UNIVERSIDADE D
COIMBRA

Critical Rides

Critical Software Carpooling System

Requirements Engineering 2024/25

"More Seats Filled, Fewer Miles Wasted"

Team Members
João Laranjeiro (uc2024181464@student.uc.pt)
Diogo Frazão (uc2024156386@student.uc.pt)
Miguel Machado (uc2020222874@student.uc.pt)
Zan Pizmoht (uc2024154128@student.uc.pt)
Marcelo Gomes (uc2021222994@student.uc.pt)

Version 1.0
December 30, 2024
University of Coimbra

Critical Rides

December 30, 2024

LaTeX Editing Tool: Overleaf

Revision History:

Version	Date	Changes Made
1.0	December 30, 2024	Initial version

Contact Email: uc2024181464@student.uc.pt

Contents

Executive Abstract

This document outlines the Software Requirements Specification (SRS) for the "Critical Rides" carpooling system, developed to improve interoffice travel efficiency at Critical Software. Aiming to reduce costs and carbon emissions, the system integrates carpooling functionalities into Microsoft Teams and Calendar. Key features include seamless ride-sharing coordination, package delivery optimization, and streamlined communication through automated prompts and centralized data.

The project addresses inefficiencies in the current travel system, primarily caused by fragmented communication and low participation in carpooling initiatives. By leveraging personas and contextual design, the proposed solution ensures user-centric functionality, such as intuitive interfaces and real-time notifications, while complying with GDPR and Portuguese transport regulations.

The system's requirements are derived from a detailed analysis of existing carpooling platforms, user stories, and workflows, ensuring alignment with Critical Software's sustainability goals. Quality attributes emphasize usability, robustness, and interoperability within the existing Teams ecosystem. Potential challenges include adoption barriers and technical integration constraints, mitigated through user engagement and iterative refinement.

This document provides a comprehensive foundation for the development, deployment, and evaluation of "Critical Rides," highlighting its contribution to environmental sustainability and enhanced organizational efficiency.

List of Figures

List of Tables

1 Introduction

1.1 Purpose

The purpose of this document is to define the Software Requirements Specification (SRS) for the "Critical Rides" carpooling system. This document serves as a guide for developers, project managers, and stakeholders, providing clear specifications for the design, development, and testing of the system.

1.2 Scope

The "Critical Rides" carpooling system is designed to address inefficiencies in interoffice travel at Critical Software. It integrates carpooling features directly into Microsoft Teams and Calendar to simplify ride-sharing and reduce environmental impact. The system will enable employees to request or offer carpool rides during meeting scheduling, ensuring effective communication and coordination. Benefits include reduced travel costs, improved employee collaboration, and alignment with the company's sustainability goals. The product will not offer real-time navigation or location tracking, as these are outside the scope.

1.3 Definitions, Acronyms, and Abbreviations

Term	**Definition**
SRS	Software Requirements Specification
GDPR	General Data Protection Regulation
UED	User Experience Design

Table 1: Definitions, Acronyms, and Abbreviations

1.4 External References

External references include:

- Microsoft Teams API documentation [3].
- GDPR compliance guidelines from the European Union [6].
- BPMN 2.0 standards documentation [2]

1.5 Document Overview

This document is organized as follows:

- Section 2 outlines the overall product description, including its perspective, major functions, and context.
- Section 3 specifies detailed requirements, categorized into functional, non-functional, business, and regulatory constraints.

- Section 4 provides an assessment of Requirements Engineering (RE) techniques.
- Annexes contain supporting information, such as the traceability matrix, larger diagrams, and a survey of existing solutions.

2 Overall Description

This section provides an overview of the general factors influencing the "Critical Rides" carpooling system and its requirements. The goal is to present the product's context and establish a background for the specific requirements detailed in Section 3. By outlining the product's relationship with the broader system, existing solutions, and its unique attributes, this section aims to enhance understanding of the requirements and their rationale.

2.1 Product Perspective

The "Critical Rides" carpooling system is not a standalone product; rather, it is an integrated component within the existing Microsoft Teams and Calendar ecosystem used at Critical Software (CSW). This integration ensures seamless functionality by leveraging established platforms already familiar to employees. The system serves to simplify interoffice travel coordination and package delivery logistics, addressing inefficiencies in communication and sustainability efforts.

2.1.1 Integration within the Larger System

The product integrates with Microsoft Teams to facilitate ride-sharing and package delivery through direct interactions with Teams channels and notifications. Employees can interact with the system by posting or browsing carpool offers and requests directly from their Teams interface. Additionally, the system integrates with the Calendar, prompting users to consider carpooling options during event scheduling.

A block diagram illustrating the system's integration and its relationship with Teams, Calendar, and employees' workflows is included in Annex C. This use-case context diagram highlights how the carpooling system aligns with existing corporate tools, enhancing usability and adoption while minimizing disruptions to current workflows.

2.1.2 Relate to existing solutions

The "Critical Rides" system builds upon insights from existing carpooling platforms, including BlaBlaCar [1], Poola [4], and TwoGo [5], to address CSW-specific challenges. A comparison of attributes and requirements from these platforms reveals key features that influenced the system's design. In table 2, you can find a detailed comparison of carpooling applications and their key attributes, highlighting the features and capabilities of each platform.

App's Name	Attributes/Requirements
Comovee	• Map Interface: Users can see colleagues' locations and find potential carpool matches. • Messaging: Users can contact colleagues directly via messages. • Publishing System: Users can either publish details of their available rides or request for rides. • Carpool Matching: The system matches employees based on their locations, displaying both a matching percentage and the number of matches found. • Destination Selection: Users can select a destination to find matching rides. • Filtering by Ride Details: Users can filter results by ride details, such as the day of the ride and preferences like car availability and smoking.

App's Name	Attributes/Requirements
Blablacar	• Create account: Allow users to create and manage their accounts. • Search for trips: Provide a feature to search for available trips. • Book trip: Facilitate the booking of a trip chosen by the passenger. • Accept booking request: Allow drivers to accept or decline booking requests. • User messaging system: Enable users to send messages to each other before or after booking. • Pay for trip: Process secure payment for the reserved trip. • Receive payments: Allow drivers to receive payments for the trips they offer. • Discounts and vouchers: Allow the use of discounts and vouchers on trip bookings. • Exchanges and refunds: Manage trip reservation exchanges and refunds. • Issue invoices and receipts: Generate invoices and receipts for payment transactions. • Manage trip (driver/passenger): Allow trip management, including communication and changes. • Rating system: Facilitate rating and reviewing of drivers and passengers after trips.
Poola	• Carpooling Calendar: Your employees can use Carpooling Calendar to update their daily schedule, change rides and inform other carpoolers. • Live map: Carpoolers see their driver's location in real-time and communicate with each other via chat and updates. • iOS and Android app. • Web admin panel with analytics & reporting.

App's Name	Attributes/Requirements
Uber for Business	• Employee Enrollment: Employees must be invited to join the company's Uber account. • Employees use their business profiles in the Uber app to request rides. • Customizable Ride Programs: Set ride availability based on times, locations. • Commuter Options: Door-to-door commute, First and last mile, etc. • Ability to invite employees manually, via CSV, or through syncing with employee management systems. • Budget Allocation: Define how much of the ride cost the company will cover, including the ability to limit ride times and vehicle types. • Safety Measures: Riders can easily access key safety information, get emergency assistance, etc.
Twogo	• User Interface: Intuitive and user-friendly mobile app and web interface. • Data Privacy: GDPR compliant, with advanced privacy controls. • Reporting & Analytics: Provides detailed reporting on user adoption, carpooling activities, and environmental impact. • Real-Time Notifications: Send alerts to users about trip status, updates, or cancellations. • Ride Scheduling: Enable employees to schedule trips in advance or on-demand according to their work schedule. • Customizable Dashboard: Admins can monitor reports on usage, user engagement, and environmental savings. • Cross-Platform Support: Support web and mobile access for users across different devices. • Corporate Branding: Offer companies the ability to customise the platform with their branding.

Table 2: Comparison of Carpooling Applications and Their Attributes

2.1.3 What can we learn from existing solutions?

After reviewing platforms like TwoGo, BlaBlaCar, Uber for Business, Poola, and Comovee, we've identified several features that could inspire a better solution for CSW.

1. Centralization and Automation

   Comovee, Blablacar, Poola, Uber for Business and Twogo offer a centralized system where users can see available rides and automatically match trips based on location and timing. This would directly address our issue of fragmented communication. A centralized carpooling system that automatically matches employees based on their schedules and destinations would save time and streamline the coordination of rides.

2. Integration with Corporate Systems

   Platforms like Poola and TwoGo integrate with corporate calendars, which is something we could implement as well. If our carpooling system were connected to employees' work calendars, they could easily input their trips, and the system would automatically suggest carpool matches. Additionally, we could link the system to tools we already use, like Microsoft Teams, for notifications and messaging.

3. Simplicity and Adoption

   The success of this system hinges on ease of use. Platforms like BlaBlaCar and Poola demonstrate the importance of having a simple, intuitive interface that allows users to quickly find and organize rides. To ensure high adoption rates, the platform should be user-friendly and feature automatic notifications when ride matches are found, keeping communication clear and timely.

4. Flexible Carpool Options

   All of the reviewed platforms allow users to customize their travel preferences, such as non-smoking or vehicle type. Incorporating such flexibility would ensure that carpooling is comfortable for everyone, without overcomplicating the process. Additionally, the system should support both pre-scheduled and last-minute rides, accommodating different travel needs.

5. Multi-Platform Support

   Comovee, Blablacar, Uber for Business and Twogo offer access via both desktop and mobile, which is essential for our use case. Our solution should provide cross-platform access, allowing employees to organize carpools from either their desktops or mobile devices, ensuring convenience regardless of location.

2.2 Product Functions

- **Allow employees to create a carpooling offer through a calendar event:** Employees can easily create a carpooling offer by confirming a calendar event. They will be prompted to provide key details such as departure time, destination, and the number of available seats. Once submitted, the offer is automatically published to the carpooling channel, making it visible to colleagues.

- **Allow employees to create a carpooling request through a calendar event:** Employees can submit a carpooling request by confirming a calendar event and filling in the necessary details. The system will automatically post the request, allowing carpool drivers to review it and consider the requester as a potential passenger.
- **Enable employees to arrange a ride via a carpooling post:** Employees can respond to carpooling posts in the channel to arrange a ride. They can communicate directly with the poster (either the driver or requester) to finalize details and confirm the arrangement.
- **Arrange package delivery to another office via a carpooling post:** Administrative staff can leverage carpooling posts to coordinate inter-office package deliveries. By assigning packages to available rides, this feature streamlines logistics and helps reduce transportation costs.
- **Allow employees to delete their own posts when the carpool is already full:** Employees can remove their carpooling posts once the ride is fully booked or no longer needed. This ensures the system remains up-to-date and free from outdated or irrelevant posts.

2.3 Product Context

2.3.1 Contextual Design

Critical Software (CSW) faces inefficiencies in its interoffice travel system, both in terms of costs and environmental impact. Currently, employees frequently travel between offices in Portugal (Coimbra, Porto, Lisbon, and Viseu) using individual transportation, leading to high expenses and unnecessary carbon emissions. Although the company has attempted to facilitate carpooling through Microsoft Teams groups, this approach has proven ineffective due to fragmented communication and a lack of centralized coordination. This inefficiency contradicts CSW's sustainability goals as a Benefit Corporation committed to reducing its environmental impact. Therefore, the company needs a more organized and user-friendly solution to enable carpooling among employees, ultimately reducing its carbon footprint and travel-related costs.

Information Flow The information flow of the current carpooling solution is illustrated in the diagram provided in figure 1. The diagram depicts three key personas: the secretary, employees with a car, and employees without a car. Each persona interacts in specific ways to facilitate both ride-sharing and package transport between offices. However, several issues prevent optimal usage, as represented by the red lightning signs in the diagram.

- **Secretary**: The secretary coordinates package delivery between offices (e.g., from Coimbra to Lisbon). When someone posts in the Microsoft Teams channel that they are traveling to an office, the secretary contacts them privately to arrange the package pickup and drop-off. However, this manual monitoring is inconsistent, leading to missed opportunities and unreliable delivery.
- **Employees With a Car**: Employees with cars are encouraged to post available rides in the Teams channel. These posts notify both colleagues and the secretary.

However, many employees do not consistently post their travel plans, resulting in missed carpooling and delivery opportunities. Coordination through private messages also creates delays, reducing system efficiency.

- **Employees Without a Car**: These employees primarily rely on public transport but may post ride requests in the Teams channel or respond to offers. Communication issues arise when ride requests are not promptly seen or responded to, which discourages usage and limits ride-sharing opportunities.

Overall, the current system is heavily dependent on Microsoft Teams for communication, leading to breakdowns in the information flow, as represented by the red lightning signs. The manual and inconsistent nature of the current process makes it unreliable, resulting in fewer ride matches and missed opportunities for package delivery.

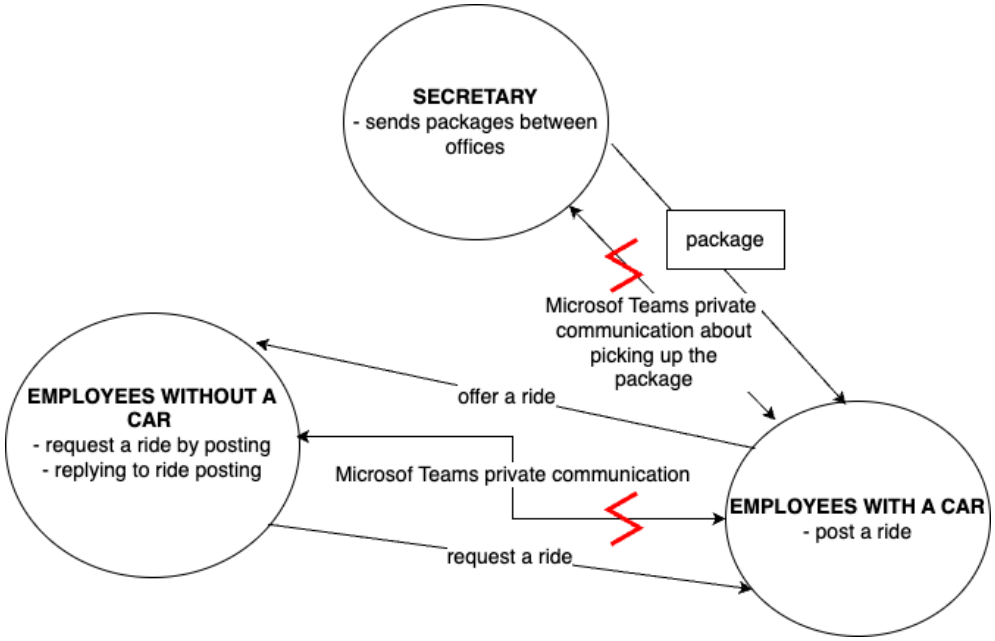

Figure 1: Information flow of Critical Software current solution using Microsoft Teams

Cultural Model This Cultural Model diagram shown in figure 2 illustrates the layers of influence surrounding an employee in the context of interoffice travel, carpooling, and package delivery coordination. Each layer represents different cultural factors that impact the employee's decisions and behavior.

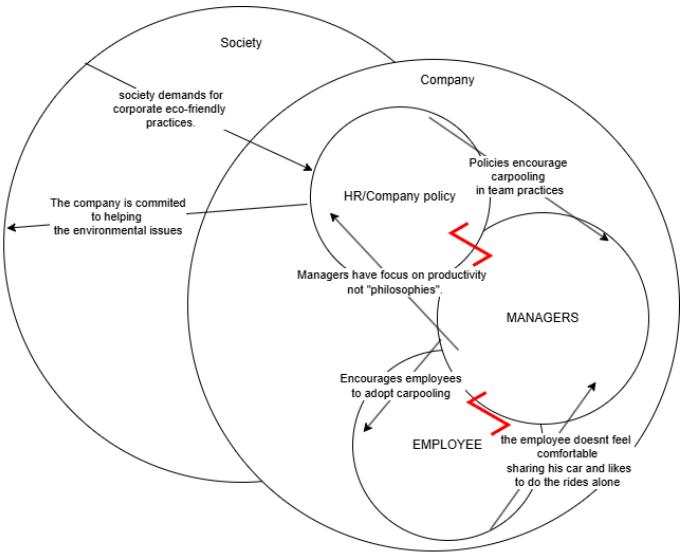

Figure 2: Cultural Model of the System

This model highlights the different influences that shape an employee's travel and carpooling choices, from environmental values to personal preferences, company policies, and social expectations.

Sequence Model The sequence model shown in figure 3 illustrates the user's journey to find a carpool option through Microsoft Teams. The process begins with accessing the Critical Software Group and navigating through specific carpooling channels (e.g., Coimbra, Lisboa, Porto). Users can either scroll to find an appropriate ride or create a post if no suitable options are available, followed by replying to a post or sending a private message to finalize arrangements. Common challenges include difficulty in filtering rides, confusion caused by multiple channels, and unclear ride availability.

The sequence model shown in figure 4 depicts the workflow for users offering carpooling options. Similar to the search model, users navigate Microsoft Teams to access the Critical Software Group and specific carpooling channels. They create a post detailing the offered ride and wait for responses. Challenges identified in this process include the complexity of managing multiple channels and ensuring that ride availability is effectively communicated to potential passengers.

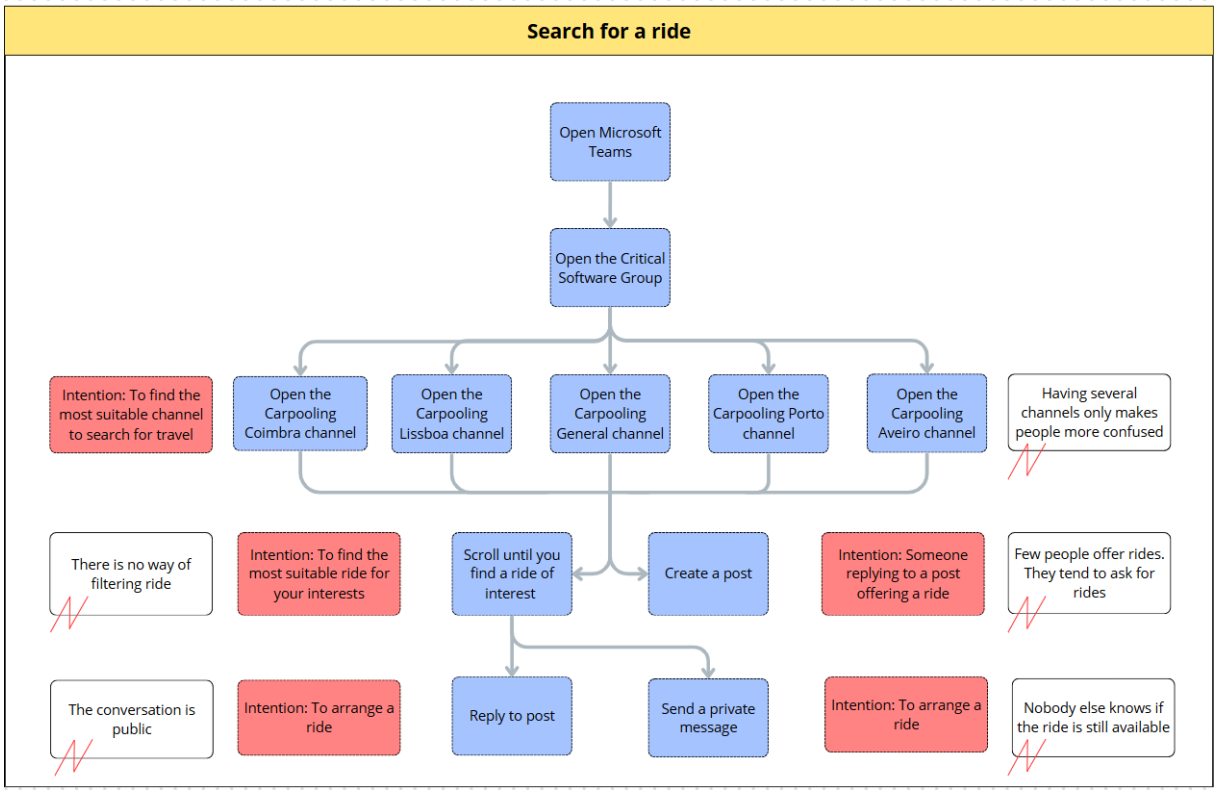

Figure 3: Sequence model - Search Ride

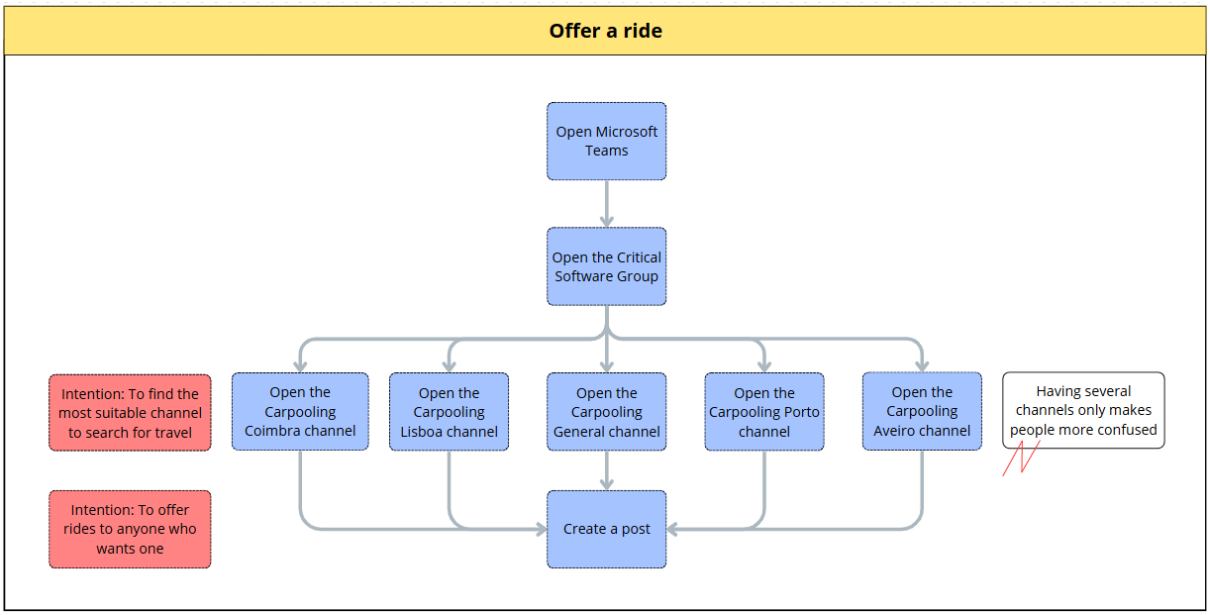

Figure 4: Sequence model - Offer Ride

2.3.2 Personas

In this section we provide a detailed description of the personas involved in our carpooling system.

Fernando (figure 5) is a 31-year-old software engineer seeking a convenient way to find rides between offices, prioritizing sustainability and social interaction with colleagues. Catarina (figure 6) is a 31-year-old people and sustainability specialist who actively promotes carpooling initiatives to reduce carbon emissions and foster a more eco-conscious workplace culture. Sara (figure 7) is a 29-year-old receptionist responsible for coordinating package deliveries between offices. She leverages colleagues' travel schedules to streamline logistics and save resources.

Fernando Hakimi

Age: 31
Occupation: Software Engineer
Location: Coimbra
Income range: 30k-40k

GOALS

- Find rides to between offices in a practical and efficient way.
- Reduce environmental impact.

DESCRIPTION

Fernando is a software engineer who values sustainability. He wants a solution that helps him find rides easily, promoting a positive environmental impact and allowing for social interactions with colleagues.

PERSONALITY

- Introverted and Reflective
- Empathetic and Eco-Conscious
- Low-Key Sense of Humor

MOTIVATION

- Help the environment through ride-sharing practices.

INFLUENCE

- Friends and colleagues who already use carpooling.
- Companies that promote sustainability and environmental responsibility.

FRUSTRATION

- Low engagement from colleagues with carpooling initiatives.
- Difficulty finding rides at specific times he needs.
- Lack of feedback on the environmental impact of his actions.

INTERESTS

- Gaming
- Tech projects
- Cycling

Figure 5: Persona requesting a ride

Catarina Fonseca

Age: 31
Occupation: People/Training and Sustainability Specialist
Location: Coimbra
Income range: 25k-35k

Post ride when she has a meeting in another office

Reduce environmental impact.

DESCRIPTION

Catarina plays a strategic role that merges People/Training with Sustainability at Critical Software. She's responsible for aligning employee well-being with environmental responsibility and creates initiatives like carpooling to foster eco-conscious habits. Catarina seeks ways to make sustainable actions engaging and accessible.

PERSONALITY

Extroverted

Highly proactive

Goal-oriented

MOTIVATION

Increase participation in sustainability initiatives.

Make environmental impact visible.

Meet colleagues and build a social network at work.

INFLUENCE

Friends and colleagues who already use carpooling.

Companies that promote sustainability and environmental responsibility.

FRUSTRATION

Low engagement from colleagues in carpooling initiatives

Difficulty finding rides at specific times he needs.

Lack of feedback on the environmental impact of his actions.

INTERESTS

Hiking

Paddel

Reading

Figure 6: Persona posting a ride

Sara Pinheiro

Age: 29
Occupation: Receptionist
Location: Coimbra
Income range: 20k - 25K

GOALS

- Streamline package deliveries between offices by leveraging colleagues' existing travel plans.
- Minimize manual work by automating delivery coordination

DESCRIPTION

Sara manages the front desk at Critical Software and is responsible for coordinating package deliveries between the company's offices. Her focus is on keeping internal logistics smooth and efficient, and she aims to save resources by arranging for colleagues to transport packages when they are already traveling.

PERSONALITY

- Organized
- Detail-oriented.
- Practical and solution-focused

MOTIVATION

- Reduce time spent coordinating deliveries manually.
- Improve the efficiency of internal logistics
- Ensure quick, reliable delivery of packages between office

INFLUENCE

- Other administrative staff who have implemented effective internal logistics solutions.
- Colleagues who frequently travel and can offer insights on ideal timing.

FRUSTRATION

- Difficulty in finding colleagues who are traveling at the right times
- Burden of individually coordinating each delivery
- Unpredictable scheduling that sometimes causes delays

INTERESTS

- Watching Movies
- Beach
- Reading

Figure 7: Secretary persona

3 Specific Requirements

3.1 Functional Requirements

This section outlines the functional requirements for a carpooling system integrated within the existing Microsoft Teams and Calendar environment used at Critical Software (CSW). This system addresses the need for efficient interoffice travel coordination, reducing individual car usage, and supporting the company's sustainability goals. The carpooling solution is intended to streamline car-sharing for employees commuting between CSW offices located in Coimbra, Lisbon, Porto, and Braga. The solution should be intuitive and integrated with Microsoft Teams and Calendar to ensure seamless user adoption without introducing additional applications.

3.1.1 User Stories

The following user stories highlight key user needs and expectations of the carpooling system:

- **As an employee with a meeting in another office, I want to choose if I'm requesting, offering or not carpooling when I accept the meeting invitation, so I can easily coordinate carpooling with colleagues.**
- **As an employee, I want to view available rides from colleagues traveling to the same location, so I can join a carpool instead of traveling alone.**
- **As a secretary, I need a way to match package deliveries with employees traveling between offices, so I can optimize delivery using carpooling resources.**

3.1.2 Workflows

This section describes the significant workflows of the system using BPMN. Each workflow combines the individual use cases into coherent sequences that represent complete processes. The actors, external systems, and data artifacts involved are explicitly identified.

Workflow 1: Publish/Offer a Carpool Ride

The workflow shown in figure 8 outlines how an employee creates a post for carpooling after confirming their participation in an event or a meeting. The workflow supports both offering and requesting rides, ensuring that posts are effectively shared with the relevant audience.

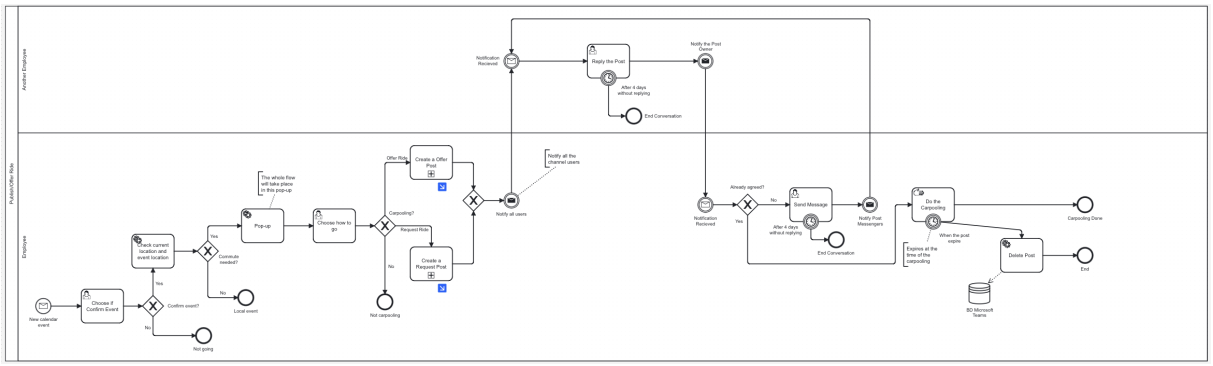

Figure 8: Workflow 1: Offer/Request a Carpool Ride

Decision Points

- **Event Confirmation:** If the employee does not confirm their attendance, the workflow stops without creating a post.
- **Event location:** If commuting is not needed to attend the event, no further actions are taken.
- **Carpool Decision:** If the employee decides not to carpool, no further actions are taken.
- **Interaction with Post:** If no users interact with the post or the carpool expires, the system deletes the post without further processing.

Workflow 2: Join a Carpool Ride Through Channel

The workflow shown in figure 9 allows an employee to browse the carpooling channel for available ride offers and join a ride by replying to a post. The system facilitates interaction between the employee and the post owner, streamlining the agreement and ride confirmation process.

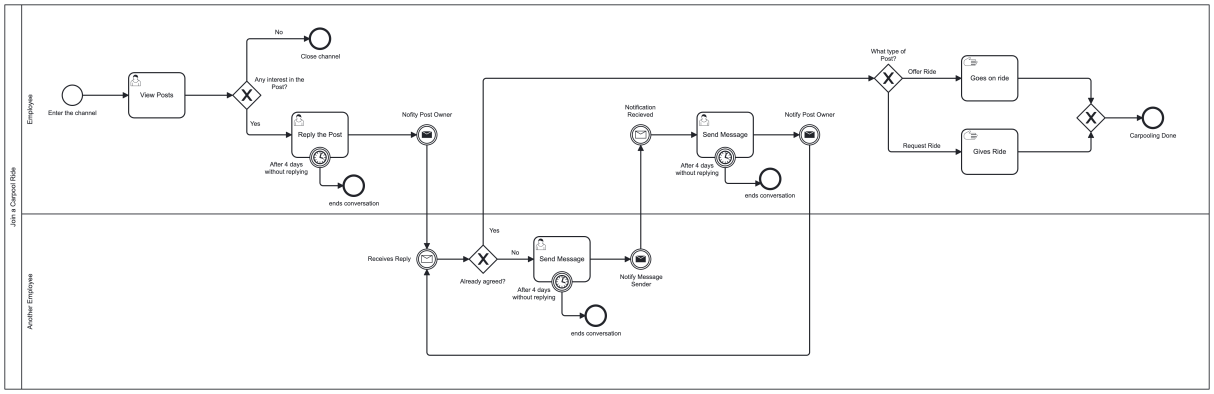

Figure 9: Workflow 2: Join a Carpool Ride Through Notification

Decision Points

- **Interest in Post:** If no suitable post is found, the employee exits the channel without initiating a request, ending the process.
- **Post Owner Response:** If the post owner does not respond within a set time frame, the conversation ends.
- **Agreement Reached:** If the post owner rejects the request, the process ends without further action.

Workflow 3: Arrange Package Delivery via Carpool

This workflow enables a secretary to schedule the delivery of a package via a carpool ride. The system supports finding a suitable ride, requesting package delivery, and coordinating the pickup and handoff between the secretary and the driver.

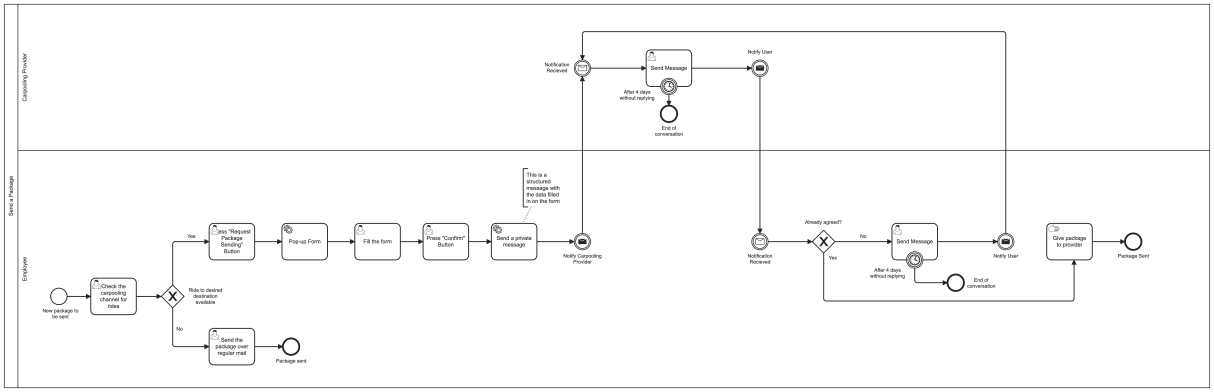

Figure 10: Workflow 3: Arrange Package Delivery via Carpool

Decision Points

- Ride Availability: If no suitable ride is available, the package is sent via an alternative method, terminating the workflow.
- Driver Response: If the driver rejects the delivery request, the secretary is notified, and the workflow ends without completing the delivery.
- Coordination of Delivery: If the secretary and the driver fail to coordinate the pickup, the delivery may be delayed or canceled.

3.1.3 Use Cases

Use Case (Level 0)

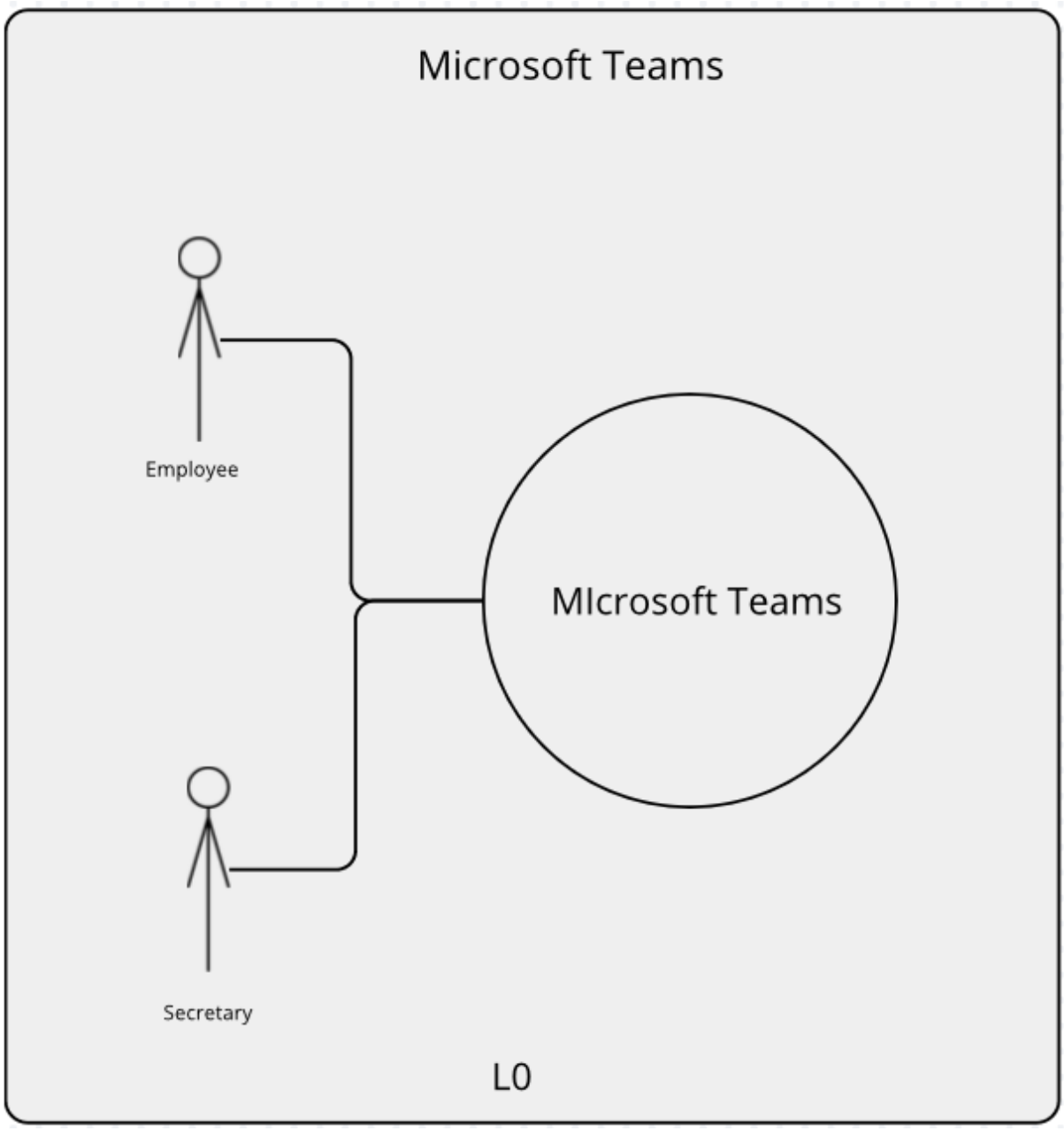

Figure 11: Use Case Diagram (Level 0)

Use Case (Level 1)

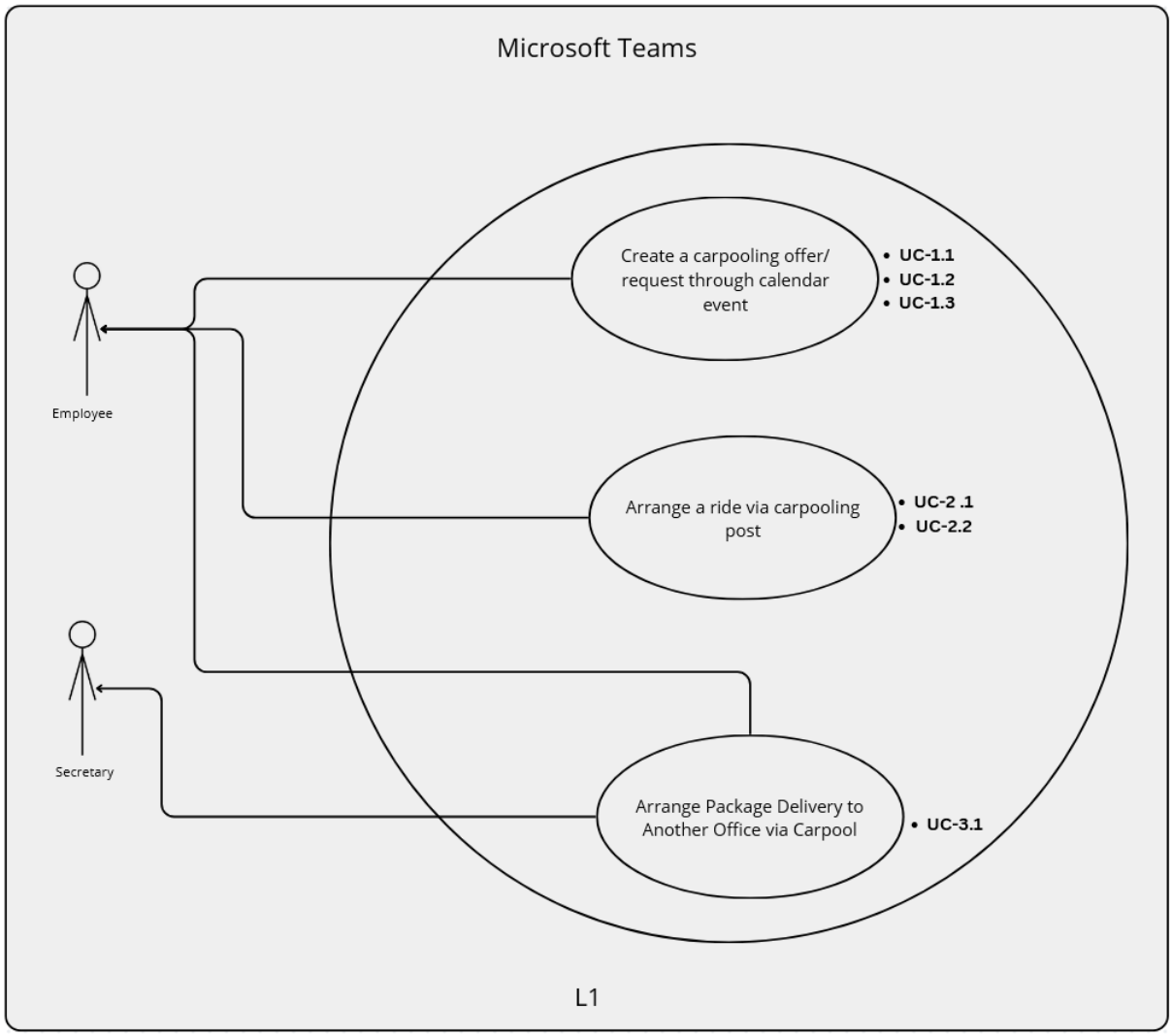

Figure 12: Use Case Diagram (Level 1)

Use Case: Offer a Ride (UC-01.1)

Actors Description

- **Employee with Car:** An employee who can offer a ride to colleagues when traveling between offices.

Use Case Details

- **ID:** UC-01.1
- **Name:** Offer a Ride
- **Level:** Sea
- **Source:** Client Requirements
- **Short Description:** When an employee confirms a meeting in another office, they are prompted to specify travel mode, either offering a ride, requesting a ride, or opting out of carpooling. They choose to offer a ride and the details are posted to a Teams channel.
- **Success Guarantees:** Successfully posts carpool details to the designated Teams channel, making the travel option visible to relevant employees.
- **Trigger:** Employee confirms a meeting scheduled in another office.
- **Main Actor:** Employee with a scheduled meeting in a different office
- **Preconditions:** Employee has an upcoming meeting in a different city office.

Use-Case Steps

Step	Actor Action (Employee)	System Responsibility
1	Employee confirms the meeting in the Calendar app.	The system prompts travel mode confirmation with three options: “Offer a ride,” “Request a ride,” or “I am not carpooling.
2	Selects "Offer a ride".	The system prompts a form to fill the details about departure time, seats available and local of departure.
3	Employee fills the forms and clicks "Confirm" button	Posts carpool details to a Teams channel, tagging relevant employees.

Exceptions:

- If the employee closes the confirmation form without submitting, the post is not created. Having the ability to do it later by going to the calendar event.

Post-conditions: Ride details are visible in Teams.

Use Case: Request a Ride (UC-01.2)

Actors Description

- **Employee:** An employee who needs to request a ride from colleagues when traveling between offices.

Use Case Details

- **ID:** UC-01.2
- **Name:** Request a Ride
- **Level:** Sea
- **Source:** Client Requirements
- **Short Description:** When an employee confirms a meeting in another office, they are prompted to specify travel mode, either offering a ride, requesting a ride, or opting out of carpooling. They choose to request a ride and the details are posted to a Teams channel.
- **Success Guarantees:** Successfully posts carpool details to the designated Teams channel, making the travel request visible to relevant employees.
- **Trigger:** Employee confirms a meeting scheduled in another office.
- **Main Actor:** Employee with a scheduled meeting in a different office
- **Preconditions:** Employee has an upcoming meeting in a different city office.

Use-Case Steps

Step	Actor Action (Employee)	System Responsibility
1	Employee confirms the meeting in the Calendar app.	The system prompts travel mode confirmation with three options: “Offer a ride,” “Request a ride,” or “I am not carpooling.
2	Selects "Request a ride".	The system prompts a form to fill the details about arrival date and time, local of departure.
3	Employee fills the forms and clicks "Confirm" button	Posts carpool details to a Teams channel, tagging relevant employees.

Exceptions:

- If the employee closes the confirmation form without submitting, the post is not created. Having the ability to do it later by going to the calendar event.

Post-conditions: Ride details are visible in Teams.

Use Case: Opt out of carpooling (UC-01.3)

Actors Description

- **Employee:** An employee who does not want to carpool.

Use Case Details

- **ID:** UC-01.3
- **Name:** Opt out of carpooling
- **Level:** Sea
- **Source:** Client Requirements
- **Short Description:** When an employee confirms a meeting in another office, they are prompted to specify travel mode, either offering a ride, requesting a ride, or opting out of carpooling. They choose to not carpool.
- **Success Guarantees:** Closes the carpooling forms.
- **Trigger:** Employee confirms a meeting scheduled in another office.
- **Main Actor:** Employee with a scheduled meeting in a different office
- **Preconditions:** Employee has an upcoming meeting in a different city office.

Use-Case Steps

Step	Actor Action (Employee)	System Responsibility
1	Employee confirms the meeting in the Calendar app.	The system prompts travel mode confirmation with three options: “Offer a ride,” “Request a ride,” or “I am not carpooling.
2	Selects "Im not carpooling".	The system closes the forms.

Use Case: Arrange a ride via carpooling post as a passenger (UC-02.1)

Actors Description

- **Employee:** An employee seeking carpool options for a meeting.

Use Case Details

- **ID:** UC-02.1
- **Name:** Arrange a Ride via Carpooling Post as a Passenger
- **Level:** Sea
- **Source:** Client Requirements
- **Short Description:** The requester and the driver communicate via the post replies to arrange the carpool.
- **Trigger:** An employee (requesting) views a carpool offer and initiates a chat with the employee offering a ride.
- **Main Actor:** Employee needing a ride.
- **Secondary Actors:** Employee offering a ride.
- **Preconditions:** The employee who offers a ride has posted a carpool offer.
- **Success Guarantees:** The driver and passenger agree on all carpool details.

Steps

Step	Actor Action	System Responsibility
1	Navigates to the carpooling channel within the Teams app and reviews available ride offers.	Displays all available rides.
2	The requester initiates a conversation with the driver in the desired post.	Facilitates real-time communication between the requester and the driver.
3	Agrees with the driver on the carpooling details.	No further system responsibility.

Use Case: Arrange a ride via carpooling post as a driver (UC-02.2)

Actors Description

- **Employee Offering a Ride:** An employee offering carpool opportunities.

Use Case Details

- **ID:** UC-02.2
- **Name:** Arrange a ride via carpooling post as a driver
- **Level:** Sea
- **Source:** Client Requirements
- **Short Description:** The driver and the requester communicate via the post replies to arrange the carpool.
- **Trigger:** An employee (driving) views a carpool request and initiates a chat with the employee wanting a ride.
- **Main Actor:** Employee offering a ride.
- **Secondary Actor:** Employee needing a ride.
- **Preconditions:** The employee who wants a ride has posted a carpool request.
- **Success Guarantees:** The driver and passenger agree on all carpool details.

Steps

Step	Actor Action	System Responsibility
1	Navigates to the carpooling channel within the Teams app and reviews available ride requests.	Displays all available rides.
2	The driver initiates a conversation with the requester in the desired post.	Facilitates real-time communication between the requester and the driver.
3	Agrees with the requester on the carpooling details.	No further system responsibility.

Use Case: Arrange Package Delivery to Another Office (UC-03.1)

Actors Description

- **Secretary:** Responsible for scheduling package deliveries between offices.
- **Driver:** Employee offering a ride to the destination of the package.

Use Case Details

- **ID:** UC-03.1
- **Name:** Arrange Package Delivery to Another Office
- **Level:** Sea
- **Source:** Client Requirements
- **Short Description:** The secretary identifies available rides for package delivery and requests delivery through a chosen ride.
- **Minimal Guarantees:** The system ensures that the secretary can manually inspect available ride offers and take further action as needed.
- **Success Guarantees:** The secretary manually identifies a ride and successfully contacts the driver for coordination.
- **Trigger:** Secretary needs to send a package to a different office.
- **Main actor:** Secretary with a package to deliver.
- **Secondary actor:** Employee with availability to deliver the package.
- **Preconditions:** The secretary has a package ready to deliver and access to the carpooling system.
- **Post-conditions:** The package is assigned to a ride, and coordination details are provided.

Steps

Step	Actor Action (Secretary)	System Responsibility
1	Navigates to the carpooling channel within the Teams app and reviews available ride offers.	Displays all available rides.
2	Selects a suitable ride offer and clicks the "Request Package Sending" button.	Opens a form in a popup window for the secretary to fill out.
3	Fills out the form with package details (size, pickup location, and delivery location) and confirms the request.	Sends a direct message to the driver with the package delivery details.
4	Waits for driver's response.	Updates the secretary with the driver's acceptance or rejection.
5	If confirmed, coordinates pickup details directly with the driver via Teams direct messaging.	No further system responsibility; relies on user interaction.

3.1.4 Navigation Layout (UED)

The User Experience Design (UED) in figure 13 outlines the navigation layout for the Critical Rides system, embedded into Microsoft Teams and Calendar. The layout ensures seamless integration and intuitive navigation for all users.

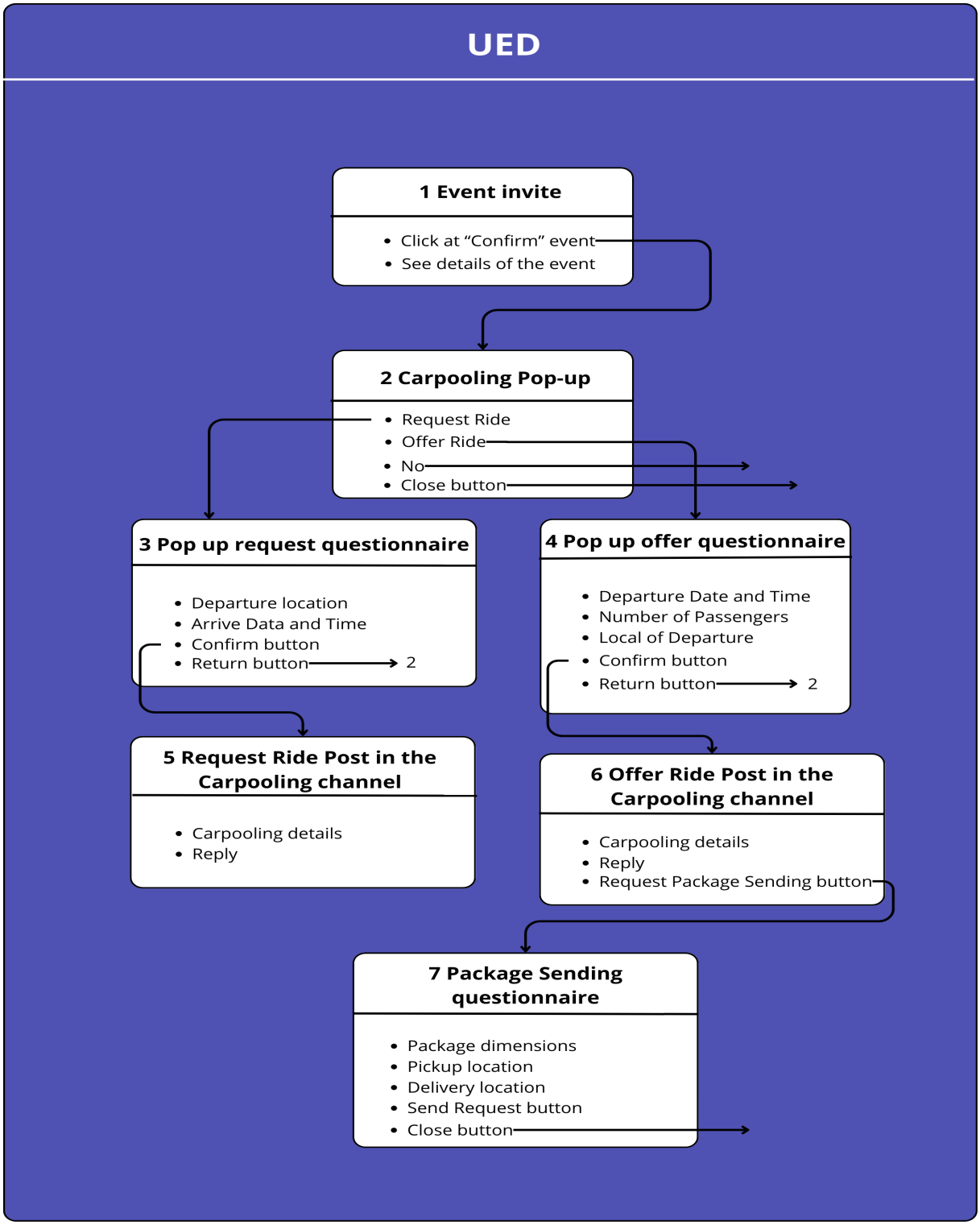

Figure 13: UED for Critical Rides

3.2 Quality Attributes

- **Usability:** The interface must be simple, intuitive, and easy to use, ensuring that users can interact with the application without difficulties or technical support.
- **Robustness:** The system must handle unexpected errors or failures while maintaining data integrity and operation. It should prevent failures from spreading and offer automatic recovery mechanisms and clear notifications.
- **Interoperability:** The application must integrate seamlessly with Microsoft Teams and external API services, ensuring consistent and reliable data exchange between tools and technologies.

Concrete Scenarios

Usability

- **Scenario:** Create a "Request" or "Offer a Ride" through a meeting in the calendar.
- **Source of Stimulus:** Employee.
- **Stimulus:** Create a request/offer ride through the calendar.
- **Environment:** Under normal behavior, the system is operational and integrated into the Microsoft Teams calendar.
- **Artifact:** System.
- **Response:** The system provides a Teams' extension to create a carpooling request/offer and guides the user through a form or questionnaire.
- **Response Measure:** The user completes the flow in less than 1 minute, without the need for additional help or instructions.

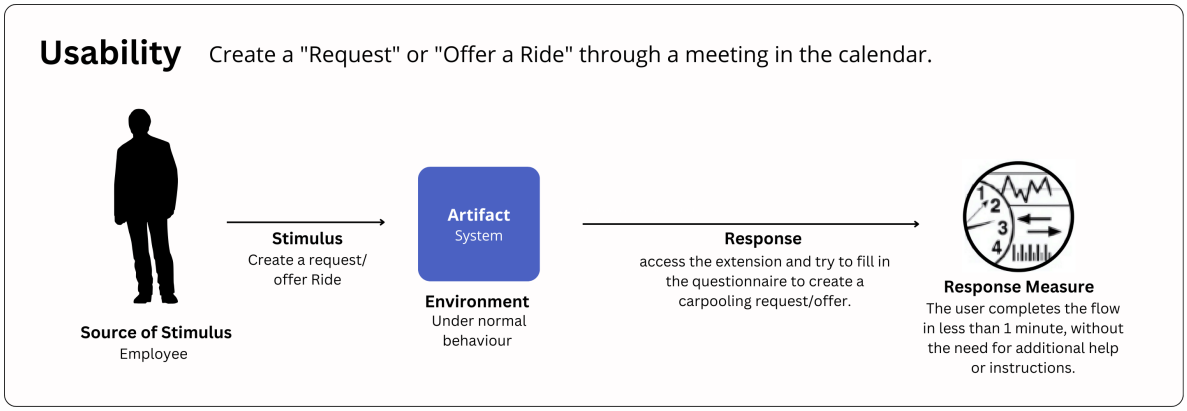

Figure 14: Usability Scenario for Request or Offer a Ride

Robustness

- **Scenario:** Publish a "Request" or "Offer a Ride" in the carpooling channel when a connectivity failure occurs.
- **Source of Stimulus:** Employee.
- **Stimulus:** Submits data from the carpooling questionnaire.
- **Environment:** The system is operational, but a call to the Teams API fails momentarily.
- **Artifact:** System.
- **Response:** The system detects the failure and retries posting the request or offer up to 3 times, with 5-second intervals. If all attempts fail, the system notifies the user and saves the data locally for later manual submission.
- **Response Measure:** 95% of operations must be successfully completed after retries, and 100% of data remains saved locally in case of persistent failure.

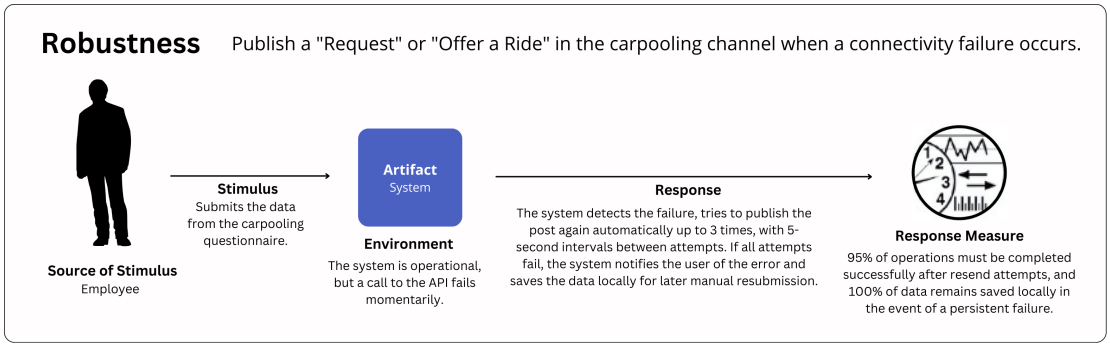

Figure 15: Robustness Scenario for Publishing a Ride

Interoperability

- **Scenario:** Send questionnaire data to the carpooling channel via the Team's API.
- **Source of Stimulus:** Employee.
- **Stimulus:** The user fills in the questionnaire and submits the data to be published on the Teams channel.
- **Environment:** Under normal behavior, the system is operational and communicates with the Teams API.
- **Artifact:** System.
- **Response:** The system sends the filled-in data to the correct Teams channel via an API call, which publishes the corresponding post (Request or Offer).
- **Response Measure:** The API publishes the post on the channel within 2 seconds of submission, without errors, and all relevant recipients receive the associated notifications.

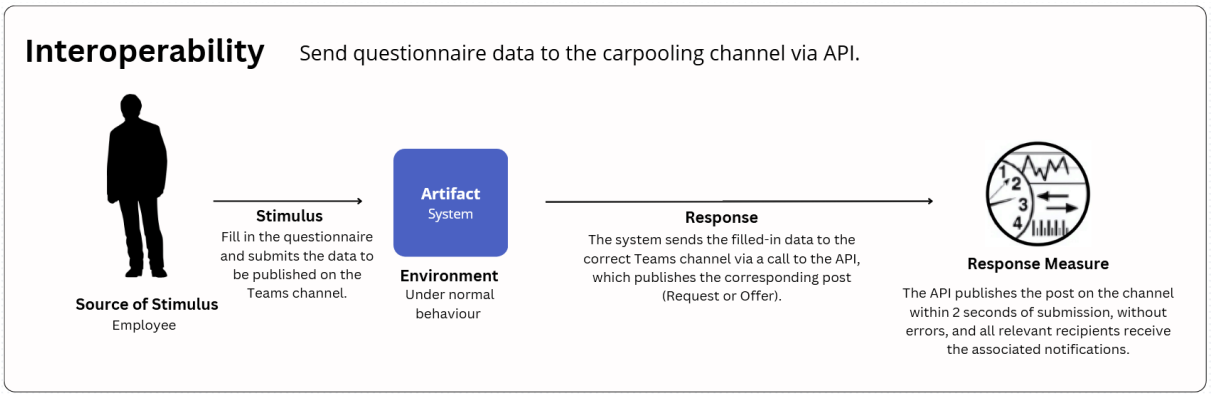

Figure 16: Interoperability Scenario for API Integration

Conflicts Between Quality Attributes

According to the Quality Attribute Interaction Matrix, there are no identified conflicts between the selected quality attributes for our system.

	Availability	Efficiency	Flexibility	Integrity	Interoperability	Maintainability	Portability	Reliability	Reusability	Robustness	Testability	Usability
Availability							+		+			
Efficiency			−		−	−	−	−		−	−	−
Flexibility		−		−		+	+	+			+	
Integrity		−			−				−		−	−
Interoperability		−	+	−			+					
Maintainability	+	−	+					+			+	
Portability		−	+		+	−			+		+	−
Reliability	+	−	+			+				+	+	+
Reusability		−	+	−	+	+	+	−			+	
Robustness	+	−						+				+
Testability	+	−	+			+		+				+
Usability		−								+	−	

Figure 17: Quality Attribute Interaction Matrix

New requirements that emerges from this dimension

ID	Requirement Description	Category	Priority
NFR-01	The system must ensure a simple, intuitive, and user-friendly interface, enabling seamless interaction for employees without requiring additional training or technical support (Usability).	Non-Functional	High
NFR-02	The system must handle unexpected errors, maintain operation during partial failures, and ensure data integrity by isolating issues and implementing automatic recovery mechanisms (Robustness).	Non-Functional	High
NFR-03	The system must integrate seamlessly with Microsoft Teams, ensuring reliable and secure data exchange between enterprise tools and services (Interoperability).	Non-Functional	High

Table 3: Non-Functional Requirements

3.3 Business Constraints

The product does not impose specific business constraints, such as requiring user registration or implementing payment systems. As a result, no additional requirements arise from this dimension.

3.4 Legal and Regulatory Constraints

The development of a carpooling solution for Critical Software introduces several legal and regulatory challenges that must be addressed to ensure compliance with local laws, company policies, and employee expectations. This section outlines key areas of concern and requirements.

Civil Liability for Passengers (LG-01)

Drivers participating in the carpooling system bear a civil responsibility for the safety of their passengers during the journey. It is essential to define and communicate these responsibilities clearly to mitigate potential disputes. Passengers should also acknowledge the risks associated with voluntary participation in carpooling trips.

Civil Liability for Transported Items (LG-02)

Drivers are responsible for safeguarding packages they agree to transport, ensuring they are handled and delivered securely. However, they should not be held liable for events beyond their control, such as accidents or theft, unless negligence can be demonstrated. Senders must agree to terms that clarify the limits of driver responsibility. This ensures mutual understanding and avoids disputes.

Transport Documentation (LG-03)

In Portugal, companies are required to issue a transport guide ("guia de transporte") when transporting goods, even between offices. This is essential to demonstrate compliance with transport regulations and clarify the purpose of the transported items if questioned by authorities, such as during a police stop.

3.5 Wireframes and Usability Checking

This section presents low-fidelity wireframes for three key focus areas identified during the design process. The wireframes are evaluated using Nielsen's usability heuristics. The evaluation includes a detailed assessment and a score on a scale of 0-10.

Wireframe 1: Confirm Travel and Offer a Carpool Ride

This wireframe (figure 18) represents the interface where an employee confirms their attendance at a meeting or event and specifies whether they will offer a carpool ride or request one. The interaction is triggered by a calendar event, and the employee can fill out the required details for the post, such as departure time and the number of available seats if offering a ride. Conversely, if requesting a ride, they must fill in the post with the location of departure, as well as the desired arrival date and time. Upon confirmation, the details are automatically posted in the carpooling channel, notifying other employees and enabling interaction.

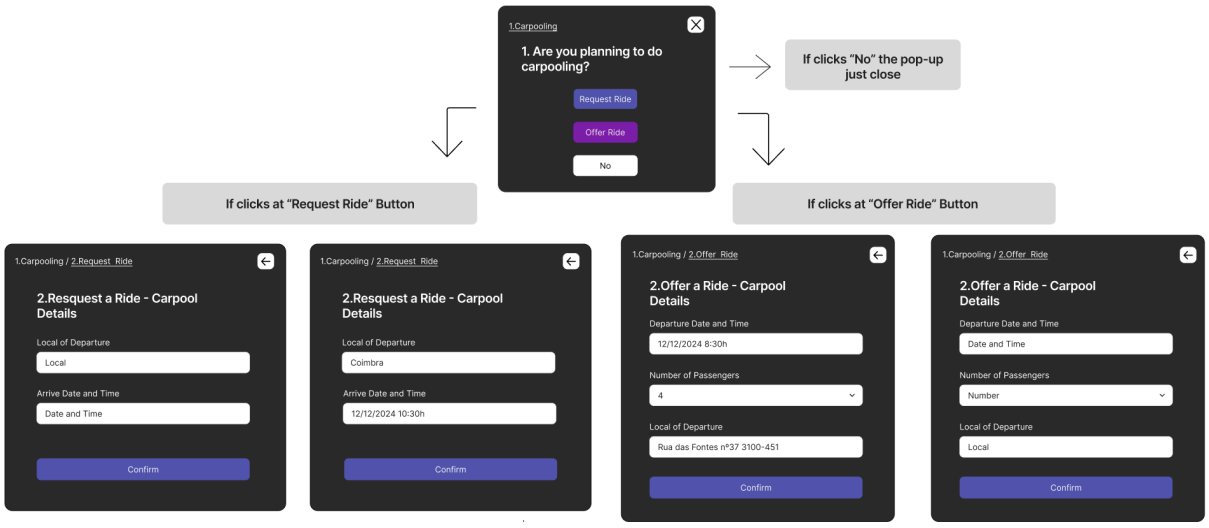

Figure 18: Wireframe 1: Confirm Travel and Offer a Carpool Ride

Wireframe 2: Join a Carpool Ride Through Notification

This wireframe (figure 19) illustrates the process of browsing and replying to ride posts in the carpooling channel. When employees receive a notification about a new event or ride post, they are made aware of the opportunity. From that point, employees can view available rides, select one, and initiate a conversation with the post owner. The interface facilitates the exchange of messages and the final agreement for the ride.

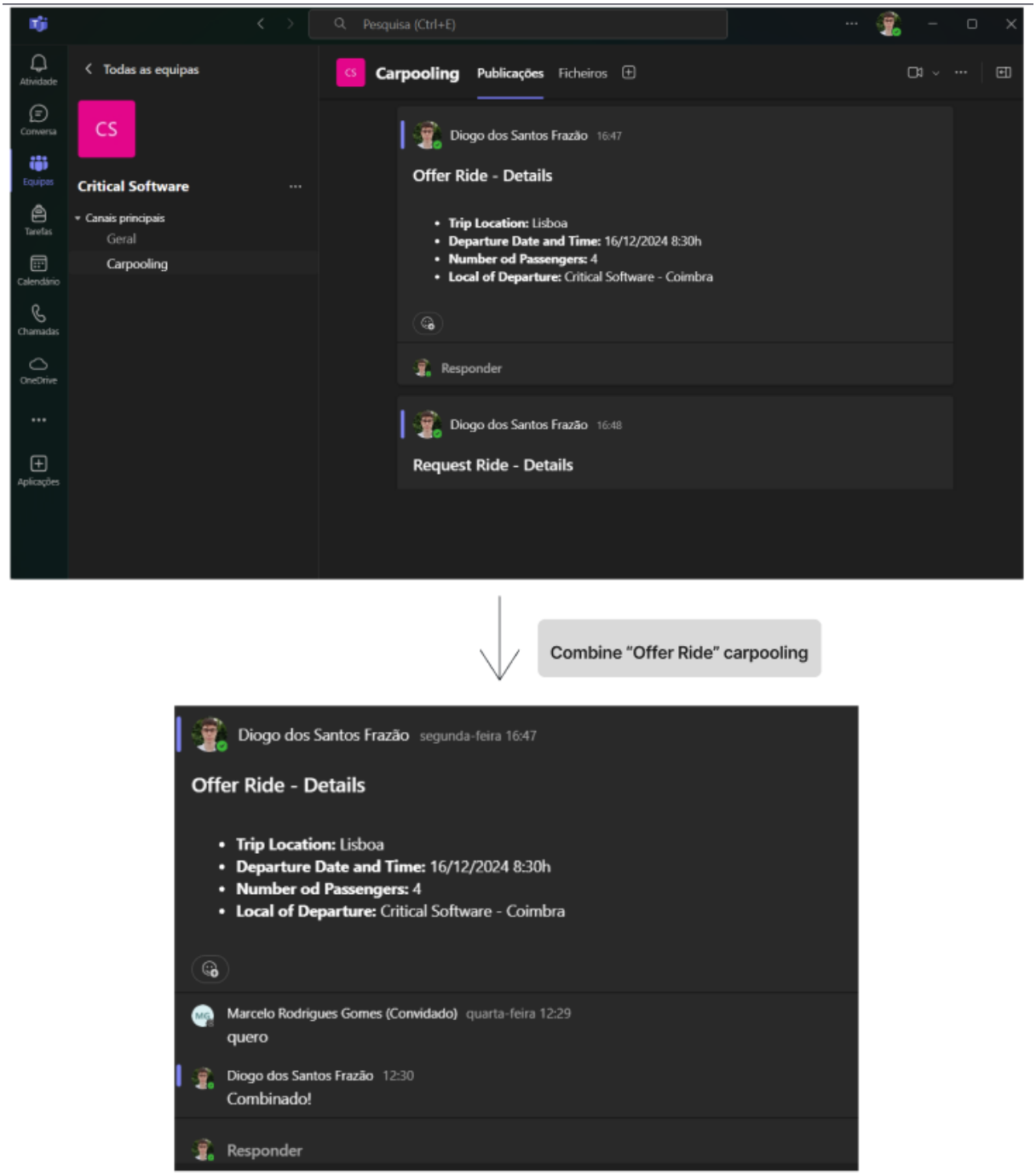

Figure 19: Wireframe 2: Join a Carpool Ride Through Notification

Wireframe 3: Arrange Package Delivery via Carpool

This wireframe (figure 20) details the process for secretaries to arrange package delivery using available carpool rides. It includes checking for suitable rides, filling out a delivery request form, and notifying the driver. The interaction ensures efficient coordination between the secretary and the driver.

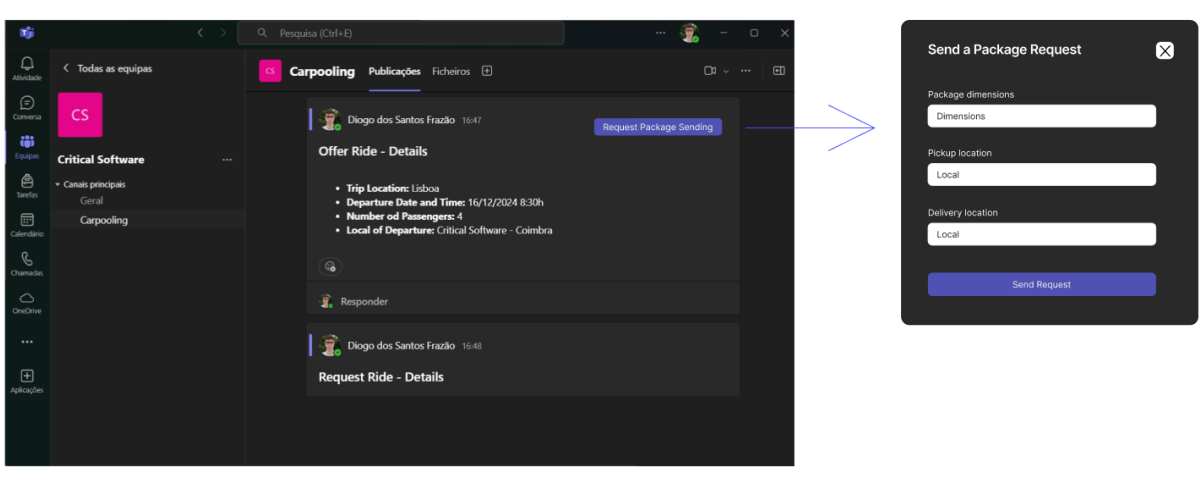

Figure 20: Wireframe 3: Arrange Package Delivery via Carpool

3.5.1 Nielsen's heuristics

The following table(table 4) presents the heuristic evaluation results based on **Nielsen's heuristics** applied to all wireframes. A single review was conducted for all wireframes, as they are nearly identical or entirely consistent in their alignment with the heuristic principles. Each heuristic is assessed for compliance, with details provided to explain how the wireframes align or fall short of the heuristic principles.

Heuristic	**Met?**	**Details**
Visibility of system status	Yes	Notifications are clear and timely.
Match between system and the real world	Yes	The language used matches user expectations.
Consistency and standards	Yes	The wireframes maintain consistent UI patterns and interactions.
Error prevention	Yes	Prompts guide users to avoid mistakes.
Recognition rather than recall	Yes	Options are visible and easy to select.
Flexibility and efficiency of use	Yes	The flow is optimized for new or infrequent users.
Aesthetic and minimalist design	Yes	The visual design is clean, uncluttered, and focused on the core functionality.
Help users recognize, diagnose, and recover from errors	Yes	Error messages are provided.
Help and documentation	No	The wireframes don't include dedicated help or documentation.
User control and freedom	No	Limited options for undoing actions.

Table 4: Heuristic Evaluation Results

The evaluation demonstrates that the wireframes successfully meet **8 out of 10** of Nielsen's heuristics, achieving a **score of 8/10**. However, the absence of dedicated help or documentation and limited user control (e.g., undo functionality) are areas for improvement. Addressing these gaps would further enhance usability and user experience.

4 RE Techniques Assessment

4.1 Problem Domain and Framework Placement

The problem domain for this project is inherently open and dynamic, as it required an in-depth exploration of user workflows, behaviors, and evolving needs within a flexible context. During interviews conducted as part of the contextual design process, a crucial insight emerged: the need to incentivize employees to carpool. Specifically, it was found that prompting employees about carpooling opportunities when a new event is added to their calendar could significantly encourage participation.

Based on these insights, the project is positioned in the dynamic and open quadrant of the framework (figure 21). This quadrant aligns with exploratory approaches that emphasize adaptability, user involvement, and iterative refinement.

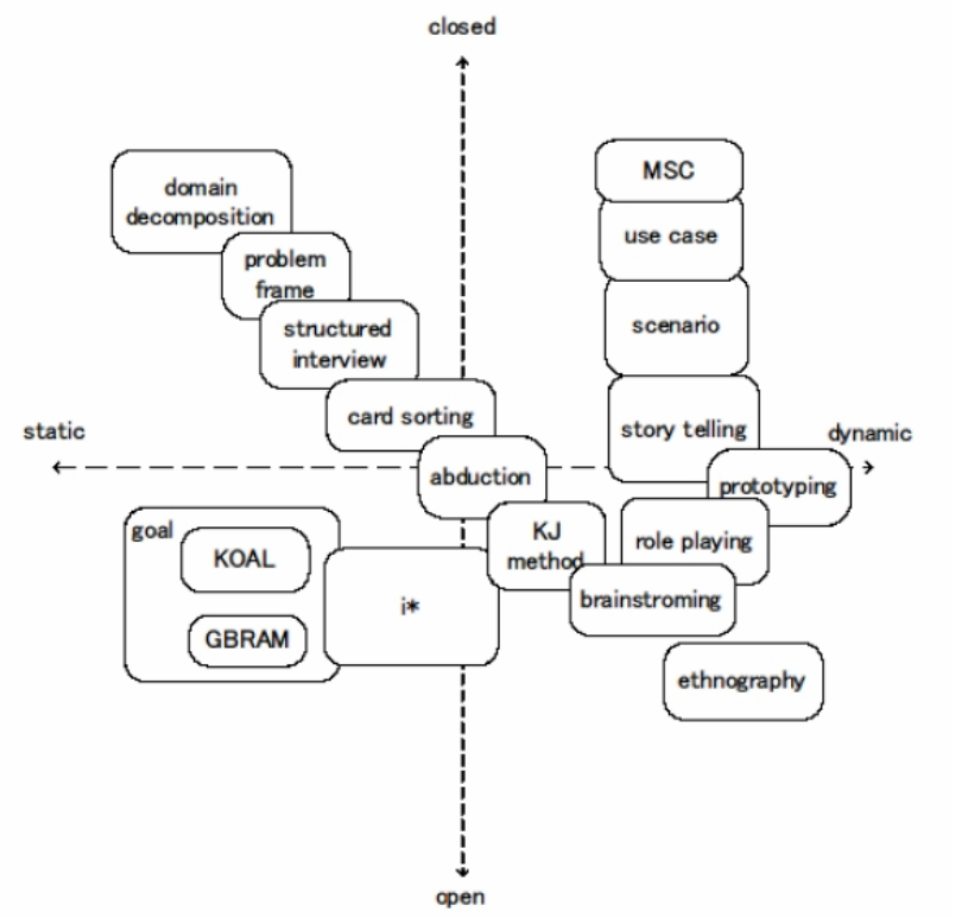

Figure 21: Framework for RE techniques

4.2 Ranking of Techniques for Requirements Elicitation

1. **Contextual Design (D2)**
   Ranking: Most Useful
   Contextual Design was the most impactful phase, as it provided critical insights into how employees use Teams in their daily workflows. Through ethnographic interviews, observations, and workflow analysis, we identified barriers to carpool adoption and recognized the necessity of integrating the solution into the Teams environment. The idea of prompting employees about carpooling during calendar event confirmations emerged directly from this analysis, forming the foundation for our final solution. Without this deep understanding of user behavior and context, we wouldn't have been able to address the core issues effectively.

2. **Workflows and Interface (D4)**
   Ranking: Very Useful
   Creating workflows using BPMN was essential for mapping and understanding the required interactions, ultimately enabling us to design a seamless solution. By visualizing how employees would request or offer rides and how these processes would integrate with Microsoft Teams, we ensured usability and alignment with existing practices. This phase also highlighted specific touchpoints, such as the calendar confirmation workflow, which were critical for promoting carpool adoption. However, these workflows were built on the foundational insights provided by Contextual Design, making it indispensable to their success.

3. **User Stories and Use Cases (D3)**
   Ranking: Useful
   User Stories and Use Cases helped define the system from the user's perspective, articulating specific features and interactions such as requesting or offering rides. This phase ensured the solution addressed real user needs and focused on achieving user goals rather than adding extraneous features. While useful for refining the scope and detailing functionality, this technique was dependent on the insights gained in Contextual Design and on the workflows developed in the Workflows and Interface phase.

4. **Non-Functional Requirements (D5)**
   Ranking: Useful, but Secondary
   Non-Functional Requirements were less central to the requirements elicitation process because our solution was integrated with Microsoft Teams, which already ensures critical aspects like performance, reliability, and security. Most NFRs were inherited from the Teams platform, making this phase less influential in shaping the solution. However, they played a role in ensuring the solution worked efficiently and aligned with the platform's constraints.

5. **State-of-the-Art/Practice Analysis (D1)**
   Ranking: Least Useful
   Reviewing the state-of-the-art and existing carpooling solutions provided general context and helped us understand trends, but it had minimal influence on our specific system. Unlike standalone carpooling platforms, our design had to be tightly integrated with Microsoft Teams, making the insights from traditional carpooling

solutions less applicable. As a result, this phase contributed the least to understanding user needs or defining the problem.

4.3 Reflection on Alignment with the Framework

The top techniques, **D2 - Contextual Design** and **D4 - Workflows and Interface**, align strongly with the dynamic and open quadrant of the framework, which emphasizes adaptability, user involvement, and iterative refinement.

- **D2 - Contextual Design** allowed for in-depth exploration of user workflows and behaviors, uncovering critical insights such as the need to incentivize carpooling. Specifically, we discovered that prompting employees about carpooling opportunities during calendar event creation would encourage participation. This exploratory approach was essential in defining the problem space and tailoring the solution to evolving user needs.
- **D4 - Workflows and Interface** translated these insights into actionable solutions. Using BPMN, we iteratively mapped and refined workflows, ensuring the system seamlessly integrated with Microsoft Teams and fit existing user practices. This step emphasized usability and adaptability, critical for fostering adoption in a dynamic organizational context.

Together, these techniques supported an iterative, user-centered approach that aligned with the open and evolving nature of the problem domain, ensuring the solution was both practical and impactful.

Annexes

Annex A - Full List of Requirements

ID	Requirement Description	Category	Priority
FR-01	Allow employees to create a carpooling offer through calendar event.	Functional	High
FR-02	Allow employees to create a carpooling request through calendar event.	Functional	High
FR-03	Enable employees to arrange a ride via carpooling post.	Functional	High
FR-04	Arrange package delivery to another office via carpooling post.	Functional	Medium
FR-05	Allow employees to delete their own posts when the carpool is already full.	Functional	Medium
NFR-01	The system must ensure a simple, intuitive, and user-friendly interface, enabling seamless interaction for employees without requiring additional training or technical support (Usability).	Non-Functional	High
NFR-02	The system must handle unexpected errors, maintain operation during partial failures, and ensure data integrity by isolating issues and implementing automatic recovery mechanisms (Robustness).	Non-Functional	High
NFR-03	The system must integrate seamlessly with Microsoft Teams, ensuring reliable and secure data exchange between enterprise tools and services (Interoperability).	Non-Functional	High
LG-01	The system must incorporate a liability disclaimer outlining drivers' obligations, passenger acknowledgment of risks, and limits on company liability, tailored to regional laws.	Legal/Regulatory	High
LG-02	The system should incorporate clear terms of use to outline liability related to the transport of packages. These terms should balance the obligations of the driver, sender, and company.	Legal/Regulatory	High
LG-03	The system should generate a digital transport guide automatically whenever an employee transports company-related items.	Legal/Regulatory	High

Table 5: Annex A - Full List of Requirements

Annex B - Traceability Matrix

REQ.ID	Description	UC	BPMN	Wireframe	UED
FR-01	Allow employees to create a carpooling offer through calendar event.	1.1	Fig. 8	Fig. 18	Fig. 13
FR-02	Allow employees to create a carpooling request through calendar event.	1.2	Fig. 8	Fig. 18	Fig. 13
FR-03	Enable employees to arrange a ride via carpooling post.	2.1 | 2.2	Fig. 9	Fig. 19	Fig. 13
FR-04	Arrange package delivery to another office via carpooling post.	3.1	Fig. 10	Fig. 20	Fig. 13

Table 6: Annex B - Traceability Matrix of Functional Requirements

Annex C - Larger Resolution Diagrams

Product Context Diagrams

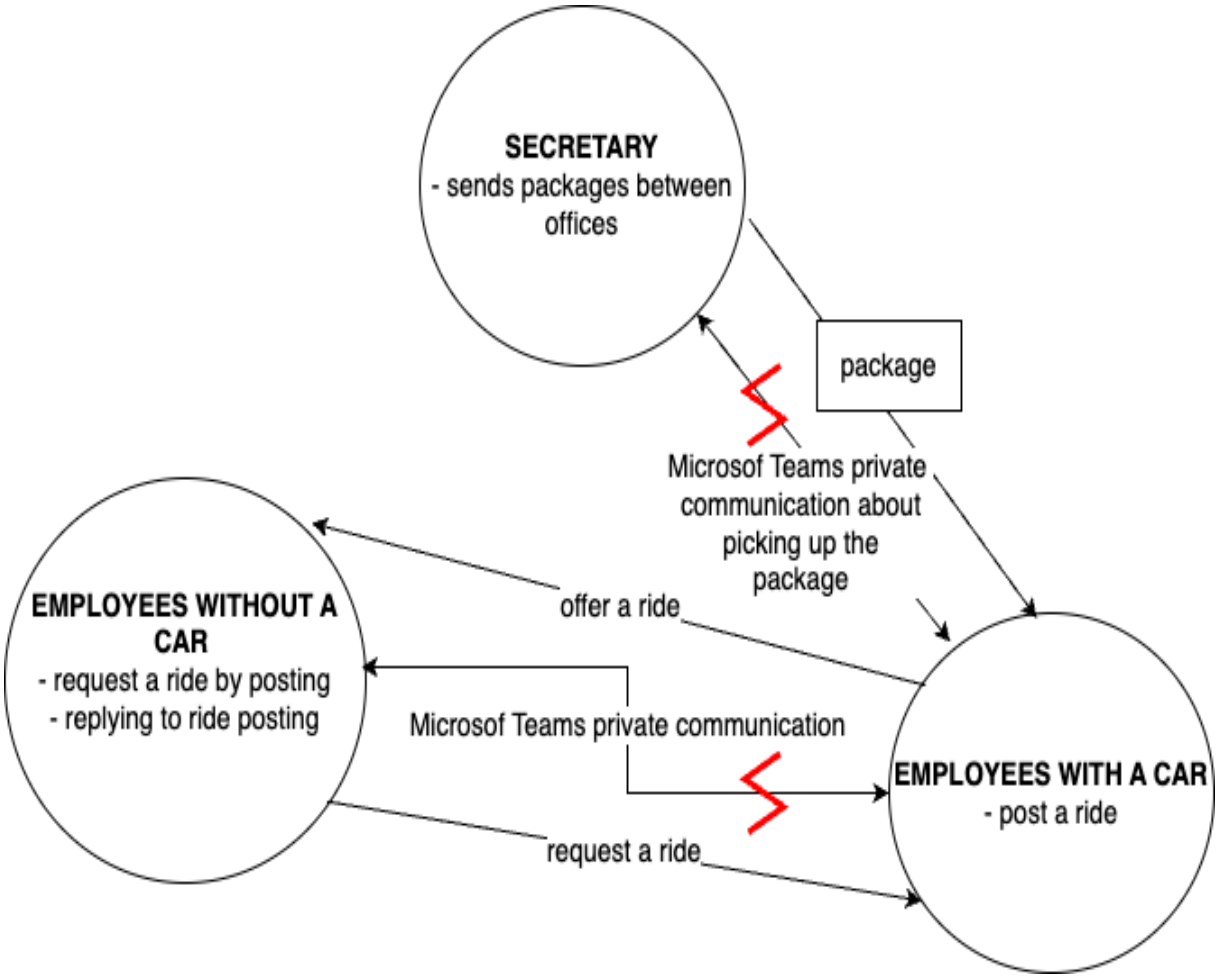

Figure 22: Annex C - Information Flow of Critical Software Current Solution

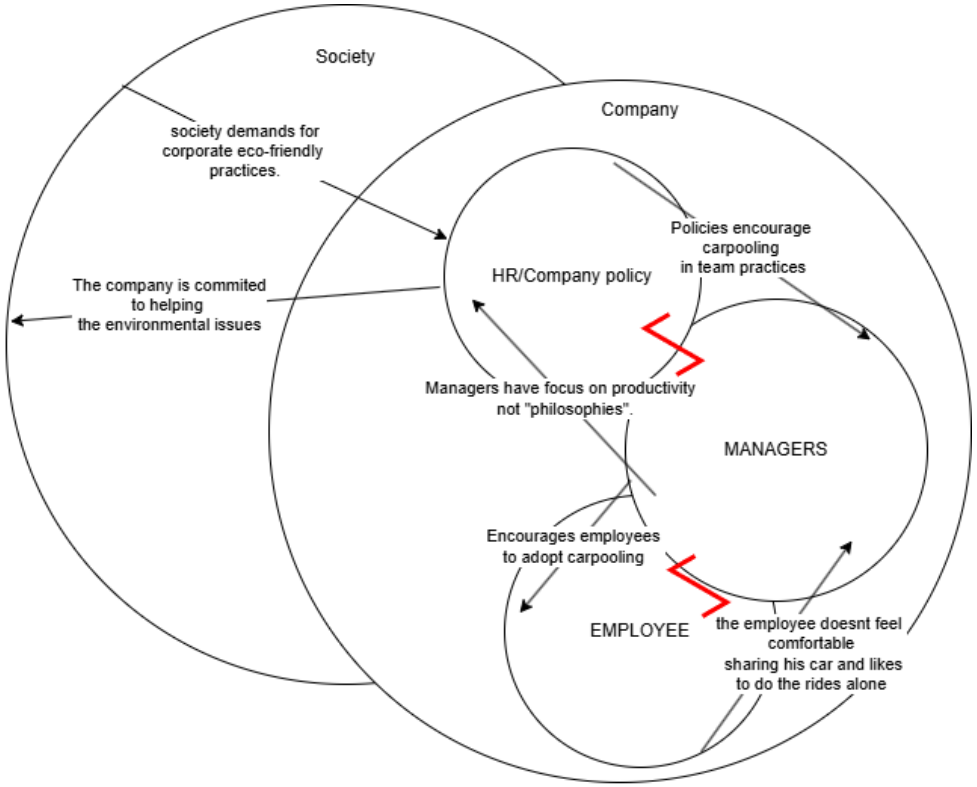

Figure 23: Annex C - Cultural Model of the System

Workflows

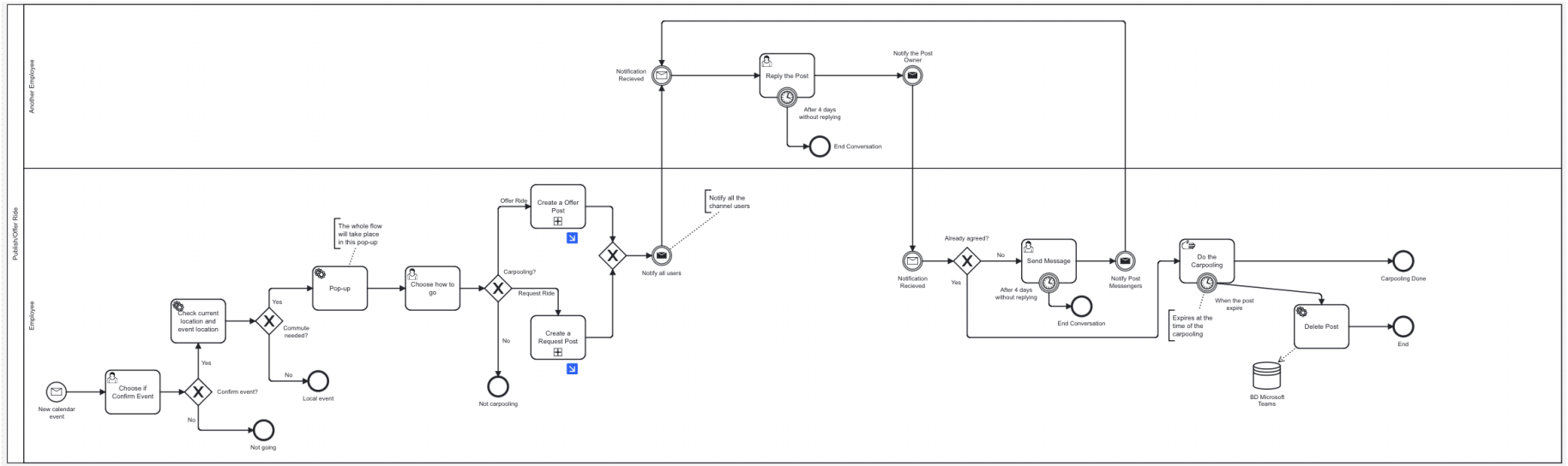

Figure 24: Annex C - Workflow 1: Offer/Request a Carpool Ride

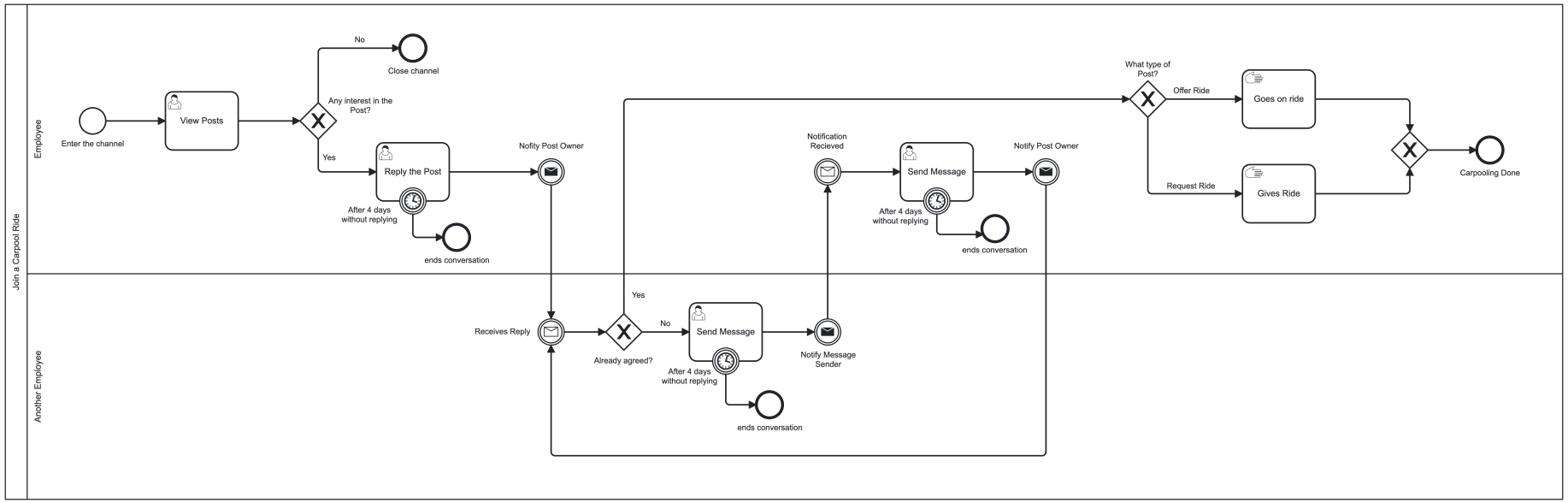

Figure 25: Annex C - Workflow 2: Join a Carpool Ride

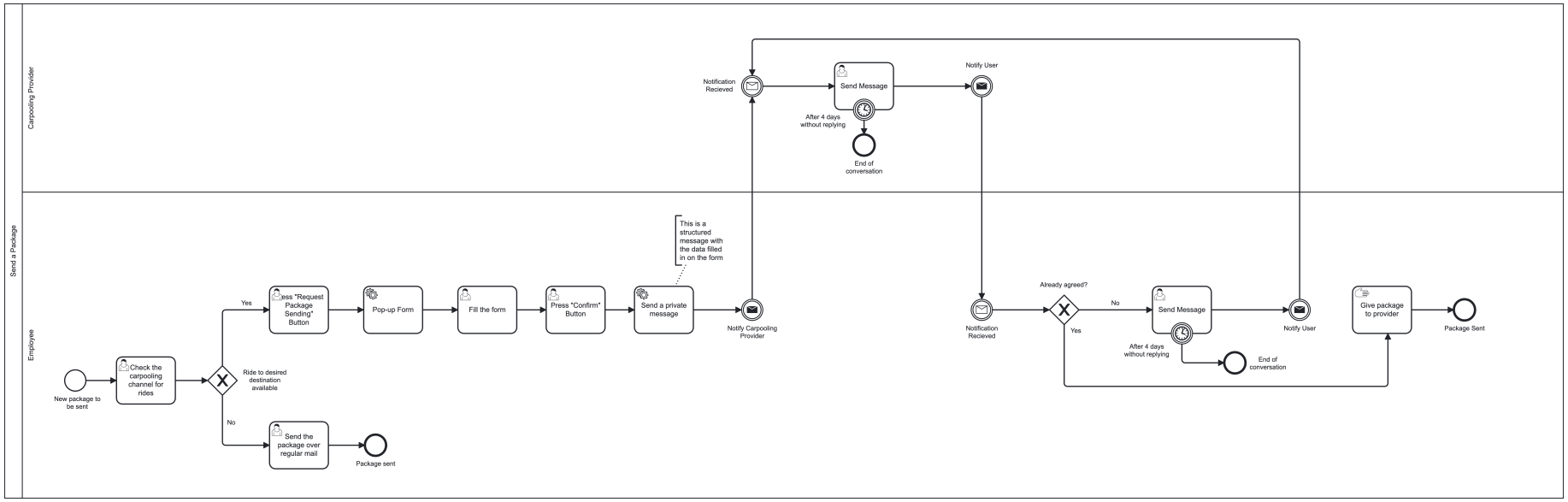

Figure 26: Annex C - Workflow 3: Arrange Package Delivery via Carpool

Annex D - Survey

Table 7: Annex D - Comparison of Features Across Carpooling Applications

Attributes/Requirements	**Comovee**	**Blablacar**	**Poola**	**Uber for Business**	**Twogo**
Map Interface: Users can see colleagues' locations and find potential carpool matches.	X		X		X
Messaging: Users can contact colleagues directly via messages.	X	X	X		
Publishing System: Users can either publish details of their available rides or request for rides.	X	X	X	X	
Carpool Matching: The system matches employees based on their locations, displaying both a matching percentage and the number of matches found.	X		X	X	X
Destination Selection: Users can select a destination to find matching rides.	X	X	X	X	X
Filtering by Ride Details: Users can filter results by ride details, such as the day of the ride and preferences like car availability and smoking.	X	X	X	X	X
Create account: Allow users to create and manage their accounts.	X	X	X	X	X
Accept booking request: Allow drivers to accept or decline booking requests.		X			
Payment System		X		X	
Discounts and Vouchers		X			
Exchanges and Refunds: Manage trip reservation exchanges and refunds.		X			
Rating System: Facilitate rating and reviewing of drivers and passengers after trips.					
Carpooling Calendar: Your employees can use Carpooling Calendar to update their daily schedule, change rides and inform other carpoolers.			X		X
Web Admin Panel			X		X

Attributes/Requirements	**Comovee**	**Blablacar**	**Poola**	**Uber for Business**	**Twogo**
Employee Enrollment: Employees must be invited to join the company's Uber account.				X	X
Commuter Options: Door-to-door commute, First and last mile, etc.				X	X
Budget Allocation: Define how much of the ride cost the company will cover, including the ability to limit ride times and vehicle types.				X	
Safety Measures: Riders can easily access key safety information, get emergency assistance, etc.				X	X
GDPR Compliance: GDPR compliant, with advanced privacy controls.	X	X	X	X	X
Reporting & Analytics: Provides detailed reporting on user adoption, carpooling activities, and environmental impact.					X
Customizable Dashboard: Admins can monitor reports on usage, user engagement, and environmental savings.					X
Cross-Platform Support (Web/Mobile): Support web and mobile access for users across different platforms.	X	X		X	X
Real-Time Notifications: Send alerts to users about trip status, updates, or cancellations.	X	X	X	X	X

References

[1] Blablacar. *Blablacar*. Accessed: 2024-12-29. 2024. URL: https://www.blablacar.com/.

[2] Object Management Group. *Business Process Model and Notation (BPMN) 2.0*. Accessed: 2024-12-29. 2011. URL: https://www.omg.org/spec/BPMN/2.0.

[3] Microsoft. *Microsoft Teams API Documentation*. Accessed: 2024-12-29. 2024. URL: https://learn.microsoft.com/en-us/microsoftteams/platform.

[4] Poola. *Poola*. Accessed: 2024-12-29. 2024. URL: https://poola.app/corporate-carpooling-new/.

[5] TwoGo. *TwoGo*. Accessed: 2024-12-29. 2024. URL: https://www.twogo.com/en/.

[6] European Union. *General Data Protection Regulation (GDPR)*. Accessed: 2024-12-29. 2018. URL: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32016R0679.