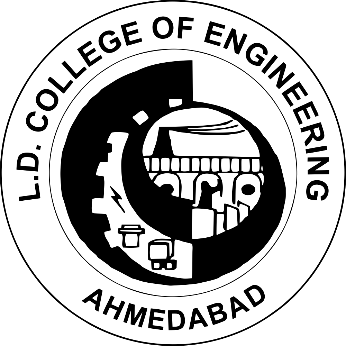
A Project Report on

The Final Final

At



### L.D. COLLEGE OF ENGINEERING [MCA]

Navrangpura, Ahmedabad.

As A Fulfilment for the Degree of Master of Computer Application (M. C. A.)

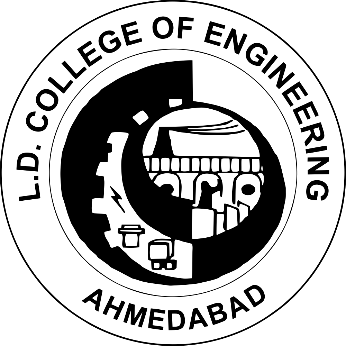
2023-2024

|  |  |
| --- | --- |
| **Guided By:**  **Prof.** Shital Solanki | **Submitted By:**  **Mr.** Abhishek Zakhariya  **Enroll no: 225160694039** |

L.D. COLLEGE OF ENGINEERING [MCA] Navrangpura, Ahmedabad.

Affiliated To

### Gujarat Technological University

# L.D. COLLEGE OF ENGINEERING

NAVRANGPURA, AHMEDABAD- 380015

**CERTIFICATE**

This is to certify that Software Development Project entitled **The Final Final** is developed and submitted to Gujarat Technological University by **Mr**. **Abhishek Zakhariya (Enrollment No. 225160694039)** fulfilment of the requirement of MCA **Semester-4** for the award of the **Master of Computer Applications (MCA)** in the year 2022- 2023.

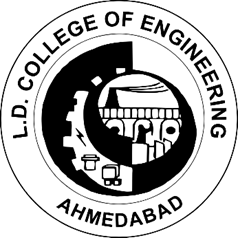
This is the original work and carried out under my guidance and supervision. We further certify that to the best of our knowledge and belief the matter presented in this project report is Bonafide work.

### DATE OF SUBMISSION:

**Prof. Shital Solanki Dr. Hitieshi Diwanji Prof.**

Asst. Professor (IT), HOD, IT-MCA

LDCE, Ahmedabad LDCE, Ahmedabad (External External)

**L.D. COLLEGE OF ENGINEERING NAVRANGPURA, AHMEDABAD 380015**

**DECLARATION**

We hereby declare that the Project report submitted with the Project entitled “**The Final Final**” submitted in partial fulfilment for the degree of **Master of Computer Application** in L.D College of Engineering-MCA Program to Gujarat Technological University, Ahmedabad, is a bonafide record of original project work carried out by us at L. D College of Engineering under the guidance of **Prof. Shital Solanki** and External Guide **Mr. Chintan Gor** (CTO, eSparkBiz Technologies PVT. LTD.), and no part of this report has been directly copied from any student’s report or taken from any other source, without providing due reference.

### Name of Student: Signature of Student

**Abhishek Zakhariya ……………………….**

### 225160694039

Company certificate

Acknowledge

It is our great Pleasure to express our deep sense of gratitude to our college L.D. College of Engineering.

We wish to express our sincere gratitude to our internal project guide **Prof. Shital Solanki** for providing her invaluable guidance, comments and suggestions on the project. Her prompt inspiration on project. Her, valuable experience and dynamism have enabled us to succeed in the project. We also Appreciate External Guide who is the owner and founder of Habri investment **Happy Abraham Manda** where the model of the system has been developed for management of Inventory of Medium and small SMES

We would also like to thank the whole MCA Department for the knowledge imparted in us that has also helped us to be able to achieve this project

#### Thank you so much,

**Mr. DANIEL OVERTON UNIQUE MAPALA: …………………………………**

#### Mr. NOOR RAHIM SHIR AFZAL: ………………………………………….

## Abstract

It is our great Pleasure to express our deep sense of gratitude to our college L.D. College of Engineering.

We wish to express our sincere gratitude to our internal project guide **Prof. Shital Solanki** for providing her invaluable guidance, comments and suggestions on the project. Her prompt inspiration on project. Her, valuable experience and dynamism have enabled us to succeed in the project. We also Appreciate External Guide who is the owner and founder of Habri investment **Happy Abraham Manda** where the model of the system has been developed for management of Inventory of Medium and small SMES

We would also like to thank the whole MCA Department for the knowledge imparted in us that has also helped us to be able to achieve this project

#### Project Profile

**Project Definition:** Logistic and Supply Chain Management

**Project Title:** The Final Final

**Company Name:** eSpakrBiz Technologies Pvt Ltd

**Tools:**

**Duration:**  January 22, 2024 to Continue

**Team Size:** 6 members

**Internal Guide:** Prof Hetal Pate (Assistance Professor, MCA DDU)

**External Guide:** Chintan Gor (CTO at eSparkBiz Technologies Pvt Ltd)

**Project Description:**

* The Final-Final is a web-based logistics platform designed to streamline delivery management for businesses of all sizes. It offers a centralized platform for managing shipments, tracking deliveries in real-time, and facilitating communication between all stakeholders involved (businesses, drivers, and customers).
* Here The User or we can say the client will be able to login or sing up for the account. Then he will be able to create their profile.
* After the login the user will be able to see the make pick up request in which he can send their household items like furniture can be deliver from one location to another.
* They will be able to specify how many items they wanted to delivery with the more precise information of items as well as destination and location.
* They able to schedule their pickups according to their comfort. They simply need to make a request based on the date and can make up the pick request for the same.
* The quotation will be generated according to the number of items they will transporting and how far the destination is from the location.
* When user makes their request, it will push the notification to the admin and all nearby drivers and whichever driver will accept the request will fulfil that request.
* The system will support the online as well as offline payment methods to provide more user conveniency.
* There will be a dedicated separate user, admin and drivers chat room in which they can communicate seamlessly.
* The admin can manage user accounts, configure system settings, oversee platform performance, generate reports for data-driven decision-making, and establish integrations with external business systems.
* The Drivers can access and manage assigned deliveries through a dedicated mobile app, utilize optimized routes for efficient navigation, update delivery statuses in real-time, and communicate with admins for clarifications or reporting any issues encountered during the route.

**SRS for The Final-Final**

**Introduction**

**1.1 Purpose**

This Software Requirements Specification (SRS) document outlines the functionalities and requirements for The Final-Final, a cloud-based logistics management application. The Final-Final aims to revolutionize the way businesses of all sizes manage their logistics and supply chain operations. It goes beyond basic tracking and scheduling, offering a comprehensive and intuitive platform designed to optimize efficiency, reduce costs, and enhance customer satisfaction.

**1.2 Definition**

The Final-Final is a web-based application accessible through a user-friendly interface. It facilitates streamlined logistics management by providing functionalities for client management, shipment scheduling, real-time tracking, driver management, performance analysis, and integration with existing business systems. It will also available as SaaS for customisable some industry requirements.

**1.3 Scope**

This initial version of The Final-Final will focus on core functionalities encompassing:

• Client Management: Create and manage client profiles, preferences, and shipping history. Generate quotes, accept orders, and maintain communication logs.

• Shipment Management: Plan and schedule shipments, defining pick-up and delivery locations, deadlines, and cargo details. Optimize routes for efficiency.

• Real-Time Tracking: Gain live GPS updates and estimated arrival times for every shipment. Monitor deviations and delays, fostering transparent communication.

• Driver Management: Assign shipments based on location, qualifications, and availability. Communicate with drivers through the platform or app for efficient delivery coordination.

• Performance Analysis: Generate reports on shipment performance, driver activity, and client engagement. Identify areas for improvement and make data-driven decisions.

• System Integration: Integrate seamlessly with popular CRM, accounting, and e-commerce platforms through APIs, eliminating data silos and automating workflows.

Additional features, such as warehousing management, advanced route optimization algorithms, and predictive analytics, may be introduced in future versions based on user needs and feedback.

**1.4 Intended Audience**

This SRS document is intended for various stakeholders involved in the development and use of The Final-Final:

• Project Managers: Overseeing the development process, ensuring timely delivery, and managing project resources.

• Software Developers: Building the platform functionalities and ensuring optimal performance.

• System Analysts: Analysing user needs and translating them into concrete system requirements.

• Quality Assurance Testers: Conducting rigorous testing to guarantee platform accuracy and reliability.

• Admin Users: Responsible for system configuration, user management, and access control.

**1.5 Intended Users**

The Final-Final caters to three primary user groups:

• Admin Users: Manage the overall platform, configure settings, create user accounts for clients and drivers, and monitor system performance.

• Customer Users: Manage their logistics operations through the platform. This includes creating shipment requests, tracking deliveries, generating reports, and communicating with drivers and admins.

• Driver Users: Utilize the mobile app to receive assigned deliveries, navigate optimized routes, track their progress, and update delivery statuses.

**2. Overall Description**

**2.1 Environmental Characteristics**

**2.1.1 Hardware/Software**

**• Hardware:**

The Final-Final is a web-based application accessible through any standard internet browser. This eliminates the need for specific hardware requirements beyond a device with a reliable internet connection. Users can access the platform from desktops, laptops, or tablets with minimal processing power requirements to run the web browser effectively.

**• Software:**

The Final-Final leverages a secure and scalable cloud-based infrastructure. This ensures platform availability, reliability, and eliminates the need for software installation on individual user devices. The platform itself is built using secure and industry-standard programming languages and frameworks.

**• Supported Browsers:**

The platform is optimized for functionality and performance on the following web browsers (latest versions recommended):

o Google Chrome

o Mozilla Firefox

o Microsoft Edge

o Apple Safari

**2.1.2 Peripherals**

While The Final-Final itself doesn't require specific peripherals, standard computer equipment is necessary for user interaction:

• Computer Monitor: To view the platform interface and interact with functionalities.

• CPU (Central Processing Unit): Provides the processing power to run the web browser and access the application effectively. Minimum processing power requirements will be documented for optimal performance.

• Internet Connection: A reliable internet connection with sufficient bandwidth is essential for accessing The Final-Final's cloud-based functionalities. Bandwidth requirements will be documented for different functionalities (e.g., basic data entry vs. real-time tracking).

• Mouse: For user interaction with the platform interface.

• Keyboard: For data entry and navigation within the application.

**2.1.3 People**

The successful implementation of The Final-Final relies on various user roles with distinct responsibilities:

**• Admin Users:**

These users act as system administrators, configuring platform settings, managing user accounts (clients and drivers), and monitoring overall system performance. They ensure smooth operation and access control within The Final-Final.

**• Customer Users:**

As the primary business users, they manage their logistics operations through the platform. This encompasses activities like creating shipment requests, tracking deliveries in real-time, generating reports for analysis, and communicating with both drivers and admins for efficient coordination.

**• Driver Users:**

These users rely on the dedicated mobile app to receive assigned deliveries. The app facilitates route navigation based on optimized routes, allows drivers to track their progress, and update delivery statuses, keeping all parties informed.

**2.2 User Needs**

• Businesses yearn for a user-friendly and intuitive logistics management system that saves time and reduces complexity.

• Real-time shipment tracking and proactive communication are crucial for exceeding client expectations and building trust.

• Optimized routes and efficient driver management are essential for minimizing costs and maximizing delivery speed.

• Data-driven insights and actionable reports empower companies to continuously improve their logistics operations.

• Mobile accessibility for drivers fosters on-the-go management and simplifies delivery execution.

**2.3 Assumptions and Dependencies**

• Users have basic computer and internet literacy.

• Reliable internet connection is available for accessing the platform.

• Mobile devices for drivers compatible with the app are readily available.

• Integration with existing business systems may be required, prompting additional analysis and configuration.

**2.4 Overview:**

The Final-Final will provide seamless interaction between all the stakeholders involved in the project and it will reduce the communication gap between them and provide the smooth transportation and logistics services to the clients. It will also be available as SaaS for the other businesses operating in the same industry.

**2.5 Definitions and Acronyms**

**SRS: Software Requirements Specification**

**API: Application Programming Interface**

**CRM: Customer Relationship Management**

**GPS: Global Positioning System**

**2.6 References:**

College provided template or guiding document to make SRS.

IEEE Guide for SRS: https://ieeexplore.ieee.org/document/278253

https://www.geeksforgeeks.org/software-requirement-specification-srs-format

**3 Functional Requirements**

**3.1 Admin Users**

**• Manage user accounts:**

o Create and manage client and driver accounts.

o Assign appropriate permissions based on user roles.

**• Oversee platform configuration:**

o Define system settings and configurations.

o Manage integrations with external business systems through

**APIs.**

**• Monitor overall performance:**

o Analyse system uptime and resource utilization.

o Review error logs and identify potential issues.

**• Generate reports:**

o Create reports on user activity, shipment performance, and other relevant data.

o Leverage reports for informed decision-making.

**• Manage Access:**

o Able to manage users access as per requirement to scale up the business.

o Able to create another admin staff for the same.

**3.2 Customer Users**

**• Manage shipment requests:**

o Create new shipment requests.

o Specify pick-up and delivery locations, cargo details, and deadlines.

o Edit or cancel existing shipment requests (within defined timeframes).

**• Track shipments in real-time:**

o View shipment location updates on a map.

o Access estimated arrival times and track delivery progress.

o Receive notifications about potential delays or unforeseen circumstances.

**• Analyse shipment data:**

o Generate reports on shipment history, associated costs, and performance metrics.

o Gain insights into logistics efficiency and identify areas for improvement.

**• Communicate with stakeholders:**

o Initiate real time communication with drivers and admins through the platform.

o Send inquiries, request status updates, and report any issues related to shipments.

**3.3 Driver Users**

**• Manage assigned deliveries:**

o Access and view assigned deliveries through the mobile app.

o Review shipment details, including pick-up and delivery locations, cargo information, and deadlines.

**• Navigate optimized routes:**

o Utilize the platform's route optimization based on traffic conditions and distance calculations.

o Receive turn-by-turn navigation instructions for efficient delivery completion.

**• Update delivery statuses:**

o Mark deliveries as picked up, in transit, and delivered in real-time.

o Provide accurate updates to keep customers and admins informed.

**• Communicate with admins:**

o Initiate communication with admins through the app for clarifications on delivery details.

o Report any issues encountered during the route (e.g., road closures, traffic accidents).

**7. System Desing**

**7.1 Architecture Diagrams**

**7.1.1 Usecase Diagrams**

**1. System Overview**

"The Final Final" is a delivery system that facilitates the process of scheduling pickups, deliveries, and managing logistics for customers, drivers, and administrators. It provides a centralized platform for all parties involved to interact, track deliveries, and manage communication.

**2. Actors**

• Customer: The individual who requests delivery services through the system.

• Driver: The person responsible for picking up and delivering items.

• Admin: The system administrator who manages user accounts, deliveries, and overall system operations.

3. Use Cases

**3.1 Customer Use Cases**

• Login: The customer enters their credentials to access the system and initiate delivery requests.

• Profile: The customer can view, edit, and update their profile information.

• Schedule Pickup: The customer initiates a delivery request by specifying pickup and delivery locations, items, and preferred service options. This use case includes a sub-use case, "Confirm Pickup," where the customer confirms the scheduled details.

• Live GPS Location Tracking (extends Confirm Pickup): Once the pickup is confirmed, the customer can track the driver's location in real-time (optional).

• Services Areas: The customer can view the geographical areas serviced by "The Final Final."

• Payment (includes Quotation): The customer can choose their preferred payment method (cash, check, or online) after receiving a cost estimate (quotation) for the delivery.

• Delivery History: The customer can access a record of their past deliveries.

• Communication (separate room in chat): The customer can communicate with the driver and admin in a dedicated chat room during the delivery process.

**3.2 Driver Use Cases**

• Job History: The driver can view a log of their past delivery jobs.

• Accept & Pickup: The driver views open delivery requests, accepts selected jobs, and picks up items for delivery.

• Open Jobs: The driver can see a list of currently requested deliveries available for them to accept.

• Communication (separate room in chat): The driver can communicate with the customer and admin in a designated chat room during the delivery.

• Live GPS Location Tracking: The driver can access their real-time location for navigation purposes.

• Login: The driver enters their credentials to access the system and view assigned deliveries.

• Profile: The driver can view, edit, and update their profile information.

**3.3 Admin Use Cases**

• Manage Customer (CRUD operations and more): The admin adds, edits, deletes, and manages customer accounts within the system. This might include additional functionalities beyond basic CRUD operations.

• Manage Employee (CRUD operations and more): The admin adds, edits, deletes, and manages driver accounts within the system. This might include additional functionalities beyond basic CRUD operations.

• Manage Calendar: The admin tracks the status of deliveries (requested, assigned, completed, cancelled, etc.) using a calendar view.

• Track Revenue: The admin monitors and analyzes the company's revenue generated through deliveries.

• Create Price Model: The admin defines different pricing structures for delivery services based on company policies.

• Payments: The admin processes and accepts payments made by customers.

• Service Areas: The admin manages the geographical areas served by "The Final Final."

• Profile: The admin can view, edit, and update their profile information.

• Login: The admin enters their credentials to access the system and manage its functionalities.

• Live GPS Tracking (for driver and items): The admin can track the location of drivers and potentially the items being delivered (optional).

• Communication (separate room in chat): The admin can communicate with customers and drivers in a dedicated chat room.

• Manage Delivery: The admin oversees the entire delivery process, including assigning deliveries to specific drivers.

• Add & Assign Delivery: The admin creates new delivery requests and assigns them to available drivers.

**4. Relationships**

• Includes: This indicates a mandatory sub-use case within a broader use case. For example, "Confirm Pickup" is included within "Schedule Pickup," meaning a customer must confirm details before scheduling a pickup.

• Extends: This shows an optional extension of a use case. In this case, "Live GPS Location Tracking" extends "Confirm Pickup," implying the customer can choose to track the driver after confirming the pickup.

**5. Conclusion**

This use case diagram description provides a comprehensive overview of the functionalities offered by “The Final Final” for customers, drivers, and admins. It outlines the interactions between actors and the system, fostering a clear understanding of the system’s capabilities.

**7.1.2 Activity Diagrams**

**1. System Overview**

**This enhanced activity diagram expands upon the previous description, detailing the entire login, pickup request, and delivery process within the "The Final Final" system. It illustrates the interactions between a user (customer) and the system for logging in (or registering), requesting a pickup, specifying delivery details, and potentially tracking the delivery.**

**2. Actors**

**• User (Customer)**

**3. Start State**

**• Activity Starts (User)**

**4. Activities**

**User Swimlane**

**• Activity 1: Start Activity**

**o Description: The user initiates the interaction with the system.**

**• Activity 2: Login / Sign Up**

**o Description: The user chooses to log in with existing credentials or sign up for a new account.**

**• Activity 14: Pickup Request (Branching point)**

**o Description: After successful login (or account creation), the user initiates a pickup request.**

**• Activity 18: Select Items (Branching point)**

**o Description: The user selects the items to be picked up and delivered.**

**o Branch 1: Select More Items (Optional)**

** Description: The user can choose to add more items (not shown in this diagram).**

**o Branch 2: Item Details**

** Description: The user specifies details about the items (e.g., quantity, description).**

**• Activity 19: Packaging Details**

**o Description: The user provides information about the packaging (e.g., size, weight).**

**• Activity 20: Destination Details**

**o Description: The user enters the delivery destination details.**

**• Activity 22: Payment Method Selection**

**o Description: The user selects their preferred payment method (cash, check, or online).**

**• Activity 23: Confirm**

**o Description: The user confirms the pickup request and delivery details.**

**• Activity 24: Track (Optional)**

**o Description: The user can choose to track the delivery progress (not shown in this diagram).**

**System Swimlane**

**• Activity 3: Check Account**

**o Inputs: Username, password (from Login activity)**

**o Description: The system verifies if a user account exists based on the provided login credentials.**

**• Activity 4 (Conditional): Check Credentials**

**o Inputs: Username, password (from Login activity)**

**o Description: (This activity is executed only if an account exists) The system validates the entered username and password against stored credentials.**

**• Activity 5 (Conditional): Verification**

**o Description: (This activity is executed only if credentials are valid) The system performs any necessary verification steps (e.g., two-factor authentication).**

**• Activity 6 (Conditional): Schedule Pickups and Deliveries**

**o Description: (This activity is executed only after successful verification) The user proceeds to schedule pickup and delivery details (partially shown in this diagram).**

**• Activity 7 (Conditional): Account Not Found**

**o Description: (This activity is executed if no account exists) The system prompts the user to sign up for a new account.**

**• Activity 8: Enter Credentials**

**o Inputs: User information (name, contact details, etc.)**

**o Description: The user enters their details to create a new account.**

**• Activity 9: Generate Code**

**o Description: The system generates a verification code (e.g., email or SMS).**

**• Activity 10: Verify Code**

**o Inputs: Verification code entered by the user.**

**o Description: The user enters the received verification code.**

**• Activity 11 (Conditional): Verification Successful**

**o Description: (This activity is executed only if the verification code is valid) The system creates a new user account.**

**• Activity 12: Create Profile**

**o Description: The user completes their profile information (optional).**

**• Activity 13 (Conditional): Show Message (Failed)**

**o Description: (This activity is executed if verification fails) The system displays an error message indicating unsuccessful verification.**

**• Activity 15: Service Availability Check**

**o Inputs: Pickup location and destination (from User activity 14)**

**o Description: The system checks if delivery service is available for the specified locations.**

**• Activity 16 (Conditional): Show Message (Service Not Available)**

**o Description: (This activity is executed if service is unavailable) The system informs the user that delivery service is not available in the requested area.**

**7.1.3 Class Diagram**

**Class Diagram Relationships:**

**• User has an association with Report. This means that a User can interact with reports in some way, such as generating or viewing them.**

**• Customer has an association with Delivery. This is a one-to-many association. A Customer can have many Deliveries, but a Delivery belongs to one Customer.**

**• Customer has an association with Communication History. This means a Customer can view their communication history related to deliveries.**

**• Delivery has an association with Shipment. This is a directed association, which is a stronger type of association. A Delivery has one Shipment, but a Shipment can be in multiple Deliveries.**

**• Delivery has an association with Communication History. This allows communication history to be viewed for a Delivery.**

**• Driver has an association with Shipment. This association likely allows a Driver to view the details of the Shipment they are assigned to.**

**• Driver has an association with Communication History. This means a Driver can view their communication history related to deliveries.**

**7.1.4 Sequence Diagram**

**7.1.4.1 Pick Up Order Sequence Diagram**

**Initiating the Pick-Up Request:**

**• The sequence diagram begins with the Customer placing a pick-up request through the system. This could involve them entering details about the pick-up location, preferred time, and any special instructions or requirements for the driver.**

**• The system acknowledges the request and retrieves information about the pick-up location, time preference, and any driver preferences the Customer specified (e.g., vehicle type, special skills).**

**• Based on this information, the system searches for a suitable driver. It considers factors like the driver's location, availability during the requested pick-up time, and whether their vehicle meets any special requirements.**

**Selecting a Driver:**

**• If a suitable driver is available, the system selects that driver and sends them a notification with the pick-up request details, including the location, estimated time, and any special instructions.**

**• The driver reviews the request and can either accept or reject it.**

**• If the driver accepts the request, the system sends a confirmation notification to the Customer, informing them of the driver's name, contact information, and estimated arrival time.**

**• If the driver rejects the request, the system sends a notification to the Customer explaining that the request is unavailable and offering options such as rescheduling the pick-up or assigning a different driver. The system might also indicate the reason for rejection (e.g., driver unavailable, vehicle unsuited for the task).**

**Possible Alternative:**

**• The sequence diagram suggests an alternative scenario where the system might not be able to find a suitable driver based on the Customer's initial request criteria (step 3). In this case, the system would likely inform the Customer about the difficulty in finding a perfect match and provide options:**

**• The Customer can wait for a suitable driver to become available.**

**• The Customer can modify their request criteria (e.g., pick-up time) to increase the chances of a quicker match.**

**• The system can propose alternative drivers who might not perfectly meet all the criteria but are still available for the pick-up.**

**7.4.1.2 Schedule Delivery Sequence Diagram**

**Initiating the Delivery Schedule Request:**

**• The sequence diagram commences with the Deliveries subsystem sending a request to the Schedule Delivery subsystem. This request includes details about the deliveries, such as origin and destination locations, desired date and time for delivery, and any special requirements or constraints (e.g., specific vehicle type needed).**

**• The Schedule Delivery subsystem receives the request and retrieves the necessary information about the deliveries.**

**Finding an Available Driver:**

**• The Schedule Delivery subsystem initiates a search for a suitable driver to handle the deliveries. It considers various factors during this search, including:**

**• Driver Availability: Drivers’ schedules are checked to find those available during the requested delivery timeframe.**

**• Driver Location: The system attempts to find drivers geographically close to the origin location to optimize efficiency.**

**• Driver Skills and Vehicle Suitability: Driver qualifications and vehicle capabilities are considered to ensure they meet any special requirements for the deliveries (e.g., temperature-controlled vehicle for perishable goods).**

**• Once a suitable driver is identified, the Schedule Delivery subsystem sends a notification to the Driver containing the delivery details and requesting confirmation of their availability.**

**Driver Confirms Availability:**

**• The Driver receives the notification, reviews the delivery details, and confirms their availability to undertake the deliveries.**

**• Upon confirmation from the driver, the Schedule Delivery subsystem transmits a confirmation message back to the Deliveries subsystem, indicating a driver has been assigned to the deliveries. The confirmation might also include details about the driver, such as their estimated arrival time at the pick-up location.**

**Possible Alternative: Driver Unavailable**

**• The sequence diagram acknowledges that a suitable driver might not be available based on the initial criteria (step 3). In such scenarios, the following alternatives are possible:**

**• The Schedule Delivery subsystem might send a notification back to the Deliveries subsystem, informing them about the difficulty in finding an immediate driver. The notification could include an estimated wait time or suggest adjusting delivery timeframes to better suit driver availability.**

**• The Schedule Delivery subsystem might attempt to find an alternative driver who partially meets the criteria (e.g., further away or with a slightly different vehicle type) and send a notification to the driver seeking confirmation for this alternative.**

**7.4.1.3 Manage Delivery Sequence**

**Initiating the Delivery Request or Modification**

**• The sequence diagram commences with the User initiating the process by either requesting a new delivery or requesting a modification to an existing delivery. The user interacts with the Deliveries subsystem, which could be a web application or another software program.**

**• The Deliveries subsystem acknowledges the user's request and retrieves the relevant information. In the case of a new delivery request, this information would include the origin and destination locations, desired delivery time frame, and any special requirements or instructions (e.g., fragile items, specific delivery time window). If it’s a modification request, the system would retrieve details about the existing delivery that the user wants to change.**

**Delivery Information Processing**

**• The Deliveries subsystem processes the delivery information, which might involve:**

**• Validating the information entered by the user, such as ensuring the addresses are complete and accurate.**

**• Checking delivery availability within the requested time frame. This might involve verifying driver and vehicle availability to handle the delivery during that time.**

**• Calculating potential delivery costs based on factors like distance, urgency, and special requirements.**

**• If the delivery request is feasible based on the system's checks (step 3), the Deliveries subsystem transmits the information to the Driver subsystem. This information includes details about the delivery, such as the pick-up and drop-off locations, the requested delivery time, and any special instructions.**

**Informing the Driver and Requesting Confirmation**

**• The Driver subsystem receives the delivery information and sends a notification to a suitable driver, considering factors like driver location, availability during the requested time frame, and driver qualifications (e.g., special vehicle types for specific deliveries).**

**• The Driver receives the notification with the delivery details and can either confirm their availability and willingness to undertake the delivery or reject the request.**

**Driver Confirms or Rejects Delivery**

**• If the Driver confirms their availability, a confirmation message is sent back to the Deliveries subsystem, indicating the driver has accepted the delivery request. The Deliveries subsystem would then likely send a confirmation notification to the User, informing them that a driver has been assigned to their delivery and providing an estimated delivery time.**

**• If the Driver rejects the delivery request, the reason for rejection might be communicated back to the Deliveries subsystem (e.g., driver unavailable during that time, vehicle not suitable for the delivery). The Deliveries subsystem would then need to find another suitable driver (steps 4-6) or inform the User about the difficulty in finding a driver immediately and offer alternative options (e.g., adjusting the delivery timeframe).**

**Handling Delivery Modifications**

**• The sequence diagram shows that if the driver confirms a modification request (step 7), the Deliveries subsystem would process the modification and update the delivery information accordingly. It would likely also inform the User about the confirmation and provide any relevant updates (e.g., revised delivery costs).**

**• If the modification requires additional action from the driver (e.g., rerouting due to traffic), the Deliveries subsystem would inform the Driver and request confirmation of the new course of action. The driver would then confirm or reject the request, and the process would continue accordingly (steps 7-10).**

**7.4.1.4 Track Delivery Sequence**

**User Requests Tracking Information: The process starts with the User requesting delivery tracking information. This could be done through a mobile app, web interface, or any other user interaction point provided by the delivery service.**

**System Provides Basic Delivery Status: The system responds by sending the user basic information about the delivery status. This might include a generic status like "pending," "picked up," "in transit," or "delivered." The system might also provide an estimated arrival time for the delivery.**

**User Requests Optional Live Tracking (if available): Depending on the service offered by the delivery company, the user might have the option to request real-time location updates for their delivery.**

**System Provides Live Location or Informs User of Unavailability: If live tracking is available, the system transmits the real-time location of the delivery to the user. This could be through a map interface within the user's app or web portal. However, the system might also inform the user that live tracking is unavailable due to privacy settings or technical limitations (e.g., GPS signal issues).**

**User Requests Detailed Tracking Information (Optional): In some cases, the user might want more detailed tracking information beyond the basic status or live location. They could request this additional information through the user interface.**

**System Provides Additional Tracking Data or Clarifies Status: The system responds by providing the user with more comprehensive tracking data, such as route history, past locations, and timestamps. If the additional information requested is unavailable, the system might clarify the current status based on the information it has access to.**

**7.2 ER Diagram**

**s**

**7.3 Data Dictionary**

**Chat Message**



**The Chat Message table is the backbone of communication within the platform. It tracks message history (text, attachments) exchanged between clients, admins, and drivers. This facilitates communication throughout the delivery lifecycle, ensuring everyone stays informed. Timestamps record the date and time of each message.**

**Companies**



**The Companies table is central to the SaaS functionality. It stores information about companies that subscribe to the logistics and supply chain management software. It might allow for some level of customization (logo, branding) and potentially integrates with payment processors (Stripe). This table enables features like subscription management and caters to multi-tenant environments.**

**Company Users**

**The Company Users table manages user accounts associated with companies. It links users to their respective companies, assigns roles (admin, staff), and tracks their status (active, inactive). It might also store user identification numbers and potentially connect to payment customer accounts, facilitating permission control and subscription management within companies.**

**Contact Us**

**The Contact Us table serves as a repository for contact information submitted by users for companies on the platform. It stores details like name, email, and phone number, allowing potential customers to initiate contact with specific companies. This facilitates communication channels between interested users and the companies that use the software.**

**Day Block Times**

**The Day Block Times is used within the calendar module. It allows companies to define specific days and timeframes where they are unavailable for deliveries. Essentially, companies can block out these periods to manage their workload and schedule deliveries accordingly. This table helps maintain clear delivery windows and avoid scheduling conflicts.**

**Decline Reasons**

**The Decline Reason table serves as a repository within the delivery module, capturing reasons for client declinations. It logs details related to delivery declines, including the associated delivery ID and reasons specified by the client. Additionally, it indicates whether to display the reason to the client. Timestamps are recorded for tracking purposes.**

**Deliveries**

**The Deliveries table is pivotal in the delivery module, storing comprehensive information related to each delivery. From pickup and destination locations to item descriptions and delivery statuses, this table ensures all crucial data pertaining to deliveries is recorded. It facilitates tracking delivery progress, assigning drivers, and managing payment details. Timestamps mark creation, modification, and deletion events.**

**Delivery Completed Notes**

**Within the delivery module, the Delivery Completed Notes table records data upon completion of a delivery. It includes comments, images, and timestamps, providing a comprehensive overview of the delivery's finalization. This information aids in verifying successful deliveries and resolving any post-delivery inquiries or issues.**

**Delivery Histories**

**Integral to the delivery module, this table documents the status history of each delivery. It tracks changes in delivery status, accompanied by relevant messages and timestamps. This comprehensive history log enables stakeholders to review the progression of deliveries and identify any potential issues or delays.**

**Delivery Items**

**This table manages the inventory aspect of deliveries, storing details of items being transported. From quantities and descriptions to associated images, it provides a comprehensive inventory record. Linked to delivery details, it ensures accurate tracking and management of items throughout the delivery process.**

**Delivery Payment Logs**

**Within the delivery module, this table logs all payment-related transactions associated with deliveries. It records payment types, statuses, transaction IDs, amounts, and additional notes for each transaction. This ensures transparent financial tracking and facilitates reconciliation processes.**

**Delivery Pickup Notes**

**As part of the delivery module, this table documents pickup-related information for deliveries. It includes pickup statuses, comments, images, and timestamps. This data aids in monitoring pickup operations and ensuring timely and accurate handling of items.**

**Delivery Reviews**

**This table within the delivery module is dedicated to storing feedback from clients regarding delivery experiences. It records review comments, ratings, and timestamps, providing valuable insights for service improvement efforts.**

**Driver Car Infos**

**A subset of the employee module, this table holds essential information regarding drivers' vehicles. It includes details such as car model, type, color, and associated company user IDs. This data aids in managing driver resources and ensuring compliance with vehicle-related regulations.**

**Forget Password Tokens**

**Essential for user account security, this table manages tokens used for password reset requests. It links tokens to user IDs and tracks expiration dates. Timestamps record token creation and modification, ensuring timely and secure password reset processes.**

**Holidays**

**This table, part of the calendar module, records company-specific holidays. It associates holidays with company IDs and stores holiday names and dates. Timestamps track holiday record creation and modification, aiding in scheduling and resource planning.**

**Items**

**This table holds essential information about items relevant to the delivery module, including item names, associated company IDs, images, status, and flat rates. It's crucial for managing and updating item details within the delivery process.**

**Message Read Statuses**

**Within the chat message module, this table is pivotal for tracking the status of messages in chat rooms. It records message delivery and read statuses for specific rooms, ensuring effective communication among drivers, clients, and admins.**

**Notification Tokens**

**In the notification system, this table stores user IDs, UUIDs, and tokens necessary for sending push notifications. It facilitates seamless communication by enabling the system to send timely updates and notifications to users.**

**Notifications**

**This table manages system-generated notifications within the platform. It includes details such as company IDs, notification titles, messages, and recipients, ensuring users receive relevant and timely information.**

**Payment Delivery Tokens**

**Integral to the payment and delivery module, this table stores tokens and expiration dates related to payment transactions. It ensures secure payment processing for deliveries by managing unique tokens and their validity periods.**

**Prices**

**Within the administrative module, this table centralizes pricing data for delivery charges and item prices. It includes company IDs, pricing keys, and values, allowing administrators to efficiently manage pricing structures.**

**Roles**

**This table, part of the employee management module, defines different roles within companies. It includes role names and company IDs, facilitating role assignment and access control for employees.**

**Room Users**

**A component of the communication module, this table tracks users associated with specific chat rooms. It records company IDs, room IDs, and user IDs, ensuring proper management of users within chat rooms.**

**Rooms**

**Service Area Restrictions**

**This table, essential for the delivery module, maintains information about service area restrictions imposed by companies. It records whether service areas are restricted and which companies implement these restrictions, aiding in efficient delivery management.**

**Service Locators**

**Within the delivery module, this table stores information about service locations, such as cities and zip codes, where a company's services are available. It helps users identify serviceable areas and facilitates efficient delivery planning.**

**Subscriptions Logs**

**Critical for managing user subscriptions, this table logs subscription-related information, including subscription IDs, payment amounts, start and end dates, and subscription statuses. It aids in tracking subscription activity and managing user subscriptions effectively.**

**User Admin Room Chat Ids**

**A vital component of the communication module, this table stores IDs related to user-admin chat rooms, facilitating communication between users and administrators. It records company IDs, user IDs, admin IDs, and room IDs, ensuring seamless communication within the platform.**

**Users**

**The "users" table stores essential information about platform users. It includes details such as user names, email addresses, phone numbers, and profile images. Additionally, it tracks user-specific data like driver license information, ZIP codes, and addresses. This table is fundamental for user management and authentication within the platform, facilitating personalized user experiences and efficient communication.**

**8 Implementation & Testing**

**8.1 Screen Snapshots**

**Login Module**

**The First Screen the Admin Encounter will be the login Screen where the user Can do the login. If not register, then he can be able to purchase plan for the system as SaaS. Here the login screen presented where the login can be done by entering email and password. If he forget the password then they can reset it.**

**The Next Screenshot represents the validation of the fields in which require field validation has been triggered here. And in addition there is email validation and password validation also been checked here. There is also multiple test validation has been checked like password security.**

**The Below Screenshot represented with values**

**Dashboard**

**Calendar Module**

**Customer Module**

**Add Customer:**

**Delivery Module**

**Employees Module**

**8.2 Reports (giving important outputs of the project)**

**8.3 Testing**

**8.3.1 Test Strategy**

**8.3.2 Test Cases and Results**

**9 Future Enhancement**

**10. Bibliography**

**W3school: https://www.w3schools.com/**

**MDN documentation: https://developer.mozilla.org/en-US**

**NPM: https://docs.npmjs.com/**

**Node JS: https://nodejs.org/docs/**

**https://nodejs.org/api/all.html**

**Express JS: https://expressjs.com/**

**React JS: https://legacy.reactjs.org/**

**https://devdocs.io/react**

**MYSQL: https://www.mysql.com/**

**https://docs.digitalocean.com/products/databases/mysql**

**ES documentation: https://ecma-international.org/publications-and-standards/standards/ecma-262**

**Quara: https://www.quora.com/**

**Stackoverflow: https://stackoverflow.com/**

**https://jsonplaceholder.typicode.com/**

**Youtube videos:**

**Harshit Vashisth**

**Chai with code and Hitesh Chaudhry**

**Web Get Simplified**

**FreeCodeCamp.org**

**Yahoo Baba**

s

jj