User Interface (UI)
User Experience (UX)
Analysis

TRANSPORT
SERVICE MATCH PLATFORM

Version	Description of Change	Author	Date
1.0	First Draft	Ahmed Berdai / Aadel Bouzambou / Maria Deffense / Moustapha El Zeir / Ophelia Karapetyan /	28/02/2025

Table of Contents

1. Introduction

This document provides essential guidelines for the UX/ UI design analysis conducted for the Transport Service Match System. Introducing user personas, visual elements such as screen prototypes for different roles: Logistic Requesters (LR) and Logistic Providers (LP), along with common interface elements and support features. It aims to convey a clear idea of the application design and the user interactions.

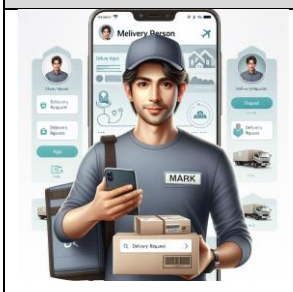
2. User's Personas

This section outlines the key user types for the Transport Service Match System. Each persona represents a distinct user role, their goals, pain points, and behaviors.

2.1 Logistic Requester (LR) - Emma

EMMA	**Role**:	♦ Creates delivery requests.
	**Goals**:	♦ Easily schedule and track deliveries. ♦ Ensure accurate and timely delivery of goods.
	Pain Points:	♦ Difficulty finding reliable logistic providers. ♦ Confusion in inputting correct delivery details.
	Behavior:	♦ Regularly uses the app to request deliveries. ♦ Prefers clear instructions and confirmation of delivery status.

2.2 Logistic Provider (LP) - Mark

MARK	**Role**:	♦ Fulfills delivery requests.
	**Goals**:	♦ Find and accept delivery requests efficiently. ♦ Ensure timely and accurate delivery of goods.
	Pain Points:	♦ Difficulty understanding delivery requirements. ♦ Lack of clear communication with requesters.
	Behavior:	♦ Frequently checks the app for new requests. ♦ Prefers straightforward request details and easy navigation.

2.3 Logistic Requester (LRi) - Ivo

IVO	Role:	♦ Starts but rarely completes delivery requests
	**Goals**:	♦ Quickly explore delivery options. ♦ Avoid committing to requests without full clarity.
	Pain Points:	♦ Finds the request process too complex. ♦ Often inputs incorrect data due to confusion.
	Behavior:	♦ Frequently abandons requests midway. ♦ Prefers simpler forms and clearer instructions.

2.4 Logistic Provider (LP) - Aga

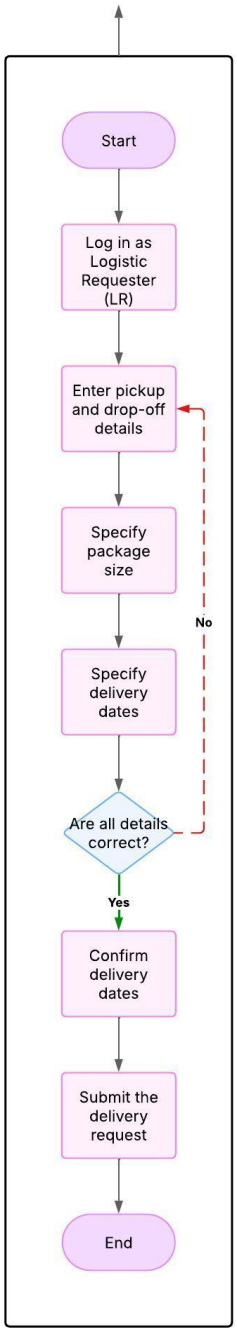
AGA	Role:	♦ Struggles to fulfill delivery requests.
	**Goals**:	♦ Accept and complete delivery requests. ♦ Avoid errors in delivery execution
	Pain Points:	♦ Difficulty understanding delivery details. ♦ Often fails to deliver due to unclear instructions.
	Behavior:	♦ Frequently makes mistakes in fulfilling requests. ♦ Needs more guidance and support from the app

2.5 Support – Adam

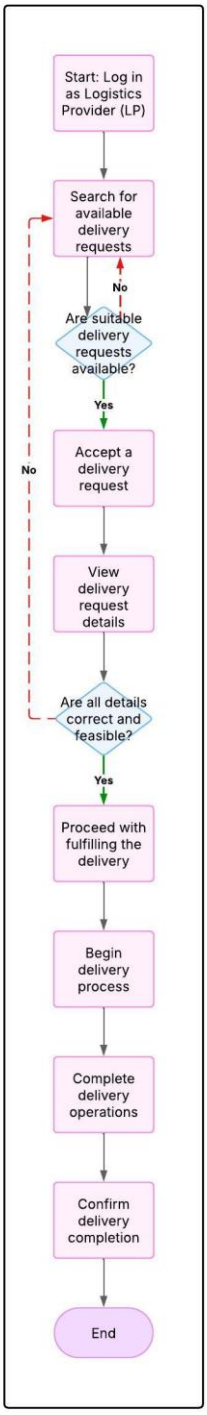
ADAM	Role:	♦ Backend support for the system.
	**Goals**:	♦ Resolve user issues quickly. ♦ Ensure smooth operation of the app.
	Pain Points:	♦ Lack of clear logs or reports to diagnose issues. ♦ Difficulty tracking user complaints.
	Behavior:	♦ Monitors app performance and user feedback. ♦ Prefers detailed logs and easy-to-use support tools.

3. User Case Journey

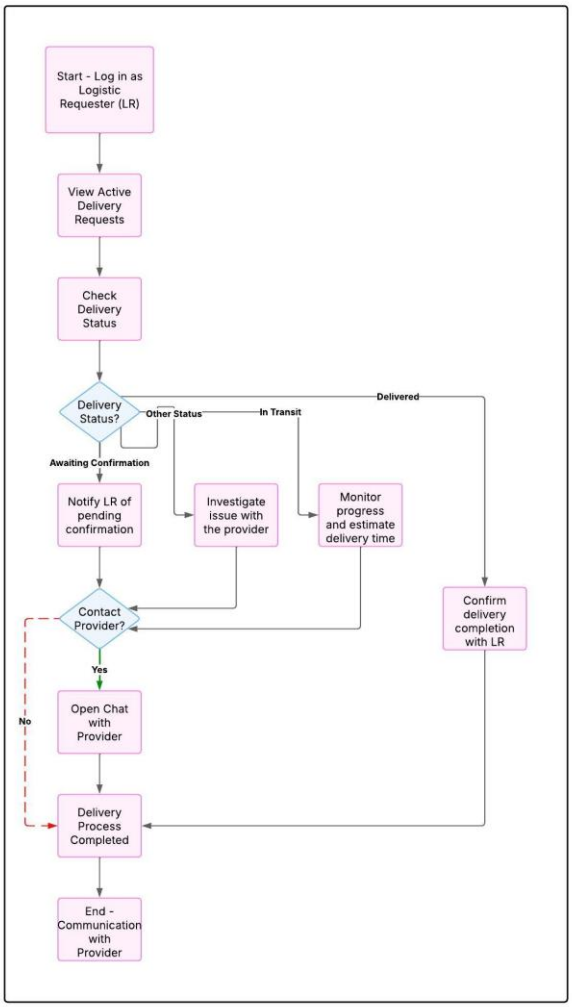
3.1 Logistic Requester (LR) - Emma Creates a Transport Request

USER JOURNEY FLOWCHART **Scenario:** Emma Creates a Transport Request	**Comments**
	♦ **Oval (Start/End):** ○ Start: Clearly marks the beginning of the user journey. ○ End: Indicates the successful completion of the transport request process. ♦ **Rectangle (Process Steps):** ○ Log in as Logistic Requester (LR): Essential first step to authenticate the user. ○ Enter pickup and drop-off details: Captures critical location information for the delivery. ○ Specify package size: Ensures the system knows the dimensions/weight of the package. ○ Specify delivery dates: Allows the user to set preferred delivery timelines. ○ Confirm delivery dates: Finalizes the timing details before submission. ○ Submit the transport request: Completes the process, sending the request to the system. ♦ **Diamond (Decision Point):** ○ Are all details correct?: A crucial checkpoint to ensure accuracy before submission. ▪ Yes: Proceeds to confirm and submit the request. ▪ No: Loops back to allow corrections (implied but not shown in detail).

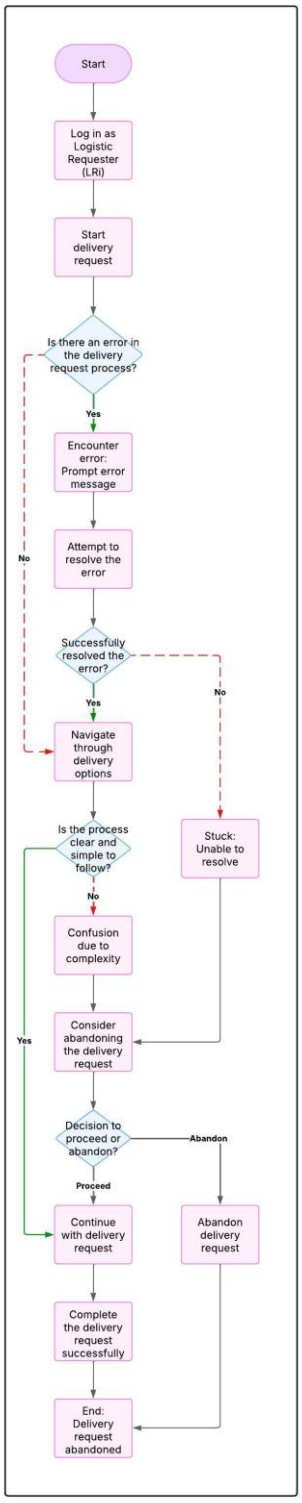
3.2 Logistic Requester (LP) - Mark Accepts and Fulfills a Transport Request

USER JOURNEY FLOWCHART **Scenario:** Mark accepts and fulfills a transport request.	**Comments**
	♦ **Oval (Start/End):** ○ Start: Clearly marks the beginning of the Logistic Provider (LP) journey. ○ End: Indicates the successful completion of the delivery process. ♦ **Rectangle (Process Steps):** ○ Log in as Logistics Provider (LP): Essential first step to authenticate the user. ○ Search for available delivery requests: Allows the LP to find relevant delivery opportunities. ○ Accept a delivery request: Enables the LP to commit to a specific request. ○ View delivery request details: Provides critical information about the delivery (e.g., pickup/drop-off, package size). ○ Proceed with fulfilling the delivery: Moves the LP into the execution phase. ○ Begin delivery process: Marks the start of physical delivery operations. ○ Complete delivery operations: Ensures the delivery is carried out as planned. ○ Confirm delivery completion: Finalizes the process, updating the system and requester. ♦ **Diamond (Decision Point):** ○ Are suitable delivery requests available?: A key checkpoint to ensure the LP finds relevant requests. ▪ Yes: Proceeds to accept and view details. ▪ No: Implies the LP may need to search again or wait for new requests (loop not explicitly shown). ○ Are all details correct and feasible?: Ensures the LP can fulfill the request before proceeding. ▪ Yes: Moves forward with delivery fulfillment. ▪ No: Implies the LP may need to reject or request clarification (loop not explicitly shown).

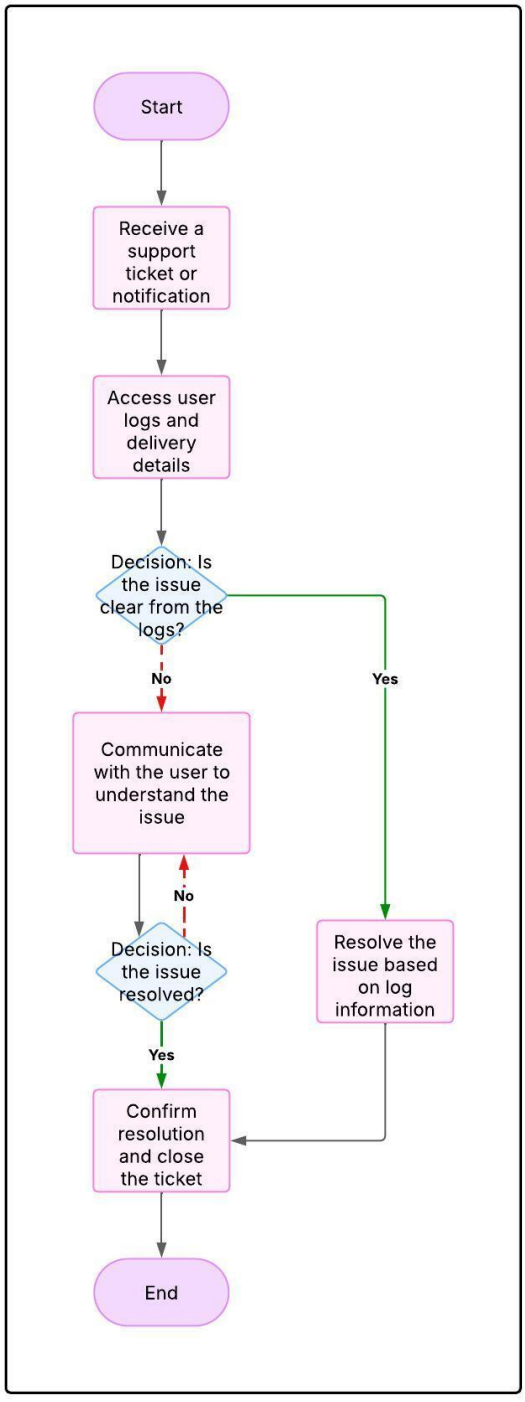
3.3 Logistic Requester (LR) - Emma Tracks Delivery Status

USER JOURNEY FLOWCHART **Scenario:** Emma Tracks Delivery Status	**Comments**
	♦ **Oval (Start/End):** ○ Start: Clearly marks the beginning of the user journey. ○ End: Indicates completion of the process, focusing on communication, confirmation. ♦ **Rectangle (Process Steps):** ○ Log in as Logistic Requester (LR): Essential first step to authenticate the user. ○ View Active Delivery Requests: Allows the LR to see ongoing deliveries. ○ Check Delivery Status: Provides real-time updates on the delivery progress. ○ Notify LR of pending confirmation: Alerts the LR if the delivery is awaiting confirmation. ○ Investigate issue with the provider: Ensures problems are addressed swiftly. ○ Monitor progress and estimate delivery time: Keeps the LR informed about timeline. ○ Open Chat with Provider: Enables direct communication between the LR and LP. ○ Confirm delivery completion with LR: Finalizes the process, ensuring the LR is satisfied. ♦ **Diamond (Decision Point):** ○ Delivery Status?: A key checkpoint to determine current state of the delivery. ▪ Awaiting Confirmation: Triggers a notification to the LR. ▪ In Transit: Leads to monitoring progress and estimated delivery time. ▪ Other Status: Implies additional actions may be needed (not explicitly detailed). ○ Contact Provider?: Decides whether the LR needs to communicate directly with the provider. ▪ Yes: Opens a chat with LP. ▪ No: Implies the LR is satisfied with the current status (not explicitly shown).

3.4 Logistic Requester (LRi) - Ivo Abandons a Transport Request

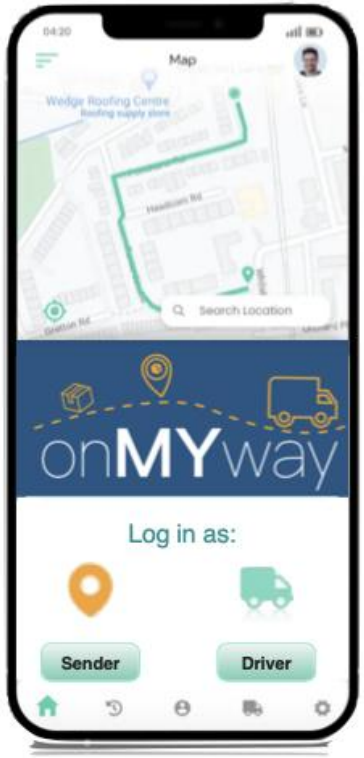
USER JOURNEY FLOWCHART **Scenario:** Ivo Abandons a Transport Request	**Comments**
	♦ **Oval (Start/End):** ○ Start: Marks the beginning of the Logistic Requester (LRi) journey. ○ End: Indicates the user abandons the process due to complexity or errors. ♦ **Rectangle (Process Steps):** ○ Log in as Logistic Requester (LRi): Essential first step to authenticate the user. ○ Start transport request: Initiates the process of creating a transport request. ○ Encounter error / Prompt error message: Highlights a potential issue in the process. ○ Attempt to resolve the error: Shows the user's effort to fix the problem. ○ Navigate through delivery options: Represents the user exploring available choices. ○ Continue with delivery request: Indicates the user decides to proceed despite challenges. ○ Complete the delivery request successfully: Marks the successful completion of the process. ○ Abandon delivery request: Represents the user giving up due to complexity or errors. ♦ **Diamond (Decision Point):** ○ Is there an error in the transport request process?: A key checkpoint to identify issues. ▪ Yes: Triggers error message, resolution attempt. ▪ No: Allows the user to proceed without issues. ○ Successfully resolved the error?: Determines whether the user can continue. ▪ Yes: Moves forward with the transport request. ▪ No: Leads to potential abandonment. ○ Is the process clear and simple to follow?: Evaluates the user's experience. ▪ Yes: Encourages the user to proceed. ▪ No: Leads to frustration and potential abandonment. ○ Decision to proceed or abandon?: Final choice for user. ▪ Proceed: Continues with the delivery request. ▪ Abandon: Ends the process without completion.

3.5 Support - Adam Resolves a User Issue

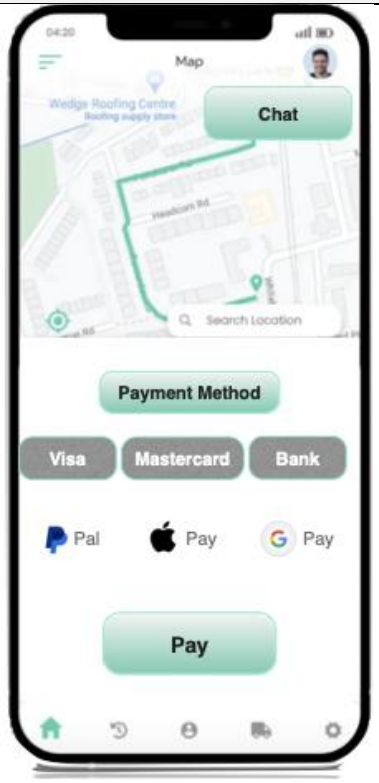
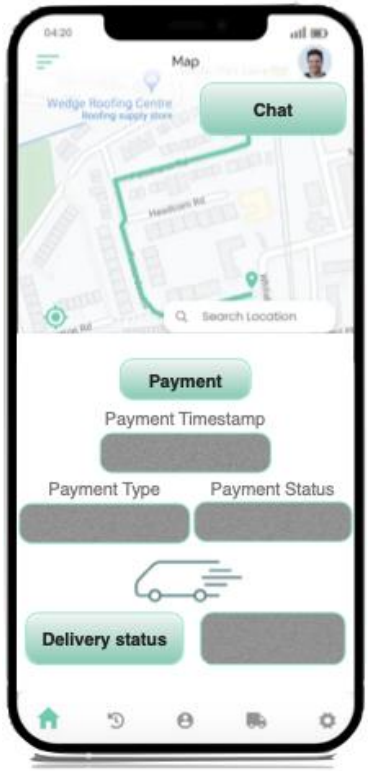
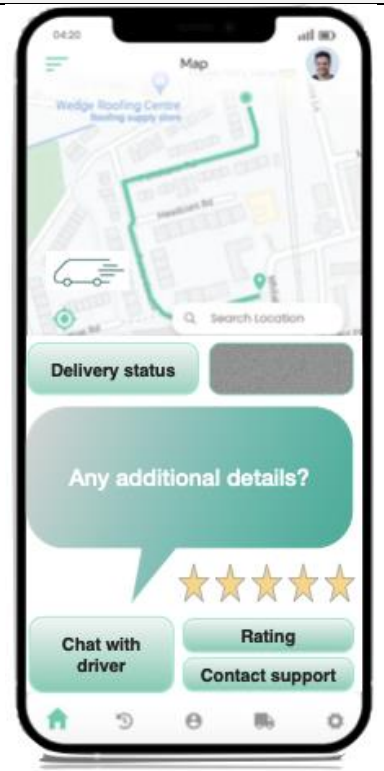
USER JOURNEY FLOWCHART **Scenario:** Adam resolves a user issue.	**Comments**
	♦ **Oval (Start/End):** ○ Start: Marks the beginning of the support process. ○ End: Indicates the successful resolution and closure of the support ticket. ♦ **Rectangle (Process Steps):** ○ Receive a support ticket or notification: Initiates the support process. ○ Access user logs and delivery details: Provides the support team with necessary information to diagnose the issue. ○ Communicate with the user to understand the issue: Ensures clarity if the logs are insufficient. ○ Resolve the issue based on log information: Directly addresses the problem using available data. ○ Confirm resolution and close the ticket: Finalizes the process, ensuring the issue is resolved. ♦ **Diamond (Decision Point):** ○ Is the issue clear from the logs?: A key checkpoint to determine if the logs provide enough information. ▪ Yes: Proceeds to resolve the issue directly. ▪ No: Requires communication with the user for further clarification. ○ Is the issue resolved?: Ensures the problem is fully addressed before closing the ticket. ▪ Yes: Confirms resolution and closes the ticket. ▪ No: Implies further action is needed (though the loop is not explicitly shown).

4. Layout Examples (Screens Simulations)

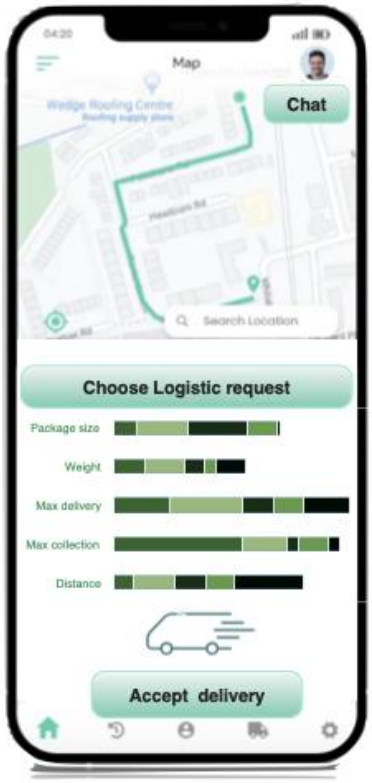
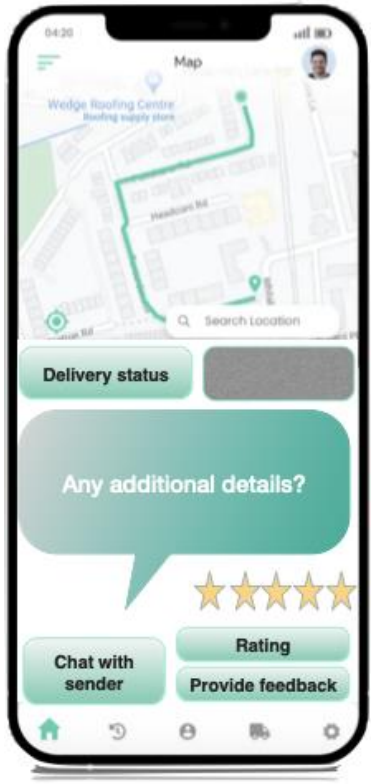
4.1 Sign In / Sign Up Interfaces

Sign In Screen	Sign Up Screen
	

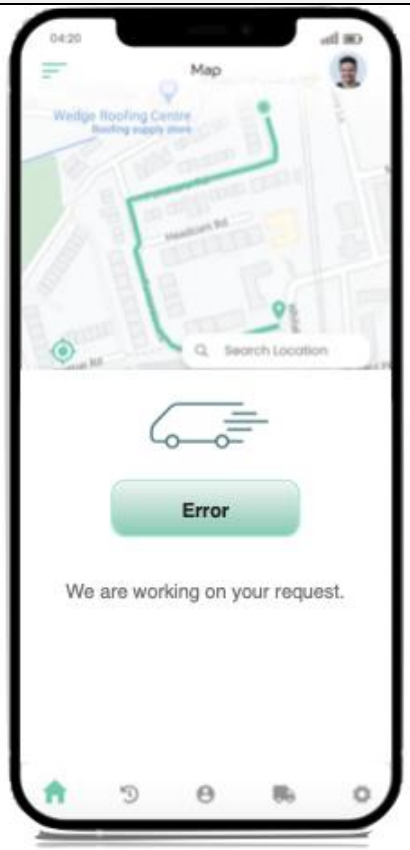
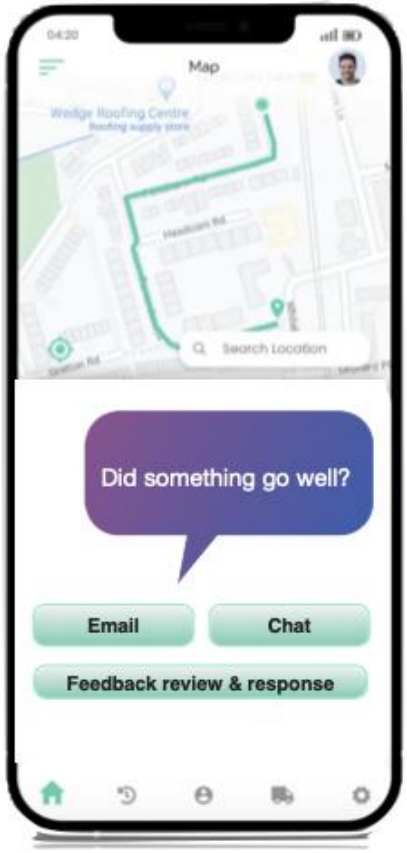
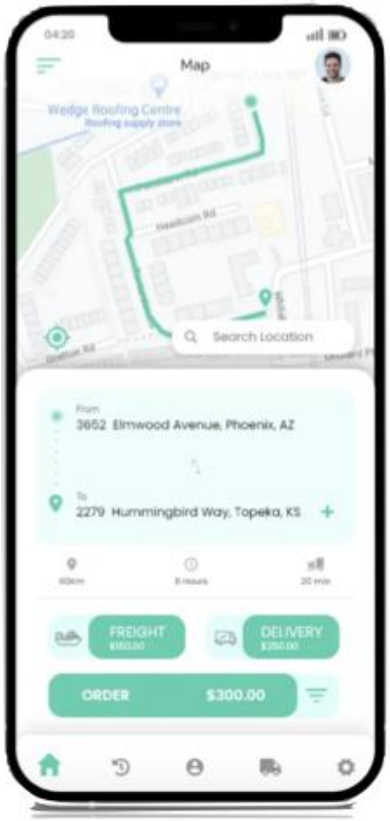
4.2 Logistic Requester (LR) Interfaces

Create a Request Screen	Payment Method Screen	Delivery Status Screen	Chat Screen
			

4.3 Logistic Provider (LP) Interfaces

Accept a Request Screen	Search Screen	Registration Screen	Chat Screen
			

5. Other Interface Elements

Error Screen	Contact Support Screen	Support Audit Screen	Order Recap Screen
			

Appendix A – Glossary

- **LR:** Logistic Requester (someone who needs transportation).
- **LP:** Logistic Provider (someone who offers transportation).
- **LRi:** Logistic Requester (Incomplete): User who starts but rarely completes Transport requests.
- **UI:** User Interface, the visual and interactive elements of the application.
- **UX:** User Experience, The Overall experience of users interactiong with the system.[1]

---

[1]Methodology Note:

This document was developed through a structured, user-centered design process by a collaborative team:

1. **User Personas:** The team identified key user roles (e.g., Logistic Requester, Logistic Provider, Support) and detailed their goals, pain points, and behaviors.
2. **User Case Journeys:** Based on the personas, the team mapped out workflows for scenarios like creating requests, fulfilling deliveries, tracking status, and resolving issues. Flowcharts were created to visualize these journeys, highlighting decision points and pain points.
3. **Screen Drafts:** The team designed draft layouts for key interfaces (e.g., Sign In, Create Request, Delivery Status, Chat) to align with the user journeys and personas.
4. **Refinement:** Through iterative feedback and collaboration, the team refined the personas, journeys, and screens to ensure they accurately reflected user needs and system goals.