID	Priority
FR-03-001	The system shall allow the warehouse loading team to view the loading schedule for each truck.	UC-24	High
FR-03-002	The system shall allow delivery teams to view their assigned delivery schedules.	UC-17	High
FR-03-003	The system shall allow delivery teams to view delivery route recommendations.	UC-22	High
FR-03-004	The system shall allow outsourced installers to view their assigned installation schedules.	UC-23	High

4.2.1.4 Manual Control and Monitoring Module

Table 4.4 outlines the functional requirements for the Manual Control and Monitoring Module, focusing on admin capabilities such as editing schedules and handling user reports.

Table 4.4: Functional Requirements for Manual Control and Monitoring Module

Req ID	Functional Requirement	Use Case ID	Priority
FR-03-001	The system shall allow admins to view and edit delivery schedules and team assignments.	UC-25	High
FR-03-002	The system shall allow admins to resolve reports submitted by users.	UC-26	High

4.2.1.5 Dashboard and Analytics Module

Table 4.5 summarizes the functional requirements for the Dashboard and Analytics Module, which allows admins to monitor performance metrics for orders and employees.

Table 4.5: Functional Requirements for Dashboard and Analytics Module

Req ID	Functional Requirement	Use Case ID	Priority
FR-04-001	The system shall allow admins to view employee performance, including the number of tasks completed and on-time completion rate.	UC-27	High
FR-04-002	The system shall allow admins to view order delivery performance, including delivery punctuality and completion status.	UC-28	High

4.2.2 Quality Requirements

Table 4.6 lists the quality requirements that define the system's non-functional expectations, focusing on security and usability.

Table 4.6: Quality Requirements

Req ID	Functional Requirement	Use Case ID	Priority
QR-01	The system shall store user credentials in encrypted form and enforce secure password policies that meet defined complexity requirements.	Security	High
QR-02	The system shall achieve an average score of at least 5.0 out of 7.0 on the PSSUQ across all sections, reflecting satisfactory user experience in terms of usefulness, information quality and interface quality.	Usability	High

4.3 Use Case Modelling (Use Case Diagram & Description)

4.3.1 Use Case Diagram

The use case diagram depicted in Figure 4.1 provides an overview of the main functionalities and user interactions within the Enhanced Logistics System for the Electrical Appliance Industry. This diagram visually represents the relationships between various actors, including Users, Customers, Delivery & Installation Team, Outsourced Installer, Warehouse Loading Team and Admin, and the use cases they are involved in.

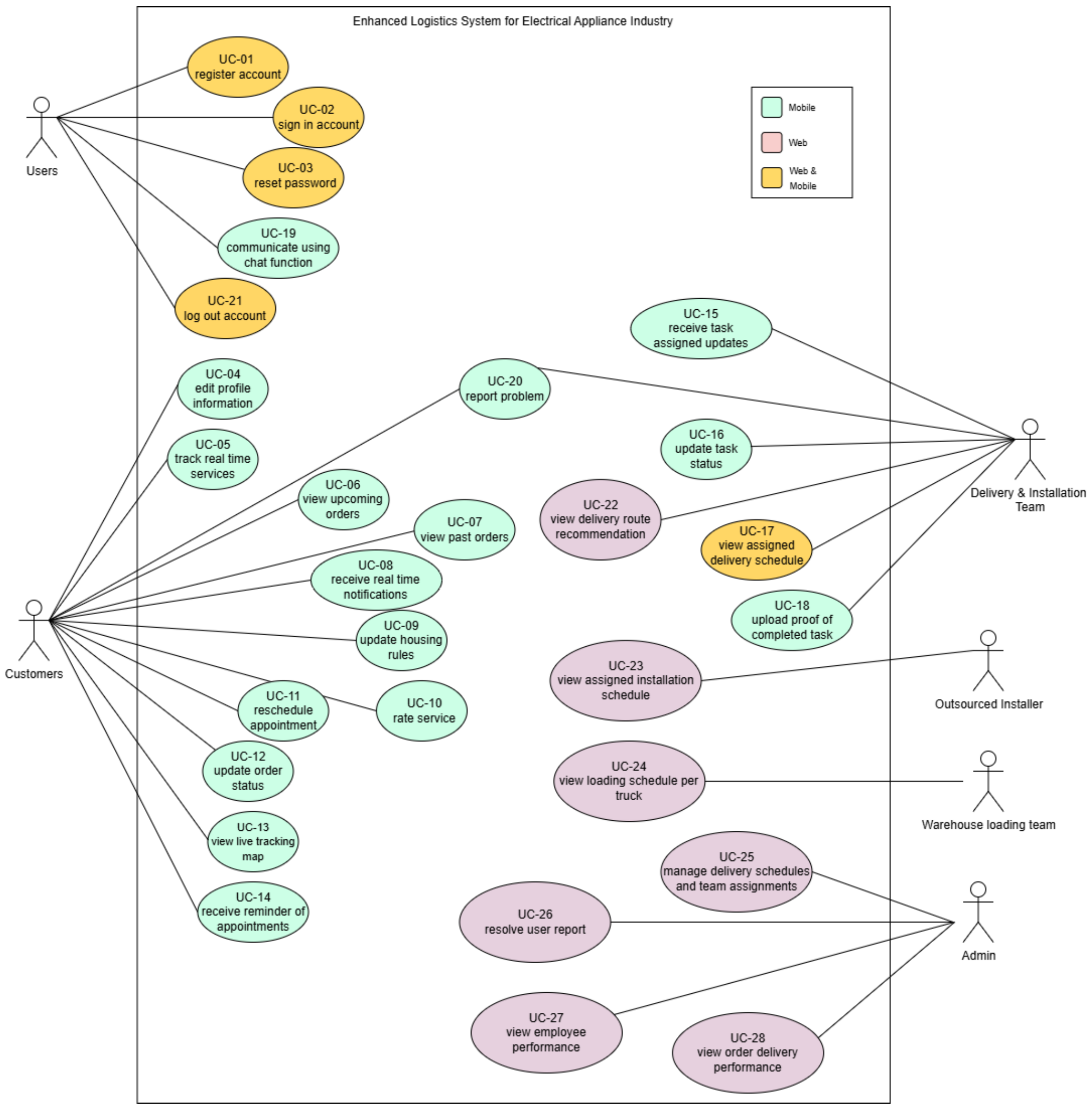

Figure 4.1: Use Case Diagram

4.3.2 Use Case Description (Web-Based Platform)

Detailed descriptions for each use case are provided in Table 4.7 to Table 4.18, covering user interactions and system responses for key scenarios such as registration, login, scheduling and resolving reports.

4.3.2.1 UC-01 Register Account

Table 4.7: Use Case Description for Register Account

Use Case Name	Register account	
Use Case ID	UC-01	
Description	The system shall allow USER to register a new account.	
Priority	High	
Actor(s)	New USER	
Triggering Event	The actor navigates to the register page.	
Pre-condition	The actor has not registered an account before.	
Post-condition	The actor successfully creates a new account.	
Flow of Events	Step	Action
	1	The actor navigates to the registration page.
	2	The actor inputs name, email and password.
	3	The actor clicks the "Next" button once all the information has been filled.
	4	The system validates the input (e.g., email format, password strength and email uniqueness).
	5	The system encrypts the password and stores the new account information.
	6	The system confirms successful registration and redirects to the dashboard.
Alternative Flow – Register with Google Account	2a	The actor clicks "Register with Google".
	3a	The system redirects to Google's authentication page.
	4a	The actor signs in and authorizes access.
	5a	The system retrieves verified email and basic profile information from Google.
	6a	The system logs the user in and redirects to the dashboard.

Table 4.7 continued.

Exception Flow	4b	If the email already exists, the system notifies the user and suggests logging in instead.
	4c	If password complexity requirements are not met, the system prompts the user to correct it.

4.3.2.2 UC-02 Sign In Account

Table 4.8: Use Case Description for Sign In Account

Use Case Name	Sign In Account	
Use Case ID	UC-02	
Description	The system shall allow USERS to sign in to an account using their registered credentials.	
Priority	High	
Actor(s)	Registered USER	
Triggering Event	The actor navigates to the login page.	
Pre-condition	The actor must already have a registered account.	
Post-condition	The actor gains access to their dashboard.	
Flow of Events	Step	Action
	1	The actor navigates to the login page.
	2	The actor inputs email and password.
	3	The actor clicks the “Login” button.
	4	The system validates the input credentials.
	5	If correct, the user is logged in and redirected to dashboard.
Alternative Flow – Login with Google Account	2a	The actor clicks "Login with Google".
	3a	The system redirects to Google’s authentication page.
	4a	The actor signs in and authorizes access.
	5a	The system verifies the account and redirects to the dashboard
Exception Flow	4b	If credentials are incorrect, the system notifies the user.
	4c	If the account is not found, the system suggests registration.

4.3.2.3 UC-03 Reset Password

Table 4.9: Use Case Description for Reset Password

Use Case Name	Reset Password	
Use Case ID	UC-03	
Description	The system shall allow USERS to reset their password when they forget it.	
Priority	Medium	
Actor(s)	Registered USER	
Triggering Event	The actor clicks “Forgot Password” on the login page.	
Pre-condition	The actor has an existing account.	
Post-condition	The actor successfully resets their password.	
Flow of Events	Step	Action
	1	The actor clicks the “Forgot Password” link.
	2	The actor enters their registered email address.
	3	The system sends a password reset link to the provided email.
	4	The actor clicks the link and is directed to the reset page.
	5	The actor inputs and confirms a new password.
	6	The system validates and updates the password.
	7	The actor is notified of success and redirected to the login page.
Exception Flow	3b	If the email does not exist, the system informs the user.
	5b	If the password does not meet complexity requirements, the system prompts correction.

4.3.2.4 UC-04 Log Out Account

Table 4.10: Use Case Description for Log Out Account

Use Case Name	Log Out Account
Use Case ID	UC-03
Description	The system shall allow USERS to log out from their current session.

Table 4.10 continued.

Priority	Medium	
Actor(s)	Authenticated USER	
Triggering Event	The actor clicks the “Logout” button.	
Pre-condition	The actor is currently logged in.	
Post-condition	The actor is securely logged out and redirected to the homepage.	
Flow of Events	Step	Action
	1	The actor clicks the “Logout” button.
	2	The system terminates the session.
	3	The user is redirected to the homepage or login page.

4.3.2.5 UC-17 View Assigned Delivery Schedule

Table 4.11: Use Case Description for View Assigned Delivery Schedule

Use Case Name	View Assigned Delivery Schedule	
Use Case ID	UC-17	
Description	The system shall allow DELIVERY TEAMS to view their assigned delivery and installation schedules	
Priority	High	
Actor(s)	DELIVERY TEAM	
Triggering Event	The actor navigates to Schedule tab.	
Pre-condition	The actor has an existing account.	
Post-condition	The actor can view their assigned schedule with complete order and timing details.	
Flow of Events	Step	Action
	1	The actor accesses to Schedule tab and select “My Tasks” section.
	2	The system retrieves all delivery and installation tasks assigned to that team.
	3	The schedule is displayed in chronological order, including estimated arrival time, product details, customer location and expected installation duration.

Table 4.11 continued.

	4	The actor uses this information to begin the delivery route.
Exception Flow	4b	If real-time updates (e.g., reassignment or changes in task order) occur, the system refreshes the displayed schedule and notifies the team.

4.3.2.6 UC-22 View Delivery Route Recommendation

Table 4.12: Use Case Description for View Delivery Route Recommendation

Use Case Name	View Delivery Route Recommendation	
Use Case ID	UC-22	
Description	The system shall allow DELIVERY TEAM to view their assigned delivery route and sequence.	
Priority	High	
Actor(s)	DELIVERY TEAM	
Triggering Event	The actor navigates to the Delivery Route tab.	
Pre-condition	The actor logs into the system.	
Post-condition	The actor accesses delivery map.	
Flow of Events	Step	Action
	1	The actor accesses the Delivery Route tab.
	2	The system displays delivery stops with optimized sequence.
	3	The actor begins the route.
Exception Flow	3b	If no route is generated, the system shows a pending message.

4.3.2.7 UC-23 View Assigned Installation Schedule

Table 4.13: Use Case Description for View Assigned Installation Schedule

Use Case Name	View Assigned Installation Schedule
Use Case ID	UC-23

Table 4.13 continued.

Description	The system shall allow OUTSOURCED INSTALLER to view their assigned installation tasks.	
Priority	High	
Actor(s)	OUTSOURCED INSTALLER	
Triggering Event	The actor logs into the system.	
Pre-condition	The actor navigates to “My Task” page.	
Post-condition	The actor sees installation tasks with location and product.	
Flow of Events	Step	Action
	1	The actor accesses the “My Tasks” tab.
	2	System displays task list with time and address.

4.3.2.8 UC-24 View Loading Schedule Per Truck

Table 4.14: Use Case Description for View Loading Schedule Per Truck

Use Case Name	View Loading Schedule Per Truck	
Use Case ID	UC-24	
Description	The system shall allow the WAREHOUSE LOADING TEAM to view loading schedules assigned to trucks.	
Priority	High	
Actor(s)	WAREHOUSE LOADING TEAM	
Triggering Event	The actor navigates to the Loading Schedule tab.	
Pre-condition	The actor logs into the system.	
Post-condition	The actor views their assigned loading tasks.	
Flow of Events	Step	Action
	1	The actor accesses the Loading Schedule tab.
	2	The system displays truck loading details by zone, product and time.
	3	The actor prepares goods for loading.
	4	The actor marks the task completed after loading finished.
Exception Flow	3b	If no schedule found, “No Schedule” message shown.

4.3.2.9 UC-25 Manage Delivery Schedules and Team Assignments

Table 4.15: Use Case Description for Manage Delivery Schedules and Team Assignments

Use Case Name	Manage Delivery Schedules and Team Assignments	
Use Case ID	UC-25	
Description	The system shall allow ADMINS to view and edit delivery schedules and team assignments.	
Priority	High	
Actor(s)	ADMIN	
Triggering Event	The actor navigates to the Schedule tab.	
Pre-condition	The actor logs into the system.	
Post-condition	Schedule and other information updates are reflected in the system.	
Flow of Events	Step	Action
	1	The actor accesses Schedule tab.
	2	The actor selects a delivery slot to edit.
	3	The actor modifies team assignment or reschedules time.
	4	System saves and broadcasts the update.
Alternative Flow - Edit Information	1a	The actor navigates to Information tab.
	2a	The actor edits information such as employee information and residential zone compliance rules.
	3a	System saves the updated information and reflects it in future scheduling logic.
Exception Flow	3b	If the schedule conflicts, system alerts admin.

4.3.2.10 UC-26 Resolve User Reports

Table 4.16: Use Case Description for Resolve User Reports

Use Case Name	Resolve User Reports
Use Case ID	UC-26
Description	The system shall allow ADMINS to resolve user-submitted reports or complaints.

Table 4.16 continued.

Priority	High	
Actor(s)	ADMIN	
Triggering Event	The actor navigates to the Reports tab.	
Pre-condition	The actor logs into the system.	
Post-condition	Reports are marked as resolved.	
Flow of Events	Step	Action
	1	The actor accesses Reports tab.
	2	The actor reviews pending reports.
	3	The actor carries out necessary action to solve the report.
	4	The actor marks report as resolved or responds with clarification.
	5	System logs and closes the report.

4.3.2.11 UC-27 View Employee Performance

Table 4.17: Use Case Description for View Employee Performance

Use Case Name	View Employee Performance	
Use Case ID	UC-27	
Description	The system shall allow ADMINS to view employee delivery and installation performance.	
Priority	High	
Actor(s)	ADMIN	
Triggering Event	The actor navigates to the Analytics Dashboard.	
Pre-condition	None	
Post-condition	Performance metrics are displayed.	
Flow of Events	Step	Action
	1	The actor accesses the Analytics Dashboard.
	2	The actor selects the Employee tab.
	3	The system displays data on completion rate and punctuality.

4.3.2.12 UC-28 View Order Delivery Performance

Table 4.18: Use Case Description for View Order Delivery Performance

Use Case Name	View Order Delivery Performance	
Use Case ID	UC-28	
Description	The system shall allow ADMINS to monitor order fulfilment and delivery status.	
Priority	High	
Actor(s)	ADMIN	
Triggering Event	The actor navigates to the Analytics Dashboard.	
Pre-condition	None	
Post-condition	Performance metrics are displayed.	
Flow of Events	Step	Action
	1	The actor accesses the Analytics Dashboard.
	2	The actor selects the Orders tab.
	3	The system displays key indicators such as success rates and average attempts.

4.4 UML Activity Diagrams

The following figures (Figure 4.2 to Figure 4.5) present the UML Activity Diagrams for key modules in the system, including authentication, scheduling and admin control. These diagrams illustrate the core process flows and decision logic across major functionalities, reflecting how different actors and system components interact throughout the delivery and installation workflow.

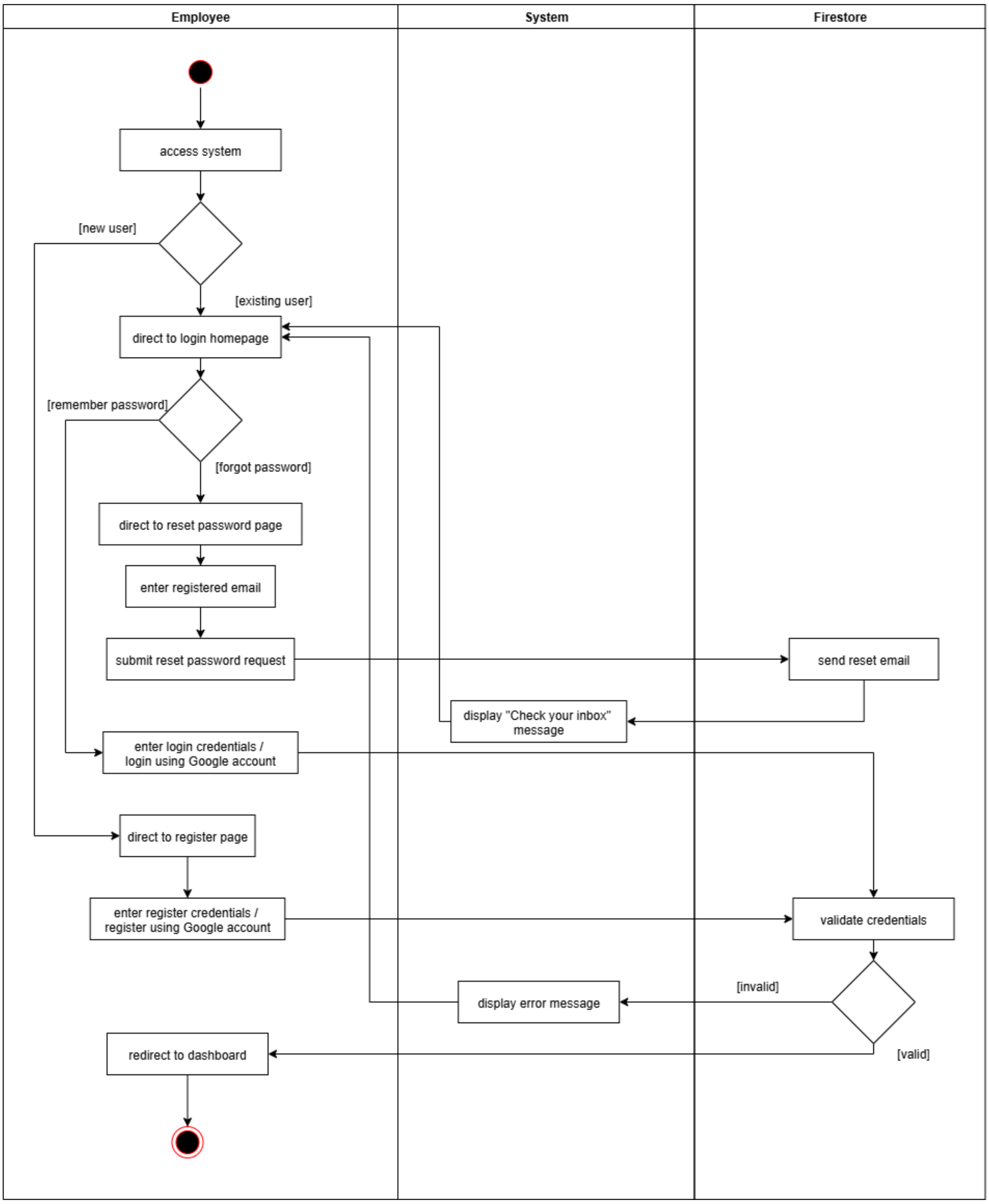

Figure 4.2: Activity Diagram for Authentication Module

Figure 4.3: Activity Diagram for Scheduling Module

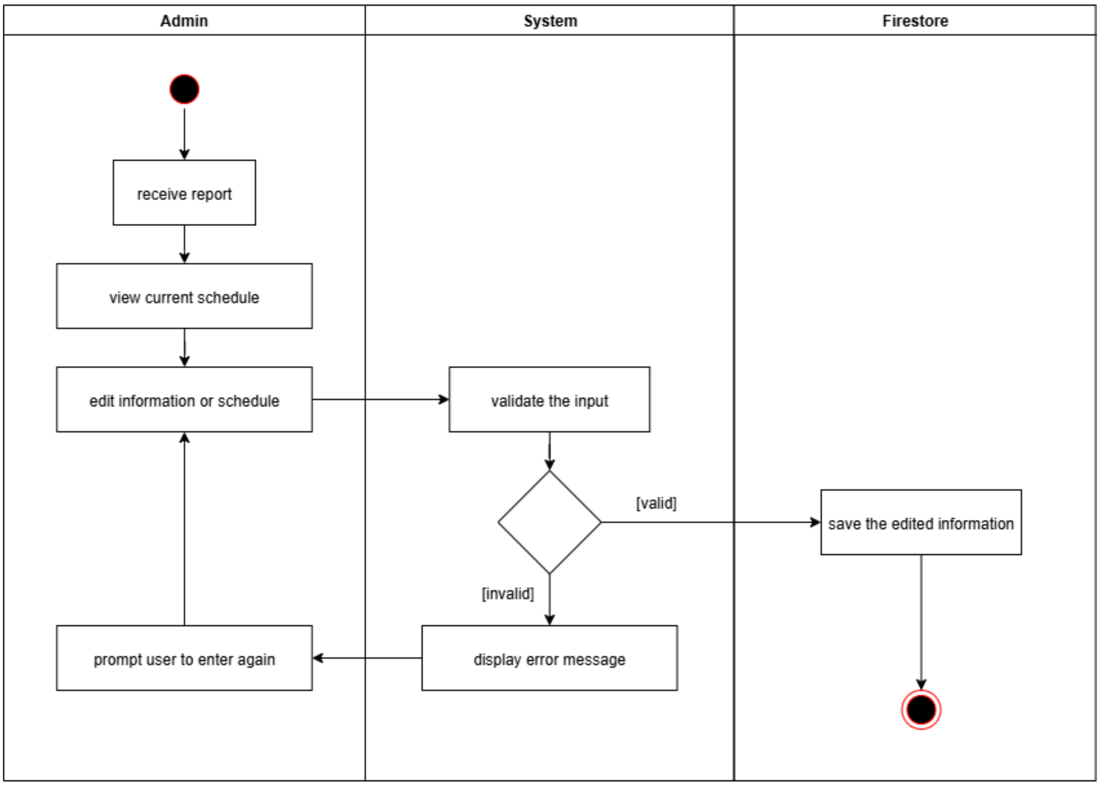

Figure 4.4: Activity Diagram for Manual Control and Monitoring Module

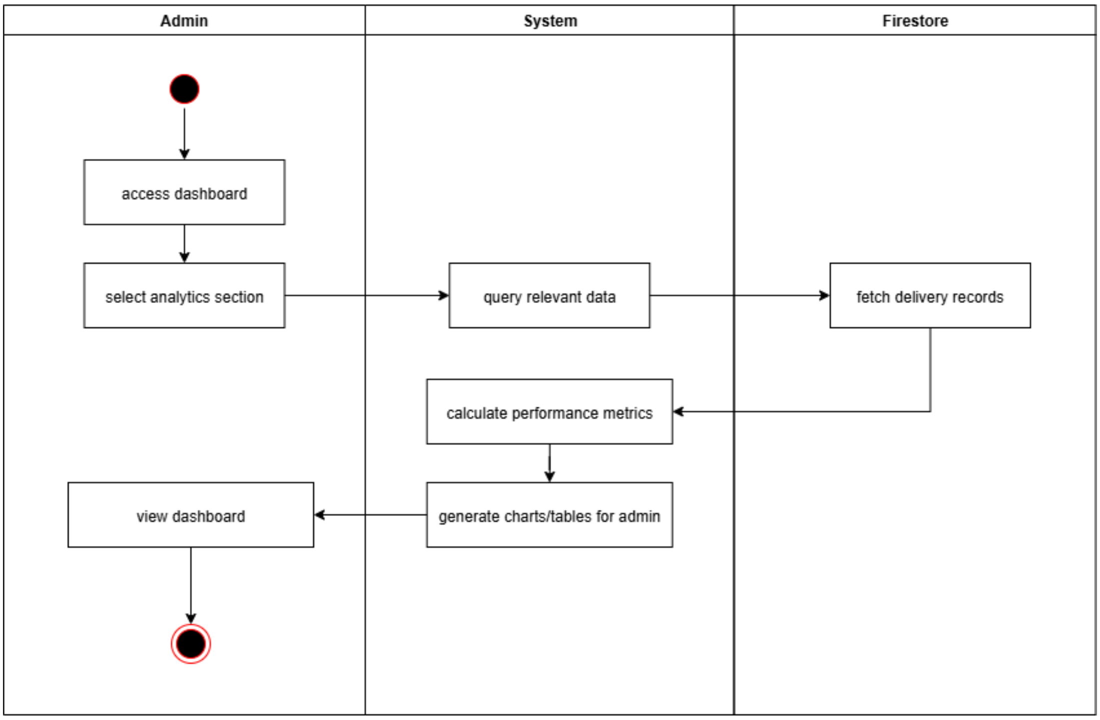

Figure 4.5: Activity Diagram for Dashboard and Analytics Module

CHAPTER 5: SYSTEM ANALYSIS AND DESIGN

5.1 Technologies Required

5.1.1 ReactJS

ReactJS was selected for the frontend development due to its component-based architecture, which enables modular, reusable and maintainable UI development. This approach is particularly advantageous for a logistics scheduling system where multiple user roles (e.g., admin, delivery teams, installation teams) require specialized interfaces. With React's reusable components, common interface elements such as navigation tabs, status badges and modals can be shared across views, reducing development redundancy and improving consistency.

Furthermore, React's use of a virtual DOM enhances performance by minimizing direct manipulation of the real DOM, allowing fast updates and dynamic rendering of delivery data, task statuses and live tracking information. This is essential for real-time systems where UI responsiveness is a key user requirement.

React also benefits from a rich ecosystem of libraries and integrations, including React Router for navigation between modules and Bootstrap for responsive, mobile-friendly UI components. These tools significantly reduced development effort and allowed rapid prototyping of complex user views tailored to each team's operational needs.

5.1.2 Tailwind CSS

To style the user interface, the system employs Tailwind CSS, a utility-first CSS framework that allows rapid development of responsive and consistent layouts directly within markup. Unlike traditional CSS or pre-built component libraries, Tailwind provides low-level utility classes that offer fine-grained control over design without enforcing fixed UI structures.

This approach is particularly effective for a logistics management system where the interface must accommodate complex tables, delivery cards, modals and dynamic indicators like delivery statuses and priority levels. Tailwind's flexibility made it easy to implement visually distinct states such as status-delivered, status-in-transit and priority-high with consistent styling.

Additionally, Tailwind integrates seamlessly with React components, supporting a clean separation of logic and presentation while keeping the styling maintainable and scalable as the project grows. Its built-in support for responsive breakpoints and dark mode also ensures that the UI remains usable across devices and user preferences.

5.1.3 Node.js with Express

Node.js with the Express.js framework is responsible for handling business logic, data routing, and backend processes. Node.js was selected for its non-blocking I/O model, which supports high concurrency that is deal for real-time scheduling applications where many users might be updating or viewing data simultaneously.

Express.js adds structure and simplicity to the backend with a clean routing mechanism and middleware system. It enables the creation of modular controllers for features like authentication, scheduling optimization, task management, manual overrides and reporting. The use of JavaScript across both frontend and backend also promotes code consistency and better team collaboration.

5.1.4 Firebase

For backend infrastructure and storage, the system uses Firebase, which includes Cloud Firestore, Firebase Authentication and Firebase Storage. Firestore acts as a real-time NoSQL database that holds all entities such as orders, employee profiles, delivery tasks and vehicle data. It supports live synchronization, meaning any changes made by the admin such as assigning a team or updating a status, are immediately reflected on user devices.

Firebase Auth manages secure logins for different roles and handles session control, ensuring that each user accesses only the appropriate views and data. This is essential for maintaining operational integrity in a multi-role system.

Meanwhile, Firebase Storage is utilized to hold binary files such as delivery proof images, uploaded signatures and photos submitted in failed delivery reports. The storage service is tightly integrated with Firestore, making it easy to retrieve media alongside related metadata during audits or performance reviews.

Overall, Firebase reduces the need for custom backend development and infrastructure maintenance, enabling rapid deployment while ensuring security, scalability and real-time performance.

5.2 System Configuration

This section describes the overall configuration of the system, providing a clear overview of the software architecture and the deployment environment. The architecture diagram illustrates the main components of the system and their interactions, while the deployment diagram shows how these components are distributed across different infrastructure layers. Together, these diagrams help to clarify the structure, technology stack and operational setup of the system.

5.2.1 Software Architecture Diagram

A software architecture diagram is a visual representation that outlines the major components of a software system and how they interact with each other. It helps stakeholders understand the structure, technology choices and communications within the system, enabling better design, development and maintenance decisions.

Figure 5.1 presents the software architecture for this project, showcasing a three-tier structure comprising the Presentation Tier, Application Tier and Data Tier. The Presentation Tier is implemented using a ReactJS web application, responsible for handling user interfaces and client-side interactions. The Application Tier utilizes a Node.js and Express server to process business logic and manage communication between the frontend and backend. The Data Tier leverages Firebase services, including Firestore, Firebase Auth and Firebase Storage, to store and manage application data securely. This architectural overview clearly demonstrates the separation of concerns and technology stack used within the system.

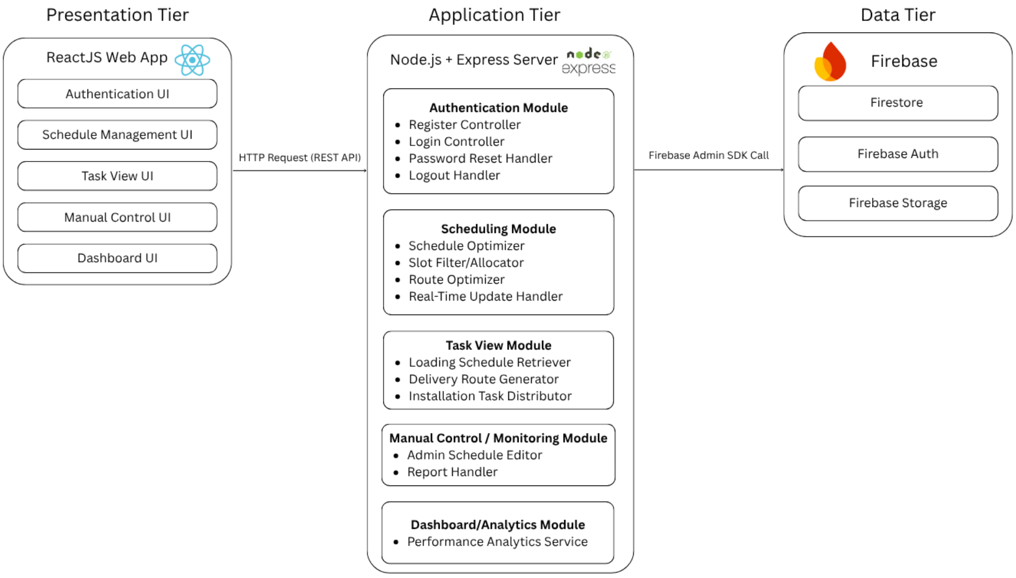

Figure 5.1: Software Architecture Diagram

5.2.2 Deployment Diagram

A deployment diagram is a type of UML diagram that illustrates the physical deployment of software components on hardware infrastructure, showing how different parts of a system are distributed across devices and environments. It helps stakeholders understand the real-world arrangement and interaction of system elements during execution.

Figure 5.2 depicts the deployment diagram for this project, detailing how the application's components are hosted and interact in a live environment. The client-side React application is deployed via Firebase Hosting and accessed through a web browser. Backend services, including authentication, scheduling and analytics modules, are implemented using Express with Node.js and deployed as Firebase Cloud Functions. These backend modules communicate with Firebase Services, such as Firestore, Firebase Authentication and Firebase Storage via the Admin SDK. This diagram provides a clear overview of the deployment strategy and the flow of communication between the system's main components.

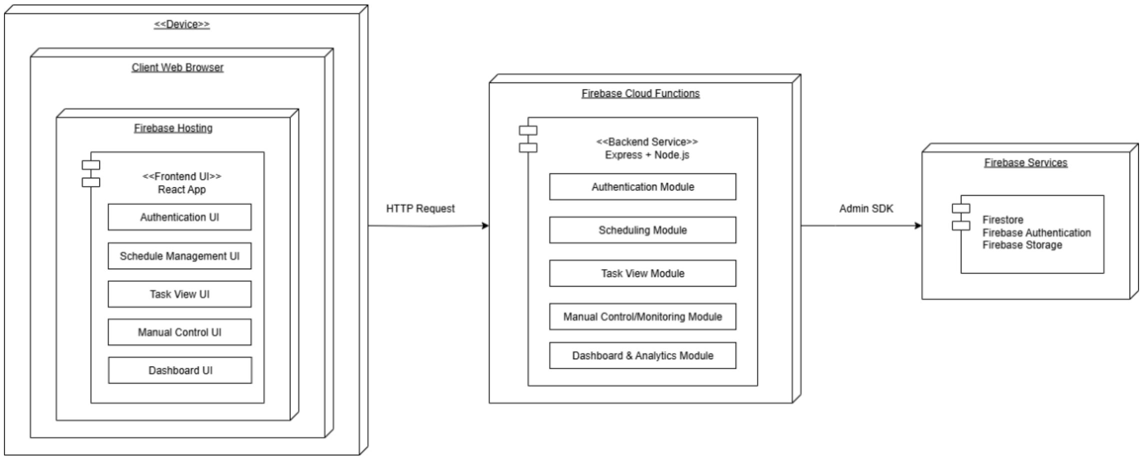

Figure 5.2: Deployment Diagram

5.3 Data Management and System Modelling

This section presents the design of how data is structured, stored and accessed within the system through two key modelling tools: the Entity-Relationship Diagram (ERD) and the Class Diagram. These diagrams serve different but complementary purposes in defining the data architecture and the interaction between different system components.

5.3.1 Business Rules and Entity Relationship Diagram (ERD)

Business rules are specific guidelines or constraints that define or restrict various aspects of business operations and processes within a system. They describe the logic, relationships and requirements that must be followed to ensure consistent data management and workflow, reflecting real-world policies or organizational practices. In the context of a database, business rules help establish how entities interact, what data is required, and what constraints must be enforced to maintain data integrity and support business objectives.

Here are the business rules for the ERD as depicted in Figure 5.3:

- Each customer (Customer) can place multiple orders (Order), but each order must be associated with one specific customer.
- An order is linked to one building (Building) where the delivery or service is to be performed; a building can receive multiple orders.
- Each order must be scheduled within one time slot (TimeSlot) and each time slot can be associated with multiple orders.
- Orders can contain multiple products (Product) and each product can appear in multiple orders, this many-to-many relationship is managed via the OrderProduct entity.
- Each OrderProduct record specifies the quantity of a specific product in an order.
- Each building belongs to a specific zone (Zone) and can only be linked to one zone, but each zone may contain multiple buildings.
- Each building may have unique access requirements, such as required pre-registration, access time windows, or special equipment needs.
- A truck (Truck) can be assigned to multiple zones via the TruckZone entity; each zone can have multiple trucks, but only one truck can be marked as the primary for a specific zone.
- Each lorry trip (LorryTrip) uses one truck and is assigned to both a warehouse team and a delivery team (Team).

- A time slot (TimeSlot) is assigned to one lorry trip, while a lorry trip may consist of multiple time slots.
- Teams (Team) can have multiple employees (Employee) and employees may be assigned to multiple teams, this many-to-many relationship is captured by the EmployeeTeamAssignment entity.
- Employees can write multiple reports (Report), but each report is associated with only one employee.
- Each product has specific handling and installation requirements, such as whether an installer team is needed or if extra time is required for installation or dismantling.
- The status, feedback and proof fields in Order capture delivery performance, customer responses and evidence of completion.
- Each zone (Zone) is uniquely identified and can be linked to multiple buildings and trucks.
- Employee records contain essential details, including their current status and team assignments.

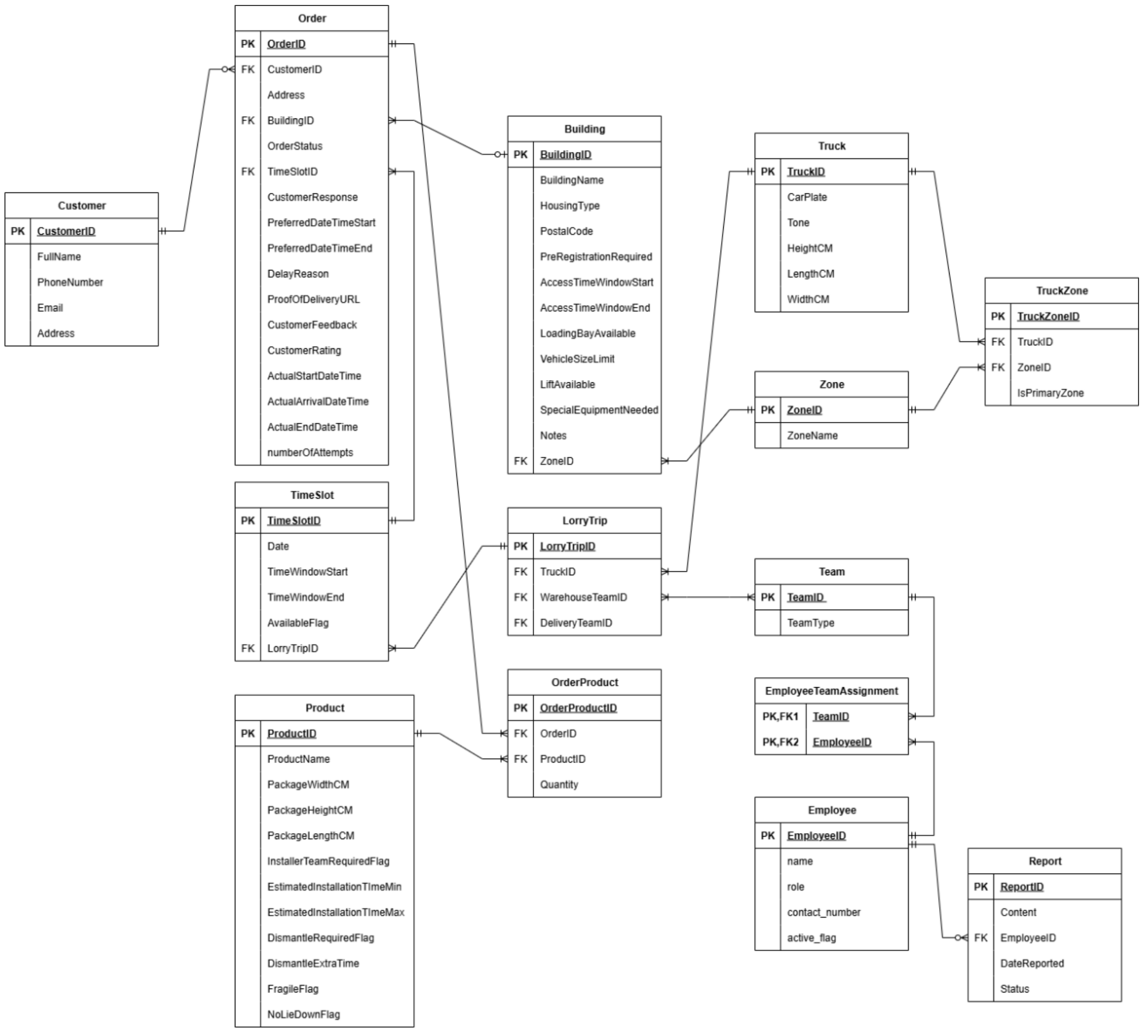

Figure 5.3: Entity Relationship Diagram (ERD)

5.3.2 Class Diagram

A class diagram is a type of static structure diagram in UML that visually represents the classes within a system, along with their attributes, operations and the relationships between them. It serves as a blueprint for system design, helping developers and stakeholders understand the core structure, data and interactions of the application.

Figure 5.4 illustrates the class diagram of the system, showcasing the main classes such as Customer, Order, Product, Building, Truck, TimeSlot, LorryTrip, Team, Employee and Report, along with their respective attributes and methods. The diagram also highlights the relationships between these classes, including associations such as "places," "includes," "is scheduled for," and "is assigned by." This class diagram provides a clear and detailed overview of the object-oriented structure of the system, defining how data is organized and how different entities interact within the software.

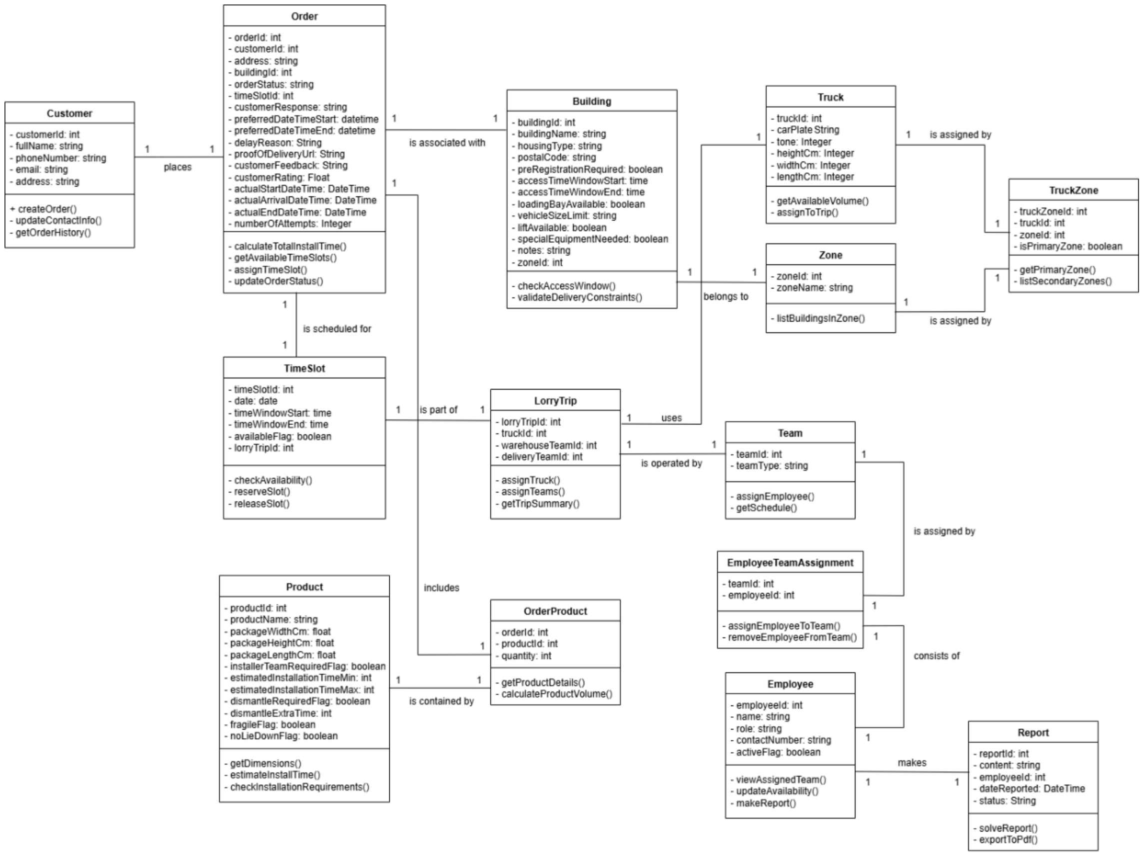

Figure 5.4: Class Diagram

5.4 User Interface (UI) Design

5.4.1 Admin Interface

The Admin Interface integrates the Manual Control and Monitoring Module with the Dashboard and Analytics Module, enabling administrators to oversee scheduling operations and team performance in one place. Admins can edit delivery schedules, reassign teams and resolve user-submitted reports as needed.

The dashboard provides real-time insights into key metrics such as punctuality and task completion rates, supporting better decision-making. Figure 5.5 to Figure 5.15 illustrate the main components of this interface, including schedule management, report handling and performance tracking.

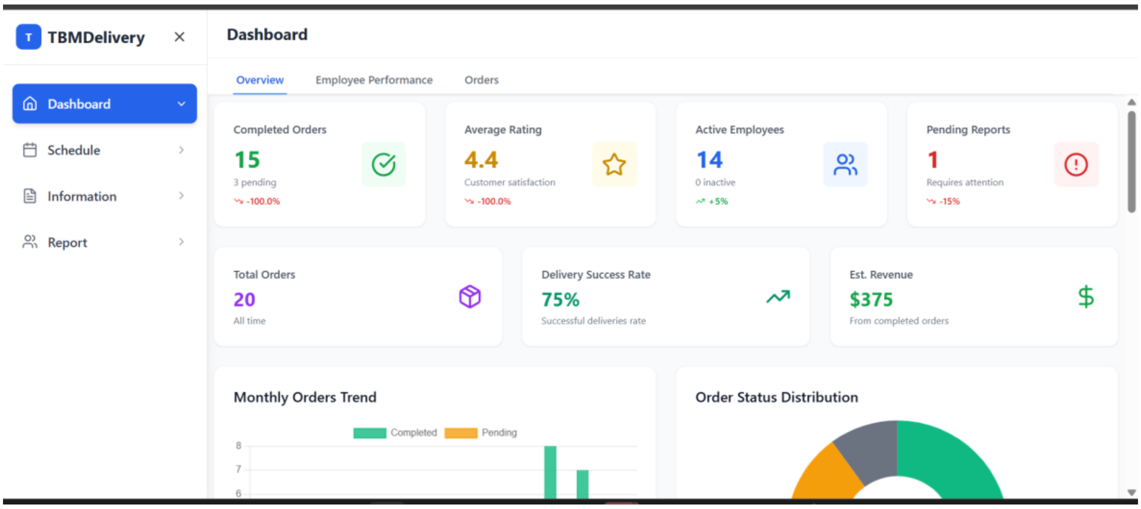

Figure 5.5: Overall Dashboard

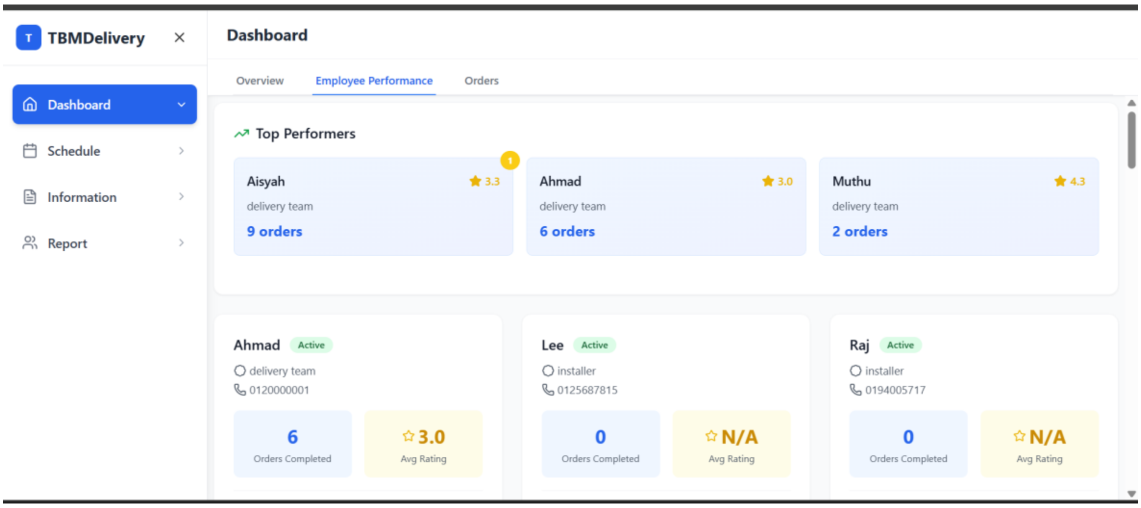

Figure 5.6: Employee Performance Dashboard

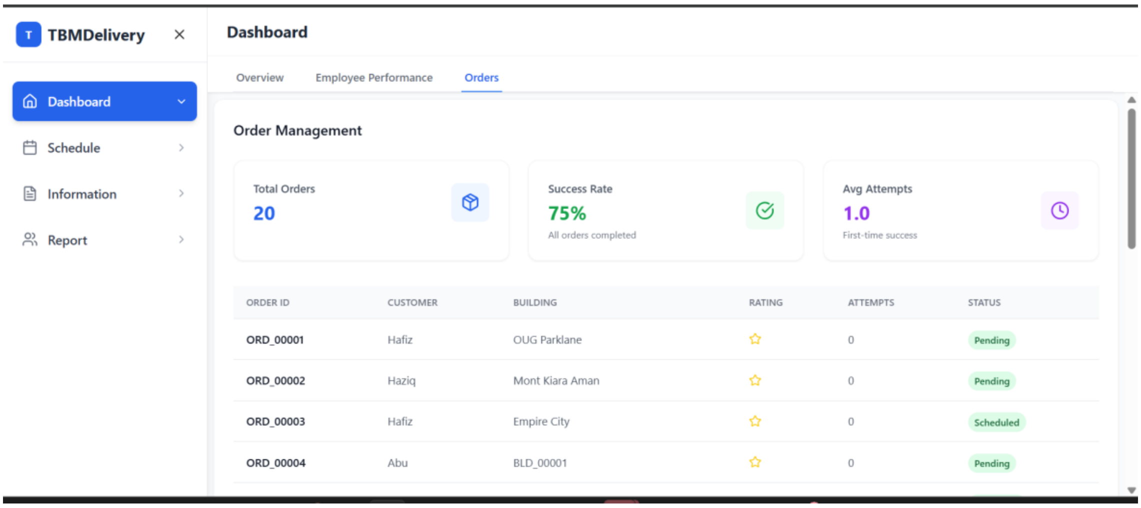

Figure 5.7: Order Delivery Dashboard

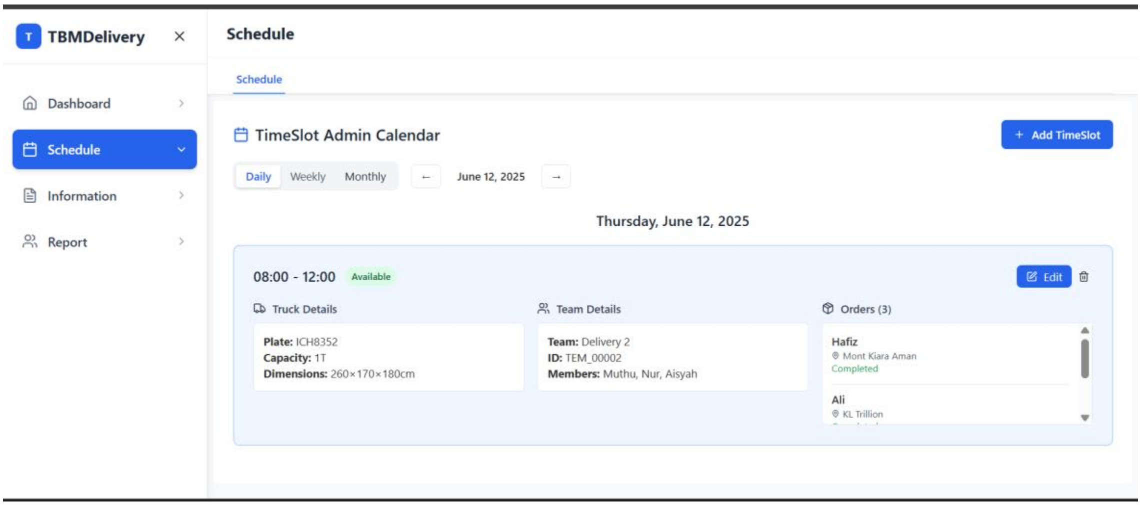

Figure 5.8: Delivery Schedule tab

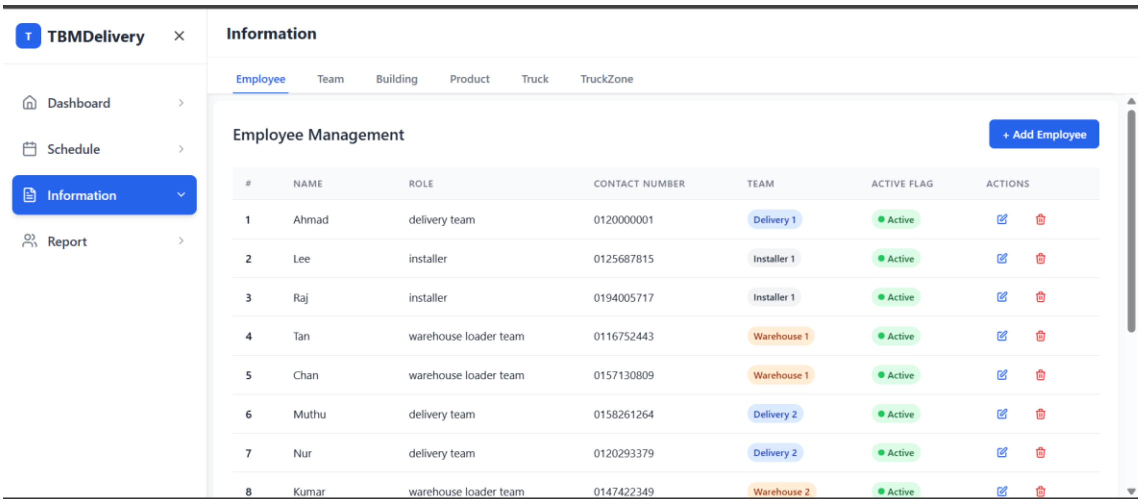

Figure 5.9: Employee Information tab

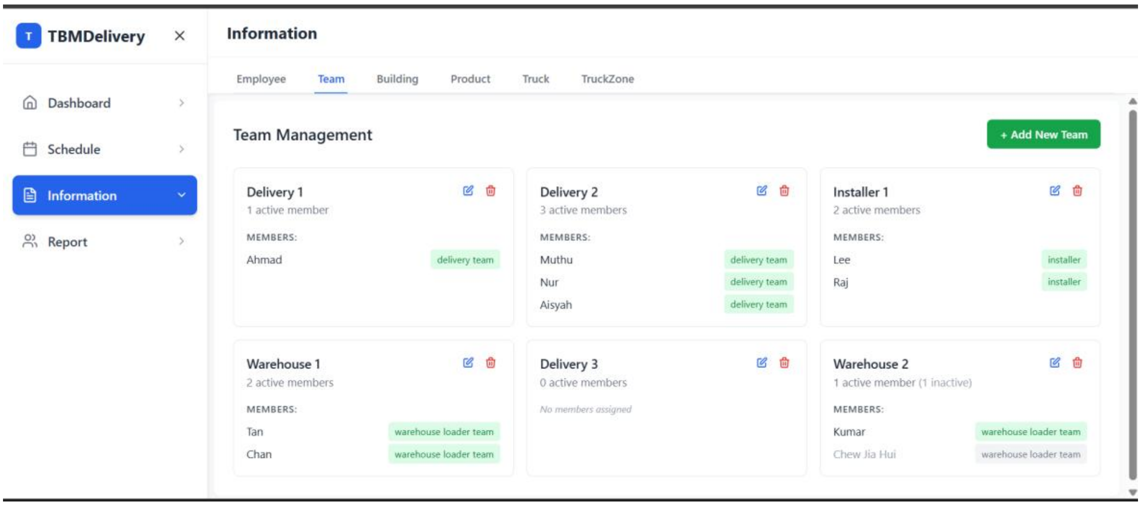

Figure 5.10: Team Information tab

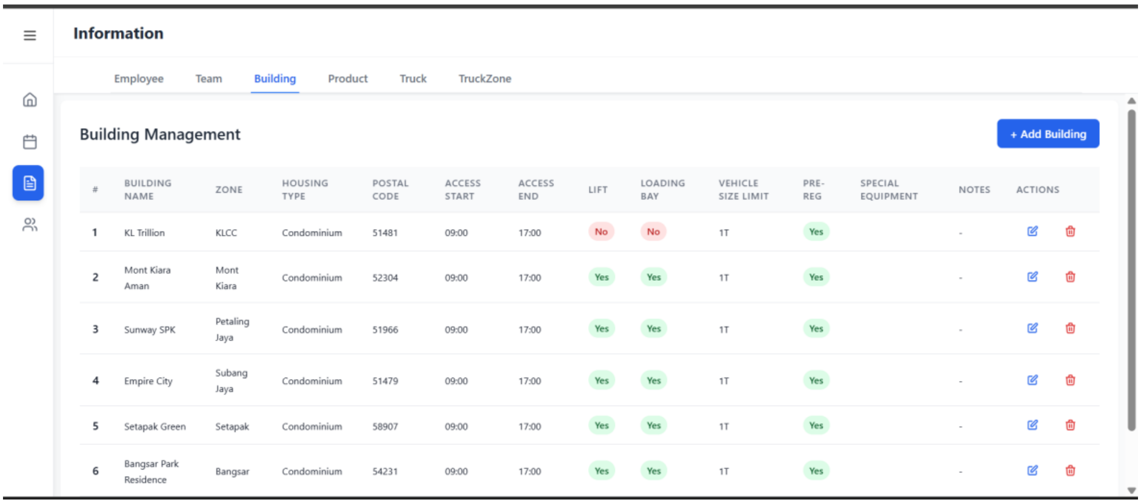

Figure 5.11: Building Information tab

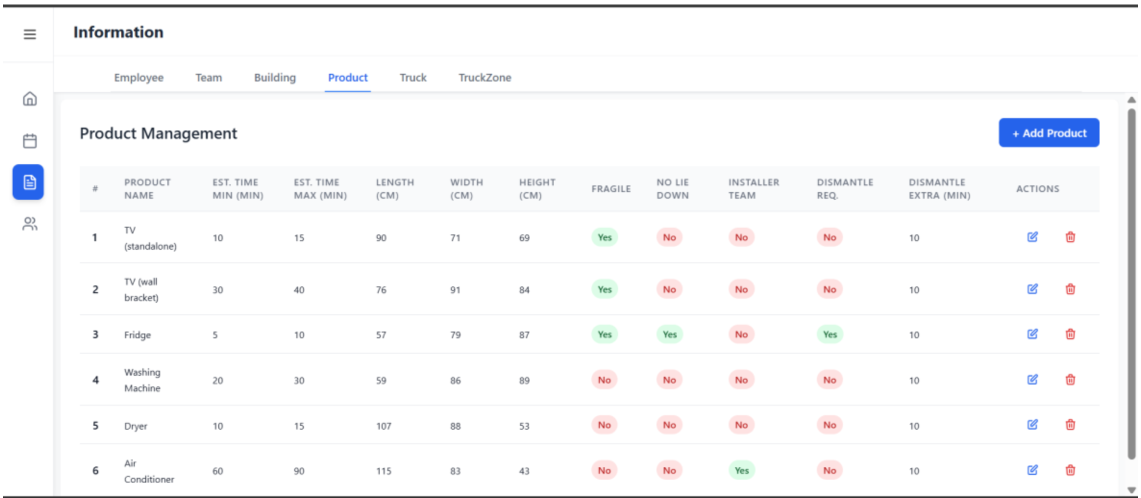

Figure 5.12: Product Information tab

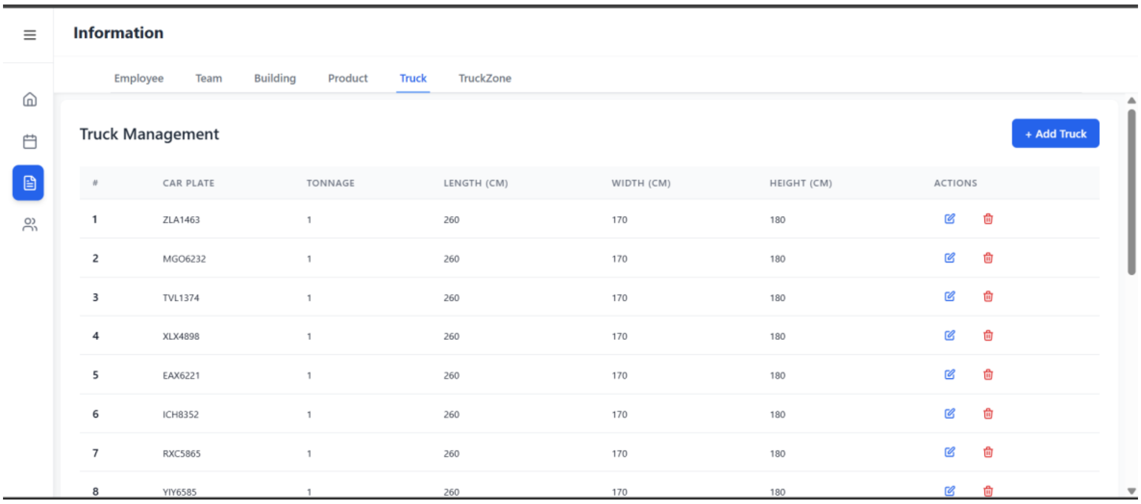

Figure 5.13: Truck Information tab

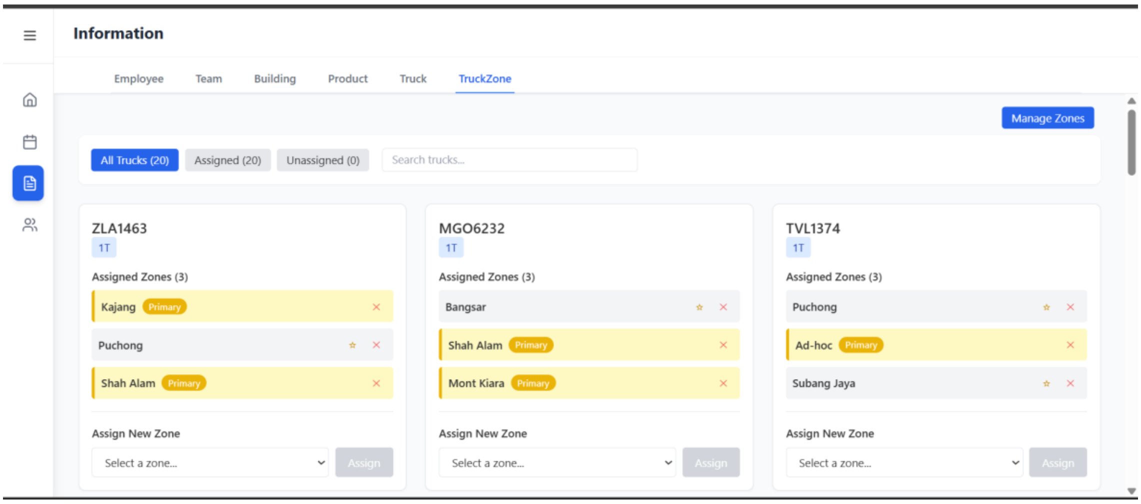

Figure 5.14: Assigned delivery zones per truck view

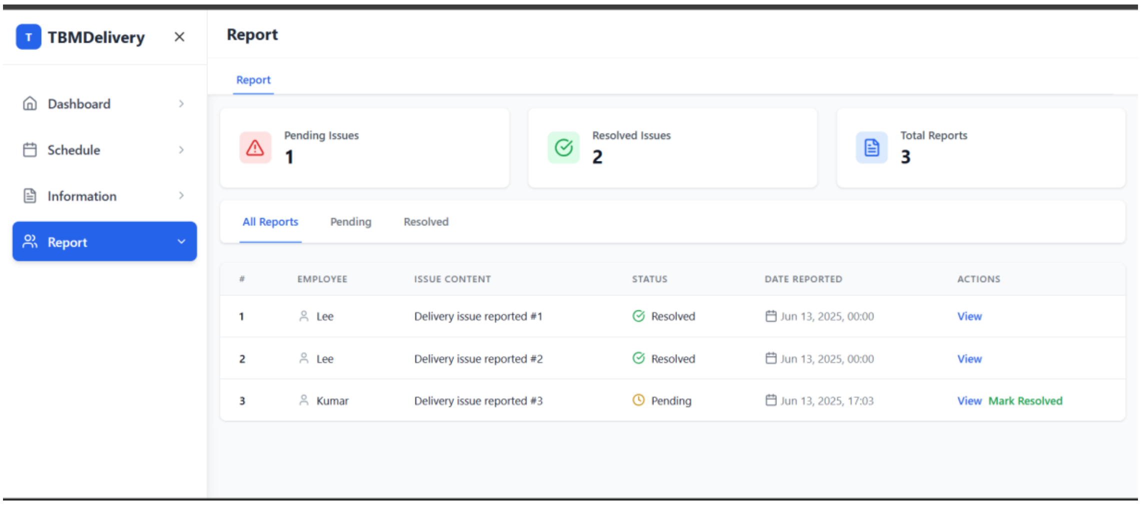

Figure 5.15: User Report Management tab

5.4.2 Delivery Team Interface

The Delivery Team Interface is designed for delivery teams to access their assigned delivery schedules and routes. It provides a clear view of daily tasks, delivery addresses and optimized travel sequences to ensure on-time service. Figure 5.16 displays the schedule view for the delivery team.

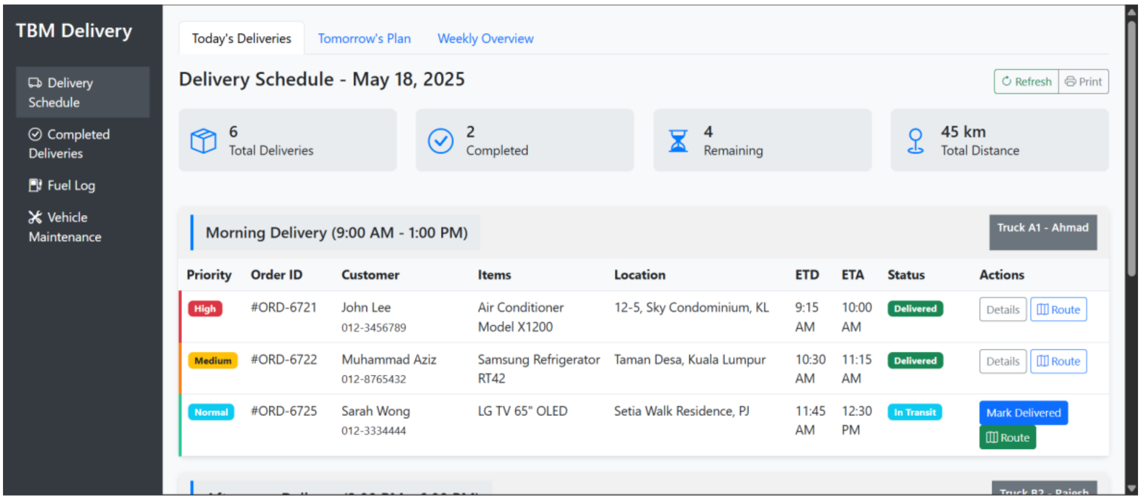

Figure 5.16: Delivery Team View

5.4.3 Outsourced Installer Interface

This interface supports outsourced installation personnel by showing their assigned tasks along with the relevant product, location and estimated installation duration. It helps the team stay informed of upcoming jobs and manage time effectively. Figure 5.17 shows the interface layout used by the installers.

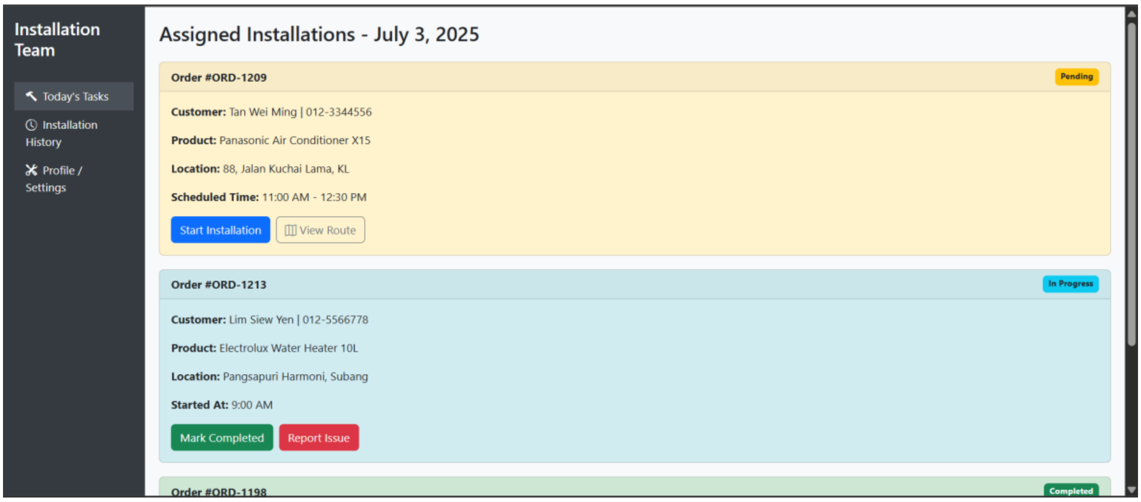

Figure 5.17: Outsourced Installer View

5.4.4 Warehouse Loading Team Interface

The Warehouse Interface enables loading staff to view which products need to be prepared for loading, sorted by truck and delivery zone. It supports efficient staging and loading operations. Figure 5.18 illustrates the loading schedule screen accessible to warehouse teams.

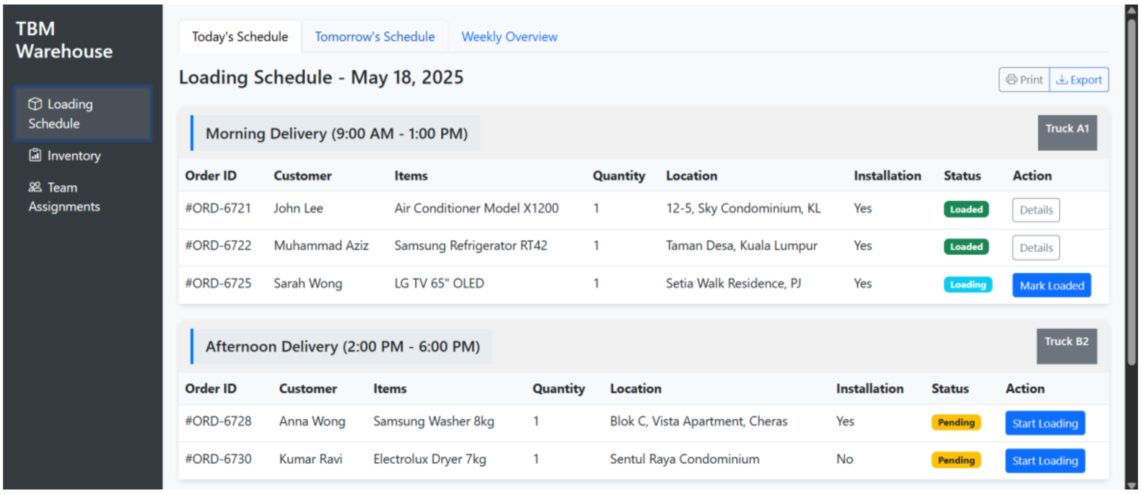

Figure 5.18: Warehouse Loading Team View

CHAPTER 6: CONCLUSION

The first phase of this project has laid a strong foundation for the development of an intelligent, constraint-aware logistics scheduling system tailored to the electrical appliance industry. Through a combination of stakeholder engagements, literature review, existing system analysis, and iterative prototyping, this project successfully identified key operational gaps and proposed a modular web-based platform that addresses the unique requirements of appliance delivery and installation.

The system design introduced several innovations that go beyond traditional delivery scheduling solutions. These include installation time estimation based on product complexity, compliance with residential zone constraints, optimized route and truck space planning, and dynamic rescheduling in response to real-world disruptions. Stakeholder interviews confirmed that such features are currently missing from the industry's existing systems, leading to inefficiencies, missed delivery windows, and poor coordination across teams.

In particular, the project addresses critical gaps observed in both academic literature and commercial tools. While many logistics systems focus on route optimization or load efficiency, few consider installation time, team reassignment, or delivery constraints imposed by high-rise residential buildings. By integrating these diverse constraints into a unified scheduling engine, this project proposes a more holistic and realistic approach to appliance logistics.

Although the system is not yet fully implemented at this stage, the first phase has achieved significant milestones. Functional requirements have been thoroughly analysed, a system architecture has been established, and early-stage UI prototypes have been created. Feedback from TBM Electrical has been continuously incorporated, ensuring that the solution remains practical and applicable in real business operations.

Moving into Phase II, the focus will shift toward full-scale system development, integration of backend intelligence, advanced testing, and stakeholder-driven refinements. The successful completion of Phase I ensures that the project is well-positioned to deliver a robust, end-to-end solution that enhances operational efficiency, reduces scheduling errors, and ultimately improves the customer experience in the electrical appliance delivery and installation domain.

REFERENCES

Alcaraz, J. J., Caballero-Arnaldos, L., & Vales-Alonso, J. (2019). Rich vehicle routing problem with last-mile outsourcing decisions. *Transportation Research Part E: Logistics and Transportation Review, 129*, 263–286. https://doi.org/10.1016/j.tre.2019.08.004

Arora, S., Pimprale, V., & Deshmukh, N. (2025). Analysis of metaheuristic optimization algorithms for solving vehicle routing problem with time windows. *2025 International Conference on Automation, Electronics and Technology (ICAET)*, 1–7. https://doi.org/10.1109/icaet63349.2025.10932238

Berhan, E., Beshah, B., Kitaw, D., & Abraham, A. (2014). Stochastic vehicle routing problem: A literature survey. *Journal of Information & Knowledge Management, 13*(03), 1450022. https://doi.org/10.1142/s0219649214500221

Engesser, V., Rombaut, E., Vanhaverbeke, L., & Lebeau, P. (2023). Autonomous delivery solutions for last-mile logistics operations: A literature review and research agenda. *Sustainability, 15*(3), 2774. https://doi.org/10.3390/su15032774

Giuffrida, N., Fajardo-Calderin, J., Masegosa, A. D., Werner, F., Steudter, M., & Pilla, F. (2022). Optimization and machine learning applied to last-mile logistics: A review. *Sustainability, 14*(9), 5329. https://doi.org/10.3390/su14095329

Gutierrez-Rodríguez, A. E., Conant-Pablos, S. E., Ortiz-Bayliss, J. C., & Terashima-Marín, H. (2019). Selecting meta-heuristics for solving vehicle routing problems with time windows via meta-learning. Expert Systems with Applications, 118, 470–481. https://doi.org/10.1016/j.eswa.2018.10.036

He, Z., Zhang, M., Lin, C., Zhao, J., Shi, K., & Ai, Z. (2024). Optimizing heterogeneous vehicle routes for urban distribution considering the three-dimensional bin packing problem of electric meters. *IEEE Access*, 1–1. https://doi.org/10.1109/access.2024.3363417

Ibrahim, A., Ishaya, J., Abdulaziz, R., & Samuel, S. (2020). Solving some variants of vehicle routing problem with branch-and-cut and column generation algorithms. *Open Journal of Mathematical Sciences, 4*(1), 63–73. https://doi.org/10.30538/oms2020.0095

Lahyani, R., Khemakhem, M., & Semet, F. (2015). Rich vehicle routing problems: From a taxonomy to a definition. *European Journal of Operational Research, 241*(1), 1–14. https://doi.org/10.1016/j.ejor.2014.07.048

Li, Y., Soleimani, H., & Zohal, M. (2019). An improved ant colony optimization algorithm for the multi-depot green vehicle routing problem with multiple objectives. *Journal of Cleaner Production, 227*, 1161–1172. https://doi.org/10.1016/j.jclepro.2019.03.185

Liu, F., Lu, C., Gui, L., Zhang, Q., Tong, X., & Yuan, M. (2023, March 1). Heuristics for vehicle routing problem: A survey and recent advances. *arXiv*. https://arxiv.org/abs/2303.04147

Liu, X., Chen, Y., Por, L. Y., & Ku, C. S. (2023). A systematic literature review of vehicle routing problems with time windows. *Sustainability, 15*(15), 12004. https://doi.org/10.3390/su151512004

Mageswari, N. (2024). Vehicle routing problem (VRP) using genetic algorithm. *International Research Journal of Advanced Engineering and Science, 9*(3), 1–3. https://irjaes.com/wp-content/uploads/2024/06/IRJAES-V9N2P260Y24.pdf

Mohammad, W. A., Diab, Y. N., Elomri, A., & Triki, C. (2023). Innovative solutions in last mile delivery: Concepts, practices, challenges, and future directions. *Supply Chain Forum: An International Journal, 24*(2), 151–169. https://doi.org/10.1080/16258312.2023.2173488

Molina, J. C., Salmeron, J. L., Eguia, I., & Racero, J. (2020). The heterogeneous vehicle routing problem with time windows and a limited number of resources. *Engineering Applications of Artificial Intelligence, 94*, 103745. https://doi.org/10.1016/j.engappai.2020.103745

Mor, A., & Speranza, M. G. (2022). Vehicle routing problems over time: A survey. *Annals of Operations Research*. https://doi.org/10.1007/s10479-021-04488-0

Rahmani Hosseinabadi, A. A., Slowik, A., Sadeghilalimi, M., Farokhzad, M., Babazadeh Shareh, M., & Sangaiah, A. K. (2019). An ameliorative hybrid algorithm for solving the capacitated vehicle routing problem. *IEEE Access, 7*, 175454–175465. https://doi.org/10.1109/access.2019.2957722

Royappa, A., Venkatesh, K., Purushothaman, N., Moorthy, A., & Solainayagi, P. (2024). AI-driven optimization for freight and logistics management using predictive analytics. *2024 International Conference on Cybernetics, Communication and Computing (CyberCom)*, 677–682. https://doi.org/10.1109/cybercom63683.2024.10803177

Simeonova, L., Wassan, N., Salhi, S., & Nagy, G. (2018). The heterogeneous fleet vehicle routing problem with light loads and overtime: Formulation and population variable neighbourhood search with adaptive memory. *Expert Systems with Applications, 114*, 183–195. https://doi.org/10.1016/j.eswa.2018.07.034

Zhang, H., Ge, H., Yang, J., & Tong, Y. (2021). Review of vehicle routing problems: Models, classification and solving algorithms. *Archives of Computational Methods in Engineering*. https://doi.org/10.1007/s11831-021-09574-x

APPENDIX

Appendix A: Collaboration Letter

The signed collaboration letter from TBM Electrical can be accessed at the following Google Drive link:

https://drive.google.com/file/d/13V90hqIQJHydP6qc-x02sMPIe1Ci_Pw7/view

Appendix B: Project Timeline Spreadsheet

The full project task breakdown and timeline used during the development process is available at the link below:

https://docs.google.com/spreadsheets/d/1CIHZBOJNcvRwD7MO_n1B1wddGCHjT6S-fZGaE3tqhIc/edit?usp=sharing