ReMuNet

Pioneering resilient and adaptive multimodal transport networks

D3.7

Collaborative platform backend and design interfaces for data and stakeholder integration

FIR e. V. RWTH Aachen University

30 / 06 / 2026

Funded by the European Union

Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Climate, Infrastructure and Environment Executive Agency (CINEA). Neither the European Union nor the granting authority can be held responsible for them.

PROJECT INFORMATION

PROGRAMME	Horizon Europe
TOPIC	HORIZON-CL5-2022-D6-02-07
TYPE OF ACTION	HORIZON Research and Innovation Actions
PROJECT NUMBER	101104072
START DAY	1 July 2023
DURATION	36 months

DOCUMENT INFORMATION

TITLE	Collaborative platform backend and design interfaces for data and stakeholder integration
WORK PACKAGE	3
TASK	3.6
AUTHORS (Organisation)	Maximilian Dicks (FIR)
REVIEWERS	All partners
DATE	30/06/2026

DISSEMINATION LEVEL

PU	Public, fully open	x
SEN	Sensitive, limited under the conditions of the Grant Agreement	
Classified R-UE/EU-R	EU RESTRICTED under the Commission Decision No2015/444	
Classified C-UE/EU-C	EU CONFIDENTIAL under the Commission Decision No2015/444	
Classified S-UE/EU-S	EU SECRET under the Commission Decision No2015/444	

DOCUMENT HISTORY

VERSION	DATE	CHANGES	RESPONSIBLE PARTNER
1	15/04/2026	First draft	FIR
2	18/05/2026	Finalization Text	FIR
3	26/06/2026	Review by consortium partners	All partners
3	29/06/2026	Final version	FIR

TABLE OF CONTENTS

1. INTRODUCTION 6
1.1 Project Background 6
1.2 Key Objectives of the ReMuNet Project 6
1.3 Objective of Deliverable 3.7 7
1.4 From previous Deliverables to Deliverable 3.7 8
1.5 Structure of Deliverable 3.7 8
2. PURPOSE AND OBJECTIVE 8
3. STAKEHOLDER INTEGRATION AND DATA SCOPE 9
4. BACKEND ARCHITECTURE 9
5. INTERFACE DESIGN AND DATA EXCHANGE 9
5.1 Exposed Backend Interfaces 9
5.2 Individual-Value Queries 10
6. EXTERNAL SYSTEM INTEGRATION 10
6.1 PTV Routing Service Integration 10
6.2 Mansio Integration 12
6.3 Mansio CLI Tool 13
7. IT SECURITY, CYBER SECURITY AND DATA PROTECTION 14
8. DEPLOYMENT AND HOSTING 14
9. RESULT AND STATUS 14
10. SUMMARY, IMPLICATIONS, LIMITATIONS AND OUTLOOK 14

LIST OF FIGURES

Table 1: interfaces of Java Spring Boot backend 10
Table 2: Sample request (Tampere – Berlin)..... 11
Table 3: Sample response excerpt (without way list)..... 12
Table 4: environment file for Mansio integration..... 13
Table 5: Python example invocation for Mansio CLI tool..... 13
Table 6: Python Example response excerpt form Mansio CLI tool 13

1. Introduction

ReMuNet – Resilient Multimodal Freight Transport Network – is financed by the European Commission through the European Union's Horizon Europe research and innovation programme under Grant Agreement No 101104072. It has been proposed by a consortium of 15 beneficiaries coordinated by FIR e. V. an der RWTH Aachen and is being carried out over 36 months, having started in July 2023. The content of the project and the aim of this deliverable will be presented in the following.

1.1 Project Background

ReMuNet is tasked with identifying and signalling disruptive events while assessing their impact on transport corridors. By communicating alternative multimodal transport routes to logistics operators and subsequently to truck drivers, train drivers, and barge captains, it facilitates a more rapid network response. Furthermore, ReMuNet orchestrates route and capacity utilisation and enables synchromodal relay transport.

Additionally, ReMuNet is responsible for establishing comprehensive standards for the transnational description of multimodal nodes and transport routes, including infrastructure elements and their parameters. This forms the basis for a unified modelling language. To ensure maximum practical applicability and acceptance, these standards are developed by evaluating and consolidating existing description standards, supplementing technical and sustainability-related descriptive attributes, and engaging in close collaboration with European transport companies, associations, and institutions.

ReMuNet's collaborative platform reduces entry barriers for regional transport companies, encouraging participation in multimodal transport, known for its lower average emissions compared to road freight transport. Intelligent routing algorithms prioritise low-emission routes and modes of transport, enhancing transport efficiency and significantly cutting emissions. ReMuNet explores the integration of alternative drives to further support sustainability goals, aiding political and economic decision-makers in expanding and maintaining vital transport infrastructure.

To make this routing functionality accessible to stakeholders, ReMuNet provides a collaborative platform consisting of a backend integration layer and a web-based frontend. This deliverable addresses the backend component of that platform.

1.2 Key Objectives of the ReMuNet Project

The concept of the Physical Internet foresees a seamless network where resources and assets can be shared openly and efficiently. ReMuNet's main objective is to support the achievement of this vision by giving the transport network the ability to prepare, respond, and adapt to disruptive events.

In practice, ReMuNet uses digital technologies to holistically improve the resilience of the logistics and freight transport network. This includes making planning more robust and secure, reducing response times, and speeding up system recovery. ReMuNet also takes environmental aspects into account and aims to provide synchromodal transport options

with the lowest emissions, thereby contributing to the implementation of the Physical Internet and reducing the environmental impact of freight transport.

ReMuNet's key objectives are to:

- Lay the foundation for a common standard to describe sustainable European multimodal transport networks for all stakeholders.
- Reduce European inland transport emissions on the main run by over 50 % in the long term.
- Enable the multimodal freight network to react and respond 20 % more quickly to disruptive events until 2026 (compared to 2021).
- Provide 50 % more alternative transport routes in the face of disruptive events and make multimodal route planning 10–20 % more accurate and efficient until 2026.
- Create new approaches for benefit-sharing, cost-efficient business models and integrate 30 % more companies (> 50 % SMEs) into the multimodal freight transport ecosystem.
- Predict the impact of disruptive events on multimodal transport corridors with 90 % higher accuracy until 2026.
- Enable civil protection organisations to set up aid-delivery logistics for crisis areas 25 % faster in 2026.
- Create a unified data pool that portrays the real-time utilisation of multimodal European transport infrastructure in compliance with data protection regulations.

The backend integration and orchestration layer documented in this deliverable directly supports several of these objectives: it makes routing and disruption information accessible to a broad range of stakeholders regardless of their own IT maturity, and it provides the technical foundation on which the project's routing-related standards, algorithms and use cases build.

1.3 Objective of Deliverable 3.7

This document describes the deliverable of Task T3.6, the collaborative platform backend of ReMuNet together with the interfaces designed for the exchange of data between the parties involved in multimodal transport planning. It is the first of two deliverables derived from the ReMuNet routing platform documentation; the second deliverable covers the web-based visualization front end built on top of these interfaces.

The backend is implemented as a Java Spring Boot service that acts as an integration and orchestration layer between ReMuNet and external routing and logistics systems. It integrates the PTV intermodal routing service and the Mansio cross-carrier logistics service, transforms their responses into a common internal data model, and exposes well-defined interfaces through which involved parties can retrieve routing scenarios, route alternatives, and handover information as structured, machine-readable data.

In line with the objectives of T3.6, the backend provides the technical foundation for making transport-related information accessible to both directly and indirectly involved stakeholders. Data exchange is currently realized as a REST API over HTTP.

1.4 From previous Deliverables to Deliverable 3.7

Deliverable 3.7 builds on the routing- and disruption-related concepts developed earlier in WP3 and establishes, for the first time, a productive backend integration layer that consolidates external routing and logistics services (PTV, Mansio) into a single internal data model. As the first of two deliverables documenting the ReMuNet routing platform, it provides the technical foundation – the interfaces and data model – on which the web-based frontend documented in the subsequent Deliverable 3.8 is built.

1.5 Structure of Deliverable 3.7

This deliverable is structured into the following chapters:

- Chapter 1 outlines the objective and scope of this deliverable.
- Chapter 2 describes the purpose and objective of the backend in more detail.
- Chapter 3 describes stakeholder integration and the scope of the exchanged master and transaction data.
- Chapter 4 describes the backend architecture, including the technology stack and hosting setup.
- Chapter 5 describes the interface design and data exchange mechanisms, including the exposed backend interfaces and individual-value queries.
- Chapter 6 describes the integration of the external PTV routing service and the Mansio cross-carrier logistics service, including the Mansio CLI tool.
- Chapter 7 addresses IT security, cyber security and data protection.
- Chapter 8 describes deployment and hosting.
- Chapter 9 summarises the result and status of the backend implementation.

2. Purpose and Objective

The objective of this deliverable is to provide the collaborative platform backend of ReMuNet and to design the interfaces through which master and transaction data can be exchanged with the stakeholders of the multimodal transport network.

The main goals are:

- Provide a backend integration and orchestration layer that consolidates routing and handover data from external services into a single, consistent data model.
- Expose clearly defined interfaces for retrieving routing scenarios, route alternatives, and handover information.
- Integrate the systems of involved service providers, in particular the PTV intermodal routing service and the Mansio cross-carrier logistics service.
- Make routing and logistics information accessible to directly and indirectly involved parties through standard, machine-readable formats.

- Enable individual-value queries so that clients can request only the specific information they need, independent of the maturity of their own IT systems.
- Ensure that data exchange is protected by appropriate IT- and cyber-security measures and complies with the General Data Protection Regulation (GDPR).

3. Stakeholder Integration and Data Scope

The ReMuNet interfaces are designed to serve two groups of parties. Directly involved stakeholders include forwarders, transport operators, infrastructure managers and similar operational actors. Indirectly involved parties include research organizations, institutions and service providers. For both groups the interfaces are intended to exchange master data and transaction data across the areas of booking, infrastructures, means of transport, environment, and status messages for loading units.

Accessibility of information for all interested parties is a deliberate design goal. Because the query needs differ between clients — a small or medium-sized enterprise may not operate an extensive Transport Management System (TMS) — ReMuNet is designed to provide individual values for specific queries rather than requiring clients to consume full datasets. Depending on client capability, the exchange is realized via API.

4. Backend Architecture

The backend is built with Spring Boot and uses Java as the programming language, running on the Java Virtual Machine. It exposes a REST API for communication with consuming clients and the web front end, exchanging data as JSON over HTTP.

Functionally, the backend acts as an integration and orchestration layer for routing and handover data. It integrates the PTV Routing Service for route calculation and Mansio for cross-carrier truck traffic and handover points. Responses received from these external systems are mapped into internal domain models before being made available through the backend interfaces, providing consumers with a consistent representation independent of the underlying source system.

The backend is hosted internally and is reachable at:

- https://digitaler-zwilling.bitsfabrik.com/api/

It integrates two external systems:

- PTV Routing Service: provides routing functionality and route calculation data.
- Mansio: supports the organization of cross-carrier truck traffic and returns Mansio Points used as part of the routing and handover information.

5. Interface Design and Data Exchange

5.1 Exposed Backend Interfaces

The Java Spring Boot backend exposes the following interfaces, which are consumed by the front end and can be used by integrated clients to retrieve routing and handover data.

Table 1: interfaces of Java Spring Boot backend

Endpoint	Purpose
GET /routes	Retrieves the available routing scenarios.
POST /route?scenario=<id>&disruption=<true|false>	Retrieves route alternatives for the selected scenario and disruption state.
POST /handovers?scenario=<id>&disruption=<true|false>	Retrieves additional tour and handover data for the selected scenario and disruption state.

API endpoints are configured through environment variables, namely MAIN_API_SERVICE_SERVER_URL for server-side requests and NEXT_PUBLIC_MAIN_API_SERVICE_URL for client-side requests.

5.2 Individual-Value Queries

The backend interfaces return discrete operational values that can be queried individually, including costs, duration, distance, CO2 emissions, number of transfers, transport modes, itinerary legs and handover points. This directly supports the T3.6 goal of providing individual values for specific queries — such as emissions, timetable and capacity information — rather than obliging clients to integrate a full dataset.

6. External System Integration

6.1 PTV Routing Service Integration

The backend integrates the PTV xServer routing service for calculating transport routes. Two PTV xServer instances are used, one for normal routes and one for disrupted routes. The backend calls the calculateRoute endpoint of the PTV xIntermodal module. A request contains routing options, start and destination stop-offs, excluded transport modes, excluded terminals or operators, and caller-context settings such as the coordinate format.

Table 2: Sample request (Tampere – Berlin)

```
{
  "routingRequest": {
                "routingOptions": {
                "startTime": "2025-08-19T12:00:00+02:00",
                "maxCosts": 40000,
                "accompanied": false,
                "timeCostWeight": 75,
                "numberOfAlternatives": 5,
                "withWayList": true,
                "excludedTransportModeCodes": ["TM_TTN_AIR"],
                "excludedTerminalCodes": [],
                "excludedOperatorCodes": ["26"]
                },
                "stopOffs": [
                {
                "name": "NordicModHomes",
                "point": { "$type": "PlainPoint", "x": 23.787, "y": 61.504, "z": 0 },
                "country": "FI", "postalCode": "33500", "city": "Tampere",
                "street": "Kalevan puistotie", "houseNumber": "7"
                },
                {
                "name": "BauModul GmbH",
                "point": { "$type": "PlainPoint", "x": 13.533, "y": 52.548, "z": 0 },
                "country": "DE", "postalCode": "12681", "city": "Berlin",
                "street": "Schwarze-Pumpe-Weg", "houseNumber": "16"
                }
                ]
  },
  "callerContext": {
                "properties": [ { "key": "CoordFormat", "value": "OG_GEODECIMAL" } ]
  }
}
```

Table 3: Sample response excerpt (without way list)


```
{
  "routes": [
            {
            "name": "R1_Road",
            "totalCosts": 2947,
            "totalDuration": 1228,
            "totalDistance": 1545,
            "totalEmission": { "carbonDioxide": 1541.49 },
            "numberOfTransfers": 0,
            "startTime": "2025-08-19T12:00:00+02:00",
            "itinerary": [ { "transportMode": { "code": "TM_ROAD", "name": "Road" },
            "distance": 1545, "duration": 1228 } ]
            },
            {
            "name": "R2_SSV",
            "totalCosts": 1305,
            "totalDuration": 3038,
            "totalTravelTimeRoad": 297,
            "totalTravelTimeIM": 2670,
            "totalDistance": 1627,
            "totalEmission": { "carbonDioxide": 1276.06 },
            "numberOfTransfers": 2,
            "itinerary": [
            { "transportMode": { "code": "TM_ROAD", "name": "Road" }, "distance": 162 },
            { "line": { "operator": { "code": "REMUNET", "name": "REMUNET" } },
            "transportMode": { "code": "TM_TTN_SHORTSEA", "name": "Timetable SSV" },
            "distance": 1176, "waitingTime": 56, "handlingTime": 15 },
            { "transportMode": { "code": "TM_ROAD", "name": "Road" }, "distance": 289 }
            ]
            }
  ]
}
```


The response contains route alternatives with aggregated values for costs, duration, transport-mode-specific travel times, waiting and handling times, distance, emissions and transfers, together with the individual itinerary legs. The backend transforms this response into the internal data model exposed through its interfaces.

6.2 Mansio Integration

The backend integrates with Mansio for the organization of cross-carrier truck traffic and the calculation of handover points. Mansio is used to import shipment and tour information and

to calculate possible handovers based on configurable parameters such as handover distance and city population. The integration requires API access credentials, which are provided through an environment file:

Table 4: environment file for Mansio integration

```
BASE_URL="https://analyse.mansio-logistics.com/api"
API_KEY=your_api_key_here
```

The external import workflow creates shipment and tour data through /external/import/tours, checks the asynchronous import result via /external/import/status/{importID}, and runs the handover calculation through /external/calculate_handover. These actions are implemented in the backend and can be executed through its exposed REST interfaces.

6.3 Mansio CLI Tool

In addition to the backend integration, a Python command-line tool was created to support faster testing of the Mansio interface. The tool requires complete address details for both the sender and the consignee, and accepts optional arguments such as the tenant ID, handover distance, city-population requirement, environment-file path and logs directory; identifiers such as case ID and external shipment and tour numbers are generated automatically if not provided.

Table 5: Python example invocation for Mansio CLI tool

```
python3 mansio.py \
  --sender-name "Acme Corp" --sender-street "Industrial Blvd" \
  --sender-city "Berlin" --sender-postal-code "10115" \
  --sender-country-code "DE" \
  --consignee-name "Beta Logistics" --consignee-street "Commerce St" \
  --consignee-city "Munich" --consignee-postal-code "80331" \
  --consignee-country-code "DE" \
  --handover-distance 200 --city-population 5000
```

Table 6: Python Example response excerpt form Mansio CLI tool

```
"handovers": [
  {
            "address": { "city": "Jüri", "postalCode": "75301", "countryCode": "EE" },
            "coordinates": { "latitude": 59.35974, "longitude": 24.92099 }
  },
  {
            "address": { "city": "Saulkrasti", "postalCode": "LV-2160", "countryCode": "LV" },
            "coordinates": { "latitude": 57.26224, "longitude": 24.41471 }
  }
  /* ... further handover points omitted ... */
```

```
]
```

7. IT Security, Cyber Security and Data Protection

IT security and cyber security comprise all technical and organizational measures that protect the systems from cyber-attacks and other threats. The measures foreseen for ReMuNet include, access controls, cryptography, rights management, firewalls, proxies, dependency scanners and vulnerability management. In addition, the requirements of the General Data Protection Regulation (GDPR) must be implemented for all exchanged data.

In the current implementation, the backend is operated in an internal hosting environment, and access to integrated external services such as Mansio is controlled through API keys held in protected environment configuration.

8. Deployment and Hosting

The backend is prepared for deployment through the existing Bitbucket Pipelines setup. The deployment process for the Java Spring Boot application is implemented and optimized, and the pipeline supports staged and production-oriented deployment flows. Environment-specific configuration is provided through deployment variables and runtime environment variables, in particular for connecting consuming applications to the backend REST API. The setup allows the backend and the front-end application to be hosted together, and the application can also be run locally for development, testing and demonstration purposes. The platform is hosted and processed within the EU and GDPR.

9. Result and Status

The implemented backend provides a functional integration and orchestration layer together with well-defined interfaces, enabling involved parties to retrieve routing scenarios, route alternatives and handover information in a consistent, machine-readable form. The integrations with the PTV routing service and the Mansio cross-carrier logistics service are in place, and exchanged data covers operational indicators such as costs, duration, distance, emissions and handover points.

Data exchange via the REST API is implemented. The backend is prepared for deployment through the existing pipeline and can be run locally for development, testing and demonstration.

10. Summary, implications, limitations and outlook

10.1 Summary

Deliverable 3.7 has provided the collaborative platform backend of ReMuNet as a Java Spring Boot integration and orchestration layer. The backend consolidates the PTV intermodal routing service and the Mansio cross-carrier logistics service into a single, consistent internal data model and exposes this data through well-defined REST interfaces that allow involved parties to retrieve routing scenarios, route alternatives and handover information as structured, machine-readable data. Individual-value queries make it possible

for clients to request specific operational indicators, such as costs, duration, distance, CO2 emissions, transfers and itinerary legs, independent of the maturity of their own IT systems. The backend is secured through appropriate access controls and protected environment configuration, complies with the requirements of the General Data Protection Regulation (GDPR), and is prepared for deployment through the existing Bitbucket Pipelines setup, with the option to run locally for development, testing and demonstration.

10.2 Implications

The backend documented in this deliverable forms the technical foundation for the web-based frontend of the ReMuNet routing platform, which is documented as the second of the two deliverables derived from the ReMuNet routing platform documentation, Deliverable 3.8. The frontend consumes the REST API exposed by this backend to retrieve routing scenarios and handover information, filters and transforms this data for presentation, and displays the resulting transport options on an interactive map, allowing stakeholders to compare normal and disrupted transport situations. In this sense, Deliverable 3.7 does not stand on its own: it is the data and integration layer on which the accessibility, usability and decision-support value of the overall platform ultimately depend.

Beyond the immediate platform, the backend contributes to the broader objectives of the ReMuNet project. By making routing and disruption information accessible through standard, machine-readable interfaces regardless of a stakeholder's own IT maturity, it lowers the entry barrier for regional transport companies and smaller operators to participate in multimodal transport planning. The individual-value queries exposed by the backend support the project's aim of providing more accurate and efficient multimodal route planning and of making low-emission alternatives more visible to logistics operators, truck drivers, train drivers and barge captains. The consolidated data model also lays groundwork for the project's longer-term goal of a unified data pool portraying the real-time utilisation of multimodal European transport infrastructure.

10.3 Limitations

The current implementation is, by design, limited to the scope defined for Task T3.6. The backend integrates two external systems, the PTV routing service and Mansio, and the resulting data model and interfaces reflect the operational indicators provided by these two sources; further routing or logistics services are not yet integrated. The backend is presently operated in an internal hosting environment rather than on a public-facing infrastructure, and access to integrated services such as Mansio depends on API keys held in protected environment configuration, which ties the operational availability of the platform to the continued provisioning of these credentials. As an integration and orchestration layer, the backend also depends functionally on the correctness, availability and response times of the external PTV and Mansio services it consumes; it does not itself perform route calculation or handover optimisation. Finally, in line with its defined scope, the backend does not provide booking functionality; it makes routing and handover information available for retrieval and visualisation, while booking and operational execution of transports remain outside the boundaries of this deliverable.

10.4 Outlook

With the backend integration and orchestration layer in place, the next step within the ReMuNet routing platform is documented in Deliverable 3.8, which builds the web-based visualisation on top of the interfaces described here. Going forward, the consolidated internal data model established in this deliverable provides a stable basis onto which additional data sources, further individual-value queries, or extended security and monitoring measures can be added as the platform matures, without requiring changes to the way consuming clients and the frontend already access routing and handover information. The backend integration and orchestration layer thereby remains the central technical foundation on which the further development of the ReMuNet routing platform, and its contribution to the project's overall objectives of a more resilient, efficient and low-emission multimodal transport network, will continue to build.