Co-funded by
the European Union

Ref. Ares(2024)4709310 - 30/06/2024

Project Acronym:	GREEN-LOG
Project Number:	101069892 (HORIZON-CL5-2021-D6-01 - Innovation action)
Project Full Title:	Cooperative and Interconnected Green delivery solutions towards an era of optimized zero emission last-mile Logistics

GREEN-LOG

DELIVERABLE

D3.1 – Real-time & Connected services V1

Dissemination level	PU - Public
Type of Document	DEM – Demonstrator, pilot, prototype
Contractual date of delivery	30/06/2024
Deliverable Leader	ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS (AUEB)
Status - version, date	Final – v1.0, 30-06-2024
WP / Task responsible	WP3 / Task 3.1 (FI), Task 3.2 (MOSAIC), Task 3.3 (FRIC), Task 3.4 (AUEB)
Keywords	Logistics-as-a-Service, Marketplace, Dynamic pricing, Nudging, On-line monitoring, Bring-Your-Own-Device, Nowcasting, Forecasting, Last-mile delivery, Dynamic routing, Multimodal transportation

Executive Summary

This deliverable reports on the services which facilitate (near-)real-time operations of the GREEN-LOG platform, in the context of work package 3 (Tasks 3.1-3.4). Following the description of the use-cases of the involved Living Labs and the problems which these bring forward, the deliverable proceeds with the presentation of the technical components of the platform, and their applicability to the use-cases. The major contribution of the deliverable is the consolidation of the services into a Logistic-as-a-Service (LaaS) Marketplace, which holistically addresses to the requirements of the stakeholders of urban logistics operations (Logistics Service Providers, local shop owners, local authorities); nonetheless, the involved components can also be applied as stand-alone services to cover the demands of the GREEN-LOG platform.

For each service, the deliverable describes the methodological framework, the technologies and the algorithms which contribute to its functionality. Also, the deliverable enumerates the functional requirements of the services, since these will facilitate their integration into the platform, which will be addressed by the upcoming deliverables of the project. For the LaaS Marketplace, the deliverable focuses on the presentation of the flow of activities, to clarify the contribution of each subcomponent to the outcome of the service. In addition, given the defined use-cases, the deliverable attempts to provide a clear understanding of the applicability of each service. To achieve this, the presentation is enhanced with indicative screens and mock-ups of the services, implementations on synthetic datasets which simulate the expected real conditions in the Living Labs, and examples of the expected input and output.

The outcome of this deliverable is expected to be evaluated by the stakeholders of the Living Labs, so that they understand the utility of the provided services and their contributions to the solution of the various operational problems. It is also expected to be connected with the Deliverable 3.2 (to be submitted in M21), which will examine the integration of the described services into a Digital Control Tower.

Deliverable Leader (organisation)	ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS (AUEB)
Contributors (organisations)	FRONTIER INNOVATIONS (FI), MOSAIC FACTOR (MOSAIC), FRONTIER KENTRO KAINOTOMIAS (FRIC), MOBY X SOFTWARE (MOBY), FACTUAL CONSULTING (FACTUAL)
Reviewers (organisations)	Ferrocarrils de la Generalitat de Catalunya (FGC), UNIVERSITY OF WOLVERHAMPTON (UoW)
Approved by (organisation)	NETCOMPANY-INTRASOFT (INTRA)

Document History			
Version	**Date**	**Contributor(s)**	**Description**
0.1	01-03-2024	AUEB	First draft of ToC
0.2	03-05-2024	FI, AUEB	Contributions for T3.1, T3.4
0.3	25-05-2024	FI, MOBY, FACTUAL, MOSAIC, FRIC, AUEB	Consolidated contributions for all tasks
0.4	08-06-2024	FI, MOBY, FACTUAL, MOSAIC, FRIC, AUEB	Internal Review (IR) version
0.5	23-06-2024	FI, MOBY, FACTUAL, MOSAIC, FRIC, AUEB	Consolidation of IR comments
1.0	30-06-2024	INTRA	Final QA'ed version

Table of Contents

Table of Figures

List of Tables

Definitions, Acronyms and Abbreviations

Acronym/Abbreviation	Title
LL	Living Lab
BYOD	Bring-Your-Own-Device
WP	Work Package
LSO	Local Shop Owner
LSP	Logistics Service Provider
LaaS	Logistic-as-a-Service
DCT	Digital Control Tower
UI	User Interface
JWT	JSON Web Token
RBAC	Role-Based Access Control
BE	Backend
SDK	Software Development Kit
NFE	Nowcasting and Forecasting Engine
ETL	Extracting, transforming, and loading
LSTM	Long Short-Term Memory
GBM	Gradient Boosting Machines
MCC	Micro Consolidation Centre
MDH	Micro Delivery Hub
MTVRPTW	Multi-Trip Vehicle Routing Problem with Time Windows
TSPTW	Travelling Salesman Problem with Time Windows

1. Introduction

1.1 Document scope

A major challenge of last-mile logistics operations is the management of real-time conditions. Such conditions may imply issues which (are expected to) disrupt the planned operations of Logistics Service Providers (LSPs), and the road transport efficiency, which concern the LSPs, the recipients of their services and the local authorities. It is noticed that the optimal approach to resolve such issues may be conflicting (e.g., a faster vehicle may imply more efficient last-mile operations for the LSPs, but additional CO2 emissions, which concern the local authorities); however, it is possible that opportunities for commonly accepted solutions will rise (e.g., the reduction of load factor of vehicles will reduce the operational costs of LSPs and the energy consumption/CO2 emissions, which is a major goal of local authorities). To respond to such conflicts and synergies in a dynamic manner, work package 3 (WP3) deploys a set of services, which consider the preferences of each actor to design globally accepted solutions.

The scope of this document is the description of the first release of the services which are designed in the context of Tasks 3.1, 3.2, 3.3 and 3.4 of the GREEN-LOG project. Beyond the description of the applied methodology and the functional requirements for each service, the deliverable aims to define the applicability of each service on the use-cases which have been brought forward in the project (as outlined in D1.2), by demonstrating a first round of implementation and mock-ups. The main objective achieved through this deliverable (as accompanying documentation to the developed services) is to provide the initial version of means that enable the management and optimization of last-mile delivery and road transport efficiency in real-time, using dynamic and interconnected services. To accomplish this, the presented services will combine historical and real-time data to react pre-emptively or dynamically to disruptions in last-mile delivery operations.

1.2 Document structure

After an introductory part in Section 1, the deliverable presents the services which are designed in the context of tasks 3.1-3.4, following the order of the work plan: Section 2 describes the LaaS Marketplace, section 3 presents the Bring-Your-Own-Device service, section 4 elaborates on the Nowcasting and Forecasting Engine, and section 5 presents the Dynamic (re)routing service.

Each section focuses on the presentation of the functional requirements of the respective service, the description of the methodological framework, and the connection of the discussed service with the use-cases of the Living Labs. The LaaS Marketplace, as a composite service, also focuses on the flow of activities and the inter-connection of the subcomponents/services which contribute to its functionality.

Finally, the conclusions are presented as the final section of this document.

1.3 Overview of Real time and connected Services

The scope of the WP3 “Real Time and connected Services” focuses on managing and optimizing last-mile delivery and road transport efficiency in real or near-real time through dynamic and interconnected services and interfaces relying on information collection and sharing across the city logistics value chain.

The objectives within this Work Package are as follows:

- Designing and implementing the GREEN-LOG last-mile logistics marketplace, embodying the Logistics-as-a-Service paradigm.
- Creating open and modular Bring-Your-Own-Device (BYOD) solutions for scalable and cost-effective online asset monitoring and tracking.
- Offering nowcasting and forecasting services to estimate transport demand and supply in the near-present and short-term.
- Developing a toolkit for reactively routing transport requests to optimize various criteria (such as total transport and delay costs, travel time, carbon emissions, vehicle loading factor, etc.), often in conflict, such as carbon emissions versus travel time.

D3.1 describes the WP3 conceptual framework for connecting the supply and demand side through an innovative LaaS Marketplace including a combination of services targeting to minimize the operational cost from the supply side and simultaneously giving capabilities of handling the demand to the end-user by the perspective of providing more information (as an outcome of WP3). The framework aims to address the requirements of the GREEN-LOG use-cases and is shown in Figure 1. It consists of four main layers offering different functionalities. These layers correspond to WP3 tasks and are derive from the advancement of the associated applications and services detailed in this deliverable (T3.1-T3.4). The forthcoming deliverable D3.2, will outline the progress of service development related to task 3.5 (DCT). Additionally, Figure 1 illustrates the interactions of the framework with the GREEN-LOG services created in WP2.

The **User Management**, which is presented in the lower layer, delineates the categories of users within the GREEN-LOG WP3 solution. The primary users consist of LSPs, LSOs, consumers, and local authorities. Pertinent details for each user profile, such as preferred delivery times and personal information like favored addresses for pickups or drop-offs, are recorded. Emphasis is placed on gathering information specifically concerning the profiles of LSPs, given that they will be visible to the demand side via the Marketplace and will significantly impact the overall functionality of services developed in the whole WP. The administrative sector encompasses the local authorities responsible for establishing transportation regulations aimed at fostering a more sustainable city logistic environment, encouraging residents to opt for greener alternatives (will be outlined in detail in upcoming D3.2).

The **Operational Services Layer** offers functionalities that facilitate crucial services that contribute to the overall operation of the LaaS Marketplace as a complete system.

The interaction among these services, such as location tracking, is presented through LaaS dashboards to end-users, with the goal of enhancing awareness regarding transportation. These services support LSPs in their everyday operational processes by taking into account the results of demand prediction and network conditions (traffic), which enables dynamic rerouting. This aids in reducing operational costs and minimizing the distance traveled as efficiently as possible.

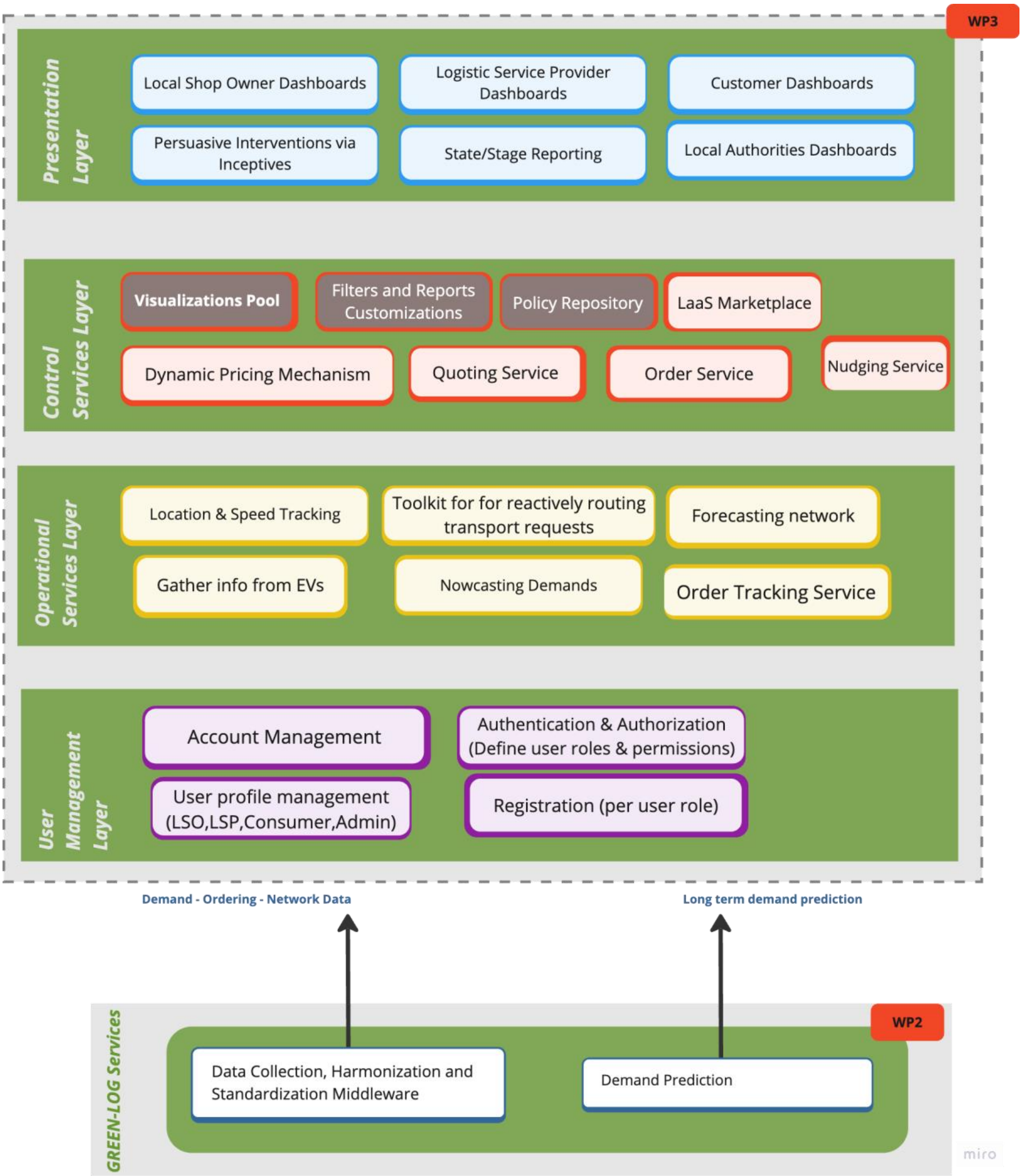


Figure 1 The GREEN-LOG WP3 System Reference Architecture

The **Control Services** offers functionalities targeted at the primary stakeholders of GREEN-LOG project aims to connect, specifically the supply side (LSPs) and demand side (consumers of supply) through the LaaS Marketplace under the oversight of the admins (local authorities). The interaction of these services, along with the operational services, improves the overall system for all categories of end-users. Particularly, the LaaS Marketplace will function as a coordinator not only for the services within this layer, such as Nudging service interacting with the Dynamic Pricing Mechanism resulting in Quoting and Order Service, but also for the Operational Services, particularly those that lower operational expenses and boost transportation efficiency. The Visualization pool in collaboration with the filters and reports customizations services, fundamental features of the Digital Control Tower[1], will provide local authorities with control over city logistics and insights to make more informed decisions leading to a more Sustainable City environment through the Policy Repository.

The outcomes of the LaaS Marketplace in conjunction with reports are displayed to users through the **Presentation Layer**.

The Layers that bridge the supply and demand together with the operational services will be presented and examined in the subsequent chapters, furnishing the reader with a deeper comprehension of how these functionalities connect the supply and demand sides.

[1] The functionalities, approach employed, and integration methodology constitute components of deliverable 3.2.

2. Logistic as a Service Marketplace

2.1 Introduction

The concept of Logistics as a Service (LaaS) revolves around a unified platform that facilitates seamless interaction and service provision between different stakeholders. These include Logistics Service Providers (LSPs) and end-users, namely Local Shop Owners (LSO) and consumers. LaaS aims to optimize the logistics and supply chain management processes through innovative, user-friendly services. In the GREEN-LOG project, the concept of LaaS is tailored to link Logistics Service Providers (LSPs) and consumers in a more sustainable urban environment by intervening in their interaction among the process of last-mile deliveries.

The implementation of the concept of LaaS presents a promising approach to modernize logistics systems. However, its adoption also highlights some challenges posed by traditional models. By examining in brief these challenges, we can better understand the necessity and potential benefits of transitioning to LaaS solutions.

Traditional logistics models are facing increasing pressure due to several challenges. One major hurdle is the lack of real-time visibility and data availability throughout the supply chain. The above as well as the fact that legacy systems often operate in silos, is making it difficult to track inventory movement, identify potential delays, or optimize routes. This can lead to inefficiencies, wasted resources, and ultimately, dissatisfied customers [REF-01].

Furthermore, the rise of e-commerce and the raised need for on-demand delivery services is placing new demands and pressure on the logistics providers. Consumers expect faster deliveries, more flexible options, and greater transparency. Traditional models are often designed to satisfy bulk shipments and long lead times, struggle to adapt to these changing consumer preferences. This necessitates a shift towards a more dynamic and data-driven approach to logistics, where real-time information and connected services are paramount [REF-02].

Under this prism of continuous requests for new services, LaaS offers a compelling solution to the challenges nesting in traditional models. By leveraging cloud-based platforms and real-time data integration, LaaS platforms create a more transparent and connected supply chain. With the use of LaaS, Businesses gain access to a wider network of logistics providers, allowing them to compare rates, optimize routes, track shipments in real-time and become more scalable allowing logistics companies to expand or shrink their infrastructure according to business needs. This not only reduces inefficiencies and hinders collaborations to become more competitive and less energy consuming but also empowers businesses to meet the growing demand for faster and more flexible delivery options. Additionally, LaaS platforms can seamlessly integrate with e-commerce platforms, streamlining processes, order fulfilment and providing customers with improved visibility into their deliveries enhancing collaboration. The LaaS platforms offers innovative strategies to tackle the challenges of urban logistics. However, it is crucial to examine how existing solutions have approached similar issues and identify any remaining research gaps.

To gain a more comprehensive understanding of the operational aspects of the existing LaaS system and how they can address the challenges posed by conventional models, it is beneficial to examine the primary functions through illustrative instances that are currently operational and documented in literature. Platforms that are available and under the LaaS concept currently exist and are called "*Freight Matching Platforms*". These are mainly meant to connect the shippers with carriers who have available capacity in their warehouses. The main exchange of information is the former to send shipment details and these platforms to respond quotes from multiple carriers.

Moreover, there are platforms that are called "*Full Service LaaS Providers*" and offer a wider range of services beyond just freight matching. These may include options such as warehousing, fulfillment, customs brokerage and other valued-added services, such as:

- Load matching: Matching shipment needs with available carrier capacity.
- Real-time tracking: Tracking shipments in real-time for increased transparency.
- Booking and management: Simplifying the booking and management of logistics services.
- Document management: Streamlining the exchange of documents between shippers and carriers.
- Data analytics: Providing insights into shipping performance and costs.
- Financial services: Offering integrated payment processing and freight financing options.

These value-added services may not only link the shippers and the LSPs but also the consumers with the LSPs depending on the type of business interactions. While existing LaaS Marketplaces offer a valuable service by connecting shippers with LSPs, there are limitations into the overall offerings they launch. These limitations include:

- Lack of connection between Shipper and Consumer.
- Consumers often being obligated to follow the shipper's preferences regarding the delivery process.
- Shippers and LSPs being unfamiliar with or ignoring restrictions that cities (authorities) want to apply for a friendlier (traffic and noise reduction) and greener urban environment.

This is where the GREEN-LOG LaaS Marketplace comes to complement and go a step further on the innovation side. It builds upon the strengths of existing solutions by presenting a more complete competitive and green offering:

- Focus on City Logistics, especially facilitating their transformation into more sustainable and eco-friendly environments.
- Transparency among all parties of a LaaS (shippers, LSPs, consumers, and Local Authorities).
- Dynamic pricing adjustments based on authorities' restrictions.

- Nudging shippers and consumers to choose a more sustainable LSP for both pick-ups and deliveries.

In conclusion, the GREEN-LOG LaaS marketplace landscape offers valuable tools for businesses to manage their logistics needs. However, limitations exist, such as (lack of transparency). This section has explored these limitations, existing solutions and what objectives GREEN-LOG LaaS.

The next sections will delve deeper into GREEN-LOG LaaS Marketplace. We will explore its functionalities based on specific use-cases in detail, the methodology followed to obtain the functional requirements and more specifications about the components that contribute to the whole solution.

2.2 Functional requirements

To develop a robust and user-centric LaaS platform within the GREEN-LOG project, a comprehensive approach to understanding the needs and expectations of key stakeholders was crucial. This sub-chapter outlines the methodology employed to capture these user requirements, which were then utilized to direct the design and development of the GREEN-LOG solution.

User-Centric Methodology:

An Agile framework throughout the process, ensuring iterative developments and continuous stakeholder engagement with interactive validation points has been employed. This collaborative approach between different stakeholders of the project involved a variety of user-centric methodologies:

- **Interactive Sessions between WP1 and WP3**: Interactive sessions were a key activity of WP1, fostering collaboration with stakeholders from each Living Lab (LL). These "Meet-ups", as they called, facilitated initial discussions and brainstorming on GREEN-LOG project and its application to each different eco-system. Through these interactive sessions, we were able to identify promising use cases for the LaaS Marketplace to be employed. Notably, the solution aligns well with the Flanders LL, encompassing three cities (Leuven, Mechelen, Ghent), and presents potential for the Ispra use case. More information can be found in D1.2.
- **Bilateral Meetings and Interviews:** In order to enhance comprehension regarding the distinct requirements and obstacles in every Living Lab (LL), the WP leader (FI) conducted bilateral meetings with key partners alongside the participants of the WP3 (AUEB, FRIC, Factual & MOBY) subsequent to the preliminary discussions..
    - Flanders Use Case: We initiated bilateral discussions with the Local Authorities of Leuven, Mechelen, and Ghent, the main partner group in Belgium for this use case. These meetings focused on exploring how the core functionalities of the LaaS platform, as described in the DoA (Document of Action), could facilitate the creation of more sustainable logistics networks in each city center. The discussions resulted in a

clearer understanding of the specific needs and priorities of each local authority.
  - Ispra Use Case: Following the same approach as the Flanders use case, we conducted bilateral meetings with key partners in Ispra to gain a deeper understanding of their specific requirements and challenges.

- **User Story Creation:** To ensure a clear understanding of how the LaaS Marketplace would function according to the needs of diverse stakeholders in the Flanders LL, we employed User Story creation. User Stories are a methodology that captures functionalities from the user's perspective. This approach proved valuable in the Flanders LL due to its complexity, involving multiple stakeholders with various needs.

- **Service Blueprint (interactive session):** For Ispra's use case, with its focus on a simpler functionality (ordering process) and a more straightforward multimodal transportation scenario, we employed a different approach that began in WP1 and tailored to the needs of the WP3. This activity was to produce in a cooperation way the service blueprint including system interactions among the process.

- **Interactive Physical Workshop:** Following productive bilateral discussions on user needs, we organized an interactive workshop to solidify a common understanding and move forward with the project. This workshop focused on user interaction through the lens of user journeys with main focus on how the User will interact with the LaaS UI. Participants actively participated in co-creating initial user journeys, providing valuable insights into how users would interact with the LaaS platform. Additionally, we presented activity diagrams developed collaboratively with all technical partners contributing to the WP. This visual representation of the system's processes further enhanced understanding for all stakeholders involved. The revised version of them will be discussed in the subsection 2.3.2.

- **Online Validation Workshop**: To ensure the LaaS platform met user needs before full-scale development, we conducted a Validation Workshop. During this online workshop, we presented stakeholders with a high-fidelity mock-up of the MVP (Most Viable Product). This MVP focused on the "Get a Quote" functionality, which allows users to request and receive quotes for deliveries/pick-ups.

  The workshop provided stakeholders with the opportunity to see the MVP in action, including detailed steps for different user types (end-user - LSP) and the anticipated results of using the platform. While some stakeholders expressed interest in additional functionalities, especially in the Flanders LL where LSPs are not part of the consortium, we determined that "Get a Quote" was the most strategic functionality to validate in the initial phase. Focusing on this core functionality allowed us to test the entire LaaS platform, including dynamic pricing and nudging features, and showcase the desired user experience.

Focus on Functional and Non-Functional Requirements:
This comprehensive methodology ensured we captured both functional and non-functional requirements. Functional requirements define the specific actions the LaaS

platform will perform to facilitate Flanders and Ispra use case for stakeholders like Local Shop Owners, Local Authorities and LSPs. Non-functional requirements address usability, performance, security, and other attributes critical to user experience for all stakeholders. By focusing on both aspects, we aim to develop a platform that is not only feature-rich but also user-friendly, efficient, and reliable for the partners contribute to LaaS Marketplace.

2.2.1 User Stories

The approach suggested by WP3 leading team and adopted by all technical partners in collaboration entails developing "*User Stories*" to accurately and concisely document user needs and expectations identified in the previous activities. This method was found to be most effective because of the different backgrounds of stakeholders, emphasizing the importance of a mutual understanding of the proposed solutions and the inclusion of diverse viewpoints on the recognized requirements.

Each user story depicted below (taking as a reference the Flanders LL user story), represents a specific functionality or feature from the perspective of the end-user, whether they are LSO, LSPs, consumers, or local authorities. In this section, we present a series of user stories that highlight the key interactions and desired outcomes for each stakeholder group, ensuring the LaaS platform is both user-centric and capable of addressing the unique challenges of each city of Flanders LL. These user stories serve as the foundation for the subsequent design and development phases, guiding our efforts to create a solution that is both practical and innovative.

In the following table, we detail the user stories created and discussed collaboratively, showing the needs and expectations of Flemish stakeholders per involved users.

Table 1 User Stories Table provided by Flanders Living Lab

Story ID	*As a user*	*I want to*	*So that*	*Priority*	*Comment (e.g status, to be detailed, comes from Story ID)*
US-FL-01	Bike Courier Manager	identify the volume and dimensions of the parcel	I can offer the appropriate solution	Medium	There is need to get data to examine how to forecast the parcels
US-FL-02		identify the type of goods	I can offer the appropriate solution	High	Their customer be able to choose the content of the parcel
US-FL-03		identify no rush deliveries (metric for urgency)	I can organize properly	Medium	Linked to US-FL-09
US-FL-04		have a seamless technical connection between GREEN-LOG solution and my software	I don't have to adjust my own systems	High	To be detailed on technical approach
US-FL-05		collaborate with technical partners to set up the dynamic pricing model	I can use the output during and after the project	Medium	To be detailed
US-FL-06		offer an adequate (dynamic) price	all costs are covered, and I can benefit from a subsidy to make a profit	High	Linked to US-FL-05

US-FL-07		have insight in the dynamic pricing for each of my orders	I can evaluate properly my benefits	High	Linked to US-FL-05
US-FL-08	Local Authorities	promote alternative delivery options to our citizens	our citizens can choose more sustainable delivery options	High	On going
US-FL-09		nudge my citizens and entrepreneurs to organize their logistics outside the school rush hour	so that de school routes are safer	High	Linked to US-FL-03
US-FL-10		impose economic incentives	I can steer the market	Medium	Stopped
US-FL-11		preselect logistic service providers that can access the LaaS	I can have more control about the quality and sustainability of the services	High	To be detailed
US-FL-12		apply our policies directly to every delivery in the city	eventually all deliveries are compliant with our policies	Medium	To be detailed
US-FL-13	Consumer	choose the no rush shipping option	I can have impact on the sustainability of my delivery	Medium	Linked toUS-FL-03
US-FL-14		see in one overview before and during the checkout what my delivery options are	I can consider them before purchasing	High	Linked to US-FL-09
US-FL-15		know which types of vehicles have carried out my sustainable delivery	I can have a better knowledge about the sustainable supply chain	Low	To be detailed
US-FL-16		avoid missed deliveries at home	I don't waste time to pick it up in a parcel shop or postal service point	Medium	To be detailed

The user stories presented above, were extracted after collaborative meetings with representatives from the Flanders LL. The initial version of the file provided by them is presented in the Appendix. It is important to note that this initial version includes requirements that impact other technical partners and offers an overview of their functionalities. Additionally, it provides input on how the interaction of different services within WP3 will occur to meet these requirements.

Moreover, it is worth noting that for the Ispra use-case, an alternative approach was followed. The Blueprints included in D1.2 were used as input, as they showcase the functionalities and system interactions in an understandable way.
In the following subsection we are detailing how user stories were translated into technicalities.

2.2.2 Activity Diagrams

Activity diagrams provide a comprehensive view of the GREEN-LOG LaaS platform's functionalities. They visually represent workflows, stakeholder interactions, and data flows across all components, including the LaaS marketplace (with dynamic pricing and nudges) and other critical functionalities like BYOD and dynamic routing. The following activity diagrams, crafted in accordance with user specifications and refined during a collaborative workshop (GA in Wolverhampton) with a view to provide a comprehensive insight into the operational dynamics of the GREEN-LOG WP3 solution.

This section presents activity diagrams that illustrate the core functionalities and system interactions of the GREEN-LOG solution based on Flanders and Ispra use case. These diagrams offer a transparent view of how the LaaS platform.

2.2.2.1 Activity Diagram of Flanders LL

The activity diagram that is provided in Figure 2 depicts the key stages involved in fulfilling a customer's delivery request through a LSP. It emphasizes the integration of sustainability considerations throughout the process.

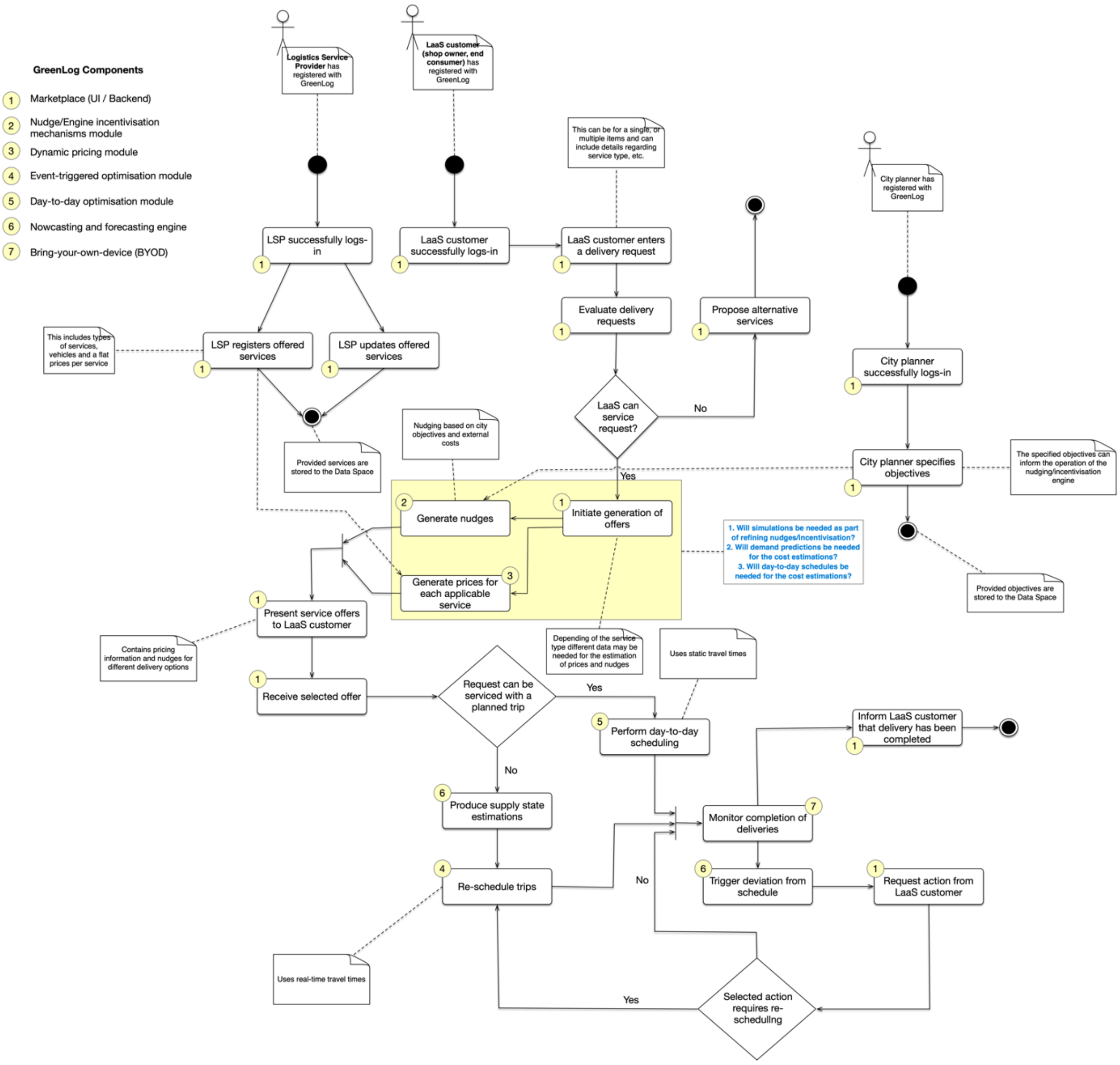


Figure 2 Activity Diagram of Flanders LL- General Approach

This activity diagram is the outcome of an interactive workshop, during which we described the following key stages that outline a typical process that will be followed and the interaction between the different services.

- Customer Request: The process begins with the customer submitting a delivery request.
- LSP Evaluation & Service Options: The LSP assesses the request and proposes alternative service options tailored to the specific needs.
- Spatial Objective Integration: City planning data is incorporated to establish spatial objectives for the delivery, prioritizing factors like minimizing emissions or traffic congestion.
- Sustainability Nudges & Incentives: The LSP leverages a nudge/incentive engine to encourage the customer to select a more sustainable service option, potentially through discounts or eco-friendly options.
- Customer Selection & Scheduling: Upon customer selection of a service, the LSP performs day-to-day scheduling and route optimization, to optimize efficiency.
- Delivery Monitoring & Rescheduling: The LSP proactively monitors delivery progress and can dynamically reschedule trips if necessary to address unforeseen circumstances.
- Delivery Completion & Customer Notification: Once the delivery is complete, the LSP informs the customer, finalizing the process.

The flowchart incorporates decision points to navigate the process based on factors like available resources and chosen service options. Additionally, it highlights key modules that contribute to efficient and sustainable logistics:

- Marketplace: Facilitates the presentation of service options to the customer.
- Nudge/Incentive Engine: Encourages customer choices that align with sustainability goals.
- Dynamic Pricing Module: Potentially used to adjust pricing based on factors like chosen service and environmental impact.

2.2.2.2 Activity Diagram of Ispra LL

The same process was followed for the Ispra use-case. The key activities are depicted in the activity diagrams in Figure 3. Since this specific use-case differs from the previous one, the target of this flowchart is not only to showcase how the actors and services will interact but also to present an alternative solution to enhance the functionalities provided to the users and leverage the operational flow.

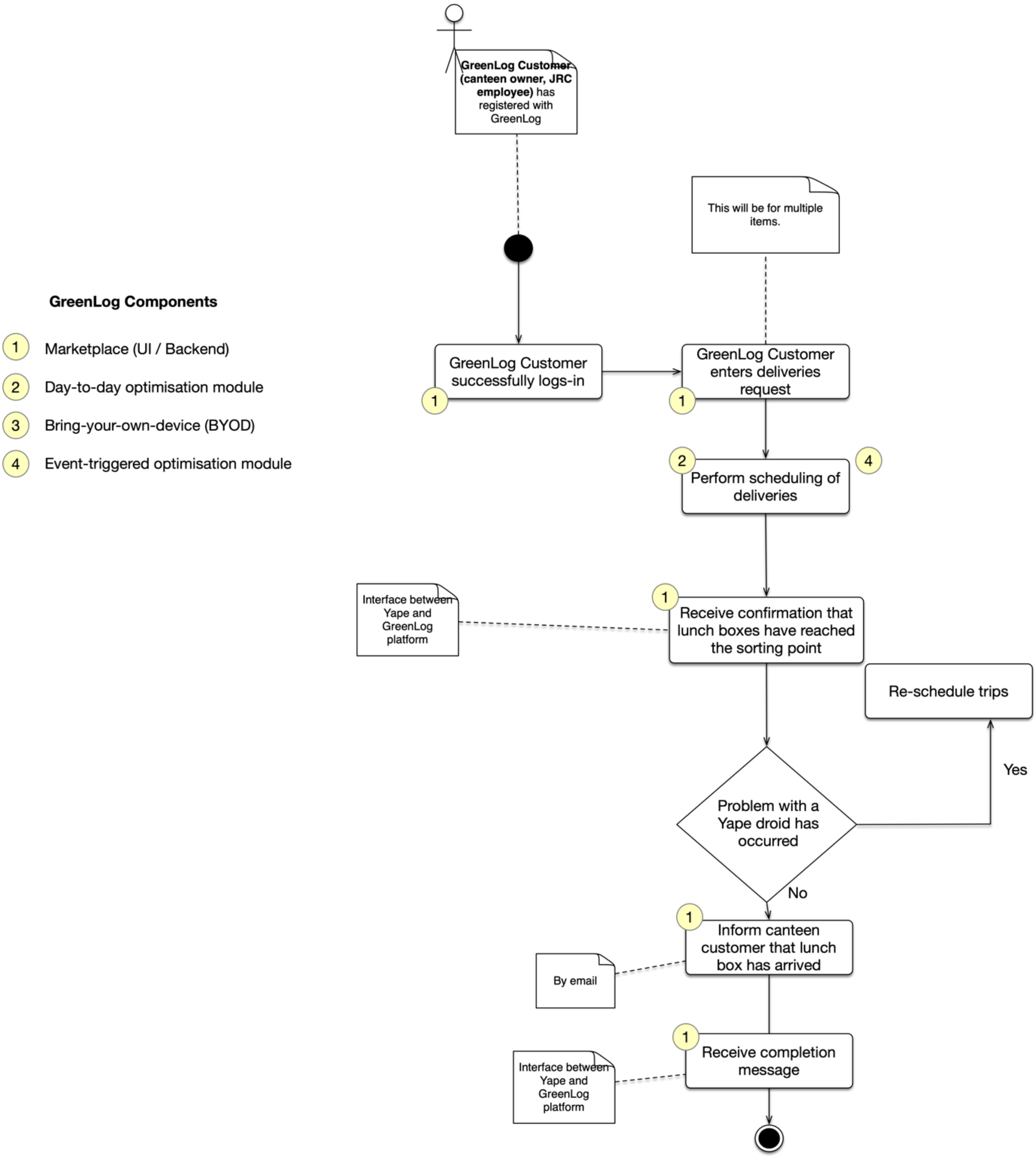


Figure 3 Activity Diagram of Ispra LL

This activity diagram outlines the core stages involved in fulfilling customer delivery requests through GREEN-LOG, where a logistics service provider offering both conventional and autonomous delivery methods. The process flow is briefly described below:

1. Main Process Flow:
   a. **Customer Request:** The process commences with the customer submitting a delivery request, potentially for multiple items.
   b. **Delivery Method Selection:** GREEN-LOG assesses the request and determines the optimal delivery method between the following:

i. **Yape Droid Delivery:** Prioritized for efficiency and potentially lower emissions.
ii. **Conventional Method:** Employed when Yape droids are unavailable or unsuitable for the delivery.

2. Yape Droid Delivery Path**:**
   a. **Existing Delivery Match:** GREEN-LOG checks if existing planned Yape droid deliveries can accommodate the new request.
   b. **Match Found:** The lunch box is integrated into the existing delivery route, maximizing efficiency.
   c. **No Match or Yape Droid Issue:** GREEN-LOG informs the customer via email about potential delays or alternative arrangements.

3. Conventional Delivery Path:
   a. **Platform Communication:** GREEN-LOG interacts with an external platform, presumably a delivery partner network, to schedule the lunch box deliveries.
   b. **Delivery Confirmation:** GREEN-LOG receives confirmation upon the lunch boxes reaching a designated sorting point within the conventional delivery network.
   c. **Customer Notification:** Irrespective of the chosen delivery method, GREEN-LOG informs the customer once their lunch box has arrived at its destination.

The activity diagram prioritizes Yape droid utilization for its efficiency and potential environmental benefits. A decision point exists to address potential Yape droid unavailability or malfunction, ensuring customer communication.

2.2.3 Functional Requirements of Subcomponents

During the workshops organized with the relevant stakeholders, we also identified the functional requirements of the subcomponents of the LaaS marketplace, including the dynamic pricing component and the nudging engine.

2.2.3.1 Dynamic Pricing Role in Specific Use-Cases

Ghent use-case: Ghent is looking into incorporating dynamic pricing to reflect the real and social costs of logistics operations, potentially influencing the choice of logistics providers based on emissions, efficiency, and compliance with city policies. This approach aims to align logistics operations with the city's sustainability goals by adjusting prices to reflect environmental and social impacts.

Mechelen Use-Case: Dynamic pricing in Mechelen can make the consumer part of the logistics equation by offering different delivery options at varying costs. More sustainable options like delayed delivery with a cargo bike on a fixed route may be cheaper than immediate delivery options. There is a strong emphasis on transparency of societal costs, which dynamic pricing could highlight. By doing so, the project seeks to incentivize consumers to choose greener delivery methods.

Leuven Use-Case: In Leuven, dynamic pricing was initially considered but later substituted with fixed pricing due to feedback from stakeholders. Instead of real-time dynamic pricing, the project may use static prices for different sustainable logistics options to encourage sustainable behavior, supplemented by financial and non-financial incentives. This approach aims to simplify the pricing model while still promoting environmentally friendly delivery choices.

Table 2: Indicative table for dynamic pricing

City	*Parameters*	*Data Requirements from LaaS*
Ghent	- Base Rate - Distance - Operational Cost per km - Emission Cost - Efficiency Discount - Time Window Multiplier	- Real-time and historical data on delivery distances - Operational costs for different vehicle types - Emission data for each vehicle type. - Delivery time preferences (peak/off-peak) - Data on delivery bundling opportunities and drop-off densities
Mechelen	- Base Rate - Distance - Operational Cost per km - Societal Cost - Sustainability Multiplier	- Real-time and historical data on delivery distances - Operational costs for different vehicle types - Societal cost data (pollution, noise, congestion) - Data on sustainability options (cargo bike availability, parcel lockers, consolidation opportunities)
Leuven	- Base Rate - Distance - Operational Cost per km - Sustainability Discount	- Real-time and historical data on delivery distances - Operational costs for different vehicle types - Data on sustainable delivery options (parcel lockers, consolidated rounds, zero-emission vehicles)

Note that if the operational cost is hard to get or access to it is restricted, the dynamic pricing module can work on existing price of delivery and then adjust it based on the rest of the parameters.

2.2.3.2 Nudging Engine incentives

For the nudging engine three measures were defined collaboratively by the three cities of Flanders Living Lab to be analysed by this component taking into account their priorities as a city authority. It is essential to highlight that while these measures have been outlined collectively, their specificities will be further refined to accommodate the diverse needs of each city. This customization process will be facilitated through the Digital Control Tower (DCT), allowing for adjustments based on local context, infrastructure, and community needs.

- Mode of Transport Incentives:

The nudging tool will incentivize deliveries made using specific modes of transport, aiming to promote sustainability and efficiency in each city's logistics operations. The implementation details of these incentives will be tailored to the transportation infrastructure and preferences of each city.

- Off-Peak Delivery Incentives:

In each city, the nudging tool will encourage deliveries during off-peak hours to alleviate traffic congestion. The specifics of these incentives will be adjusted to fit the unique traffic patterns and peak hours of each city, optimizing delivery schedules accordingly.

- School Zone Delivery Incentives:

Delivery incentives targeting school areas outside of peak school hours will be implemented with consideration for the safety and convenience of each city's school communities. The nuances of these incentives will be adapted to accommodate the varying school schedules and locations across the three cities.

By tailoring the implementation of these measures to the specific characteristics and priorities of each city, the nudging tool aims to maximize its effectiveness in optimizing delivery logistics while addressing local challenges and opportunities.

2.3 Description of components

The GREEN-LOG LaaS Marketplace is an innovative platform designed to streamline logistics and shipping processes for local shop owners and logistics service providers. This chapter serves as a comprehensive guide to understanding, using, and managing the application’s various features and functionalities. To provide clear insight into how the different services contribute to the LaaS Marketplace and its distinct functionalities, we have developed the following Information Architecture, Figure 4. This architecture details each component, explaining its role and how it integrates with other parts of the system to enhance operational efficiency and user experience.

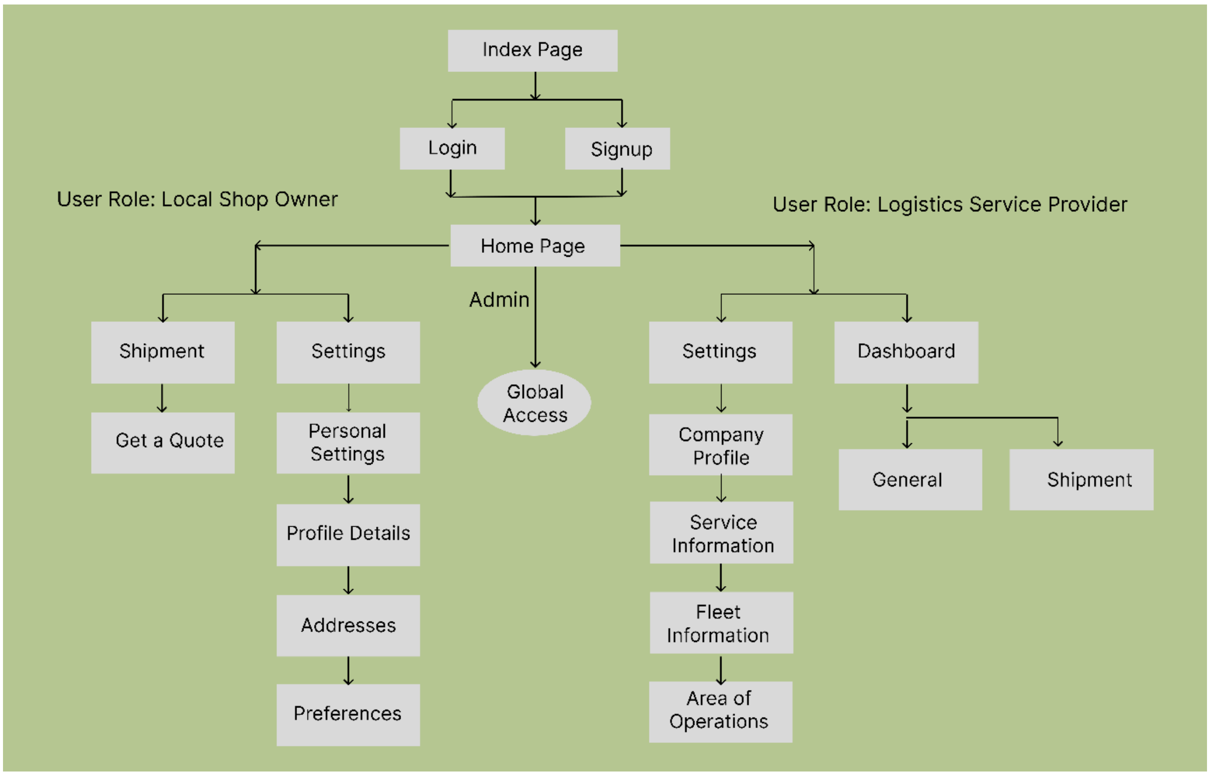


Figure 4 Information Architectureof LaaS Frontend

This is the Schema of the User Story Roadmap. It depicts the way the users can navigate through the platform and the abilities they can access. Each of the characteristics will be described in detail in the following sections.

2.3.1 LaaS Marketplace Frontend

2.3.1.1 Index Page Overview: Laas Marketplace

The Index Page serves as the primary landing page for users accessing the LaaS Marketplace. This page is structured to provide both functional and informative elements as outlined below:

1) **Hero Section:** This prominent area displays essential information about the Marketplace, succinctly informing users of the platform's purpose and scope.
2) **Partners Section:** This section showcases the organizations collaborating with the Marketplace, highlighting their logos and brief descriptions to emphasize community and support.
3) **Navigation Bar:** Located at the top of the page, the navigation bar includes the Marketplace logo, ensuring brand visibility. It also features a button that directs users to the login/sign-up page, facilitating easy access to user accounts.
4) **Footer:** The footer provides additional navigational aids and links to pertinent information.
5) **Translation Feature:** A key functionality of Laas Marketplace is the ability to translate the entire application's content, enhancing accessibility and user engagement across diverse linguistic demographics.

Upon selecting the designated button, users are directed to the initial step of the Login/Sign-Up menu, which is presented within a modal. This method not only streamlines user interaction by maintaining the underlying page visibility but also enhances the user interface (UI) experience.

The modal incorporates dynamic URL parameters to facilitate tracking and state management during the login or registration process. Displayed prominently within this interface are options allowing users to register or log in as either a Logistics Service Provider (LSP) or a Local Shop Owner. This dual functionality is designed to accommodate the distinct needs of our diverse user base.

2.3.1.2 Login Form Functionality

Users are required to enter their valid credentials, which trigger a conditional redirection to the main page of the application, based on their server-side authorization. Authentication and data integrity are managed via JSON Web Token (JWT) technology, ensuring secure and efficient data transfer from the server to the frontend application.

In instances of incorrect credential input, access to the application is denied, maintaining security and user integrity. For enhanced user navigation, the modal provides a 'Back' button, allowing users to revert to the initial state of the modal. Additionally, users can opt to close the modal by clicking the 'Exit' button located at the top-right corner.

2.3.1.3 Registration Process and User Role Selection

Upon opting to register via the step-1 Menu, the modal dynamically transforms into a signup form. This form mandates the input of various pieces of user data, which are essential for advancing to subsequent stages of the application. During this phase, users also select their desired role—either as a Local Shop Owner or a LSP—from a dropdown list.

User navigation options within the modal include a 'Back' button, enabling a return to the initial menu, and an 'Exit' button to close the modal entirely, similar to the functionalities described in Login functionality.

2.3.1.4 Access Control and Navigation Policy

Should a user attempt to access the application directly via URL without proper authentication, access is limited to the Index Page or the Login/Sign Up Menu. Any attempt to navigate beyond these points without logging in results in redirection to the Login Modal State. This routing policy is enforced client-side to enhance security; nevertheless, server-side measures are in place to restrict unauthorized actions, ensuring robust security even if initial client-side controls are bypassed.

2.3.1.5 Role-Based Access Control (RBAC)

Following the login or signup phase, the application implements Role-Based Access Control (RBAC). This system customizes the user interface based on the user's designated role, ensuring a tailored experience that aligns with the specific functionalities and privileges appropriate to either Local Shop Owners or LSPs.

2.3.1.6 Features and functionalities of each user

2.3.1.6.1 LSO

Upon successful login, the landing page tailored for Local Shop Owners is presented, as illustrated in Figure 5. This interface provides access to specific sections relevant to their operations:

- Shipment Section: This section, which will be detailed later, allows Local Shop Owners to request, manage or inspect pickups and deliveries.
- Settings Section**:** Here, users can modify various application settings. The options available are conditionally displayed based on the user's preferences and role-specific requirements, ensuring a customized user experience that adapts to the individual's operational needs.

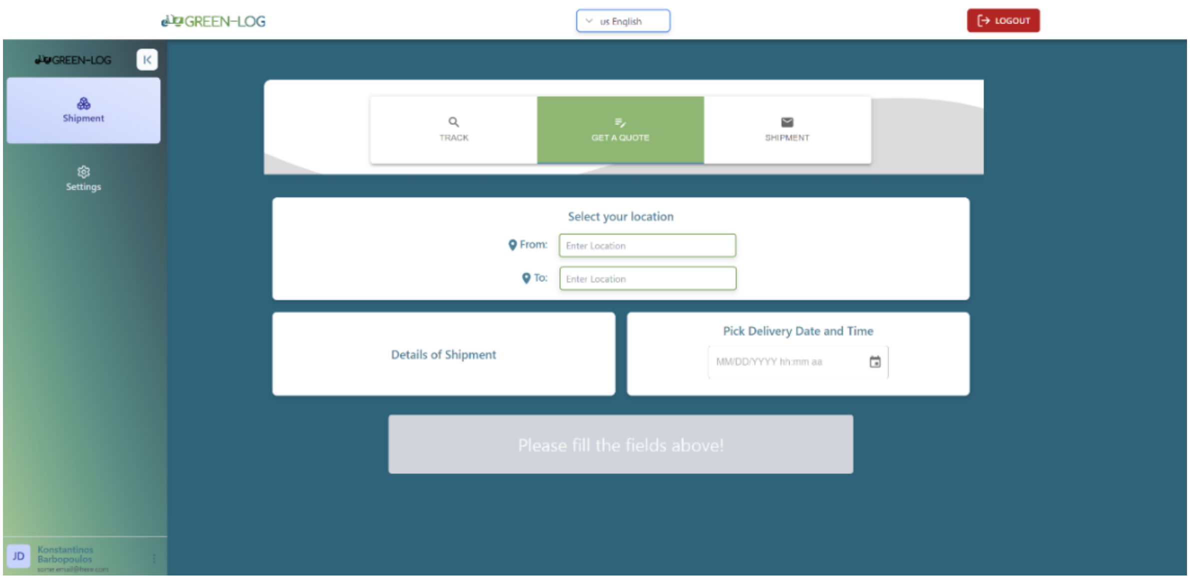


Figure 5 Landing page after login as a Local Shop Owner and "Get a Quote Service"

"Get a Quote" Service Overview

Local Shop Owners utilizing the Laas Marketplace have exclusive access to the "Get a Quote" service (Figure 5), a fundamental feature within the Shipment section. This service is tailored to facilitate efficient logistics management by enabling users to solicit competitive delivery offers.
The "Get a Quote" service operates by gathering essential shipment details from the user, which include:

- **Location:** The starting point for the shipment, where the package will be delivered.
- **Details:** Pertinent information about the package, such as its size, weight, and

the nature of its contents.

- **Delivery Date and Time:** The desired schedule for when the package should be delivered.

Comprehensive Review and Submission Process

Upon entering these inputs, the Marketplace provides a comprehensive view of the gathered information for:

- **Review of Information:** The screen displays all previously entered information by the user—validated as correct by the application—for final review. This allows users to verify details and make any necessary adjustments to ensure accuracy before proceeding.
- **Date and Time Picker:** This interface enables users to specify the preferred date and time for the pickup and delivery of the shipment, facilitating precise scheduling.
- **Activation of the 'Get a Quote' Button:** The button transforms in appearance once all required inputs (Location, Shipment Details, and Delivery Date and Time) are correctly filled. This visual change indicates that the user can proceed to request quotes.

API Interaction and Quote Retrieval

Upon clicking the 'Get a Quote' button, an API call is initiated. This involves the frontend of the Laas Marketplace collating all entered data into a structured format and sending it to the backend API. The backend then re-evaluates the provided information to ensure accuracy.

Visualization of Quote Results

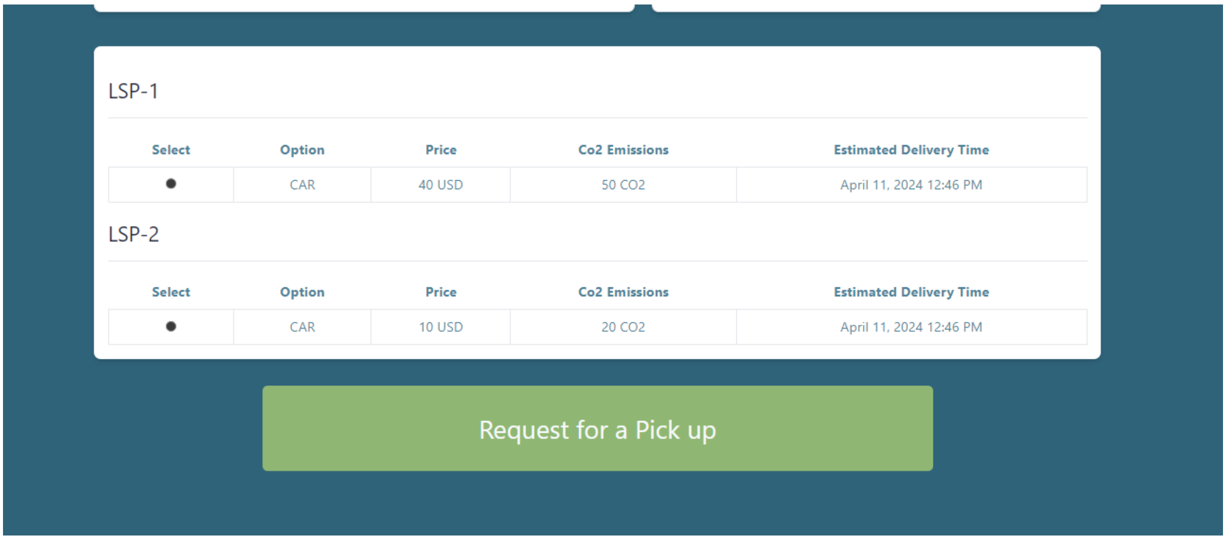


Figure 6 Results from the previous API call for the Get a Quote service

Figure 6 illustrates the outcome of the API call within the "Get a Quote" service, showcasing how the Laas Marketplace backend API presents the response data. This visualization is crucial as it provides users with clear and actionable information regarding different transportation options for their package.

Steps Following the Visualization:

1) **Selection of Transportation Option:** Users are shown various offers, each detailing aspects like cost, delivery time, and carrier details. This enables them to make informed decisions on how to proceed with the shipment.
2) **Order Placement:** After selecting a preferred transportation method, the user finalizes the process by placing an order. This action triggers another interaction with the backend, sending the detailed order information for processing.

Settings Interface for Local Shop Owners

The settings interface for Local Shop Owners in the Laas Marketplace is specifically designed to cater to their business needs. This interface is segmented into several sections, each dedicated to different aspects of the user's profile and preferences:

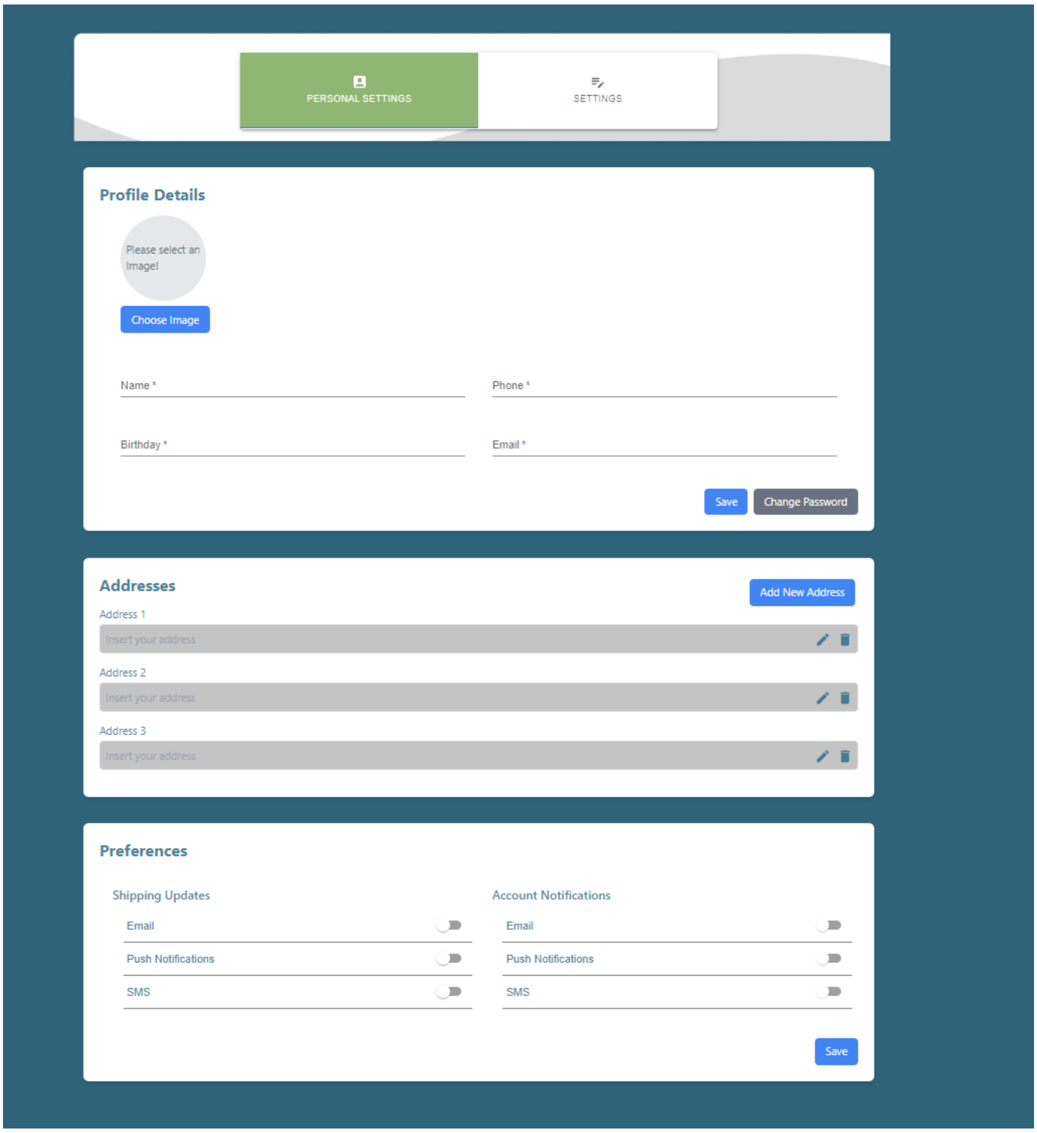


Figure 7 Settings Interface for Local Shop Owners

Profile Details:

- **Image Upload:** Users can upload or update their profile image, adding a personal touch to their account.
- **Name**, **Phone**, **Birthday**, and **Email**: Essential personal details that can be updated to keep the user's profile current and accurate.

Addresses Section:

- **Default Shop Address:** This critical setting allows users to define or update the primary address where their Local Shop is located, ensuring accurate service and delivery mapping.

Communication Preferences:

- **Shipping Updates:** Users can opt-in to receive notifications related to the status of shipments, keeping them informed of all logistics movements.
- **Account Notifications:** Preferences can be set for receiving updates regarding account changes and other notifications, enabling users to stay informed about important account activities.

2.3.1.6.2 LSP

The interface designed for Logistics Service Providers (LSPs) in the Laas Marketplace is distinctively tailored to meet the unique needs of logistics operations. This interface offers a different array of functionalities and settings, enabling LSPs to efficiently manage and adjust their service offerings and operational preferences.

Landing Page-Dashboard Interface

Figure 8 Landing page after login for an LSP

Following the login, Logistics Service Providers (LSPs) are directed to a bespoke landing page, depicted in Figure 8. This page is designed to facilitate efficient navigation and management of their logistical tasks, featuring the following key sections:

- **Dashboard Section:** This central area provides LSPs with a comprehensive overview of their operations, including real-time updates and analytics. Details of this section will be elaborated upon subsequently.
- **Settings Section:** Similar to other user roles, the Settings section for LSPs allows for the customization of various application parameters. The availability and type of settings are conditionally displayed, tailored to meet the specific needs and preferences of LSPs.

Dashboard Interface for Logistics Service Providers

The Dashboard Interface is a crucial component of the LSP’s user experience within the Laas Marketplace, designed to provide comprehensive insights into their logistical operations. This interface is tailored to offer both general and specific performance metrics, facilitated through visually informative charts and graphs. Key Features of the Dashboard Interface Include:

Performance Charts:

- **General Performance:** These charts display overarching metrics that help LSPs gauge their overall operational efficiency and effectiveness.
- **Individual Performance:** Focused metrics allow for monitoring of specific aspects of services, such as delivery times, customer satisfaction, and operational cost efficiency.

History Navigation Menu:

- **Scheduled Shipments:** Listings of all upcoming deliveries scheduled by the LSP.
- **On Delivery:** Real-time data on shipments currently in transit, providing updates on progress and any potential delays.
- **Delivered**: A historical archive of completed deliveries, useful for performance review and customer follow-up.

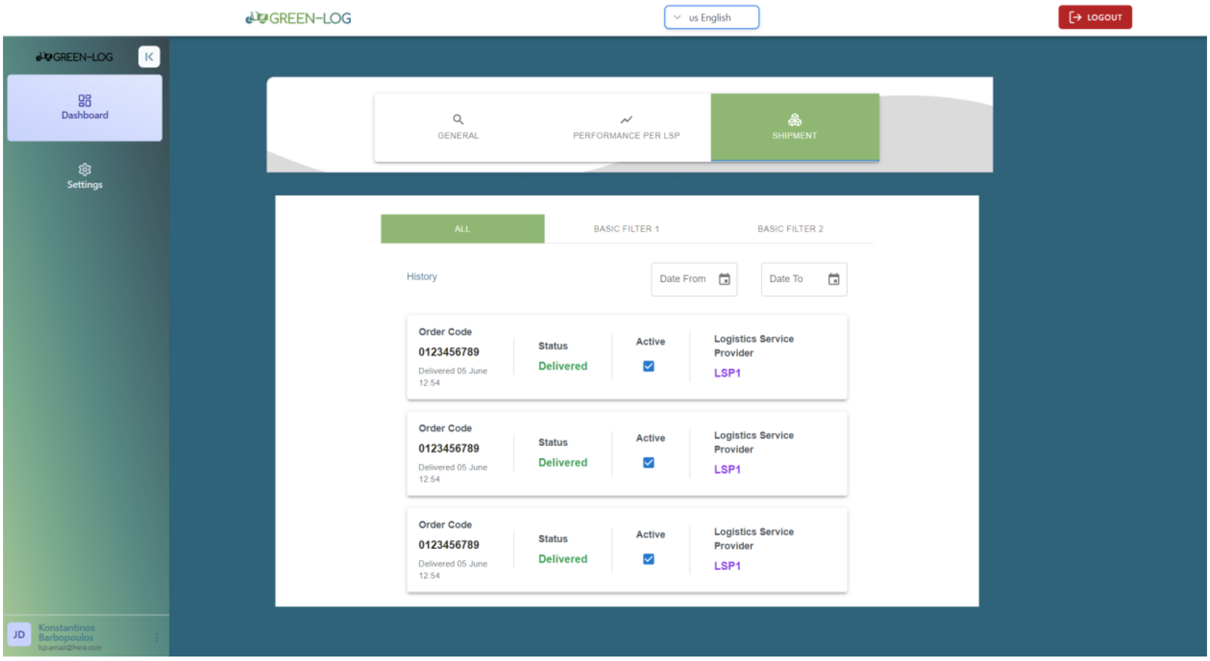


Figure 9 Shipment Data of LSP

Settings Interface for Logistics Service Provider

The settings interface for LSPs within the Laas Marketplace is meticulously designed to support the specific requirements of logistics companies, facilitating customization and enhancement of their service offerings.

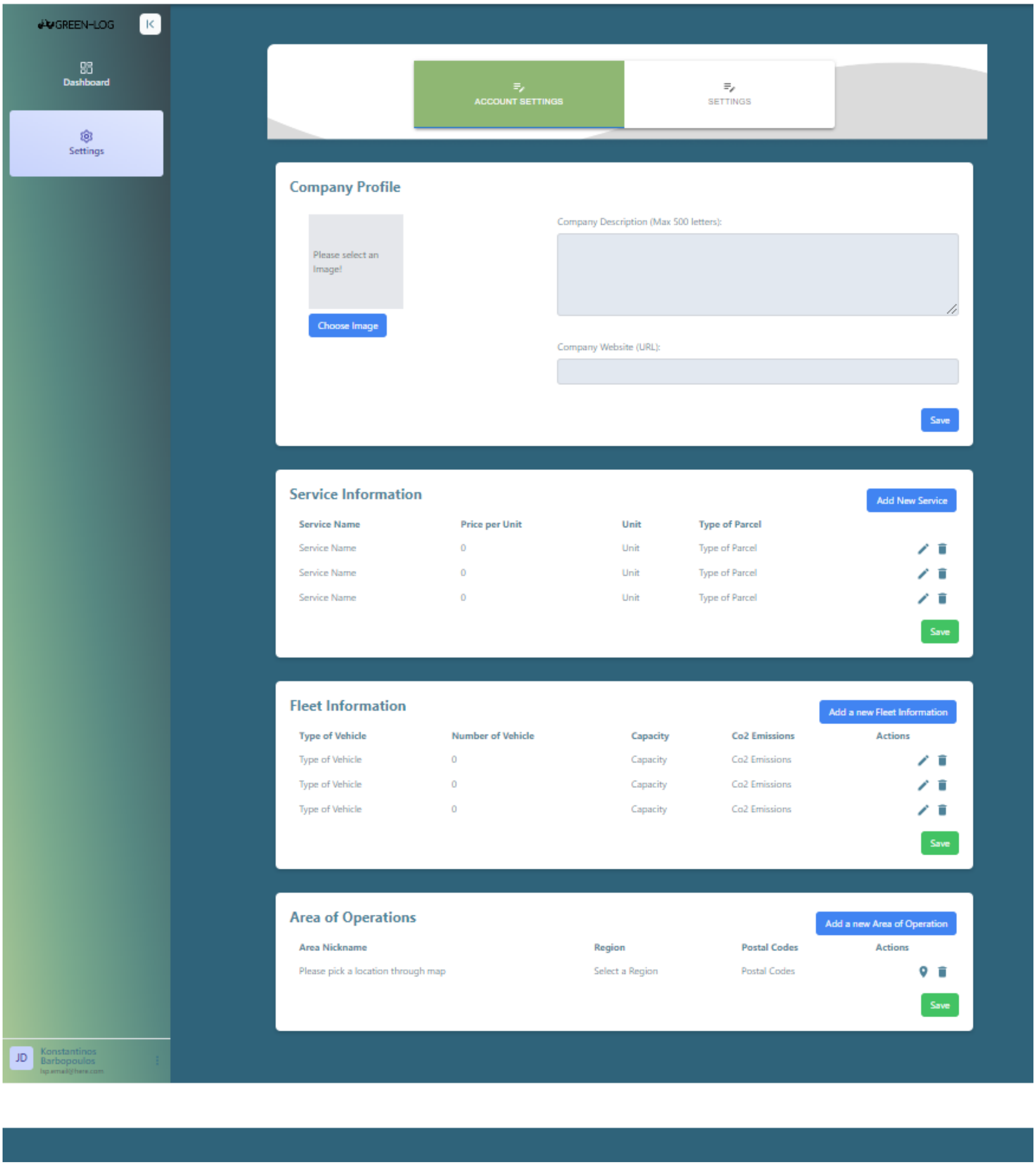


Figure 10 Settings interface for LSPs

Detailed features of the settings interface include: (1) company profile customization, (2) service offerings …. They are

1) Company Profile Customization:

- **Image Upload:** LSPs can upload or update their company logo or an image that best represents their brand, enhancing their profile's visual appeal.
- **Company Description:** A text field allows LSPs to provide a detailed description of their company, outlining their mission, values, and key differentiators.

- **Company URL:** This input allows for the inclusion of the company's website, facilitating direct access for users seeking more information about the LSP.

2) Service Offerings:

Via a dedicated modal, LSPs can add and describe their logistical services, tailoring their profile to match the services they offer.

Service Name:

- **Door to Door:** Standard shipping service receiving the package from the local shop and delivering to door of customers.
- **Door to Pickup Point:** Expedited services receiving the package from the local shop and delivering by using of pickup points around the city
- **Same Day Peak Hours:** Specialized services designed for peak hours, receiving the package from pickup point and delivering to pickup point too.

Unit Measurement: LSPs can specify the unit of measurement relevant to their services, such as kilograms (kg) for weight, liters for volume, and other applicable units. This setting ensures that pricing and service descriptions are aligned with industry standards and customer expectations.

Price per Unit: Enables LSPs to set specific pricing based on the chosen unit of measurement. This feature supports transparent and customizable pricing strategies, allowing LSPs to adjust their rates according to the type of service or client requirements.

Parcel Type: LSPs can define the types of parcels they handle, such as letters, boxes, pallets, and more. This categorization helps streamline operations and allows customers to choose the appropriate service based on the nature of their shipment.

3) Fleet Information

Via a dedicated modal, LSPs can comprehensively manage and update data about their delivery fleet. This section is designed to support the operational needs while aligning with the marketplace's commitment to promoting eco-friendly delivery options.

Type of Vehicle: LSPs can specify the types of vehicles available in their fleet, such as bicycles, scooters, vans, or trucks. This information helps in aligning the service offerings with the delivery requirements and environmental goals of the marketplace.

Number of Vehicles: This input allows LSPs to indicate the number of each type of vehicle available. Managing this data helps in operational planning and ensures that capacity meets customer demand.
Capacity of Each Vehicle: LSPs can detail the capacity of each vehicle type, whether in terms of volume, weight, or number of parcels it can carry. This information is crucial for optimizing delivery loads and routes.

CO2 Emissions: Reflecting the marketplace's dedication to sustainability, LSPs are required to provide information on the CO2 emissions for each type of vehicle. This data supports the promotion of greener delivery methods and helps customers make environmentally conscious choices.

LSPs have the ability to add new vehicles, update existing fleet information, or remove vehicles as needed. These updates ensure that the fleet data remains current and reflective of the LSP's capabilities, enhancing service accuracy and environmental compliance.

4) Area of Operations

2.3.1.7 Technical Approach of Development

Laas Marketplace Dashboard is a web-based application conceived to operationalize the Logistic-as-a-Service (LaaS) model in last-mile logistics. It provides a cohesive platform with advanced functionalities derived from the requirements specified in WP3. The Dashboard integrates several key features:

- **User Authentication:** Secure login and registration mechanisms.
- **Authorization:** Robust security using JSON Web Tokens (JWT) for maintaining secure sessions.
- **User Management:** Role-based access control to enforce appropriate user permissions.
- **Routing Policy:** Efficient navigation and access control within the application.
- **Connection to Backend:** Seamless backend integration for real-time data retrieval and management.
- **Internationalization:** Support for multiple languages to cater to a diverse user base.
- **Data Visualization:** Utilization of charts to display data analytics and insights.

2.3.1.8 Technical Architecture

The architecture of the Laas Marketplace Dashboard leverages modern web technologies[2] and frameworks to ensure a scalable, maintainable, and efficient application:

Table 3:Technologies Used in Development of LaaS_FE

Feature	Technology	Details
Frontend Framework	React.js	Enhances development robustness by enforcing type safety.

[2] https://github.com/frontier-innovations/greenlog-frontend

Containerization	docker Docker	Encapsulates the application in containers for consistency across environments, including a Dockerfile for building images and a docker-compose.yml for orchestrating services and managing volumes.
Styling	Tailwind CSS Tailwind CSS	Tailwind CSS provides a utility-first approach to styling, while Material-UI components offer ready-to-use UI elements.
Bundling and Optimization	VITE Vite js	Known for its fast, lean builds of React applications into plain JavaScript, enhancing performance and development experience.
State Management	React JS Context API ContexAPI	Manages state across components without prop drilling, facilitating a clean data flow architecture.
Localization	i18next I18next	Allows dynamic changes between English and Greek, ensuring all UI strings are localized.
Routing and Navigation	React Router React Router Dom	Manages application routing and parameter handling, securing routes based on authentication tokens.

Data Visualization	Recharts	Renders responsive and interactive charts, integral for displaying analytical data.
Backend Integration	Axios	Interfaces efficiently with the backend API to fetch and manage data.

This structure and choice of technologies ensure that the Laas Marketplace Dashboard is not only robust and secure but also adaptable to the evolving future of coding.

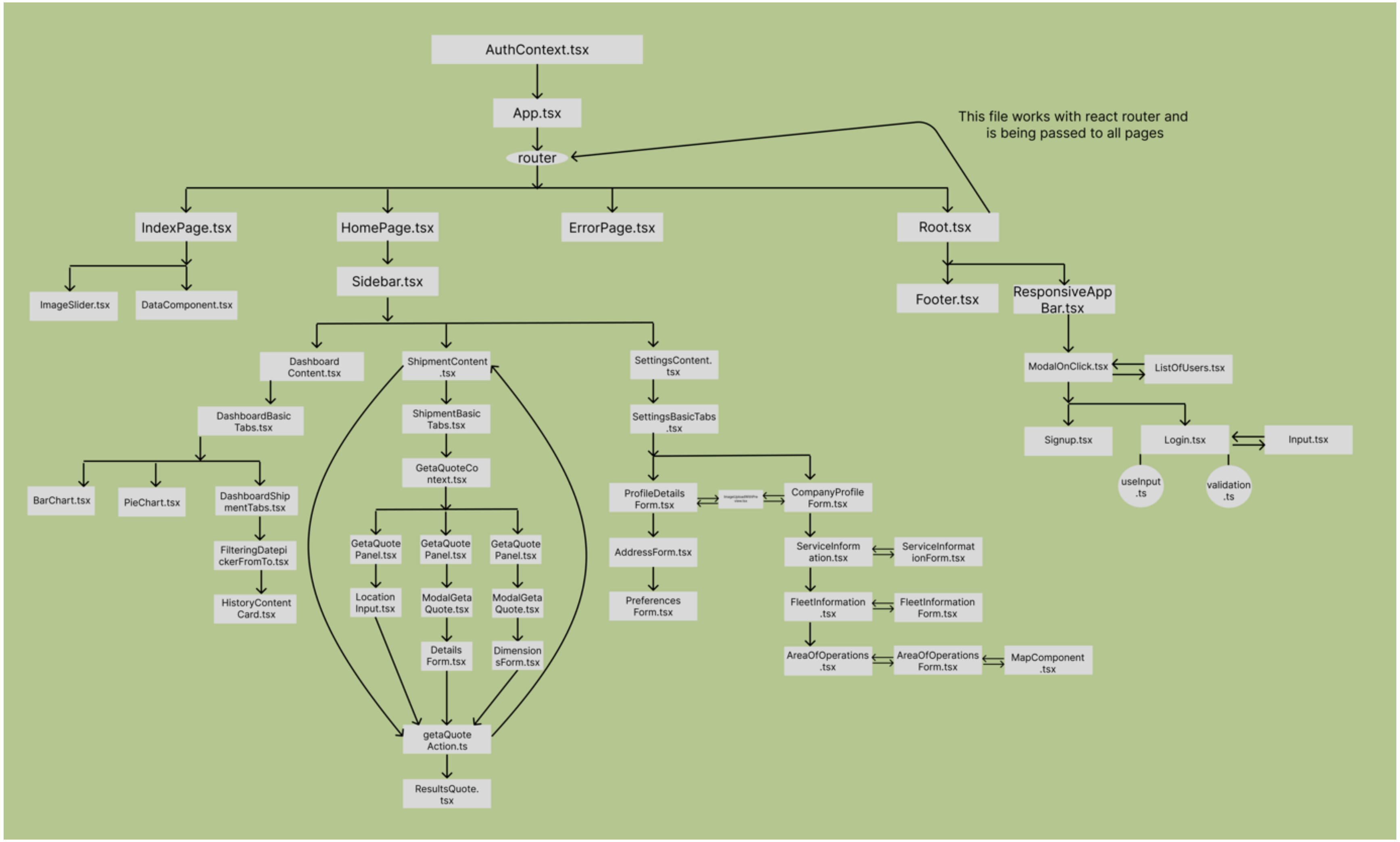


Figure 11 Architecture Component Tree of Laas Marketplace.

2.3.2 LaaS Backend

This section introduces LaaS marketplace's backend system, a critical component responsible for facilitating core functionalities and ensuring smooth operation among the components that are part of the LaaS. The backend system serves as the technological foundation, supporting the following functionalities and key features:

- Logistics Service Management: Catalogs and manages available logistics services offered by various providers within the marketplace. This includes basic pricing, delivery types, emissions for the fleet of vehicles and operating regions.
- *Matching and Optimization Engine*: Matches business requirements with the most suitable logistics providers based on factors like cost, service level, green incentives, and real-time data. Obtain different quotes based on those factors and allow the business to select one that fits their needs.
- *Data Management and Analytics*: Stores and manages all platform data, including user information, service details, and order history. Provides data analytics capabilities to optimize marketplace operations and user experience.
- *Integration with External Systems*: Connects with external systems for seamless data exchange and functionality.
- *User Management*: Creates and manages user accounts, profile information and details for both businesses and logistics providers.
- *Authentication/Authorization*: The system uses JWT (JSON Web Tokens) for stateless authentication and authorization between the backend and the frontend of the marketplace. This ensures secure access control over the system's resources.
- *Email Verification/Reminder Service*: User verification and password reminders are handled via an integrated email service for reliable email delivery. Either Logistics Service Providers or Local Shop Owners can register, verify or get reminders through email.

2.3.2.1 LaaS Backend Integrations:

- Marketplace Frontend: The first and foremost integration of the marketplace backend is the user interface. They communicate through secure RESTful APIs.
- Nudging Engine: Integrates via RESTful APIs to prioritise and incentivise green delivery options.
- Dynamic Pricing: Provides dynamic pricing quotes based on delivery options and parameters.

The sequence diagram in Figure 12 depicts the interactions between the LaaS platform and the components needed to generate the quotes. The generation of quotes goes through the (Re)routing engine to calculate the distance and the alternative routes, the Dynamic Pricing module that generates the price based on the given parameters, and the Nudging engine that prioritizes and incentivizes green delivery options.

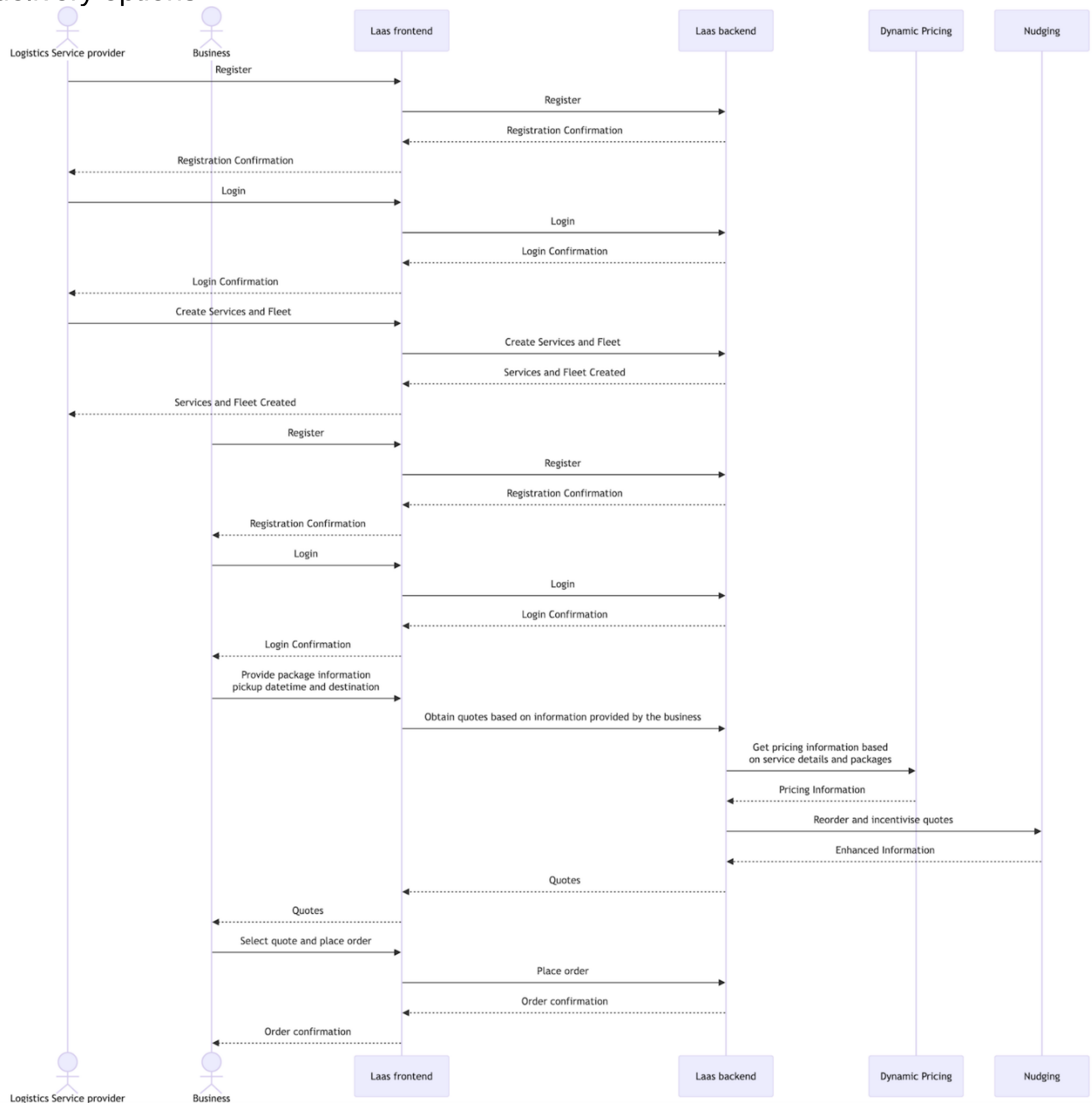


Figure 12 Sequence Diagram of LaaS BE as an orchestrator

2.3.2.2 LaaS_BE Technologies

The architecture is built to be scalable, and secure, ensuring seamless integration and interoperability with various components of the LaaS ecosystem. Key technologies[3] presented in Table 4 Technologies Used in Development of LaaS_BE include PostgreSQL for the persistence layer, Java 17/Spring Boot for the application layer, OpenAPI 3.0 for API standardization, and Docker for containerization.

[3] https://github.com/frontier-innovations/greenlog-backend

Table 4 Technologies Used in Development of LaaS_BE

Java 17	
Spring-Boot 3.2.3	
Springdoc 2.4.0, OpenAPI v3	
Hibernate 6.x	
PostgreSQL 16.2	
JSON Web Tokens	
Docker	

Database Schema: The backend services utilize PostgreSQL as the primary database. The schema includes tables for users, logistics service providers, delivery requests, vehicle details, and pricing models.

Security and Compliance: The LaaS backend adheres to industry-standard security practices, including secure communication protocols (https) and secure authentication and authorization mechanisms (JWT). Compliance with relevant data protection regulations (e.g., GDPR) is ensured through strict data management policies.

Continuous Improvement: The backend architecture is designed to be easily updated regularly, based on user feedback and evolving business requirements. This approach ensures the LaaS platform remains adaptive to new challenges and requirements.

2.3.2.3 API Endpoints Summary: Marketplace Backend – Frontend

Method	***Endpoint***	***Description***
POST	/api/v1/auth/register	Call to register a new user
POST/GET	/api/v1/auth/register/confirm	This is sent through an email, for verifying the email of the user
GET	/api/v1/auth/reset	Reset password call
POST	/api/v1/auth/reset/save	Save new password

POST/GET	/api/v1/auth/reset/confirm	Confirm new password for user
POST	/api/v1/auth/login	Login to authenticate the user and get a json web token for authorization
GET	/api/v1/users/{id}	Get user profile details
PUT	/api/v1/users/{id}	Update user profile details
POST	/api/v1/auth/login	Login to authenticate the user and get a json web token for authorization
GET	/api/v1/lsp-services	Get all LSP services available. As an LSP, you can only get your own services.
GET	/api/v1/lsp-services/{id}	Get details for a specific LSP service. As an LSP, you can only get your own services.
PUT	/api/v1/lsp-services/{id}	Update service provided by an LSP. This includes options about delivery types (for example, same day), base pricing options and others.
DELETE	/api/v1/lsp-services/{id}	Delete provided LSP service. An LSP can remove only their own.
GET	/api/v1/lsp-vehicles	Get all fleet information about an LSP. As an LSP, you can only get your own services.
GET	/api/v1/lsp-vehicles/{id}	Get details for a specific LSP vehicle fleet. As an LSP, you can only access your own.
PUT	/api/v1/lsp-vehicles/{id}	Update LSP's vehicle fleet. This includes options about transport types (for example, e-bike, car), base emissions and other options.
DELETE	/api/v1/lsp-vehicles/{id}	Delete LSP vehicle fleet. An LSP can remove only their own.
GET	/api/v1/lsp-areas	Get all areas that an LSP operates in. As an LSP, you can only access your own data.
GET	/api/v1/lsp-areas/{id}	Get details for a specific area an LSP operates in.
PUT	/api/v1/lsp-areas/{id}	Update an LSP's operating area.
DELETE	/api/v1/lsp-areas/{id}	Remove an area from an LSP's list of operating areas.
GET	/api/v1/orders	Get all orders that you have made in the past (or are yet in any state). Only available to Local Shop Owners.
POST	/api/v1/orders	Create order. Only available to Local Shop Owners.
GET	/api/v1/orders/{id}	Get order by ID. Only available to Local Shop Owners.
GET	/api/v1/statistics/users	Get user statistics
GET	/api/v1/statistics/orders	Get order statistics

2.3.3 Subcomponents of LaaS Marketplace

2.3.3.1 Dynamic Pricing Module

The Dynamic Pricing Module is responsible for calculating delivery prices based on various parameters, ensuring optimized, fair, and sustainable pricing strategies. It comprises several sub-components:

2.3.3.1.1 Price Calculation Engine

The Price Calculation Engine serves as the core component of the Dynamic Pricing Module. Its functionalities include:

- Receiving input data from the Dynamic Pricing API, including vehicle type, distance, operational cost per kilometer, and preferred delivery time window.
- Utilizing predefined algorithms to calculate the delivery price, incorporating factors like operational costs, distance, vehicle type, and delivery time windows.
- Communicating the finalized price back to the Dynamic Pricing API.

2.3.3.1.2 Dynamic Pricing API

The Dynamic Pricing API acts as the interface between external systems and the Price Calculation Engine. It handles:

- Receiving input data in JSON format from external systems requesting price calculations.
- Validating input data to ensure all necessary parameters are provided and correctly formatted.
- Sending validated data to the Price Calculation Engine for processing.
- Returning the calculated price to the external system in a structured JSON response.

2.3.3.1.3 Input Parameters and Data Sources

The Dynamic Pricing Module requires various inputs to accurately calculate delivery prices. These include:

- Vehicle Type: Specifies the type of vehicle used for delivery, such as bike or van.
- Distance: The distance of the delivery in kilometers.
- Operational Cost per Kilometer: The cost associated with operating the vehicle per kilometer, which varies by vehicle type.
- Time Window: The preferred time window for delivery, indicating whether the delivery is during peak or off-peak hours.

These parameters are essential for the accurate functioning of the Price Calculation Engine. More parameters will be added based on the requirements and the further refinement of the use-cases.

2.3.3.1.4 Data Handling and Processing

The Dynamic Pricing Module processes data as follows:

- Receive Input: Input data is received through the Dynamic Pricing API.

- Validation: The API validates the input data to ensure all required parameters are provided and correctly formatted.
- Price Calculation: The Price Calculation Engine computes the price based on the validated input data using predefined algorithms that consider various factors such as distance, operational costs, vehicle type, and time windows.
- Return Output: The calculated price is returned to the requesting system via the Dynamic Pricing API.

These functional requirements ensure seamless communication and coordination among the sub-components of the Dynamic Pricing Module and the LaaS marketplace.

2.3.3.2 Nudging Engine

The Nudging Engine is tasked with facilitating the interaction between various components to implement credit mechanisms for delivery services. It comprises several sub-components:

2.3.3.2.1 Incentives and Credits engine

The Incentives and Credits Engine serves as the intermediary between the Incentive Measures Management API and the LaaS delivery API. Its functionalities include:

- Receiving information regarding incentive measures from the Incentive Measures Management API.
- Utilizing the LaaS delivery API to retrieve data pertaining to delivery trips.
- Communicating the finalized credits per delivery option back to the LaaS platform.

2.3.3.3 Incentive Measures Management API

The Incentive Measures Management API serves as a crucial data source for the Nudging Engine. It receives inputs from the DCT, including:

- Information on city programs, referred to as constraints, specifying various parameters such as off-peak hours during weekdays, designated delivery areas utilizing geofencing (e.g., city centers), and prioritization of delivery modes (e.g., cargo bikes).
- Credit allocation rules for each program dictating the number of credits earned based on specific conditions such as peak or off-peak hours.

2.3.3.4 LaaS Delivery Options API

The LaaS Delivery Options API provides essential inputs for the Nudging Engine to optimize delivery choices. It receives the following inputs for each new delivery:

- Trip identification.
- Final delivery address.
- Available delivery options generated through dynamic pricing.

  - Mode of delivery/transport.
  - Estimated date and time of delivery.

Additionally, it outputs the following data to the LaaS platform:

- Trip identification.
- Credit amounts allocated to each delivery option.

These functional requirements ensure seamless communication and coordination among the Nudging Engine and its associated components, enabling efficient and incentivized delivery operations within the urban environment.

3. Bring-Your-Own-Device Service

3.1 Description of service

Small last mile delivery companies often face significant challenges due to limited resources. One of the most basic needs of the LSPs is to have a real-time tracking system that helps the courier monitoring the status of each parcel to coordinate and replan the deliveries if needed.

The inability of some small couriers to provide their drivers/riders with dedicated devices equipped with custom applications is one of the major issues they face. The lack of resources to manage and track deliveries can lower their operational efficiency and competitiveness in the market.

The set of SDKs developed in the BYOD addresses this problem by enabling small companies to integrate crucial functionalities into their available mobile applications. These SDKs are designed to be modular and affordable, allowing companies to select and implement only the features they need. This flexibility ensures that even companies with limited resources can enhance their delivery operations.

While the main focus of the tool is on small delivery companies, larger companies can also benefit from this solution for their combined operations. Future iterations will include the development of more complex functionalities tailored to the needs of larger operations, such as advanced route optimization (developed in T3.4) or dynamic scheduling and their integration with sophisticated logistics platforms.

The solution includes three essential SDKs designed to enhance the online asset monitoring and tracking of last mile deliveries. The first is the Event Registration SDK, which enables drivers to log and manage key delivery events such as parcel pickup, enroute, delivery completed, and delays. The second, the QR or barcode Label Scanning SDK, which allows the riders/drivers to scan QR codes on packages to quickly retrieve and update package information, simplifying the identification and monitoring process. Finally, the third is the Rerouting SDK, which offers real-time re-routing options based on different events defined in T3.4, optimizing delivery routes for improved efficiency and flexibility.

The instance of an application simulating an LSP system will be provided by MOSAIC (assigned to this partner from M6 as a mitigation action due to the withdrawal of the one responsible for providing the app and backend to conduct the demonstrations). As a result, in the first iteration, the SDKs will be provided in their simplified version. Certain functionalities such as obtaining information from electric vehicles will be added at a later stage. Additionally, the Rerouting functionality will be also developed during the second iteration, once the routing has been thoroughly tested during the first demonstrations.

3.2 Functional requirements

The following table outlines the functional requirements for the mobile application and associated SDKs, detailing the necessary features to match the user needs. These functional requirements have been defined by LSPs and technical developers in the context of co-designing sessions.

Functional requirements (FR) for the app			
ID of FR	Name of FR	Description of FR	Level of necessity
FR-1	Delivery task Management	The app shall display a list of assigned delivery tasks to the logged-in user, including details such as recipient address, delivery time, and special instructions (defined in the data models T2.1)	Could
FR-2	Real-Time Status Updates	The app shall enable deliverers to update the status of their deliveries in real-time, by registering events such as package pickup, delivery completed, and failed attempts.	Must
FR-3	Real-Time Status Updates	The app shall send real-time updates to the backend system for tracking and reporting purposes.	Must
FR-4	QR/barcode Code Scanning	The app shall include functionality for scanning QR and barcodes on packages to retrieve package information.	Must
FR-5	QR/barcode Code Scanning	The app shall ensure accurate retrieval of package information and seamless. Integration with theLaaS.	Must
FR-6	Optimal Route Display and Rerouting	The app shall display the optimal delivery route to the driver.	Should
FR-7	Optimal Route Display and Rerouting	The app shall provide the option to trigger a rerouting based on events (information of the electric vehicle, e.g., low battery)	Must
Functional requirements for the SDK1 (Event Registration SDK)			
FR-8	Event logging	The SDK shall provide APIs for registering a delivery event such as package pickup, on	Must

		the way, delivery competition or failed attempt. Including the parcel ID, status, timestamp and location (coordinates).	
FR-9	Event logging	The existence of the parcel in the database shall be confirmed before sending the update.	Must
FR-10	Event logging	The SDK shall ensure that event registration updates are sent in real-time to the backend system for tracking and reporting purposes.	Must
Functional requirements for the SDK2 (QR Label Scanning SDK)			
FR-11	QR Code Scanning Integration	The SDK shall integrate seamlessly with the app to enable QR and barcode scanning functionality for package identification	Must
FR-12	QR Code Scanning Integration	The SDK shall ensure accurate retrieval of package information and update delivery status accordingly.	Must
Functional requirements for the SDK3 (Rerouting SDK)			
FR-13	Real-Time Rerouting	The SDK shall provide APIs for generating alternative delivery routes based on real-time conditions.	Must
FR-14	Real-Time Rerouting	The SDK shall allow the app to present rerouting suggestions to drivers in a clear user-friendly manner, facilitating informed decision-making.	Must
Integration with other components of the project			
FR-15	Green-log Integration	The tool shall integrate the real-time services developed in Green-log that are essential for the last mile deliverer.	Must
FR-16	LaaS contact point	The tool shall serve as a contact point to provide to the Logistics as a Service component information regarding the parcels, ensuring comprehensive data	Should

		exchange and operational efficiency.	

3.2.1 Functional Architecture and relation with other components

The functional architecture of the BYOD tool is organised into four levels: hardware, frontend (mobile app), backend (instance of the backends of the LSPs), and dataspace. Each level has specific components and functionalities essential for the efficient real-time delivery tracking.

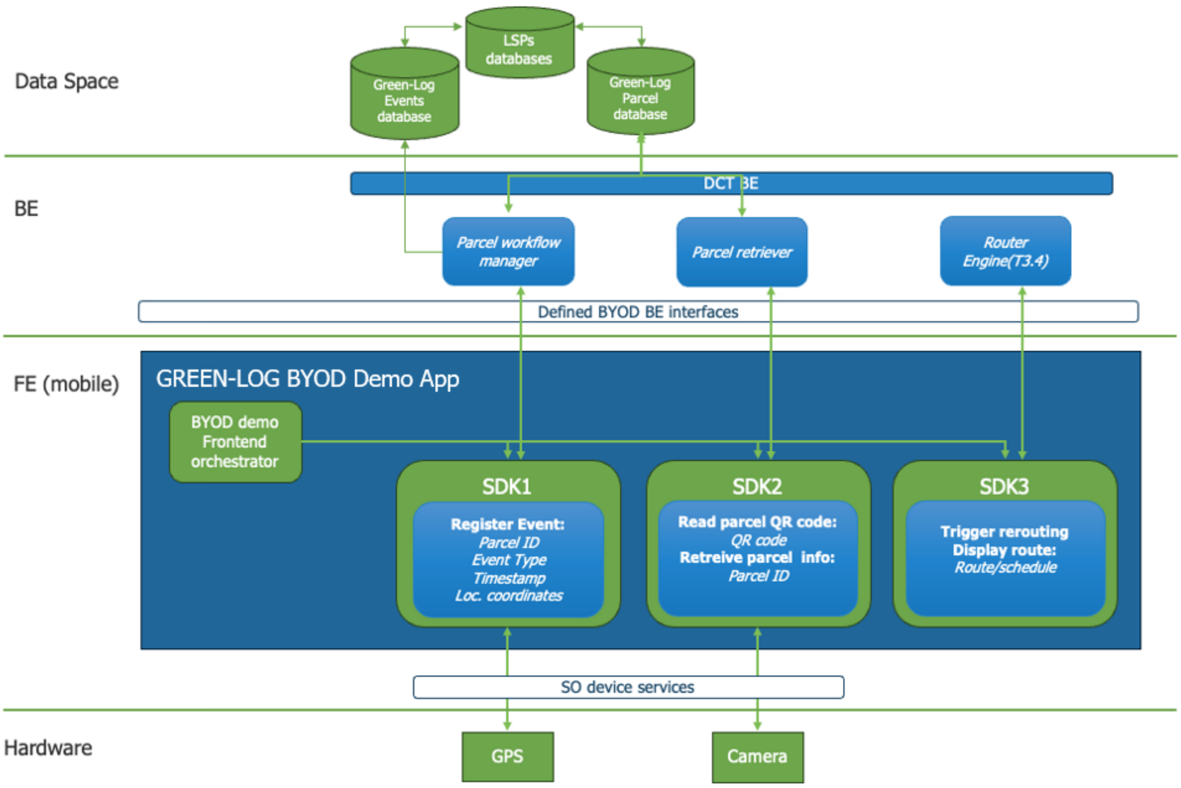


Figure 13. Functional architecture of BYOD and relation with other components (DCT and Re-routing)

At the hardware layer, we can find the GPS module and the camera of the mobile phone. The GPS module is used to determine the exact location of the parcels by providing its coordinates, which is crucial for real-time tracking and optimizing the delivery routes. The camera is used to scan QR codes or barcodes on parcels, enabling quick and accurate retrieval of the parcel information.

At the next layer, the frontend, there is an instance of the GREEN-LOG BYOD App developed for demonstration purposes where the three SDKs developed in the project are integrated. This app includes a BYOD demo frontend orchestrator, which triggers the workflow and manages the interaction within the app.

The backend layer includes two types of backend, the instance of the LSPs backend (due to the absence of an actual LSP backend) and the DCT platform backend. The LSP backend is responsible for managing the registration of events handled by SDK1 and includes a Parcel Workflow Manager to process and manage the parcel workflows. The SDK2 backend includes a Parcel Retriever, which verifies the existence of parcels before registering or retrieving any information, such as delivery location (coordinates), status of the parcel, receiver information, and others (defined in the T2.1 data models). The information about the events is sent to, consolidated and processed by the LaaS platform backend.

Finally, at the Data Space layer, the Green-Log dataspace serves as the central repository for all parcel related information. It stores standardised data in the Green-log models, ensuring consistency and accessibility. The Green-Log Parcel Database within the dataspace contains detailed information about each parcel, including coordinates for delivery, status, receiver information and others. And the Green-Log Events dataspace stores the historic events of a specific parcel. LSPs will provide their data to the dataspace, which will standardise it into the Green-Log models. This structured information flow ensures that the system operates efficiently, with accurate and real-time data available across all living labs and components.

The technology used to develop the tool is React Native to cover both, Android and IOS devices.

3.2.2 Mock-ups

The following images show a first version of the flow of the mobile application and a mock-up (including SDK1 and SDK2). The User Interface design will be adapted depending on the progress of the task.

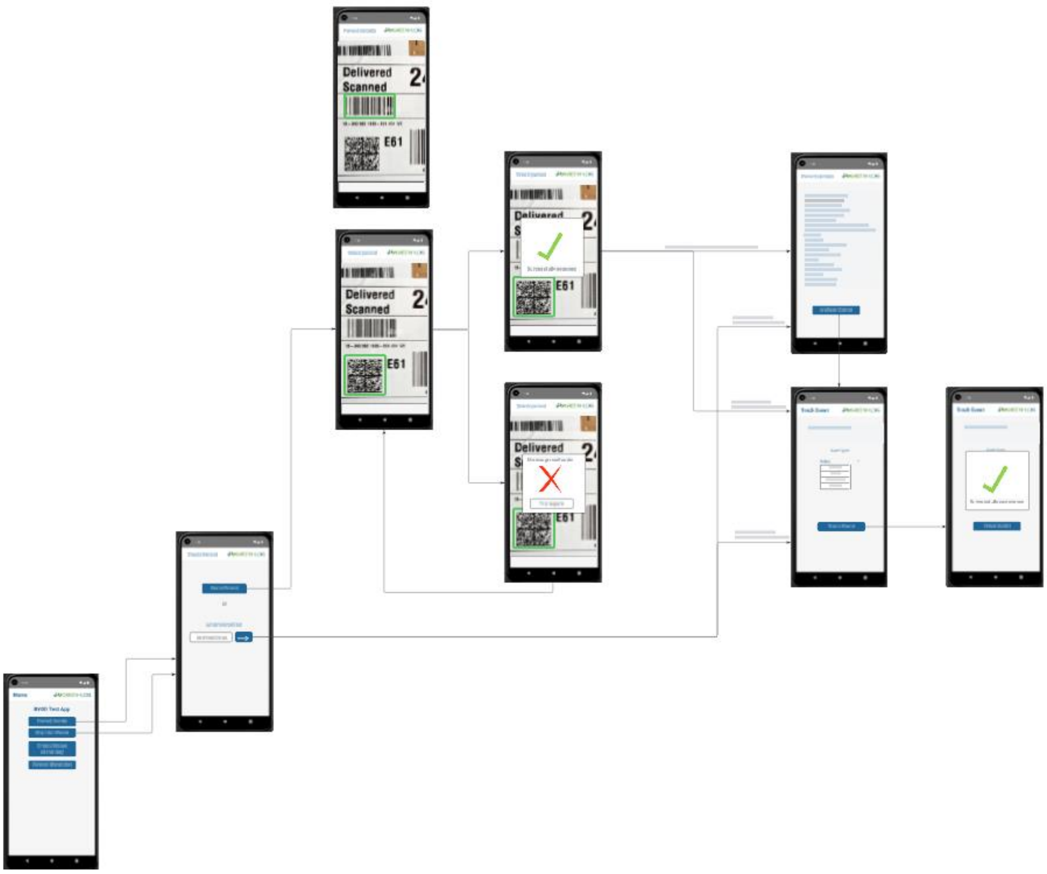


Figure 14. Overview of the workflow of the BYOD

The first screen (**Screen 1**) serves as a welcome screen, featuring the logo of the project.

Below the logo there are four buttons:

- Parcel details: This button allows the deliverers to enter or scan the parcel ID to retrieve detailed information about the parcel.
- Register event: This button enables deliverers to log a new event related to the parcel, such as pickup, delivery, or other status updates.
- Check delivery route: This button enables the user to check the list of deliveries with the location and timestamps provided in T3.4.
- Report disruption: This button enables the deliverer to report any disruption on the delivery course. For example, the electric vehicle has run out of battery, the road is closed, etc.

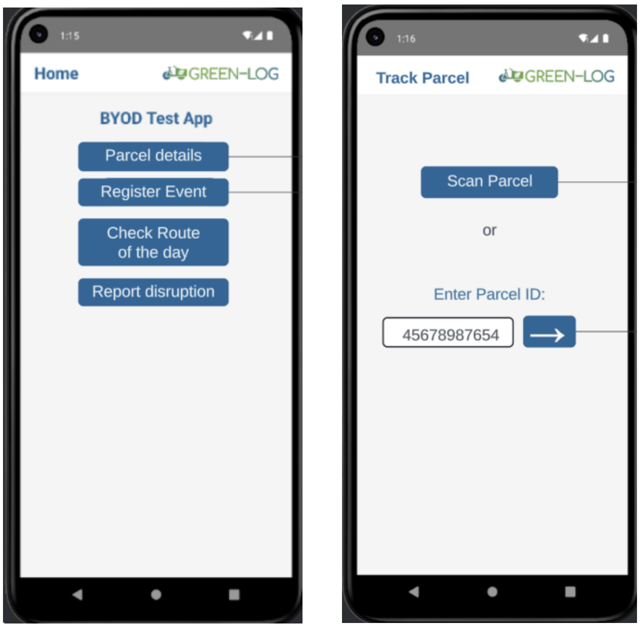


Figure 15. Screen 1, 2 of the BYOD tool mock-up

Screen 2: The deliverer has the option to enter the Parcel ID manually using the keyboard of the mobile phone or to scan the QR code or barcode when pressing the "Scan Parcel" button **(Screen 2)** using the mobile phone camera (**Screen 3 and Screen 4).**

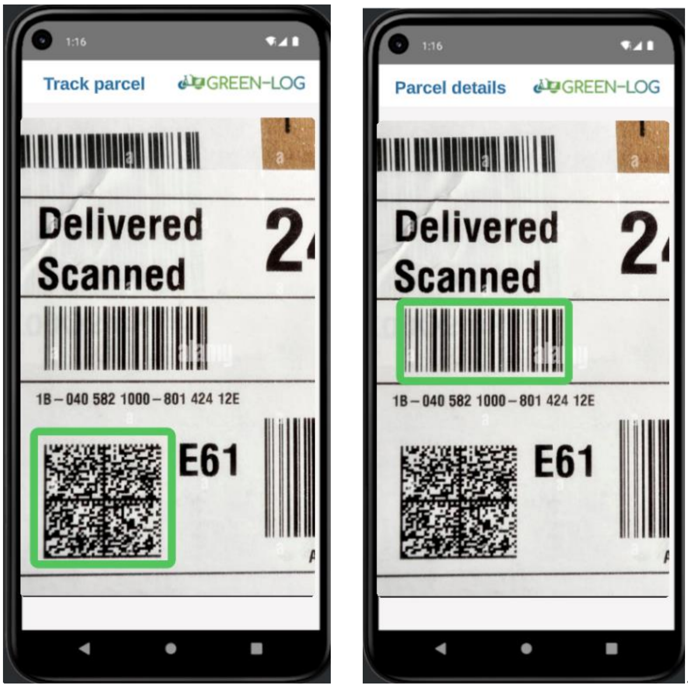


Figure 16. Screen 3 and 4 of the mock-up of the BYOD.

If the codes are not well scanned, and error will appear and the deliverer will have the option to try again (**Screen 5)**, if the code is correctly scanned, an approval message will appear (**Screen 6**).

Figure 17. Screen 5 and 6 of the BYOD mock-up

If the deliverer has selected “Parcels details”, a list of the details related to the parcel will appear on the screen **(Screen 7**). From that screen, the deliverer can also update the Status of the selected parcel. If the deliverer has selected “Register Event”, the user will have to select the type of event to be registered from a list (**Screen 8**), and the Parcel ID, the event type, the coordinates and timestamp will be updated in the Parcel and Parcel Events database **(Screen 9).**

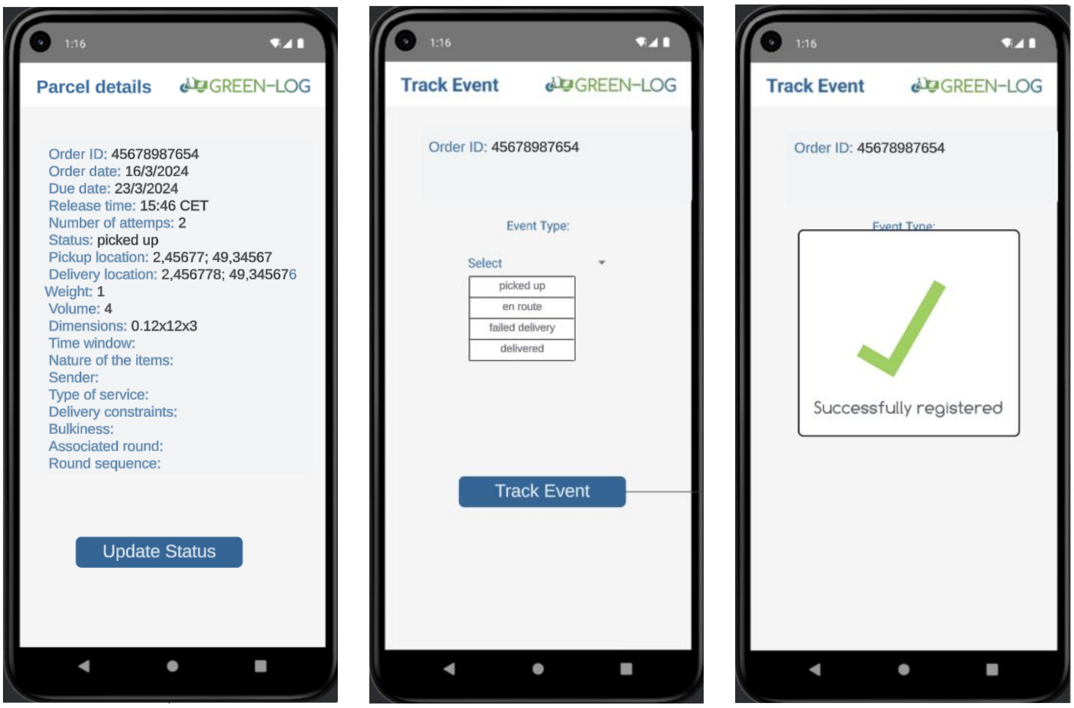


Figure 18 Screen 7, 8 and 9 of the BYOD mock-up.

The remaining screens (Check route of the day and Report disruption) will be designed once the service is more advanced.

3.3 Implementation and Evaluation

The BYOD will be tested in two phases. During the first phase, it will be trialed in Barcelona and Ispra by two small LSPs partners: e-Novia and VANaPED. In the second phase, two large companies, ACS SMSA and FedEx, will also test the advanced features of the tool for their combined operations.

As mentioned earlier, since the LSPs in Ispra and Barcelona do not have mobile applications where the SDKs can be installed, an instance of a tracking app will be developed by MOSAIC. This app will be used during the demonstration activities to showcase the usability of the SDKs.

3.3.1 Ispra LL

In Ispra, the rider of the bike, will go at the canteen and will register the first event for the selected parcels by entering the ID manually or scanning the QR/barcode. The rider will go to the intermediate point and register another event to the database, announcing that the lunchbox was placed on the droid.

The rider can also use the rest of the functionalities such as retrieve information of the parcel, check the optimal route and trigger a rerouting if something unexpected happens (to be included during the second round of demonstrations).

3.3.2 Barcelona LL

In Barcelona, we will have a different use case, but the same principles (of event registration) will be applied. Parcels are previously organized into two delivery plans – one for a rider using cargo-bike, and another for a rider using public transport modes. These plans will be defined by the Dynamic Routing Service (DRS) and will consist of predicted times at which the rider reaches check points along the route (e.g. changes of mode) and a registration of arrival at receiver locations and changes in status of each parcel (in transit, delivered, undelivered etc.). One of the aims of the pilot is to use registered events to assess how accurate the DRS plans are (this is crucial if LSPs are to use rail as a potential alternative to using vehicles from their own fleet). Following the Phase 1 pilot (City-based use cases), an Interurban Use-Case will be examined; this investigates how a prototype robot device could be integrated, and how the LSPs of the Barcelona and Terrassa city use-cases could collaborate. The events for this pilot phase are still being discussed by the BCN LL partners.

Each rider will check the parcels into the container that will be used with the assigned mode (e.g.: the rider using public transport uses a rucksack and/or a trolley). The time of leaving the hub will be registered. Each rider will have a BYOD-equipped device for registering events.

3.3.3 Athens LL

In Athens, the BYOD will be tested during the second round, and it will include, apart from the previously mentioned functionalities, the routing visualization and event-triggered optimization of delivery routes. Details of this demonstration will be defined after the first round of tests.

4. Nowcasting and Forecasting Engine

4.1 Description of Components

4.1.1 Problem Description

Urban areas face significant challenges in managing traffic congestion and logistic demands, a problem which impacts both city authorities and LSPs. The current situation is characterized by highly unpredictable traffic conditions and fluctuating logistics demands, which are mitigated by arbitrary experience-based predictions of key personnel, making it difficult for city authorities to ensure smooth traffic flow calculation, and for LSPs to optimize their resource allocation based on demand. This unpredictability leads to inefficiencies, increased operational costs and compromised service quality. In order to mitigate the aforementioned issues, the is a need for assistive tools that can take advantage of already existing data and can provide accurate, near real-time predictions for both road network traffic and geospatial logistics service demands.

The Nowcasting and Forecasting Engine (NFE) is designed to address this need and aims to enhance operational efficiency, strategic decision making and overall urban mobility and urban logistics performance overall. The NFE serves as a powerful standalone tool that can significantly enhance the operational efficiency and strategic decision-making capabilities of both Logistics Service Providers (LSPs) and city authorities. This engine is designed to deliver near real-time predictions and short-term forecasts for both road network traffic and logistics service demand. By leveraging historical data and near real-time inputs attained from the GREEN-LOG Dataspace, the engine serves the users as an additional assistive tool allowing city authorities to improve traffic flow and enhance road safety and LSPs and shop owners enlisted in the GREEN-LOG LaaS Marketplace to manage logistics resources efficiently and enhance the overall operational efficacy.

For city authorities, forecasting traffic conditions helps in managing urban mobility more efficiently. It allows for proactive measures to mitigate congestion, improve traffic flow, and enhance road safety. The traffic short term forecasts apply mostly to use-cases where main roads are utilized by LSPs, hence Athens and Barcelona use-cases will be mostly benefitted from this objective of the NFE.

By predicting traffic conditions and parcel demand, forecasting also enables LSPs to optimize delivery routes, ensure timely deliveries, and allocate resources more effectively. It also allows shop owners using the Laas Marketplace, to see visualization of forecasted demand for specific areas, allowing them to optimize their scheduling of parcel pickups, especially for cities where LSPs use alternate modes of transport like the use-cases of Mechelen, Leuven and Ghent. This capability is especially crucial in urban environments where dynamic changes in traffic patterns and fluctuating demand can pose substantial challenges.

Overall, the integration of forecasting tools aligns with the diverse requirements of different cities by promoting safer, more efficient, and collaborative logistics operations. With the use of predictive analytics, both LSPs and city authorities can achieve a higher level of service quality, meet customer expectations, and contribute to the sustainable growth of urban logistics ecosystems. Forecasting not only supports LSPs in their operational endeavors but also empowers city authorities to make informed decisions that enhance urban living conditions for all residents.

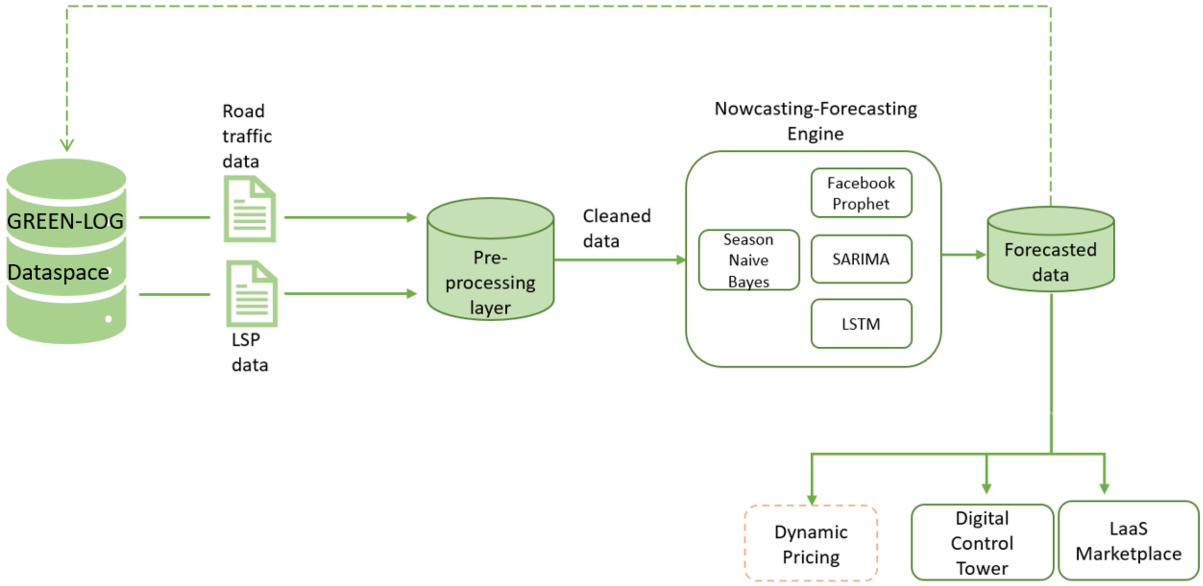


Figure 19 Nowcasting and Forecasting Engine high level architecture

As mentioned above, the NFE serves a dual objective, to provide forecasts as output that will be visualized in the form of plots by the two interfaces of the GREEN-LOG Platform: (a) in the Digital Control Tower to serve the city authorities as an assistive tool for traffic management improvement,(b) in the LaaS Marketplace to aid the users in their selection of the best fitting LSP (see chapter 3). The two problems that the NFE is assistive to are:

1. **Traffic Forecasting:**
    - To predict road network traffic conditions, including sudden congestion, based on historical and near real-time data.
    - To provide actionable insights for dynamic re-routing and traffic management.
2. **Logistics Demand Forecasting:**
    - To predict the demand for parcel pickups, enabling logistics service providers to allocate resources efficiently.

The engine must integrate various data sources, process large datasets, and produce accurate short-term forecasts to support real-time decision-making.

4.1.2 Definition of Nowcasting and Forecasting

Before we extensively inspect the functionality of the NFE it is important to depict the difference between the forecasting and nowcasting techniques, since there are two GREEN-LOG services with similar functionality but serving different purposes in the scope of the project.

Forecasting involves predicting future values based on past and present data. It is typically used for longer-term predictions, providing insights over extended periods. Forecasting models analyze historical trends and patterns to project future conditions, which is essential for strategic planning and resource allocation [REF-12]. Mathematically, forecasting can be represented as:

$$\hat{y}t + h = g(yt,\ yt - 1,\ \ldots,\ yt - n; \theta)$$

where:

- $\hat{y}t + h$ is the forecast for h steps ahead,
- $yt, yt - 1, \ldots, yt - n$ are the historical observations,
- g is the forecasting function,
- θ represents the parameters of the model.

Nowcasting, on the other hand, refers to the prediction of the present and the very near future. It is particularly useful for short-term forecasting, such as predicting traffic and logistics conditions within a 6-hour time window. Nowcasting relies heavily on real-time data and provides immediate, actionable insights for dynamic decision-making [REF-13]. The mathematical representation of nowcasting is:

$$\hat{y}t = f(yt - n,\ yt - n + 1,\ \ldots,\ yt)$$

where:

- $\hat{y}t$ is the nowcast for time t,
- $yt - n, yt - n + 1, \ldots, yt$ are the observations from past time steps,
- f is the nowcasting function, typically involving real-time data integration.

The key difference between nowcasting and forecasting lies in their time horizons and data dependencies. While forecasting uses historical data to predict future trends over days, weeks, or months, nowcasting focuses on the immediate future, leveraging real-time data to make predictions for the next few hours [REF-15].

The NFE is more focused on nowcasting since the predictions will be based on near real-time data for dynamic decision making, whilst there is another service in the project (i.e Spatio-temporal demand prediction) which focuses on longer time horizon forecasts for scheduling purposes. This dual approach ensures that both long-term planning and short-term operational adjustments are supported, enhancing the overall efficiency of logistics and traffic management systems.

4.1.3 Data Sources

The success of the Nowcasting and Forecasting Engine (NFE) hinges on the quality and comprehensiveness of the data it utilizes. Two main dataset types are crucial for implementing the NFE: traffic data and parcel demand data. These datasets provide the necessary inputs, containing the most critical features to train and validate the forecasting models, ensuring accurate and timely predictions. Those datasets will be substituted by data stored and in GREEN-LOG Data Space, optimizing the outputs of the NEF and potentially expand its functionality. For the initial implementation of the NEF a use of synthetic data based on real datasets was used.

Traffic Data

The traffic data is essential for forecasting road network conditions. The dataset used, includes detailed records of traffic flow, average speeds, and vehicle counts at various locations and times. The dataset typically has the following columns:

- **appprocesstime**: The timestamp when the data was processed.
- **average_speed**: The average speed of vehicles at the recorded time and location.
- **countedcars**: The number of vehicles counted at the recorded time and location.
- **deviceid**: The identifier of the device that recorded the data.
- **road_info**: Detailed information about the road segment where the data was recorded.
- **road_name**: The name of the road where the data was recorded.

The data used for the initial implementation based on data from loop detectors of the Attica Peninsula and are openly accessible by the Greek Government.

	appprocesstime	average_speed	countedcars	deviceid	road_info	road_name
0	2023-01-05T08:00:00	42.000000	305	MS450	ΚΥΡΙΟΣ ΔΡΟΜΟΣ 80 Μ. ΠΡΙΝ ΑΠΟ ΤΗ Λ. ΜΕΣΟΓΕΙΩΝ	ΞΕΝΟΠΟΥΛΟΥ
1	2023-01-05T08:00:00	101.250000	1318	MS144	ΚΥΡΙΟΣ ΔΡΟΜΟΣ ΜΕ ΚΑΤΕΥΘΥΝΣΗ ΓΛΥΦΑΔΑ ΠΡΙΝ ΑΠΟ Τ...	ΠΟΣΕΙΔΩΝΟΣ
2	2023-01-05T08:00:00	87.173913	2730	MS292	ΚΥΡΙΟΣ ΔΡΟΜΟΣ ΜΕ ΚΑΤΕΥΘΥΝΣΗ ΚΕΝΤΡΟ 120 Μ. ΠΡΙΝ...	Λ. ΚΗΦΙΣΙΑΣ
3	2023-01-05T08:00:00	39.250000	1885	MS838	ΚΥΡΙΟΣ ΔΡΟΜΟΣ ΜΕ ΚΑΤΕΥΘΥΝΣΗ ΠΡΟΣ ΒΟΥΛΙΑΓΜΕΝΗ, ...	Λ. ΒΟΥΛΙΑΓΜΕΝΗΣ
4	2023-01-05T08:00:00	45.500000	2950	MS291	ΚΥΡΙΟΣ ΔΡΟΜΟΣ ΜΕ ΚΑΤΕΥΘΥΝΣΗ ΚΕΝΤΡΟ 120 Μ. ΠΡΙΝ...	Λ. ΚΗΦΙΣΙΑΣ
...	...	...	...	...	...	...
1094172	2023-05-07T11:00:00	42.282443	15720	MS523	ΚΥΡΙΟΣ ΔΡΟΜΟΣ ΜΕ ΚΑΤΕΥΘΥΝΣΗ ΠΡΟΣ Λ. ΓΑΛΑΤΣΙΟΥ ...	Λ. ΟΜΟΡΦΟΚΚΛΗΣΙΑΣ/ΒΕΪΚΟΥ
1094173	2023-05-07T11:00:00	57.797710	20960	MS535	ΚΥΡΙΟΣ ΔΡΟΜΟΣ ΜΕ ΚΑΤΕΥΘΥΝΣΗ ΟΑΚΑ 170 Μ. ΠΡΙΝ Α...	ΚΥΜΗΣ
1094174	2023-05-07T11:00:00	45.919192	3960	MS556	ΚΥΡΙΟΣ ΔΡΟΜΟΣ ΜΕ ΚΑΤΕΥΘΥΝΣΗ ΠΡΟΣ Ε.Ο ΑΘΗΝΩΝ-ΛΑ...	ΑΧΑΡΝΩΝ
1094175	2023-05-07T11:00:00	28.953846	5200	MS582	ΚΥΡΙΟΣ ΔΡΟΜΟΣ ΜΕ ΚΑΤΕΥΘΥΝΣΗ ΠΡΟΣ ΠΑΤΗΣΙΩΝ 80 Μ...	ΗΠΕΙΡΟΥ
1094176	2023-05-07T11:00:00	33.584046	14040	MS584	ΚΥΡΙΟΣ ΔΡΟΜΟΣ ΜΕ ΚΑΤΕΥΘΥΝΣΗ ΠΡΟΣ ΑΧΑΡΝΩΝ 70 Μ....	ΙΟΥΛΙΑΝΟΥ

Figure 20 Sample of input of Loop Detector Traffic Data

This dataset provides the necessary inputs for the NFE to predict traffic conditions, including average speeds and vehicle counts, which are crucial for accurate traffic forecasting.

Parcel Demand Data

The parcel demand data is critical for forecasting logistics service demand. This dataset includes detailed records of parcel pickups and deliveries to specific delivery locations, capturing information about the parcel size, weight, locations, and the logistics service provider (LSP) responsible for the delivery. The dataset typically has the following columns:

- **parcel_id**: The unique identifier of the parcel.
- **pickup_location**: The GPS coordinates of the pickup location.
- **delivery_location**: The GPS coordinates of the delivery location.
- **parcel_size**: The size category of the parcel (e.g., small, medium, large).
- **parcel_weight**: The weight of the parcel.
- **LSP**: The identifier of the logistics service provider handling the parcel.
- **delivery_location_name**: The name of the delivery location.
- **timestamp**: The timestamp of when the data was recorded.

	parcel_id	pickup_location	delivery_location	parcel_size	parcel_weight	LSP	delivery_location_name
timestamp							
2023-01-01 11:49:28.888513	7967d016-0271-4ccb-909a-104c403a3c6e	(37.93535833629881, 23.62184300889548)	(37.9838, 23.7275)	medium	7.03	LSP1	a
2023-01-01 12:05:22.739821	479a8267-673f-4b3b-8cb0-6fd740d9f7b0	(38.03523657486071, 23.742603884039596)	(37.9786, 23.7125)	medium	16.83	LSP1	b
2023-01-01 10:02:53.196118	d2a4f0ba-d52a-4062-b20c-f957ef15ed7a	(37.949220474802104, 23.600235023363346)	(37.9842, 23.7281)	medium	4.66	LSP2	c
2023-01-01 16:27:32.041724	c97874fb-294d-472f-8996-35766438c837	(38.00223242811099, 23.73228195217173)	(37.9838, 23.7275)	small	2.54	LSP1	a
2023-01-01 13:56:45.744065	6b7020be-3672-4370-8ae9-787d136b3e0f	(38.075320104683605, 23.75316478721698)	(37.9838, 23.7275)	small	10.30	LSP1	a
...	...	...	...	...	...	...	...
2023-01-01 12:45:36.211099	1987b2f2-13de-465b-a381-e4a4f721f0ed	(37.94976664737857, 23.602815329127257)	(37.9715, 23.7267)	medium	11.74	LSP2	d
2023-01-02 12:36:33.649953	549a4992-181f-4a57-8631-b07e7547470e	(37.95786958330991, 23.659017537903512)	(37.9715, 23.7267)	small	4.24	LSP2	d
2023-01-01 11:42:32.143114	efd3611d-65f6-4df6-90b1-3dc6de001ebc	(37.968057621890104, 23.656179169417026)	(37.9838, 23.7275)	small	13.09	LSP1	a
2023-01-02 16:05:06.545413	00a27115-727a-4eb3-aec4-8b5c6d4a1e43	(37.94199952865072, 23.62821156358557)	(37.9842, 23.7281)	large	19.08	LSP2	c
2023-01-01 11:52:45.294513	d97f1b31-366e-469e-bf6f-fbb62fd48907	(37.958471778622815, 23.682028815364657)	(37.9838, 23.7275)	large	7.66	LSP1	a

Figure 21 Sample of input of parcel demand data from LSPs

This dataset provides the necessary inputs for the NFE to forecast the parcel pickup demand for specific locations, including timestamps of the pickup orders and the locations where the parcel will be picked up from, which are crucial for accurate parcel pickup demand forecasting.

All the aforementioned data are completely synthetic for the initial implementation of the service using Faker (previously called fake-factory), a Python library that generates fake data, but basing them on real data samples acquired from partners of the project. For the parcel pickup data strict locations based on geospatial characteristics where defined since in the real use-cases similar well-defined areas will be depicted for the demonstration purposes of the project.

4.2 Functional Requirements

The NFE is a critical component of the GREEN-LOG project, designed to enhance operational efficiency for both city authorities and LSPs by providing real-time and near real-time predictions. The functional requirements as depicted after careful consideration of stakeholders and project experts' opinions, outline the necessary capabilities and performance standards that the NFE must meet to ensure it delivers accurate and timely forecasts. These requirements are categorized into data acquisition, data processing, forecasting, output generation, and integration with other components.

4.2.1 Data Acquisition

4.2.1.1. Traffic Data Collection

- The NFE will acquire historical and near real-time traffic data from the GREEN-LOG Dataspace.

4.2.1.2. Parcel Demand Data Collection

- The NFE will acquire historical and near real-time parcel demand data from the GREEN-LOG Dataspace.
- The data must include parcel IDs, pickup and delivery locations (GPS coordinates), parcel size, parcel weight, logistics service provider (LSP) IDs, delivery location names, and timestamps.
- The system must handle data from multiple LSPs and ensure consistency in data formats.

4.2.2 Data Processing

4.2.2.1. Data Preprocessing

- The NFE will perform data cleansing to remove inconsistencies, outliers, and irrelevant information from both traffic and parcel demand datasets.
- The system must transform raw data into a format suitable for model training and prediction, including aggregating data by time intervals and geographic segments where necessary.

4.2.2.2. Near real-time Data Handling

- The NFE must be capable of integrating and processing near real-time data feeds to ensure predictions are based on the most current information available.
- The system shall handle data latency and ensure near real-time data is incorporated into forecasts without significant delays.

4.2.3 Forecasting

4.2.3.1. Traffic Forecasting

- The NFE shall predict road network traffic conditions, including average speeds and vehicle counts, for the next 6 hours with predictions updated at fixed intervals.
- The system must utilize time series forecasting algorithms such as FB Prophet and ARIMA to generate traffic forecasts.
- The forecasts should be at a section level, providing aggregated predictions for specific road segments.

4.2.3.2. Parcel Demand Forecasting

- The NFE shall predict parcel pickup demand for the next 6 hours, with forecasts generated at 1-hour intervals.
- The system must use time series and machine learning models, including FB Prophet and Gradient Boosting Machines (GBM), to forecast demand for specific areas.
- The forecasts should be at the level of specific pickup locations, allowing LSPs to allocate resources effectively.

4.2.4 Output Generation

4.2.4.1. Forecast Data Output

- The NFE shall generate forecast outputs in a structured format, such as JSON, suitable for integration with other systems.
- The system must provide both numerical predictions and confidence intervals for the forecasts to indicate the reliability of the predictions.

4.2.4.2. Visualization Support

- The NFE must output data in a format that can be easily visualized by the Digital Control Tower for city authorities and the LaaS Marketplace UI for LSPs and shop owners.
- The system should support the generation of GEOJSON files for geographic visualization of traffic and parcel demand forecasts.

4.2.5 Integration

4.2.5.1. Green---Log Dataspace Integration

- The NFE shall integrate seamlessly with the GREEN-LOG Dataspace to retrieve historical and real-time data.
- The system must support API-based data extraction and ensure secure and efficient data transfer.

4.2.5.2. LaaS Marketplace Backend Integration

- The NFE shall push forecast outputs to the LaaS Marketplace backend to support the generation of visualizations and reports.
- The system must ensure that data is transferred in real-time to enable dynamic decision-making by users.

4.2.5.3. Future Integration Potential

- The NFE should be designed with the flexibility to integrate with future modules such as the Smart Pricing Module and the Dynamic (Re)routing Service.
- The system must be scalable to incorporate additional data sources and expand its forecasting capabilities as needed.

4.2.6 Performance and Reliability

4.2.6.1. Accuracy

- The NFE must achieve high accuracy in its predictions, with performance metrics regularly evaluated and models retrained as necessary to maintain prediction quality.

4.2.6.2. Scalability

- The system must be scalable to handle increasing volumes of data as more sensors and LSPs are integrated into the GREEN-LOG platform.

4.2.6.3. Reliability

- The NFE must operate reliably, ensuring continuous availability and timely updates of forecasts even under high data loads.

4.2.6.4. Latency

- The system must minimize latency in data processing and forecast generation to provide near real-time predictions.

These functional requirements ensure that the Nowcasting and Forecasting Engine delivers precise, timely, and actionable insights, supporting efficient traffic management and logistics operations. By adhering to these requirements, the NFE will enhance the strategic and operational capabilities of city authorities and LSPs, contributing to the overall success of the GREEN-LOG project.

4.3 Technology Stack

The following technologies are used for the development of the Nowcasting and Forecasting Engine (NFE):

Technology	Version	Description
Python	3.x	Primary programming language for data processing and machine learning.
Flask	2.x	Micro web framework for building the API.
Docker	20.x	Containerization platform for deploying the application.
openAPI (Flask-RESTPlus)	3.0.x	To create API documentation.

Table 5 NFE Technology Stack

By using these technologies, the Nowcasting and Forecasting Engine ensures robust data handling, efficient model training, and seamless API integration for delivering forecasts and supporting dynamic operational adjustments.

4.4 Methodology

The methodology for the NFE is designed to ensure accurate and timely predictions for both traffic conditions and parcel demand. Although synthetic data were initially fed to the NFE for initial implementation, the data pipeline of the GREEN-LOG project is already designed and after this initial implementation phase, the data integration process begins with the acquisition of the data through an API from the GREEN-LOG dataspace, which provides GEOJSON files containing timeseries data of the types mentioned above. This data forms the backbone of our forecasting models and are crucial for the reliability of the predictions.

4.4.1 Data Acquisition and Preprocessing

The first step involves extracting, transforming, and loading (ETL) the data from the GREEN-LOG dataspace. The ETL process is essential for cleaning and preparing the data, ensuring it is suitable for analysis and model training. Historical and near real-time traffic data is processed to remove any inconsistencies, outliers, and irrelevant information. Similarly historical and near real-time parcel demand data is cleansed to create a robust dataset that accurately reflects past demand patterns.

4.4.2 Forecasting Algorithms

After the data are cleansed and transformed, they are ready to be given as input in the forecasting algorithms. Based on the use-cases, several algorithms have been researched, and a comparison of the best models was selected to achieve the objectives of the NFE. Several algorithms were implemented during the exploration phase that included:

Time Series Analysis

For nowcasting, which focuses on predicting the near future, specific time series algorithms have proven to be particularly effective. These algorithms are designed to handle the intricacies of time-dependent data, making them suitable for short-term forecasting:

- **FB Prophet[4]**: A powerful forecasting tool developed by Facebook, Prophet is specifically designed for handling time series data that may have strong seasonal effects and multiple seasonality with daily, weekly, and yearly cycles. It is well-suited for nowcasting because of its ability to model and forecast short-term trends with high accuracy.
  $y^t = g(t) + s(t) + h(t) + \epsilon \mathrm{t}$
  where:
    - $g(t)$: models non-periodic changes
    - $s(t)$: represents periodic changes (e.g., weekly and yearly seasonality)
    - $h(t)$: accounts for holidays and events
    - ϵ: the error term

- **ARIMA (AutoRegressive Integrated Moving Average)**: Used for modeling and forecasting time series data based on historical patterns. ARIMA is effective for capturing the autocorrelations in the data.
  $\phi(B)(1 - B)^d\, yt = \theta(B)\epsilon \mathrm{t}$
  where:
    - $\phi(B)$ is the autoregressive part,
    - $\theta(B)$ is the moving average part,
    - $(1 - B)^d$ represents differencing.

- **SARIMA (Seasonal ARIMA)**: An extension of ARIMA that accounts for seasonality in the data. This model is particularly useful when the data shows seasonal effects that recur at regular intervals.

  $\phi(\mathrm{B})\Phi(Bs)(1 - B)d(1 - Bs)Dy_t = \theta(B)\Theta(Bs)\epsilon \mathrm{t}$

  where:
    - $\Phi(Bs)$ and $\Theta(Bs)$ are the seasonal components,
    - $(1 - Bs)D$ represents seasonal differencing.

[4] https://github.com/facebook/prophet

This model is not serving the purposes of nowcasting and was purely implemented during the exploitation phase for research purposes.

Machine Learning Models

Machine learning models were also adapted for time series forecasting by leveraging their ability to handle large datasets and capture complex patterns.

Gradient Boosting Machines (GBM)

which are essentially an ensemble learning technique that builds models sequentially, with each new model correcting errors made by the previous ones. GBM can be effective in capturing nonlinear relationships in time series data.

Deep Learning Models

Deep learning models have shown great promise in time series forecasting, particularly when dealing with large datasets and capturing long-term dependencies:

- **LSTM (Long Short-Term Memory):** A type of recurrent neural network (RNN) capable of learning long-term dependencies, particularly useful for time series forecasting where patterns can span long durations. LSTMs are effective in handling the temporal dependencies in traffic and parcel demand data.

$$ft = \sigma(Wf \cdot [ht - 1, xt] + bf)$$

$$it = \sigma(Wi \cdot [ht - 1, xt] + bi$$

$$c{\sim}t = \tanh\,(WC \cdot [ht - 1, xt] + bC)$$

$$ct = ft \cdot ct - 1 + it \cdot c{\sim}t$$

$$ot = \sigma(Wo \cdot [ht - 1, xt] + bo)$$

$$ht = ot \cdot \tanh\,(ct)$$

where ft, it, ot are the forget, input, and output gates, respectively, σ is the sigmoid function, and tanh is the hyperbolic tangent function.

- **CNN (Convolutional Neural Network):** While typically used for image data, CNNs can be adapted for time series forecasting to detect patterns and trends over time. CNNs can capture local dependencies and are particularly useful for spatial-temporal data such as traffic patterns.

Using Deep Learning models proved to be too time consuming for the purposes of nowcasting, and the results yielded by those models did exceed the results of simpler models hence, we refrained from using them on the first iteration of the implementation.

By using an ensemble of the time series analysis and machine learning models, the NFE leverages the unique strengths of each algorithm, resulting in highly accurate and reliable predictions for both traffic conditions and parcel demand. This multi-model approach ensures that the system is robust and can handle various types of data and forecasting challenges effectively.

4.4.3 Traffic Forecasting

For traffic forecasting, the methodology leverages historical data to train predictive models. Initially, the cleaned historical data undergoes an extensive training phase where the selected models are employed (Prophet, GBM). These models are selected for their proven effectiveness in time series forecasting and their ability to handle the complexities of traffic data and sudden fluctuations.

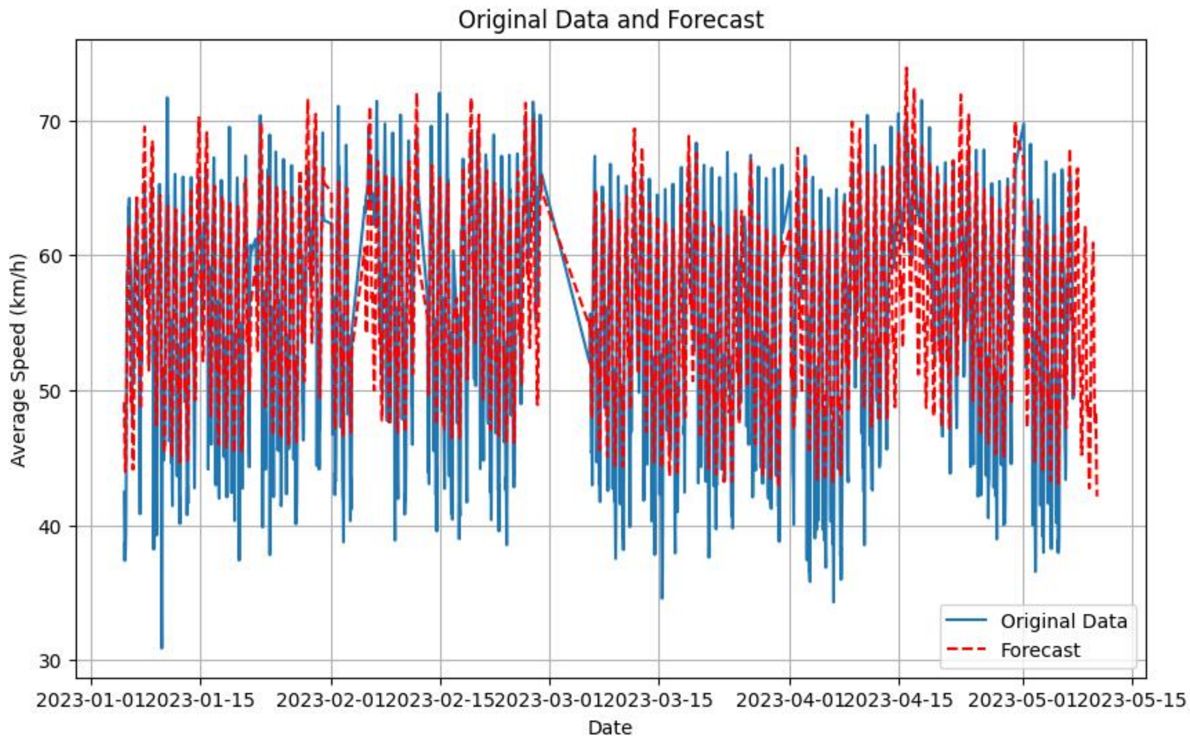


Figure 22 Evaluation of Traffic forecasting service with historical data

Once the initial training is complete, the models are continuously updated using the most recent data, specifically the last 24 hours of traffic information. This ensures that the models remain current and capable of adapting to recent trends and anomalies. The trained models then predict traffic conditions for the next 6 hours at fixed time intervals, which are determined based on the granularity of the input data. These predictions are pushed from the services through an API and are provided to the LaaS Marketplace backend, to provide actionable insights for traffic management with the use of visualizations (see Figure 22), assisting authorities to mitigate congestion and enhance road safety.

The traffic forecasting service provides outputs in section level based on aggregated data acquired from multiple loop detectors in that specific section, hence the result provide an overall estimation of traffic in section level, with the potential expansion of the NFE capabilities to provide forecast on even higher granularity on specific road locations.


```json
{
    "road_name": "Vouliagmenis",
    "data": [
        {
            "datetime": "2023-05-07 12:00:00",
            "avg_speed": 49.656235,
            "speed_category": "Moderate"
        },
        {
            "datetime": "2023-05-07 13:00:00",
            "avg_speed": 50.580764,
            "speed_category": "Moderate"
        },
        {
            "datetime": "2023-05-07 14:00:00",
            "avg_speed": 52.411205,
            "speed_category": "Moderate"
        },
        {
            "datetime": "2023-05-07 15:00:00",
            "avg_speed": 55.033824,
            "speed_category": "Moderate"
        },
        {
            "datetime": "2023-05-07 16:00:00",
            "avg_speed": 58.113360,
            "speed_category": "Moderate"
        },
        {
            "datetime": "2023-05-07 17:00:00",
            "avg_speed": 61.009421,
            "speed_category": "Fast"
        },
```


Figure 23 Sample JSON output for the Traffic service of NFE

4.4.4 Parcel Demand Forecasting

In parallel, the parcel demand forecasting component follows a similar methodology. Historical data on parcel pickups and deliveries is subjected to the ETL process, resulting in a clean and comprehensive dataset. This dataset is used to train models that forecast demand for specific areas.

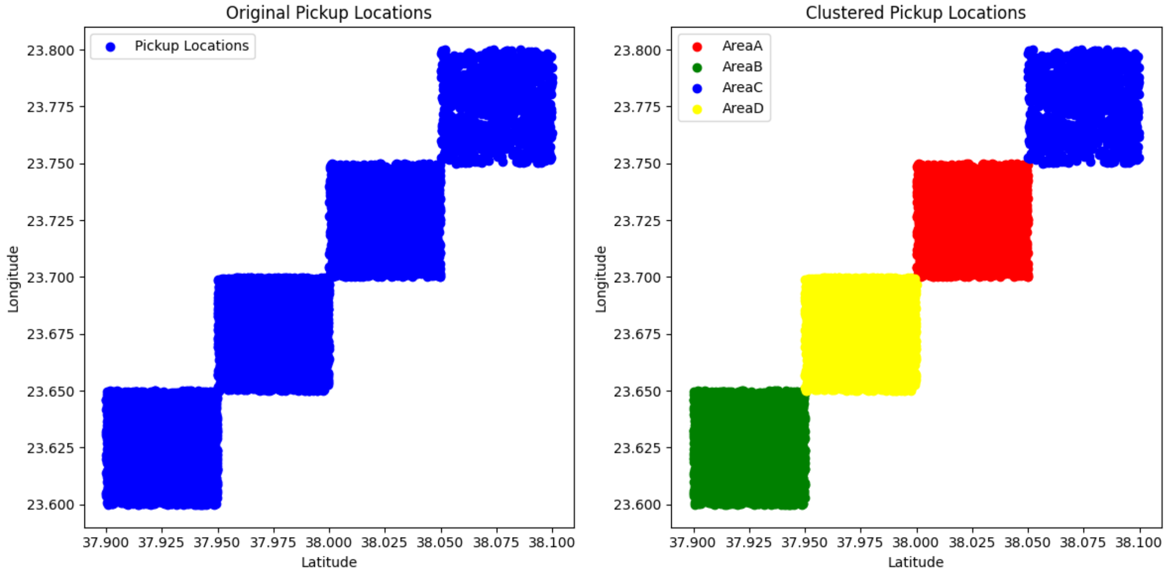


Figure 24 Evaluation of Clustering of pickup locations

The forecasting process involves predicting the parcel demand for the next 6 hours, with predictions made at 1-hour intervals. This granularity allows logistics service providers to allocate resources effectively and respond to fluctuating demand in a timely manner.

The parcel pickup demand forecasting is now implemented at a specific pick-up location level, essentially predicting the potential pickup demand in a specific area. By comparing the predicted demand in each area, the LSPs are able to potentially allocate resources accordingly and shop owners can optimize their scheduling of pick-ups from LSPs according the forecasted area congestion.

```
{
    "type": "FeatureCollection",
    "features": [
        {
            "type": "Feature",
            "geometry": {
                "type": "Point",
                "coordinates": [23.7275, 37.9838]
            },
            "properties": {
                "timestamp": "2023-01-02 19:00:00",
                "forecast": 226.0
            }
        },
        {
            "type": "Feature",
            "geometry": {
                "type": "Point",
                "coordinates": [23.7275, 37.9838]
            },
            "properties": {
                "timestamp": "2023-01-02 20:00:00",
                "forecast": 225.0
            }
        },
        {
            "type": "Feature",
            "geometry": {
                "type": "Point",
                "coordinates": [23.7275, 37.9838]
            },
            "properties": {
                "timestamp": "2023-01-02 21:00:00",
                "forecast": 95.0
            }
        },
```


Figure 25 GEOJson sample of the pick-up demand forecast for specific area

4.4.5 Evaluation Metrics

The effectiveness and accuracy of the Nowcasting and Forecasting Engine (NFE) were rigorously evaluated using standard metrics in time series forecasting. These evaluations were essential to ensure that the NFE meets the required performance standards and delivers reliable predictions for both traffic conditions and parcel demand. The following subsections outline the evaluation methodology and the results obtained from the testing phase.

4.4.6 Evaluation Methodology

To evaluate the performance of the NFE, we utilized a set of widely accepted statistical metrics. These metrics provide a comprehensive assessment of the prediction accuracy and the robustness of the forecasting models. The primary metrics used for evaluation were:

- **Mean Absolute Error (MAE):** Measures the average magnitude of the errors in a set of predictions, without considering their direction. It provides a straightforward measure of forecast accuracy.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}|$$

where:
$\hat{y} - predicted\ value\ of\ y$

- **Mean Squared Error (MSE):** Measures the average of the squares of the errors. It is more sensitive to larger errors, thus penalizing them more than smaller errors.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y})^2$$

- **Root Mean Squared Error (RMSE):** The square root of the mean squared error, providing a measure of the standard deviation of the prediction errors. It is useful for understanding the spread of the errors.

$$RMSE = \sqrt{MSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y})^2}$$

- **Mean Absolute Percentage Error (MAPE):** Measures the average absolute percentage error of the predictions, providing a relative measure of the prediction accuracy as a percentage.

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}|\frac{\hat{y}_i - y_i}{y_i}| \times 100$$

These metrics are calculated based on a test dataset, for both traffic and parcel demand data, to ensure a comprehensive evaluation of the NFE's performance across different scenarios.

4.4.7 Evaluation Results

The NFE's performance is assessed using the aforementioned metrics, and the results are used for model evaluation, performance and model fine-tuning. These results reflect the engine's accuracy in predicting traffic conditions over the evaluated period.

Table 6 Evaluation result on traffic forecasting

Metric	Value
Mean Absolute Error (MAE)	4.7226
Mean Squared Error (MSE)	35.3543
Root Mean Squared Error (RMSE)	5.9459
Mean Absolute Percentage Error (MAPE)	8.78%

The evaluation results are able to indicate if the NFE achieves a high level of accuracy in its predictions.

The MAE of 4.7226 and RMSE of 5.9459 demonstrate that the engine can consistently provide precise forecasts, minimizing the average prediction errors. The MAPE of 8.78% further confirms that the forecast errors are relatively small in comparison to the actual values, highlighting the reliability of the NFE for both traffic and logistics demand forecasting.

Overall, these evaluation results validate the effectiveness of the Nowcasting and Forecasting Engine in providing accurate and actionable insights, supporting the operational and strategic decision-making processes of city authorities and Logistics Service Providers.

4.5 Data Output and Integration

After generating the forecasts, the NFE outputs the predicted data through an API to the LaaS Backend. This integration enables the visualization of forecasts by the Digital Control Tower and the LaaS Marketplace UI. These interfaces allow city authorities and logistics service providers to access and utilize the forecasted data for decision-making and operational planning.

Additionally, the forecasted data is intended to be used by the Dynamic Pricing component in a later stage of the project. By incorporating real-time demand and traffic predictions, the dynamic pricing mechanism can adjust service prices in response to current conditions, optimizing both cost-efficiency and service delivery.

In summary, the methodology of the Nowcasting and Forecasting Engine is a comprehensive process that integrates data acquisition, preprocessing, model training, and real-time forecasting. By leveraging advanced predictive models and continuously updating them with the latest data, the NFE provides valuable insights for traffic management and logistics optimization, contributing to the overall efficiency and sustainability of urban logistics ecosystems.

4.5.1 Interaction between Components

The NFE interacts mainly with specific components within the GREEN-LOG Project:

1. **GREEN-LOG Dataspace:**
   - For historical data from traffic management systems and logistics service records.
   - For near real-time data streams from traffic sensors logistics management systems.
2. **LaaS Marketplace backend:**
   - To push the outputs of the NFE for production of visualizations and reports for traffic forecasts and logistics demand.

Potentially there could be an interaction with two additional modules of the GREEN-LOG platform which are the Smart Pricing Module (see chapter 3) and the Dynamic (re)routing service (see chapter 6), but this is an option still explored and will be implemented in a later phase of the project.

4.6 Implementation and Evaluation

The functionality of the NFE remains unchanged regardless of the specific use-case that it will be implemented, providing consistently output that are later on visualized in the GREEN-LOG LaaS Marketplace and the DCT. The NFE will be initially implemented in Athens and Oxfordshire, addressing both traffic forecasting and parcel demand forecasting objectives in Athens, and focusing on parcel demand forecasting in Oxfordshire. Following this initial implementation, we plan to extend at least one of the two objectives to all Living Labs (LLs) in Flanders, Barcelona, Athens, and Oxfordshire, excluding Ispra as it does not derive added value from the NFE in its current use-case. The implementation will be meticulously planned and tailored to meet the specific needs and requirements of each use-case.

4.6.1 Athens LL

In Athens, the NFE will serve dual purposes:

1. **Traffic Forecasting:**

    - The NFE will utilize historical and near real-time traffic data to predict road network conditions, focusing on improving the safe movement of drivers.
    - Data from various traffic sensors across the city will be collected and processed to provide accurate traffic forecasts, which will be visualized in the Digital Control Tower interface.

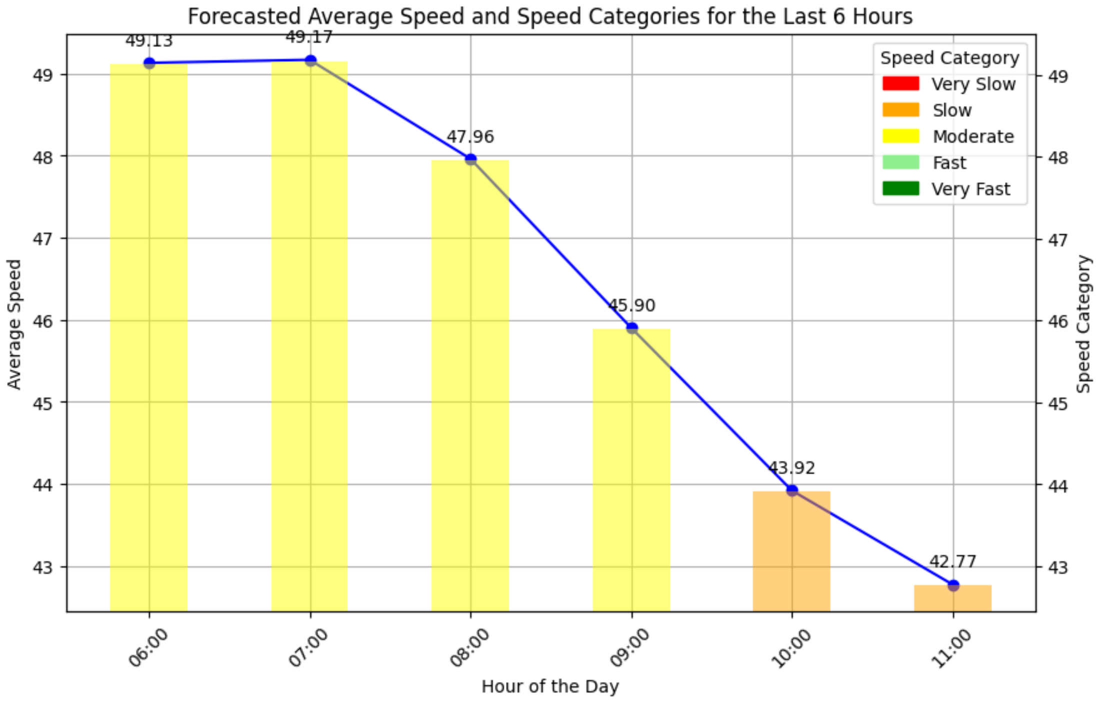


Figure 26 Sample output of the traffic forecast service of the NFE.

2. **Parcel Demand Forecasting:**

    - The NFE will predict parcel pickup demand, enabling Logistics Service Providers (LSPs) to optimize resource allocation and delivery routes.
    - Historical parcel data and real-time demand inputs will be integrated to forecast demand accurately, aiding in efficient and cost-effective deliveries.

This dual implementation aims to enhance urban mobility and logistics efficiency in Athens, addressing the specific needs of the Living Lab.

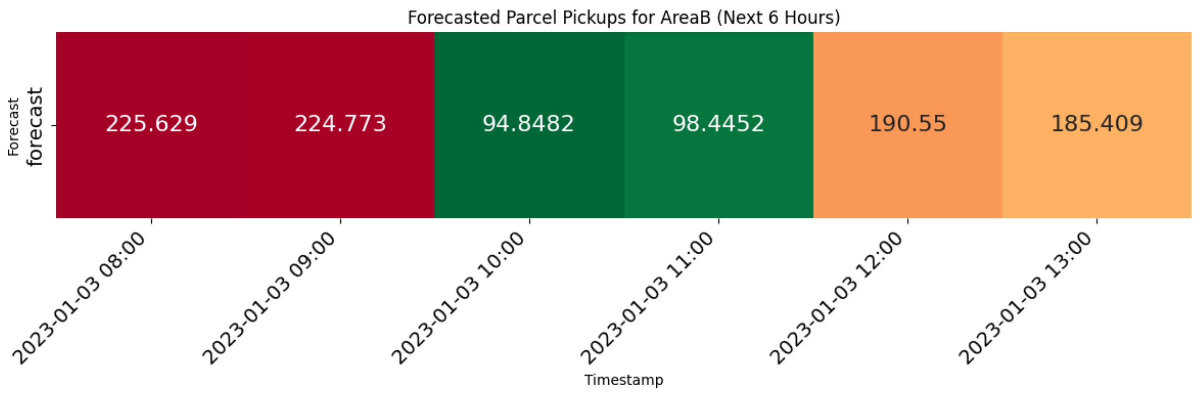


Figure 27 Sample Visualization of Predicted pickup demand for specific area

4.6.2 Implementation in Oxfordshire

In Oxfordshire, the initial focus will be on:

- **Parcel Demand Forecasting:**

    - The NFE will predict parcel pickup demand for LSPs operating in the region.
    - The forecasts will be used to optimize delivery schedules and routes, improving overall delivery efficiency.

By focusing on parcel demand forecasting, the NFE will contribute to more efficient logistics operations in Oxfordshire, addressing the specific needs identified in the Living Lab.

4.6.3 Future Implementations in Other Living Labs

Based on the requirements of the Living Labs where the NFE has applicability, the following implementations are planned:

- **Barcelona LL:**
    - **Objective:** Efficient and cost-effective deliveries.
    - The NFE will be used to predict traffic conditions and parcel demand, aiding in optimizing delivery operations and reducing costs.

- **Flanders LL:**
    - **Objective:** Increased efficiency for deliveries.

- The NFE will focus on predicting parcel demand and optimizing delivery routes, improving the overall efficiency of logistics operations.

- **Athens LL:**
    - **Objective:** Safe movement of drivers.
    - Continued use of the NFE for traffic forecasting to enhance road safety and manage urban mobility effectively.

In conclusion, the implementation and evaluation of the NFE are designed to meet the diverse needs of various Living Labs, ensuring enhanced operational efficiency and strategic decision-making capabilities for both city authorities and Logistics Service Providers. By leveraging advanced predictive models and continuously updating them with the latest data, the NFE contributes significantly to the overall efficiency and sustainability of urban logistics ecosystems.

5. Dynamic (re)routing service

5.1 Description of components

5.1.1 Problem Description

Once the freight is allocated to the Micro-Consolidation Centres (MCCs), a fleet of cargo bikes is tasked with collecting the parcels and executing the last-mile delivery. Each delivery point is linked to a specific parcel and a designated time window within which the delivery must occur. Each vehicle is defined by its capacity, indicating the maximum number of parcels it can transport in a single trip. Due to the capacity of the cargo bikes, each vehicle may be required to undertake multiple trips within a workday to fulfil the delivery demand.

The overarching objective is to determine the trips for each vehicle, ensuring compliance with capacity and time-window constraints, while minimizing the total travel distance. This problem description aligns with the Multi-Trip Vehicle Routing Problem with Time Windows (MTVRPTW) [REF-03][REF-03]. Several approaches to solving the MTVRPTW have been proposed in the literature, that employ either metaheuristics [REF-04][REF-05], [REF-06] or exact methods [REF-06][REF-07].

The problem specifics presented above directly align with the requirements of the Oxfordshire case. In the case of Barcelona, the delivery of specific parcels is preassigned to each delivery person, reducing the overall problem to multiple Travelling Salesman Problems with Time Windows (TSPTW) [REF-08]. Hence, there is no need to optimize the allocation of the parcels among the employees performing the last-mile delivery.

It is possible that the (re)routing service succeeds the stage of MCC deployment (see *D2.1 - Services for orchestrated deployment of new delivery methods V1* for more details). The optimization engine includes an internal orchestrator, which ensures the interconnection of the stages of optimization. Therefore, if the (re)routing service is the second stage of an optimization problem, the solution of the first stage is converted into a predefined format which is applicable on the (re)routing algorithms. An indicative example of the `.txt` file which is provided to the (re)routing service is demonstrated below:

```
ID_of_route

NUMBER CLIENTS
4

VEHICLE
NUMBER          CAPACITY
10              1500

VEHICLE RELEASE
0       420
1       420
2       420
```

```
CUSTOMER
CUST NO.DEMAND1 READY TIME      DUE DATESERVICE TIME
0               0               420             600             0
1               162             520             973             3
2               462             575             1008            3
3               299             528             1013            3
4               267             573             1029            3

TRAVEL TIMES
0       0       3       3       14      6
1       3       0       1       14      4
2       3       1       0       14      3
3       14      14      14      0       14
4       6       4       3       14      0
```

Example of the .txt file which provides input to the (re)routing service.

In return, the output of (re)routing is provided to the internal orchestrator, which is converted into the output json file (see 5.3.2 for more details). It is possible that the result of (re)routing modifies the MCC deployment schedule of the first stage of optimization. For example, if the indicated time window of a mobile depot (i.e., the Micro Delivery Hub in the use-cases of Oxfordshire and Athens) is not adequate for all operations to be completed in time, this will be extended to ensure the feasibility of the daily plan.

The algorithms of the (re)routing service are the same for the Day-to-Day Optimization and the Event-Triggered Optimization. The difference lies in the input txt file which is provided by the internal orchestrator:

- If a new (pickup) request has triggered the (re)routing service, then the algorithms are implemented for all previously scheduled vehicles. For each vehicle, the input data (i.e., the `NUMBER_CLIENTS` parameter of the txt file and the respective travel times) will include the assigned clients of the previous plan, added with the new request. The new request will be integrated into the vehicle which implies the route of the minimum travel time (or any other objective of interest).
- If a scheduled vehicle is unavailable, all assigned operations (pickups or deliveries) are considered as new requests. If the number of unserved operations is large and there are vehicles which remain idle, the previous routing plan is assigned to the available vehicles; otherwise, the unserved operations are integrated into the scheduled routes of the active vehicles.
- If a scheduled operation is cancelled, then the respective route will be re-computed, neglecting the respective operation.

For all the above scenarios, the starting point (i.e., node 0 of the txt file) will be the current location of the vehicle when the re-routing was triggered.

The source Python code for the implementation of the designed algorithms is available upon request.[5]

5.1.2 Interaction between stages

As mentioned above, the optimization engine consists of two stages, if the MCC deployment scheduling applies on the studied use-case. Therefore, the methods

[5] https://github.com/aueb-greenlog/opt-service

which address to each stage should interact: the outcome of MCC deployment should be provided to the (re)routing service, then the routing plan should modify the MCC deployment schedule, if additional delays appear. The following scheme demonstrates the internal flow diagram of exchange of knowledge between the optimization stages for the Day-to-Day optimization module:

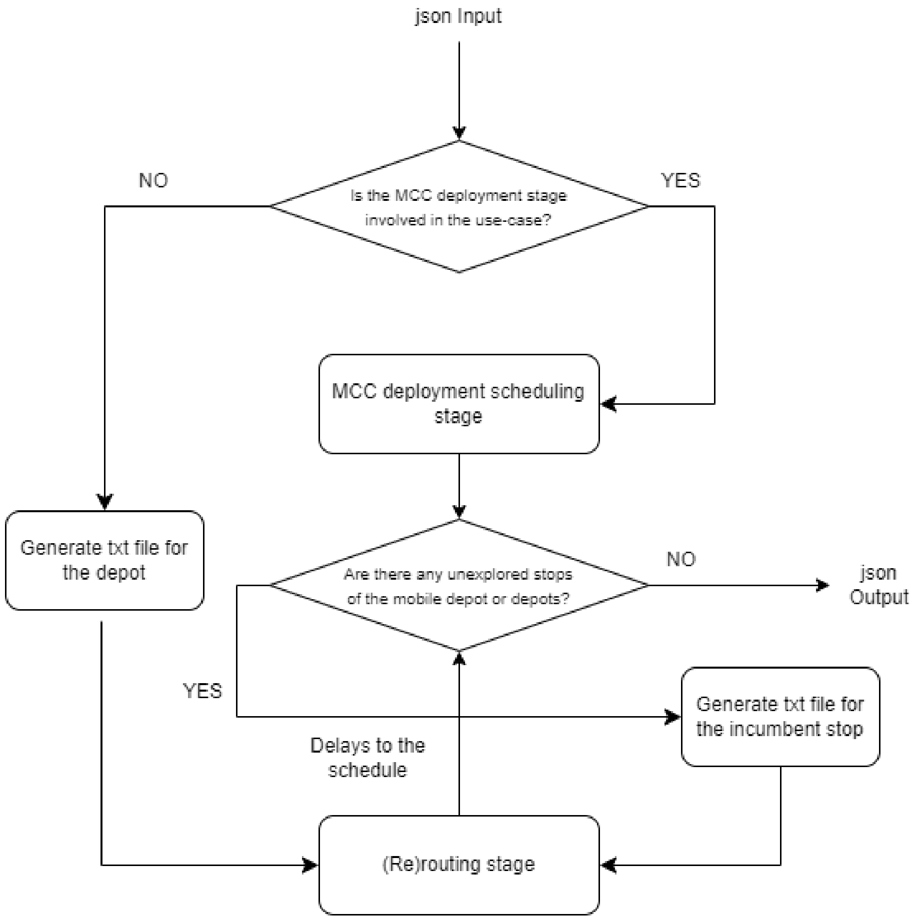


Figure 28 Flow diagram of internal interactions in the Day-to-Day optimization module.

If the MCC deployment stage is involved, then for each stop of the mobile depot, a different txt file is generated, assuming that the starting point (i.e., node 0) is the location of the incumbent stop. During the local routing, the mobile depot remains in the stop, to ensure that the remaining daily schedule will not be significantly disrupted. The waiting time of the mobile depot is determined by the result of the first stage (see D2.1 for more details); however, it is possible that the estimated waiting time is not sufficient to cover the assigned demand. If this is the case, then the waiting time of the mobile depot will be extended, hence the remaining daily schedule will be modified.

If the MCC deployment stage does not apply (e.g., in All-the-Way use-cases in Barcelona), then a single txt file, assuming that node 0 is the central depot of the LSP will be directly generated and the (re)routing stage will be instantly triggered.

5.1.3 Event-triggered optimization module

It is possible that urgent events will disrupt the implemented daily schedule. The Event-triggered optimization module is assigned to modify provided routing schedules to handle such disruptions. In this deliverable, we consider two types of events which trigger the module: a) the appearance of new pickup (and probably delivery) requests during the day, and b) the unexpected malfunction of a vehicle. For both of these cases, the Event-triggered optimization module will attempt to compute new routes for the vehicles of last-mile operations. It is noted that the schedule of mobile depots will not be affected by the module, unless new requests or additional trips of the scheduled vehicles require additional waiting time of the mobile depot. In any case, the sequence of stops will remain intact.

a. New requests:

   If any new requests appear during the execution of the daily plan, then these will be assigned to the nearest (in travel time) of the remaining stops. The time of event occurrence will determine which orders are considered as "*completed*" or "*pending*" (e.g., if new requests are submitted at 12:30, then all orders which are connected with stops which are visited at any later time instance, including the new ones, are set as "*pending*").

b. Unavailable vehicles:

   If any of the vehicles which has been assigned to conduct last-mile operations is out-of-order, then the assigned parcels should be picked up by a different vehicle. This procedure will extend the waiting time of the mobile depot in the stop which relates to the incident (i.e., where the mobile depot was when the malfunction was detected); the scale of delay depends on the availability of vehicles. Therefore, if a vehicle is unavailable, then all parcels which were loaded at the time of the event will be picked up from the last location of the involved vehicle, and the release time of the vehicle will be set to a very large value, thus the routing service will never assign it to any operation.

5.1.4 Optimization method

The optimization process developed is able to tackle both the Oxfordshire and Barcelona cases, without a modification of the details of the underlying methodology. The input to the method consists of details on the delivery points, namely the demands, time windows and the travel times between them, as well as the specifics of the vehicles (distributors) to perform the last-mile delivery, such as their number and their capacity. The method's output comprises a set of workdays (one for each vehicle), each of which consists of a sequence of visits to a subset of the delivery points, as well as the depot.

The devised optimization scheme (**Algorithm 3**) is an iterative process consisting of two subprocesses. Firstly, a randomized construction heuristic (**Algorithm 1**) produces an initial solution to the problem. Subsequently, the initial solution is improved by a local search procedure (**Algorithm 2**). A local search procedure [REF-08] starts with an initial solution $\boldsymbol{s}$ and considers a subset of solutions structurally similar to $\boldsymbol{s}$. If a better solution $\boldsymbol{s'}$ is found, it substitutes $\boldsymbol{s}$ and the search is repeated, until no improving solution is found. The overall process is repeated a predefined number of times (restarts) and the best solution encountered is retrieved. In the following sections, the core components of the devised methodology are presented in detail.

Algorithm 1: *RandomConstruct* **-** Randomized Initial Solution Construction

Input: clients $\boldsymbol{C}$**,** number of available vehicles $\boldsymbol{V}$
Output: initial solution $\boldsymbol{s}$

1. $\breve{\boldsymbol{C}} \leftarrow$ create a random ordering of the clients $\boldsymbol{C}$;
2. $\boldsymbol{s} \leftarrow$ initialize a solution with $\boldsymbol{V}$ empty workdays;
3. **for** every client $\boldsymbol{c}$ in $\breve{\boldsymbol{C}}$ **do**
  3.1. insert $\boldsymbol{c}$ into its minimum cost position in the partial solution $\boldsymbol{s}$;
4. return $\boldsymbol{s}$;

Algorithm 1. Construction of Randomized Initial Solution

Algorithm 2: *VND* - Variable Neighbourhood Descent

Input: initial solution $\boldsymbol{s}$, $\boldsymbol{k}$ neighbounerhoods $\boldsymbol{N} = \{\boldsymbol{N_i}, i = 1, \dots, \boldsymbol{k}\}$
Output: improved solution $\boldsymbol{s}^*$

1. $\boldsymbol{s}^* \leftarrow \boldsymbol{s}$;
2. $\boldsymbol{i} \leftarrow \mathbf{1}$**;**
3. **while** $\boldsymbol{i} \leq \boldsymbol{k}$ **do**
  3.1. $\boldsymbol{m} \leftarrow$ find the minimum cost move in neighbourhood $\boldsymbol{N_i}$;
  3.2. **if** $\boldsymbol{m}$ improves $\boldsymbol{s}^*$ **then**
    3.2.1. $\boldsymbol{s}^* \leftarrow$ apply $\boldsymbol{m}$ to $\boldsymbol{s}^*$;
    3.2.2. $\boldsymbol{i} \leftarrow \mathbf{1}$;
  3.3. **else**
    3.3.1. $\boldsymbol{i} \leftarrow \boldsymbol{i} + \mathbf{1}$;
4. return $\boldsymbol{s}^*$;

Algorithm 2. Variable Neighbourhood Descent Algorithm

Algorithm 3: Multi-restart metaheuristic

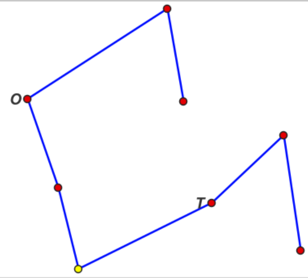
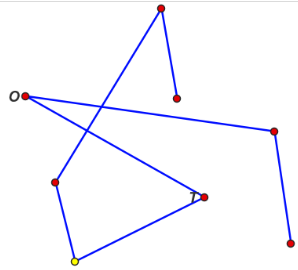
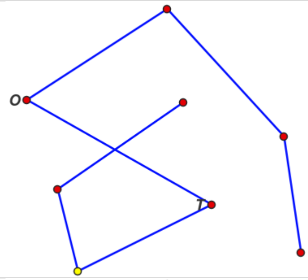
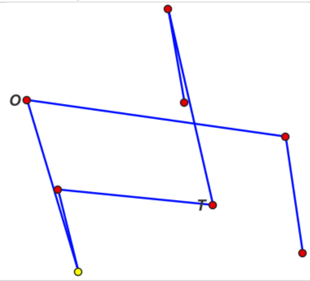
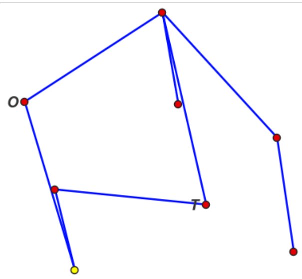
Input: clients $\boldsymbol{C}$, number of available vehicles $\boldsymbol{V}$, number of restarts $\boldsymbol{R}$, neighbourhoods $\boldsymbol{N}$
Output: best solution found $\boldsymbol{s}^*$

1. $\boldsymbol{s}^* \leftarrow$ initialize best solution;
2. $\boldsymbol{i} \leftarrow 1$**;**
3. **while** $\boldsymbol{i} \leq \boldsymbol{R}$ **do**
  3.1. $\boldsymbol{s} \leftarrow RandomConstruct(\boldsymbol{C}, \boldsymbol{V})$;
  3.2. $\boldsymbol{s} \leftarrow VND(\boldsymbol{s}, \boldsymbol{N})$**;**
  3.3. **if** $\boldsymbol{s}$ is better than $\boldsymbol{s}^*$ **then** $\boldsymbol{s}^* \leftarrow \boldsymbol{s}$;
  3.4. $\boldsymbol{i} \leftarrow \boldsymbol{i} + 1$;
4. return $\boldsymbol{s}^*$;

Algorithm 3. Multi-start heuristic component of the (re)routing service

5.1.5 Definition of Local Search Neighbourhoods

A local search neighbourhood comprises a set of moves on the current solution. Each move is uniquely described by two reference clients, namely the origin $\boldsymbol{O}$ and the target $\boldsymbol{T}$. The neighbourhood, as well as the reference clients, define in what way a move modifies the current solution. The local search neighbourhoods employed for the metaheuristic are presented below:

Move	Description	Illustration
Reference	The initial solution	
Exchange-10	$\boldsymbol{O}$ is relocated next to $\boldsymbol{T}$.	
Exchange-20	the subsequence consisting of $\boldsymbol{O}$ and its next client is relocated next to $\boldsymbol{T}$.	
Exchange-11	$\boldsymbol{O}$ is swapped with $\boldsymbol{T}$.	
Exchange-21	the subsequence consisting of $\boldsymbol{O}$ and its next client is swapped with $\boldsymbol{T}$	

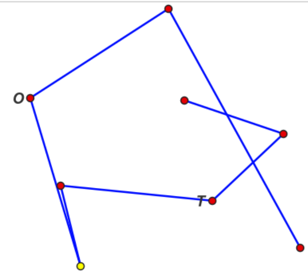
Exchange-22	the subsequence consisting of $\boldsymbol{O}$ and its next client is swapped with the subsequence consisting of $\boldsymbol{T}$ and its next client	

Table 7 Local Search Neighbourhoods

5.1.6 Definition of acceleration technique

As the scale of the problem increases, i.e. the number of clients, the evaluation of a local search neighbourhood tends to be too time-consuming. For this reason, several acceleration techniques are proposed in the literature to boost the efficiency of the search. The speed-up technique employed for the metaheuristic is defined as *Lists of Neighbours [REF-08]*[REF-08].

Let the total number of clients be $\boldsymbol{C}$. For every client $\boldsymbol{i}$, its list of neighbors $\boldsymbol{L}(\boldsymbol{i})$ comprises the other $\boldsymbol{C}-1$, $\boldsymbol{j} \neq \boldsymbol{i}$ clients, sorted in an increasing order of travel times $\boldsymbol{t_{ij}}$. The list of neighbors for each client is precomputed at the beginning of the metaheuristic. A threshold value $\boldsymbol{\theta}$ is selected, e.g. $\boldsymbol{\theta} = (\boldsymbol{C}-1)\,/\,2$, and then the only moves evaluated are the ones for which the target $\boldsymbol{T}$ belongs to the first $\boldsymbol{\theta}$ elements of $\boldsymbol{L}(\boldsymbol{O})$, where $\boldsymbol{O}$ is the origin.

5.1.7 Segments – Move evaluation

A core component of both the construction heuristic and the local improvement procedure is the move evaluation process. For each candidate move, it is required to compute the travel time of the resulting solution, as well as to check if the move leads to a violation in the time window and capacity constraints.

VRPs with time windows are generally deemed very constrained problems. The time window constraints make it difficult to move between feasible solutions, and thus to explore the solution space. The most prevalent technique utilized in the literature to tackle this challenge, is to relax the problem's constraints and penalize their violation in the objective function [REF-08]. Allowing infeasible solutions facilitates navigating the solution space, providing shortcuts between quality solutions. The minimization of the penalized objective function will lead to the elimination of the constraint violations and, consequently, to feasible solutions.

The cost of a move is thus defined by the resulting solution's travel time, time window violation and capacity violation. Accordingly, the move evaluation process requires the computation of these quantities. To address this requirement, the approach outlined in [REF-09] is implemented within the metaheuristic. This method capitalizes on the insight that any move can be conceptualized as breaking down the solution's trips into subsequences, which are thereafter combined to form new trips. To expedite the process, the desired quantities are computed for all possible subsequences identified by the current solution as a preprocessing step. Consequently, during the search process, the evaluation of any move can be conducted in constant time by concatenating the appropriate subsequences in accordance with the move.

5.2 Functional requirements

5.2.1 Use-cases

The use-cases of the shell are listed below:

- Use-case 1 (UC1): Authenticate user.
- Use-case 2 (UC2): List the available optimization services.
- Use-case 3 (UC3): Submit a new optimization job.
- Use-case 4 (UC4): Get the solution of a completed optimization job.
- Use-case 5 (UC5): Store the solution of a completed optimization job.

The listed functional requirements are also demonstrated in the image below:

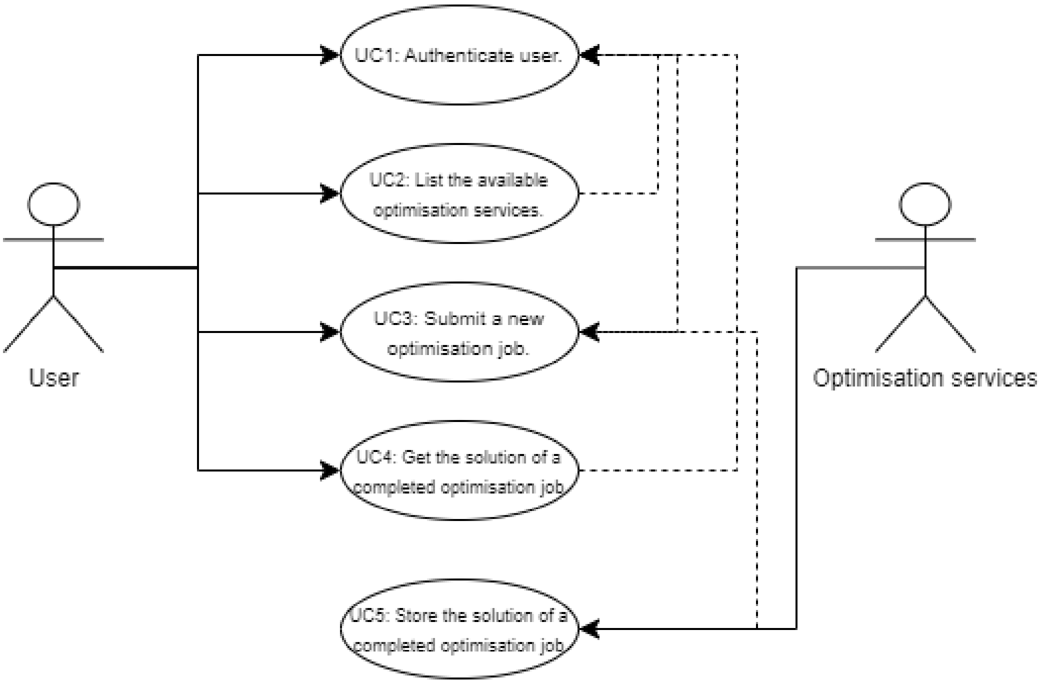


Figure 29 optEngine Functional Requirements

The following class diagram demonstrates the data requirements of *optEngine*:

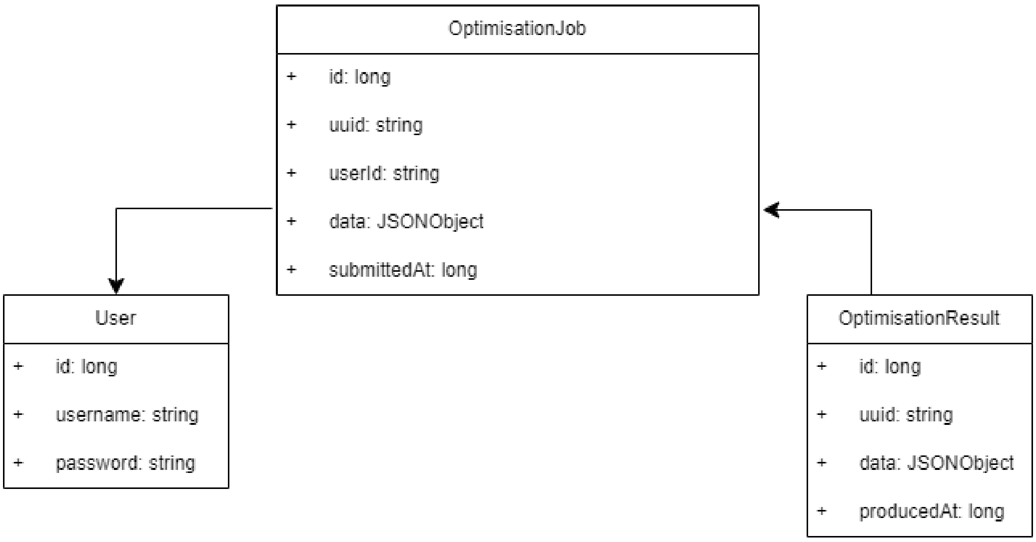


Figure 30 optEngine Data Requirements

5.2.2 Architectural View

OptEngine works as a shell around the optimization. Its architecture follows an asynchronous approach and is agnostic to optimization-specific data requirements. That is, *optEngine* receives and forwards the optimization data to the optimization service requested by the end-user, then it stores the computed solution to the Dataspace.

Optimization requests along with the respective data are received via a web API. The communication with the API requires authentication, is encrypted (https) and asynchronous, meaning that the user does not have to wait for the completion of a submitted optimization job.

Data stores within *optEngine* work in a twofold manner:

Permanent storage via a dataspace: the requests and the related data are permanently stored or retrieved and updated, when necessary, from a Dataspace. For example, the Day-to-Day optimization module stores the acquired solution to the Dataspace. If a disruption occurs during the implementation of the daily schedule, the Event-triggered optimization module retrieves the stored solution to update it.

Temporal storage via the use of queues: optimization data are stored to the Dataspace until they are retrieved from the optimization services that read these queues.

Regarding the optimization data, both the dataspace and the queues are data-agnostic following a general json schema. This allows the storage, permanent and temporal, of different data structures required from different optimization services.

The employed queues allow the asynchronous processing of an optimization job. Additionally, by being durable they ensure that when *optEngine* or an optimization service fail, the job along with data are available in the respective queue. This means that *optEngine*, upon reception of a new optimization job, forwards it to the requested optimization service via a queue. Each optimization service listens for a new optimization job to a specific queue and writes status/progress updates to another queue. Last, *optEngine* listens to (a) a queue for status/progress updates and (b) multiple queues for optimization results.

The flow of data and the architectural approach of *optEngine* are depicted below.

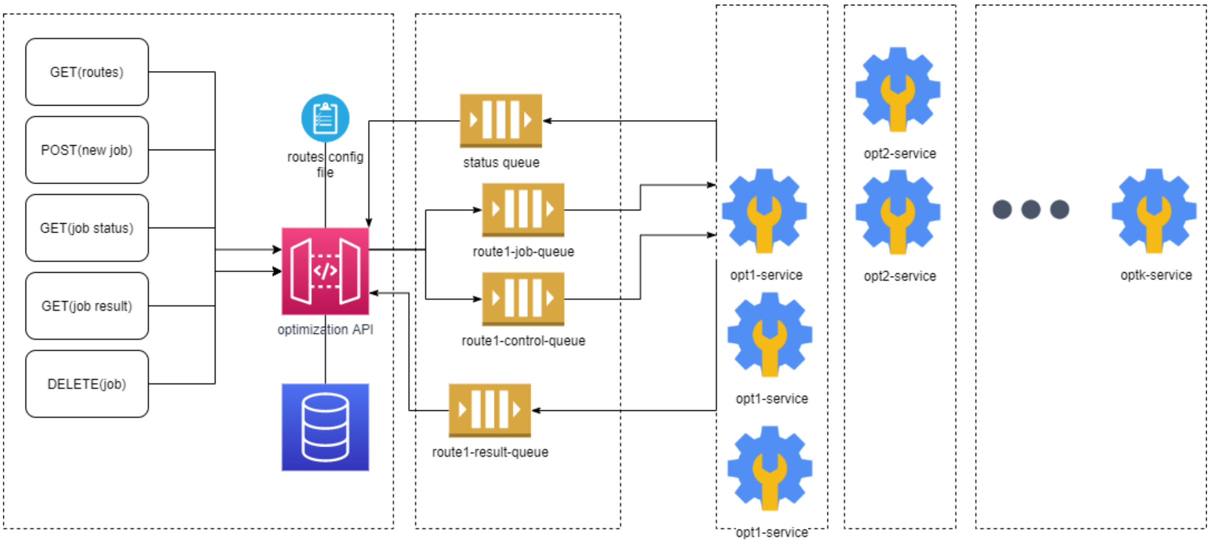


Figure 31 optEngine Data Flow

The optimization engine has been originally designed for the 2020 Horizon project FACTLOG (Grant agreement: 869951). An elaborate description of *optEngine* is presented in D5.1[6].

5.2.3 Technology Stack

The following technologies are used for the development of *optEngine*:

Java 8	
Spring-boot 2.4.5	
Springdoc openAPI 1.5.2	
Hibernate 1.0.0	
PostgreSQL 11	

6 https://www.factlog.eu/about/deliverables/

RabbitMQ 3.8.16	
Docker 18.09.7	

Figure 32 optEngine Data Stack

5.3 Implementation and evaluation

5.3.1 Athens LL

The optimization engine will be implemented in Athens for both indicated use-cases. The first one considers a single vehicle (which is the white-label van), starting from the MCC location in Klafthmonos square, and completing all deliveries. It is possible that the van will conduct multiple trips if the total weight or volume of the parcels exceed the capacity of the vehicle. This use-case indicates a straightforward implementation of the (re)routing service, skipping the MCC deployment stage.

For the second use-case, the white-label van is used as a mobile depot, which will follow a route of stops during the daily shift. The schedule of the mobile depot is determined by the MCC deployment stage; hence it is out of the scope of this deliverable. The customers are expected to receive a message, so that they are informed that their parcels are available at the mobile depot nearby – then, they will collect their parcels within the indicated time window. Consequently, the (re)routing service is not expected to apply on the second use-case.

For the involved use-case, we consider a dataset with the following attributes:

Use-case 1:

- Area of deployment: Historical centre of Athens
- Location of MCC/Depot: Klafthmonos square
- Number of deliveries: 50 – a larger number cannot be addressed during the operating hours of LSPs
- Number of mobile depots: 0 (i.e., direct routing of the white-label van)
- Number of last-mile vehicles: 1 van for the first use-case (i.e., the white-label van)
- Capacities of vehicles: Estimated for realistic assumptions per type of vehicle
- Candidate stops: 0 – the mobile depot is not involved.
- Travel times: Obtained by OpenStreetMap, using the OSMNX [REF-11], for a radius of 3000 metres from the centre of Athens

The output of the first use-case is a single route of the white-label van, as summarised below:

Vehicle ID (Operator)	Daily route

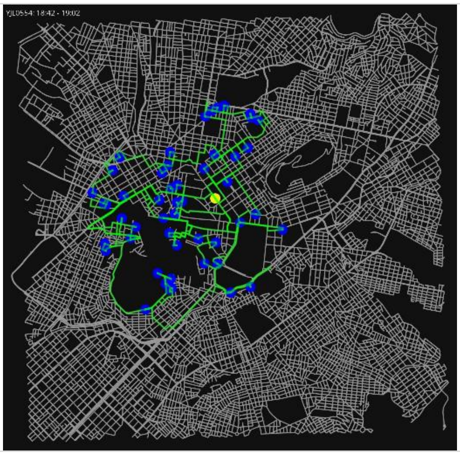
YJL0554 (White-label)	

Table 8 (Re)routing results for the first use-case in Athens

The total weight and volume of the parcels plays a significant role in the complexity of the problem. To evaluate the performance of the proposed method, we generate delivery requests which exceed the capacity of the mobile depot, meaning that at least one reload should occur during the daily shift.

The blue nodes indicate demand points, and the yellow node indicate the location of the MCC.

5.3.2 Barcelona LL

In Barcelona, we examine the combination of last-mile delivery operations using public transport with regular routing. The use-cases are divided into “All-the-way” routing using the fleet of VANaPED, multimodal delivery using the rail network of FGC, and a combination of these two. For this stage of implementation, we examined the third scenario, as it is inclusive for all applied cases.

Using public transport is expected to imply additional travel time: each carrier should either walk or ride a bike to the station in Plaça de Catalunya, then use the next trip to the area of interest, and finally complete the deliveries on foot. Nevertheless, there are several reasons why examining the multimodal deliveries may bring forward potential profits for the operators:

- no resources are consumed: the fleet of vehicles remain available for other delivery requests.
- accessibility to distant districts: the elevation of the city does not ensure accessibility of all districts by bike. Also, distant areas (e.g., in Terrassa) are not easily accessible by electric vehicles, as the recharging requirements will be higher.

Given these, we implement the (re)routing service sequentially: at first, we address as many deliveries as possible using the rail network, then we plan the routing of the remaining demand using the fleet of vehicles.

For the assignment of carriers to rail trips, we use the MCC deployment stage, assuming that each carrier is a mobile depot – more details are elaborated in D2.1. Then, for each station, we generate a txt file for each carrier, considering only singular fleets of vehicles (consisted of the carriers themselves).

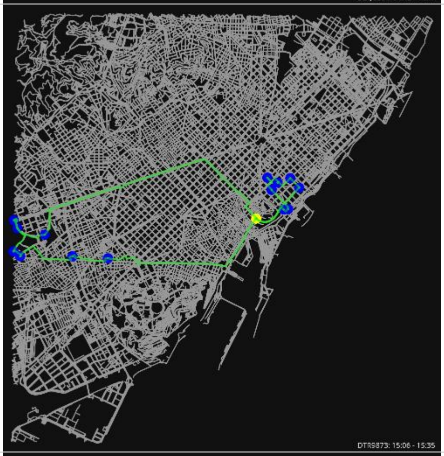
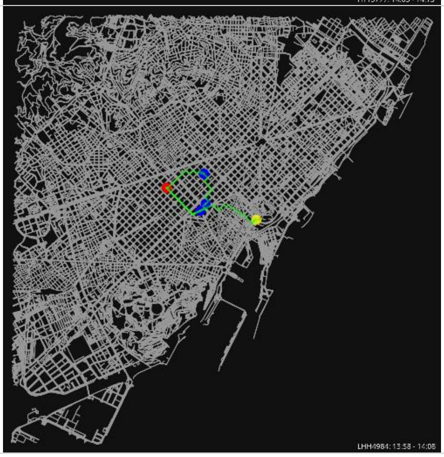
For the remaining demand, we consider a direct implementation of the (re)routing service, using the depot of Vanapedal as the starting point of the vehicles (which are cargo-bikes).

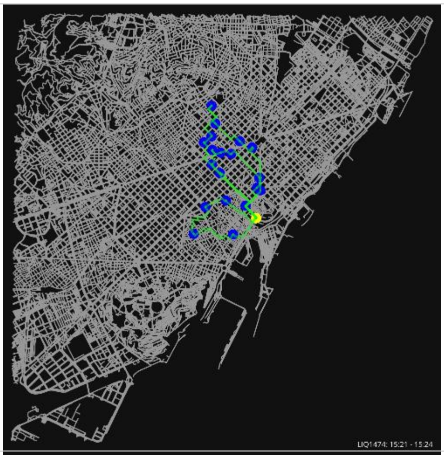
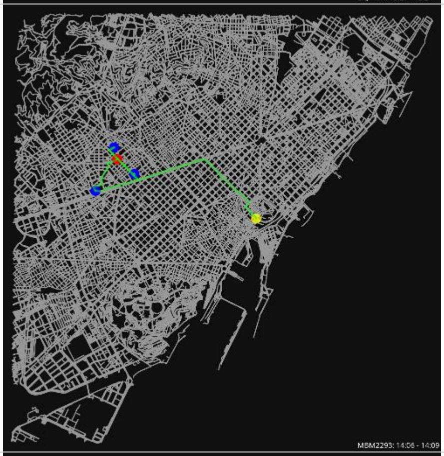
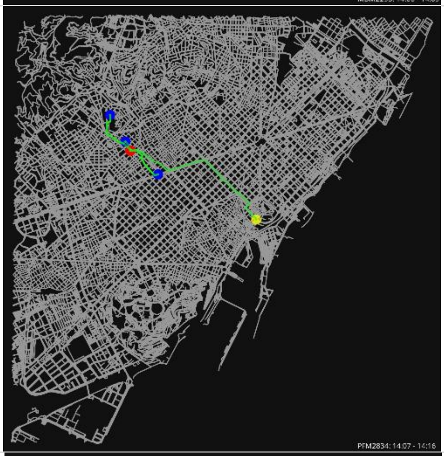
For this stage of implementation, we consider the hybrid scenario of multimodal delivery planning and direct routing from the depot. The applied instance has the following features:

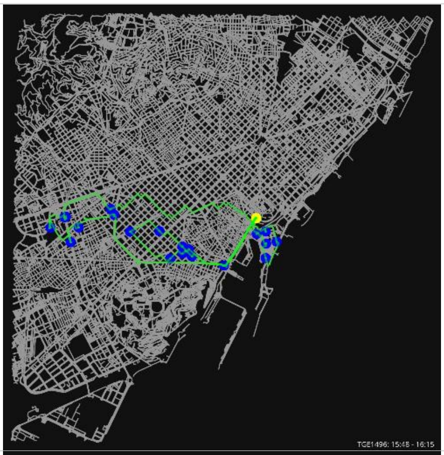
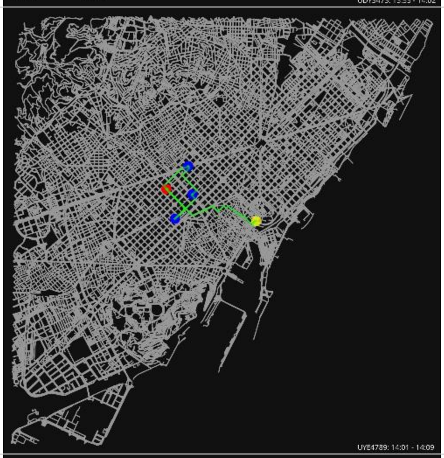
- Area of deployment: Centre of Barcelona – Sant Gervasi and adjacent districts
- Location of MCC/Depot: Depot of VANaPED
- Number of deliveries: 100 – randomly assigned to nodes within the area of interest
- Number of mobile depots: 15 – we assume that the carriers are used as mobile depots to apply the MCC deployment stage
- Number of last-mile vehicles: 15 carriers and 5 cargo-bikes
- Capacities of vehicles: Indicated by the stakeholders of the Living Lab
- Candidate stops: 8 stations of Line 6 of FGC
- Travel times: Obtained by OpenStreetMap, using the OSMNX [REF-11], for a radius of 6000 metres from the centre of Barcelona.

The routing plan is computed sequentially: at first, all parcels which are within a 10-minute walk catchment area from a candidate station can be potentially addressed by carriers; then all remaining parcels are assigned to the cargo-bikes. A summary of the routes is illustrated in Table 9.

Vehicle ID (type)	Daily route
CES2611 (cargo-bike)	

DEQ6804 (carrier)	
DTR9873 (cargo-bike)	
HYY5777 (carrier)	
LHH4984 (carrier)	

LIQ1474 (cargo-bike)		
MBM2293 (carrier)		
PFM2834 (carrier)		
PMV1083 (carrier)		

TGE1496 (cargo-bike)	
UDY3473 (carrier)	
UYE4789 (carrier)	
VRH9503 (carrier)	

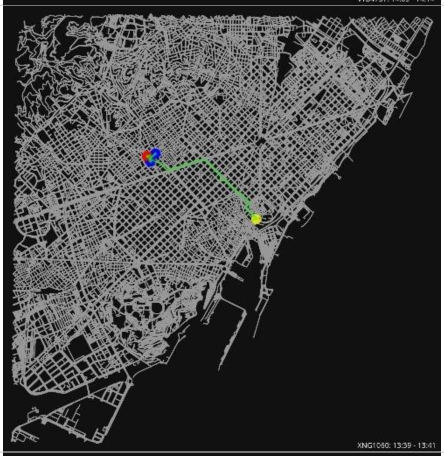
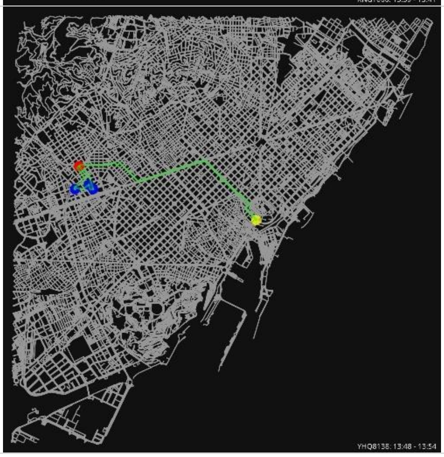
VYB4737 (carrier)	
XNG1060 (carrier)	
YHQ8138 (carrier)	

Table 9 (Re)routing results in Barcelona

The blue nodes indicate delivery points, the red nodes are stations of Line 6 and the yellow node is the depot of VANaPED. We notice that most of carriers are assigned to multimodal trips, as the exploitation of public transport is prioritized to minimise the usage of fleet resources. We also note that almost all orders within Sant Gervasi are conducted by carriers, as the 10-minute walk catchment area from the FGC stations covers the district. On the contrary, the locations outside Sant Gervasi are covered by the fleet of cargo-bikes.

The major metrics, as printed in the json output file, are the following ones:

```
"general": {
```

```
    "uuid": "9ee2f69e-be6a-43a8-968e-fc966db89ba2",
    "userName": "AUEB_optEngine",
    "producedAt": 1714606702,
    "KPIs": {
        "numRoutes": 16,
        "totalDistance": 179.61,
        "parcelsPerRoute": 2.4,
        "totalEmissions": 0.0,
        "totalDelay": 0.0
    }
```


The implementation will be extended to the wider metropolitan area of Barcelona. In particular, the district of Terrassa, outside the center of the city, will also be involved, following a similar approach to the one presented in this section. The involvement of Terrassa will allow us to examine the potential of the integration of public transport into freight urban operations, as a mechanism which unifies entire metropolitan areas.

5.3.3 Oxfordshire LL

In Oxfordshire, we examine the potential benefits of the usage of a mobile depot (namely MDH) to minimise the travel times of last-mile operations. The scope of (re)routing service is significantly extended compared to the aforementioned applications in Living Labs: multiple operations should be completed, meaning that there are pickup and delivery requests (or even both of them for some parcels), and new requests may appear during the day, forcing the Event-triggered module to be urgently activated.

For the Day-to-Day optimization module, each stop of the MDH generates a new txt file, which is provided to the (re)routing service. Then, a local routing plan will assign pickup and delivery requests to the fleet of cargo-bikes of Pedal & Post. The routing problems are subject to precedence constraints, since there are parcels which should be picked up and delivered. Now, these precedencies are integrated into the MDH schedule of the first stage, meaning that each parcel which requires both operations is assigned to preceding stops of the MDH. In the upcoming updates of the method, these operations will be eligibly assigned to the same stop and the precedence constraints will be applied on the (re)routing service.

For the Event-triggered module, the new pickup (and delivery) requests are assigned to the nearest of the remaining stops of the MDH (given that the daily schedule before the time of event occurrence is executed as planned). Then, the (re)routing service is re-implemented from scratch, meaning that completely new routing plans will be computed for each vehicle. Since the duration of the algorithm is no longer than a minimal number of seconds, the new plan will be provided to the LSPs and the other tasks of the platform in time.

Day-to-Day optimization: The first iteration of implementation in Oxfordshire considers the following dataset:

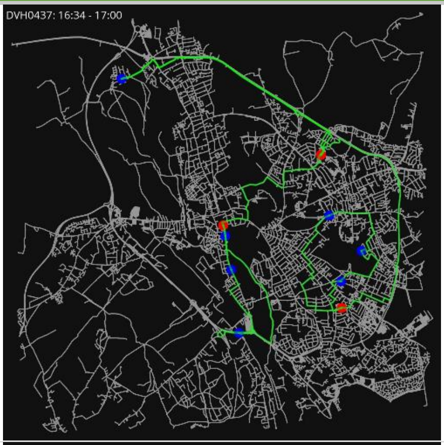
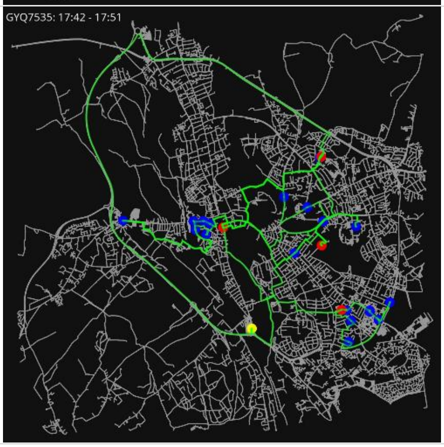
- Area of deployment: Oxford
- Location of MCC/Depot: Redbridge Park-and-Ride
- Number of deliveries: 100 – picked up from the MCC (75% probability) or a random node in the network (25%), delivered to the MCC (25% probability only if not picked up

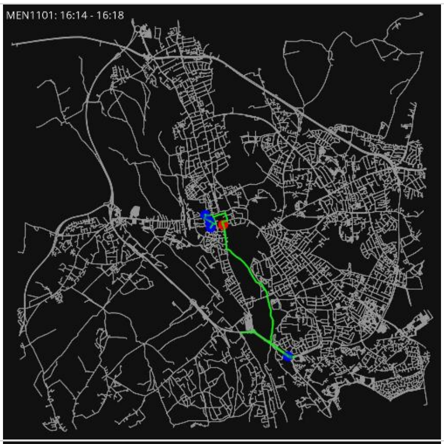
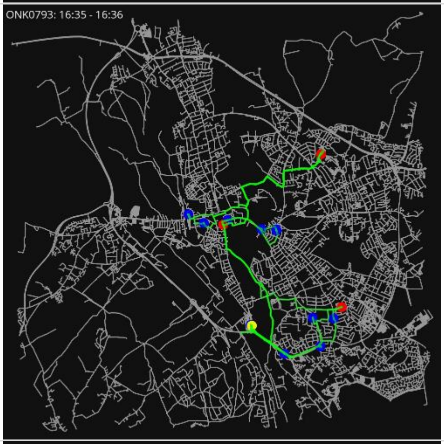
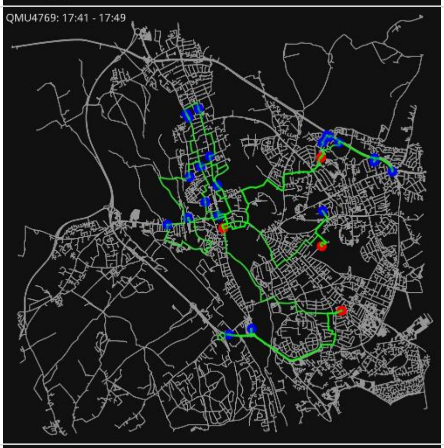
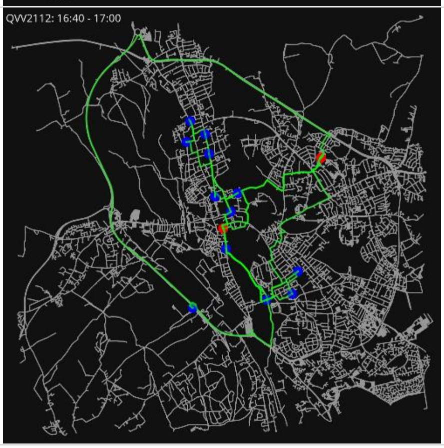
from the MCC) or a random node in the network (75% probability if not picked up from the MCC, 100% probability if picked up from the MCC).

- Number of mobile depots: 1
- Number of last-mile vehicles: 10 cargo-bikes
- Capacities of vehicles: Indicated by the stakeholders of the Living Lab
- Candidate stops: 10
- Travel times: Partially obtained by AIMSUN, completed by OpenStreetMap using the OSMNX [REF-11], for a radius of 3000 metres from the centre of Oxford

We note that each delivery request has a restricted time window. The (re)routing service allows violations of this time window, since it is possible that the daily demand cannot be feasibly covered within the indicated time windows for all parcels; nonetheless, a possible violation will be penalised in the objective of the algorithm to ensure that it will be avoided as much as possible. Also, as indicated by the stakeholders, at least one reload is expected to take place during the shift, therefore the total weight and volume of the parcels will exceed the capacity of the mobile depot.

The results of the last-mile operating fleet (the results for the MDH are presented in D2.1) are as it follows:

Vehicle ID	Daily route
DVH0437	
GYQ7535	

MEN1101	
ONK0793	
QMU4769	
QVV2112	

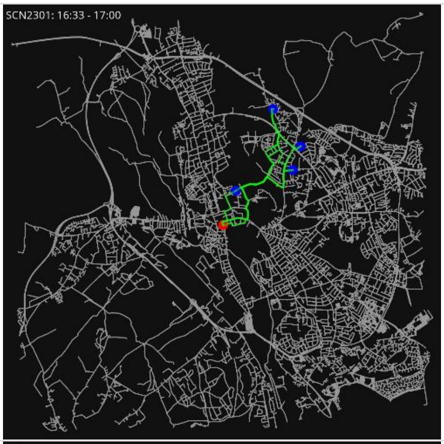
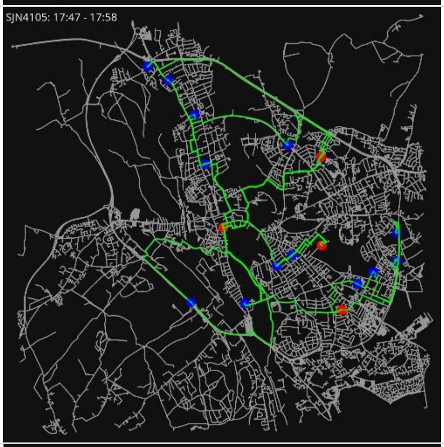
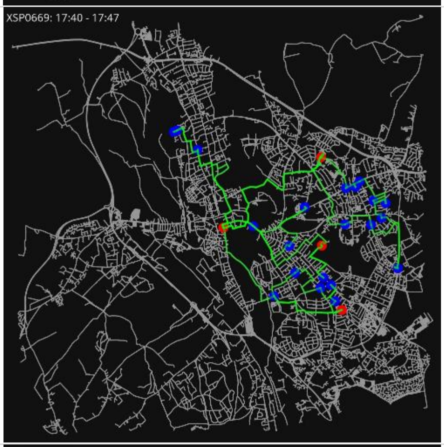
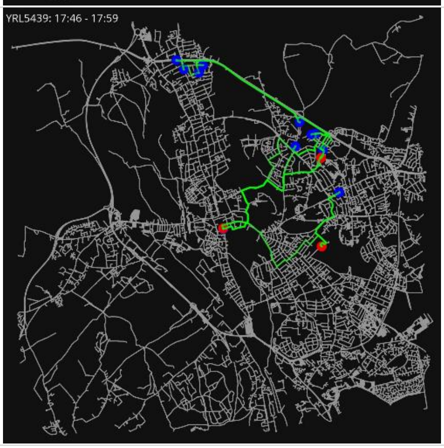
SCN2301	
SJN4105	
XSP0669	
YRL5439	

Table 10 (Re)routing results of Day-to-Day Optimization in Oxford

The demand nodes are coloured in blue, the red nodes indicate the stop of the MDH and the yellow node is the Redbridge Park-and-Ride, where the MCC is located. As the local authorities are interested in examining the installation of MCC in different locations within the city, the following iterations of the method will consider alternative options, comparing the efficiency of each one of them.

The summary of the json output file is the following one:


```
"general": {
        "uuid": "5d9a85b3-1976-407c-a7d6-2135c7a508b6",
        "userName": "AUEB_optEngine",
        "producedAt": 1714658190,
        "KPIs": {
            "numRoutes": 11,
            "totalDistance": 388.54,
            "parcelsPerRoute": 10.0,
            "totalEmissions": 0.0,
            "totalDelay": 0.0
        }
```

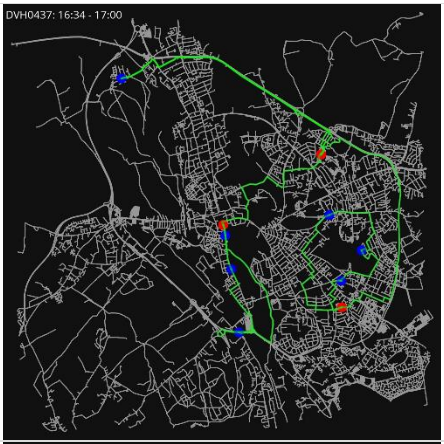
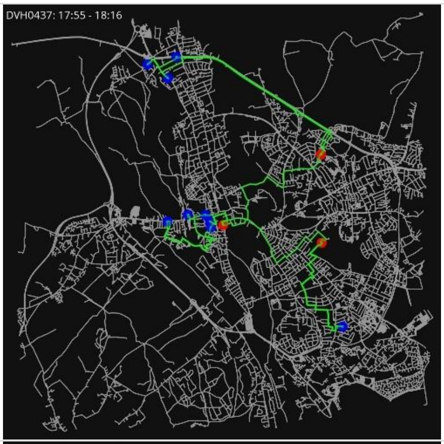
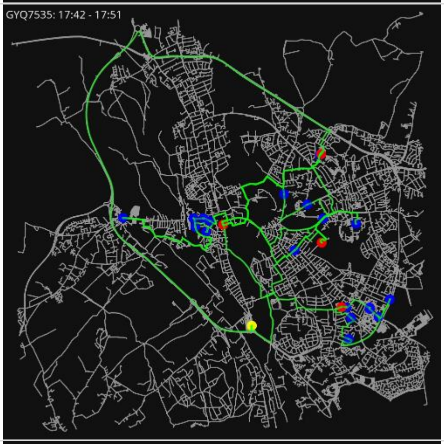
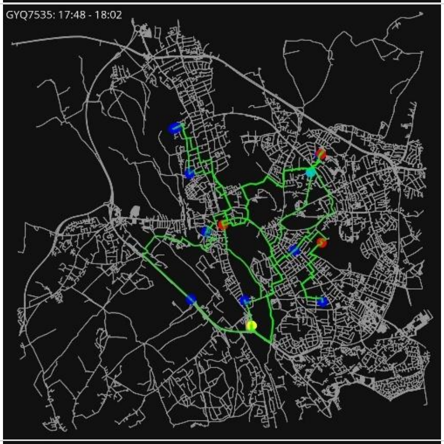
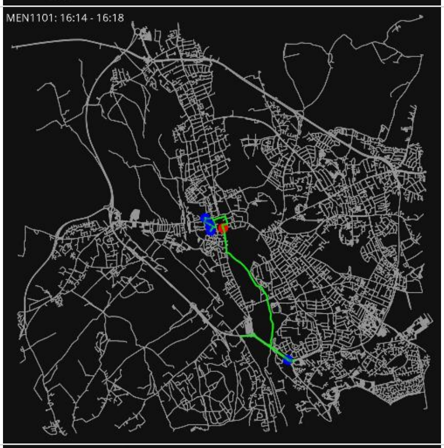
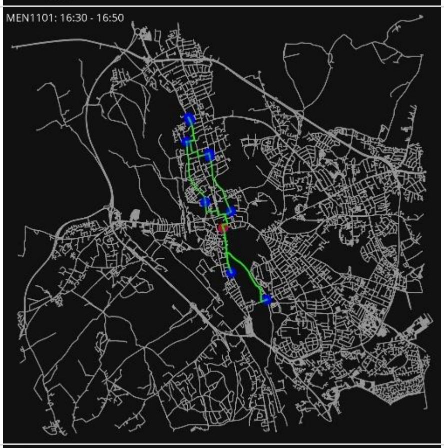
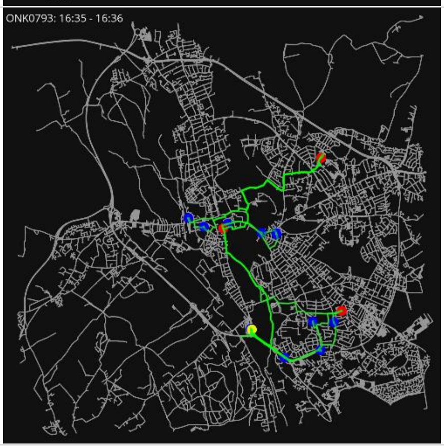
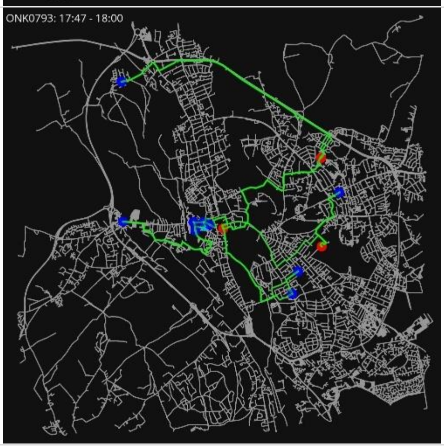

It is noted that the number of routes does not include the route of the MDH, as it is not connected with the completion of any request. Therefore, the average number of parcel of routes (i.e., 10.0) is computed by the division of the total number of parcels (100) with the number of last-mile routes (10).

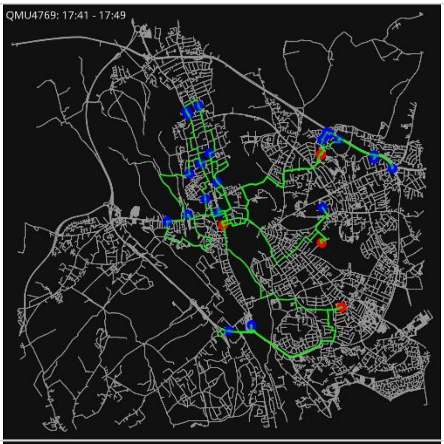
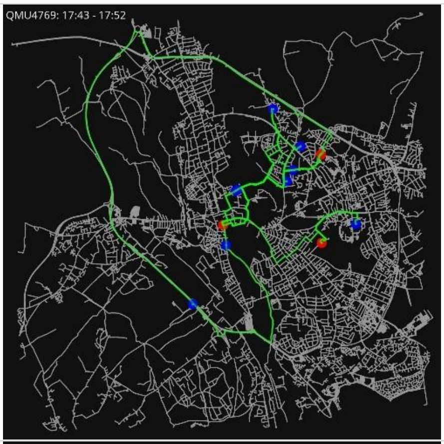
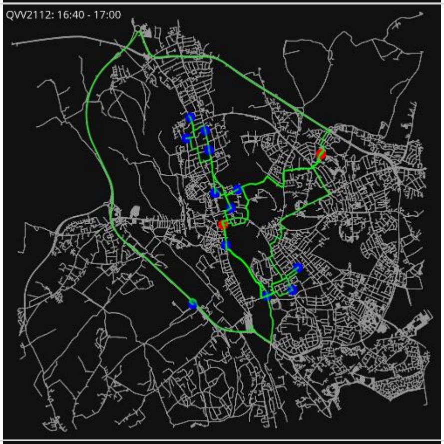
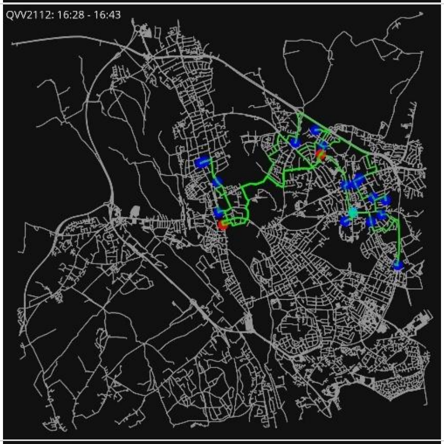
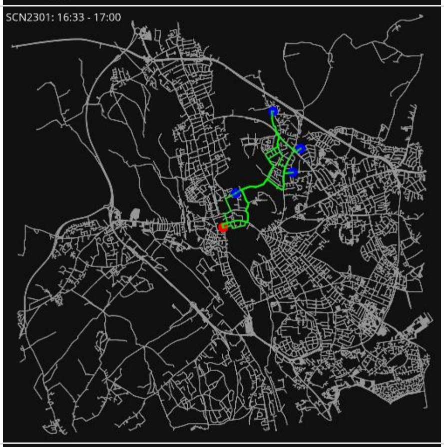
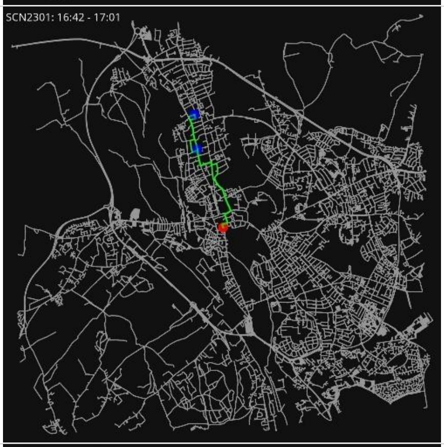
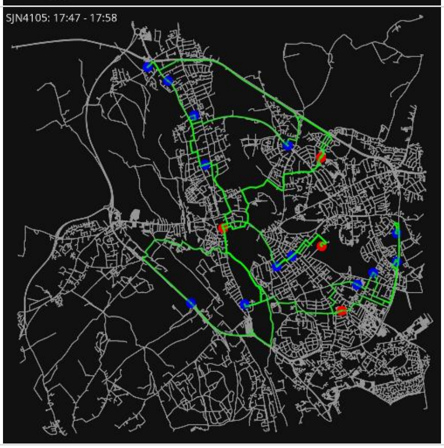
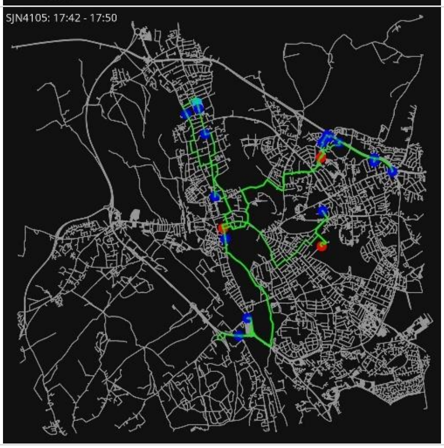
Event-triggered Optimization: After the computation of the daily schedule, the dataset is updated with 10 additional parcels (all of them should be picked up, some of them should also be delivered). We simulate a scenario of event-triggering at 10:30: all scheduled operations which are linked with stops of the MDH before the time of trigger are completed as planned, but the remaining ones should also integrate the newly arrived requests. Each parcel is assigned to the nearest stop of the MDH, then a complete re-routing of the remaining daily schedule is conducted.

- Area of deployment: Oxford
- Location of MCC/Depot: Redbridge Park-and-Ride
- Number of deliveries: 10 – all picked up, then delivered to the MCC (25% probability) or a random node in the network (75% probability).
- Number of mobile depots: 1
- Number of last-mile vehicles: 10 cargo-bikes
- Capacities of vehicles: Indicated by the stakeholders of the Living Lab
- Candidate stops: 10
- Travel times: Partially obtained by AIMSUN, completed by OpenStreetMap using the OSMNX [REF-11], for a radius of 3000 metres from the centre of Oxford

The updated routing plan is as follows:

Vehicle ID	Previous plan	New plan

DVH0437		
GYQ7535		
MEN1101		
ONK0793		

QMU4769		
QVV2112		
SCN2301		
SJN4105		

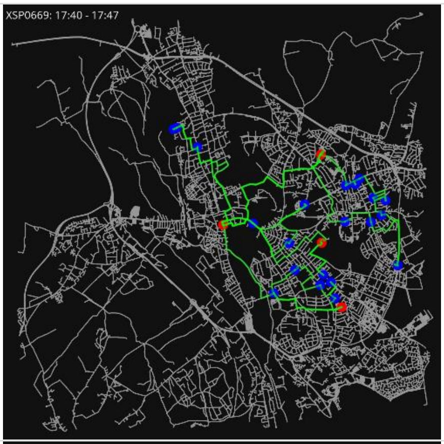
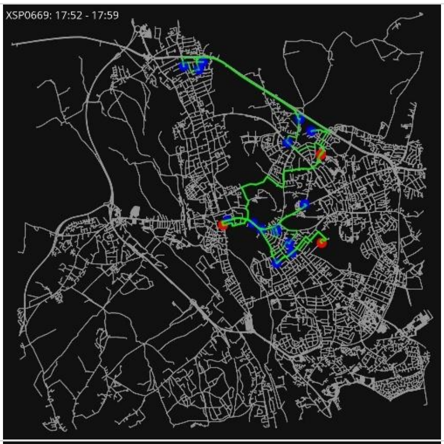
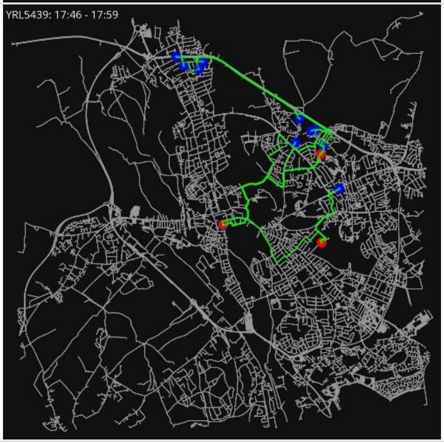
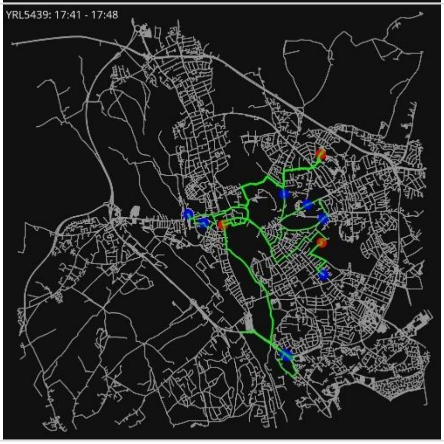
XSP0669		
YRL5439		

Table 11 (Re)routing results of Event-triggered Optimization in Oxford

The annotations are the same as in previous experiments, except for the nodes of the new requests, which are shown in light blue. We notice that the daily plan of several vehicles is extended, as new demand requests are added. We note that several nodes which were illustrated in the initial daily plans are now neglected, as they were completed before the time of re-triggering. Finally, the total distance shown in the json output file below only considers the routes which were conducted later than the time of re-triggering.


```
"general": {
        "uuid": "6ac53f4e-5350-45b6-ac5c-79b52537228c",
        "userName": "AUEB_optEngine",
        "producedAt": 1714659797,
        "KPIs": {
            "numRoutes": 11,
            "totalDistance": 330.94,
            "parcelsPerRoute": 10.5,
            "totalEmissions": 0.0,
            "totalDelay": 0.0
        }
```


5.3.4 Ispra LL

In Ispra, the bike is considered as a mobile depot, which will be scheduled by applying the MCC deployment stage. For each stop of the bike in the intermediate point, the fleet of autonomous droids will pick-up the lunchboxes from the bike and deliver them to the receivers. After the completion of each round of deliveries, the bike will depart from the intermediate point, re-load in the canteen and repeat the procedure.

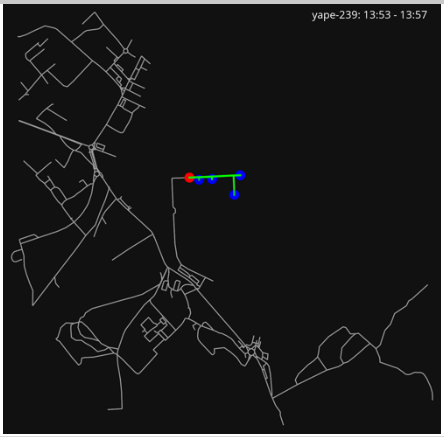
Although the scale of the problems is not as large as the one in the previous Living Labs, the time windows are significantly stricter: lunch-time is between 12:00 and 14:00, therefore all lunchboxes should be delivered within this 2-hours interval. Also, we examine the second type of disruption, which is the malfunction of a vehicle. In particular, whenever a droid is out-of-order, the event-triggered module should re-assign the affected deliveries to the remaining vehicles; in this case, the bike can facilitate the delivery procedure.

Due to the fact that the area of deployment is the enclosed campus of Joint Research Centre (JRC), there are no open-source geographic data. After the network of interest has been indicated by the stakeholders, a network has been constructed from scratch. The speed of the droids has been estimated using historical data of e-Novia.

Day-to-Day optimization: The rest of the parameters are as follows:

- Area of deployment: JRC in Ispra
- Location of MCC/Depot: Canteen
- Number of deliveries: 30 – randomly assigned to the four candidate demand points
- Number of mobile depots: 1 – the bike
- Number of last-mile vehicles: 3 Yape droids
- Capacities of vehicles: Indicated by the stakeholders of the Living Lab
- Candidate stops: 1 – the intermediate point
- Travel times: Estimated using input from e-Novia

The daily plan of the droids is the following one:

Vehicle ID	Daily route
yape-239	

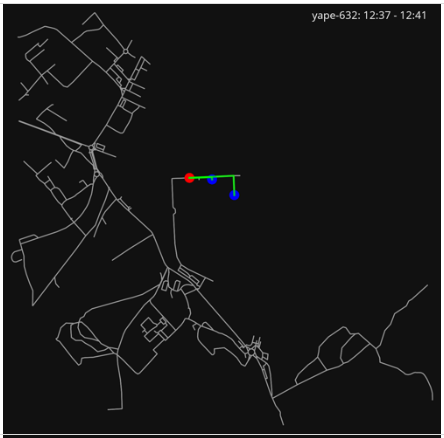
yape-632	
yape-853	

Table 12 (Re)routing results of Day-to-Day Optimization in Ispra

Event-triggered optimization: A randomly selected droid (i.e., yape-853) is out-of-order at 13:07. Three lunchboxes will be affected, as they cannot be delivered by the assigned vehicle; a re-routing procedure will assign the lunchboxes to the remaining vehicles, given that these are also in route to deliver when the malfunction occured, therefore they will be available after their return in the intermediate point. The involvement of the bike in the last-mile delivery operations in case of malfunction is critical for the completion of the daily plan in time, since it will always be available at the intermediate point if a routed droid is out-of-order.

- Area of deployment: JRC in Ispra
- Location of MCC/Depot: Canteen
- Number of deliveries: 3 – the lunchboxes which have not been delivered at the time of the event
- Number of mobile depots: 1 – the bike
- Number of last-mile vehicles: 2 Yape droids and 1 bike
- Capacities of vehicles: Indicated by the stakeholders of the Living Lab
- Candidate stops: 1 – the intermediate point
- Travel times: Estimated using input from e-Novia

After the implementation of the (re)routing service, the daily plan of the bike is extended, as it will be unsurprisingly the one which will conduct the remaining deliveries:

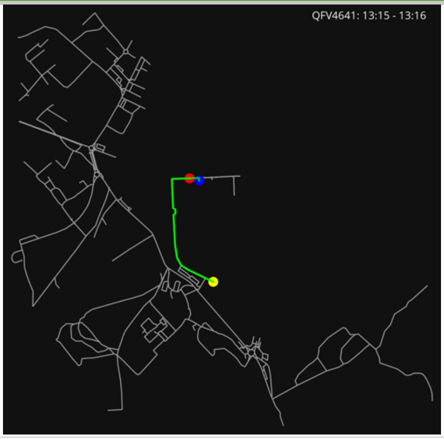
Vehicle ID	Daily route
QFV4641	

Table 13 Results of Event-triggered optimization in Ispra

Conclusions

Deliverable 3.1 introduces the methodological framework of the technical components, which are linked to Tasks 3.1 - 3.4 of the GREEN-LOG platform. Each section is dedicated to one of the services involved, describing its capabilities and demonstrating its potential applicability to the Living Labs which participate in the project. A consolidation of indicative screens, computational experiments on randomly generated benchmarks and mock-ups provides comprehensible illustrations of the expected outcome of the services for the implementation stages, which are planned to take place in the upcoming months.

The services which relate to Tasks 3.2 - 3.4 are mostly functional as standalone components. On the contrary, the Logistic-as-a-Service Marketplace, which is linked to Task 3.1, is a composite platform which exploits the functionalities of the components to facilitate the interaction between multiple stakeholders of the urban last-mile ecosystem. The deliverable elaborates on the synergies of the subcomponents of the LaaS Marketplace, and the representation of the interfaces expected to be accessible for the platform's users.

The outcome of this deliverable addresses to the developers of the technical services, which are expected to be integrated in the context of the upcoming deliverables of GREEN-LOG, and the Living Labs, which will benefit from the solutions of the operational problems, as acquired from the results of the services during the upcoming implementation stages. The complete potential of the devised services will be deployed in the upcoming months, since most of them require large volumes of historical and real-time data to be accurate (i.e., Nowcasting and Forecasting Engine, Dynamic (re)routing service) or implementation on real last-mile operations to demonstrate their functionalities (i.e., LaaS Marketplace, BYOD service). Therefore, the future work for the components of this deliverable includes an elaborate evaluation of their performance, after their implementation on the use-cases of the participating Living Labs, and their integration into the platform, so that their output can be exploited by any involved stakeholder (e.g., LSPs, local authorities of the Living Labs, or other technical services of GREEN-LOG).

References

Reference	Name of document
[REF-01]	Gartner Research, "Infographic: Supply Chain Visibility Is Fundamental to Resilience in Supply Ecosystems", 2022
[REF-02]	Shopify, 13 Ecommerce Challenges and How To Overcome Them, 2024
[REF-03]	E. D. Taillard, G. Laporte, and M. Gendreau, "Vehicle routing with multiple use of vehicles," *Journal of the Operational research society*, vol. 47, no. 8, pp. 1065–1070, 1996.
[REF-04]	D. Cattaruzza, N. Absi, and D. Feillet, "The multi-trip vehicle routing problem with time windows and release dates," Transportation Science, vol. 50, no. 2, pp. 676–693, 2016.
[REF-05]	N. Azi, M. Gendreau, and J.-Y. Potvin, "An adaptive large neighborhood search for a vehicle routing problem with multiple routes," Comput Oper Res, vol. 41, pp. 167–173, 2014.
[REF-06]	F. Hernandez, D. Feillet, R. Giroudeau, and O. Naud, "Branch-and-price algorithms for the solution of the multi-trip vehicle routing problem with time windows", Eur J Oper Res, vol. 249, no. 2, pp. 551–559, 2016.
[REF-07]	Y. Dumas, J. Desrosiers, E. Gelinas, and M. M. Solomon, "An optimal algorithm for the traveling salesman problem with time windows," Oper Res, vol. 43, no. 2, pp. 367–371, 1995.
[REF-08]	N. Labadie, C. Prins, and C. Prodhon, Metaheuristics for vehicle routing problems. John Wiley & Sons, 2016.
[REF-09]	T. Vidal, T. G. Crainic, M. Gendreau, and C. Prins, "A hybrid genetic algorithm with adaptive diversity management for a large class of vehicle routing problems with time-windows," Comput Oper Res, vol. 40, no. 1, pp. 475–489, 2013.
[REF-10]	Open Geospatial Data and Services for Greece - http://geodata.gov.gr/en/
[REF-11]	Boeing, G., "OSMnx: New Methods for Acquiring, Constructing, Analyzing, and Visualizing Complex Street Networks." Computers, Environment and Urban Systems. 65, pp. 126-139, 2017.
[REF-12]	Alexander Hess, Stefan Spinler, Matthias Winkenbach, "Real-time demand forecasting for an urban delivery platform.", Transportation Research Part E: Logistics and Transportation Review. 145, 2021.
[REF-13]	Hyndman, R. J., & Athanasopoulos, G., "Forecasting: principles and practice.", OTexts, 2018.
[REF-14]	Gneiting, T., & Katzfuss, M., "Probabilistic forecasting. Annual Review of Statistics and Its Application", 1, 125-151, 2014.
[REF-15]	Ord, J. K., Fildes, R., & Kourentzes, N. "Principles of Business Forecasting". Wessex Press Publishing, 2017

Appendix: User-stories in Flanders

As a User	I want to	So that	Explanation/Comments
Bike courier	have a manageable workload (limited km)	I'm not stressed out	Get demands for pick-up near to my position.
	To have information on what is or is not allowed in the city	I can avoid retributions	Get informed with restrictions (Hours permitted for delivery, etc)
Bike courier manager	receive demands that can be added to my existing route	I can optimise my drop density and do more business in less time	Demands=Pick-ups(mostly) but deliveries too. Define time-frames constrains
	test if I can extend my activity zone (cycle distance) if the Laas allows me to do substantially more business when cycling a bit further		25min distance from the Hub. Calculate if it is worth to cycling more or not. Check if there is a need to make some operational transformation (put an another Hub to a point)
	get more business opportunities (new clients/points) identified through the LaaS	I can do more business	Get insights for the existing situation (the deliveries that happen via Vans potential will be done via cargo bikes)
	receive demands in real-time	I can raise my drop density real-time	
	identify the volume and dimensions of the parcel	I can offer the appropriate solution	There is need to get data in order to examine this functionality
	profit from the optimisation features of GREEN-LOG	my business gains efficiency	Insight of their operations - KPIs from GA-
	identify the type of goods	I can offer the appropriate solution	Fragile - Waste goods - Pharmacy - Identifications of types
	I want to deliver/pick-up the parcels in time	I meet my service level agreement	If there is a SLA - need to check companies requirements
	identify no rush deliveries (metric for urgency)	I can organise properly	Get the priority of delivery according to the Consumer needs. Ie urgent or pick a time frame (2-3 days)
	the LaaS to show me which demands can optimise my planning	I can do more business in less time	
	plan my couriers efficiently, with or without LaaS	I can cover all costs	Check if there is a operational cost with or without LaaS
	connect the client with the courier who will pick-up/deliver the parcel	I can offer a transparent, simple, efficient service	Set the appropriate courier to do the demand according to the distance. Need to inform both shop-owner /consumer with the person that is going to do it
	know which goods I can consolidate in my city hub if possible (last or first mile)	I can limit the km driven	Rooting optimization according to the delivery address. Linked to the no -rush time deliveries.
	automatically feed orders in my own planning system	my normal business process is not interrupted	
	receive all the data of the parcel automatically	I can deliver correctly and in time	Re combination 8-11 and 22 (Rows)
	have a seamless technical connection between GREEN-LOG solution and my software	I don't have to adjust my own systems	Real -time connection
	enter my hub location in the system	the now and forecasting model is customised	Initialization of hub locations. To enter the information to a system.

	I want to be clear about our service level	my client knows what to expect (and what not)	Put information according to the service provision to fulfil LaaS -
	share my data with GREEN-LOG	I can optimise my delivery with the output of the data processing	Data exchange
	collaborate with technical partners to set up the dynamic pricing model	I can use the output during and after the project	
	offer an adequate (dynamic) price	all costs are covered	Limits to Dynamic pricing so as not compromise any courier company
	Get a fair price for my service	I can offer my couriers a fair salary	
	profit from subsidies to partially cover inter-city costs	I can test its operations	
	offer an adequate (dynamic) price	all costs are covered and I can benefit from a subsidy to make a profit	Limits to Dynamic pricing so as not compromise any courier company (Same as 28)
	test how my business can act as part of a LaaS	I can use the output during and after the project	Non-technical
	collaborate with technical partners to share my knowledge	the solution is in line with how the market works	Non-technical
	I want to co-exist in a fair ecosystem with other cycle logistics service providers	we don't have to compete too strongly and race for the lowest price	Non-technical
	collaborate with nudging partners to set up the nudging model	I can use the output during and after the project	Non-technical
	have insight in my efficiency gain using the LaaS	in order to evaluate properly	Non-technical
	have insight in the dynamic pricing for each of my orders	I can evaluate properly	Non-technical
	get insight in the cost per kilometer using GREEN-LOG solution	I can evaluate properly	Non-technical
	evaluate how ad hoc demand can optimise my existing routes (reporting tool)	I can evaluate properly the value of the Laas partnership	Non-technical
	have insight in delayed deliveries (according to sender/receiver)	to adjust and offer a better service	Non-technical
	have my service rated	to adjust and offer a better service	Non-technical
	have my service rated	to identify which client could benefit from extra follow-up	Non-technical
	identify my best clients	I can evaluate properly and have a long term partnership with them (after the project's end)	Non-technical

	get a better idea of/more information on what services are needed	I can develop new business opportunities	Non-technical
	promote my trailer park amongst local businesses	I can offer the solution they need	Non-technical
	promote my services amongst local businesses	I can offer the solution they need	Non-technical
	have my brand on the project's communication channels	I can promote my business	Non-technical
	inform new customers about my service of offering cargo-bike delivery for large volumes	I can increase my market share	Non-technical
	inform new customers about my service of offering cargo-bike delivery for heavy parcels	I can increase my market share	Non-technical
Local authorities	have an overview of which goods and Deliveries flow to, within and out the city (reporting tool)	we can implement data-driven policies	Non-technical
	find out if and how the Laas can be a solution for the challenges entrepreneurs in the city-centre or car free areas are coping with concerning their logistics	we can scale up and let more partners use the LaaS	Non-technical
	have better insight in he societal cost calculations of city logistics	we can have a more solid argumentation for investments in sustainable logistics by the city	Impact on Society such as carbo emissions. Type of report.
	promote alternative delivery options to our citizens	our citizens can choose more sustainable delivery options	
	nudge my citizens and entrepreneurs to organise their logistics outside the school rush hour	so that the school-routes are safer	Give time-window and locations constrains
	impose economic incentives	I can steer the market	Reduce the price so intervene the market
	nudge my citizens and entrepreneurs to organise their logistics more sustainable		
	impose restrictions through business rules	I can steer the market	How the intervention with businesses is going to happen? Part of co-creation business model/
	centralise all demands	the LaaS becomes more efficient and attracts more (diverse) stakeholders	Business transformation - Part of co-creation -
	centralise the operations of several sustainable logistic service providers	All demands can be covered	Business transformation - Part of co-creation
	preselect logistic service providers that can access the LaaS	I can have more control about the quality and	Application form- SLA -> Their request was to be able to evaluate which LSP can enter the LaaS Marketplace

		sustainability of the services	according to some criteria, i.e., Cargo bikes, low emission vehicles, etc.
	apply our policies directly to every delivery in the city	eventually all deliveries are compliant with our policies	Deep into more detail - Same as 59
	facilitate bundling	there are less vehicles in the city-centre	Co-operational business model of logistics?
	make sure all actors pay/receive a fair price	I change the system in an appropriate way	How the intervention with businesses is going to happen? Part of co-creation business model
Local shop owner	I want to see which LSP is able to deliver my goods and for which price	So that I can have my goods delivered even during road works	Restrictions: City Zones - Time windows - Local Authorities restrictions
	track the deliveries and receive notifications	I am informed about the estimated delivery scheme	We need to identify if any LSP willing to exchange information
	get transparent invoices of the deliveries	I know what I'm paying for and to calculate future costs	Capture the analysis of the price after Dynamic Pricing will be applied.
	get notified if a request hasn't gone through	I can find a solution	Communication between LSP -customer (Examine if this will be a part of the System)
	get an overview of all requests that have gone trough	I can double-check when in doubt	
	get reports on how many deliveries were carried out by shipper X, Y, Z	I know which LSP's are most suitable for me	
	track real-time when the shipper is on his/her way to my shop to pick up the goods	I can get organised efficiently	Extended functionality of row N.69
	get a message when the goods or waste are picked up in my shop	I don't have to worry of it is picked up yet or not	Proof of Pick-up
	track the deliveries and receive notifications	inform my customer about the estimated delivery scheme	Same as 74
	inform my customers in the check-out about the societal cost of the delivery options	I can induce behavioural change while still offering e-commerce and letting my business grow	Communicating between LSP -customer (Examine if this will be a part of the System)
	thank my customer by e-mail after delivery for choosing the most sustainable delivery option	I can extend the user journey after the delivery and engage them to change behaviour/to come back to my (web)shop	Communicating between LSP -customer (Examine if this will be a part of the System)
	thank my customer for repeated sustainable deliveries by offering an advantage (promotion/gadget/budget to spend in the community of we.deliver)	I can extend the user journey after the delivery and engage them to change behaviour/to come back to my (web)shop	Communicating between LSP -customer (Examine if this will be a part of the System)
	to use a user friendly system	I don't loose time finding out how everything works	

	have my parcel orders entered automatically in the LaaS, without manual intervention from my side	I can work more efficiently	Need to specify the rules in order to fulfill this. Deep into more details
	label my parcel as time-sensitive or not (chocolate e.g.)	shippers can choose no rush shipping whenever possible	Non-technical. Co-operation with LSP for goods categorization. LSP need to inform which goods can I pick-up
	decide which vehicles can carry out my deliveries	I don't have to consider vehicles that don't comply with my approach	Non-technical. Co-operation with LSP for further discussion.
	decide which shippers can offer deliveries through the LaaS for my demand	I don't have to consider shippers that don't comply with my service level	We need to identify if any LSP willing to exchange information (Can provide staff's name, etc.)
	access the LaaS on multiple devices (phone, tablet, computer)	I can organise my work wherever, whenever	
	be able to let urgent goods be delivered within the time window of the car free area	I can have my urgent goods	Non-technical. Co-operation with LSP for further discussion. Possible Part of Co-creation
	send only confirmed packages	the package is in stock	Deep into more details. Mostly has to do with the user of the web shop. When he/she will inform for readiness of the product to be delivered
	get priority scheduling as a good customer by LSP's that often ship my goods		Non-technical. Co-operation with LSP for further discussion.
Private consumer	choose the no rush shipping option	I can have impact on the sustainability of my delivery	Same as 16
	see in one overview before and during the checkout what my delivery options are	I can consider them before purchasing	
	decide to receive all my orders from the webshop in one delivery	I can get organised efficiently	Operational
	to pay whatever is needed (willingness to pay) for a service	I can get the shipping done as I want it	Non-technical.
	decide that I want to receive my parcel in a locker close to my home, job...	I don't have to feel bad about ordering online	Delivery options. Depends on the service that LSP can provide
	decide when I want to receive my parcel	I'm home to receive the parcel	Delivery options. Depends on the service that LSP can provide
	pay whatever it takes to get my express delivery shipped in a sustainable way	I can combine comfort and sustainability (whatever the price)	Same as 94
	choose less packaging	I don't waste materials and feel proud	Non-technical. It depends on Local Shop owner
	decide that I want to pick-up my parcel in the store	I don't have to feel bad about ordering online	Extension of 95
	choose the most sustainable delivery option in terms of emissions	I don't contribute to the overload of delivery vans in my city and feel proud	Non-technical. It depends on the user's preferences.

	know what emissions I've saved by choosing my delivery option	I can feel proud and continue my behaviour	Reporting if data available
	know how my no rush delivery has saved emissions and vehicle movements	I can feel proud and continue my behaviour	Reporting if data available. Hard to achieve
	see in one overview before and during the checkout what the cost of each delivery option is	I can consider this before purchasing	Extension of 92
	choose the most sustainable delivery option in terms of noise	I don't contribute to the overload of delivery vans in my city and feel proud	
	choose the most sustainable delivery option in terms of fair pay	I don't contribute to the race to the bottom when it comes to pricing	
	get rewarded for my sustainable choices by the local shop owner	I'm motivated to continue my changed behaviour	Non-technical. It depends on Local Shop owner. Does the front-end will have CRM functionalities. Rewards- Complaints, etc.? It sounds to me out of scope. During discussion this was highlighted by our side
	track my parcel	I can get organised efficiently	Same as 93
	be informed about delays	I can get organised efficiently	Discussion with LSP. How often can a delay happen? Usually, the alerts and live tracking is functionality of a mobile. GreenLog doesn't have mobile app implementation for customers
	rate the delivery service	I can give feedback (good or bad)	Same description as 106
	know which types of vehicles have carried out my sustainable delivery	I can have a better knowledge about the sustainable supply chain	Part of LaaS Marketplace delivery options.
	be able to complain about the delivery service	it won't happen again	same 104 - 105
	know how I've supported fair working conditions by choosing my delivery option	I can feel proud and continue my behaviour	same 104 - 105
	know how my delivery option has reduced noise pollution	I can feel proud and continue my behaviour	same 104 - 105
	be able to hold someone responsible when something goes wrong (delay, lost or damaged goods, wrong goods...) and contact this contact person	I can complain and get my problem sorted	Communication/Same description as 106
	avoid missed deliveries at home	I don't waste time to pick it up in a parcel shop or postal service point	Live tracking - Commucation issues and Operational Issues (Part of WP1)