I2PANEMA

Innovation Report v2

Deliverable No.	D5.2	**Due Date**	01.04.2022
Type	Document	**Dissemination Level**	Public
Version	2.1	**Status**	Submitted
Description	Innovation Report		
Work Package	WP5		

Authors

Name	Partner	e-mail
Krishna Sudhakar	NXP	krishna.sudhakar@nxp.com
Felix Manthey	NXP	felix.manthey@nxp.com
Jannis Schneider	CAT	jannis.schneider@catkin.eu
Yildiray Kabak	SRDC	yildiray@srdc.com.tr
Julia Wernecke	HPA	julia.wernecke@hpa.hamburg.de
Otto Klemke	NautilusLog	otto@nautiluslog.com
Valentin Mees	FhG LBF	valentin.mees@lbf.fraunhofer.de
Maximiliane Lorenz	FhG IML	maximiliane.lorenz@iml.fraunhofer.de
Achim Klukas	FhG IML	achim.klukas@iml.fraunhofer.de
Björn Krämer	FhG IML	Bjoern.kraemer@iml.fraunhofer.de
José A. Clemente	prodevelop	jclemente@prodevelop.es
Christophe Joubert	prodevelop	cjoubert@prodevelop.es

History

Date	Version	Change
04.11.2020	V1	input chapter 1 by NXP
06.11.2020	V1	input chapter 2 by CAT
09.11.2020	V1	complete version V1 (small adjustments by NXP)
12.11.2020	V1	first review by Franz-Josef Stewing
16.11.2020	V1	Revision by CAT
20.11.2020	V1	Revision by NXP
25.11.2020	V1	2nd review by Maximiliane Lorenz
30.11.2020	V1	V1 completed by CAT & NXP
31.08.2021	V2	Update to V2 by CAT
26.11.2021	V2	Finalized input for Spanish BS in chapter 1 by NXP
29.11.2021	V2	Finalized current input in chapter 2 by CAT
30.11.2021	V2	Creation of Spanish version by CAT
02.03.2022	V2	Finalized current input in chapter 2 by CAT
03.03.2022	V2.0	Finalized input for Turkish BSs in chapter 1 by NXP
01.04.2022	V2.1	Public version of the latest version for up-load to the ITEA portal

Key Data

Keywords	I^{2}PANEMA, Security, Workflow
Lead Editor V2	NXP

Internal Reviewer(s)	

Abstract

This document aims to reflect the results of tasks T5.2 and T5.3 at the stage of publishing V2 of D5.2.

Due to the different project timespans of the respective country subprojects of I^{2}PANEMA the updates in V2 of D5.2 will be populated in incremental manner. The first subproject to close is the Spanish subproject, ending in December 2021. The Turkish subproject ends in March 2022, whereas the German subproject will be the last one to finish, with its duration currently being extended to December 2022. At this stage of the deliverable D5.2 it is submitted for the Spanish and the Turkish subprojects' ends. Consequently, only the Spanish and the Turkish parts of it will reflect the complete updated V2, i.e., only the general sections and those for business scenario Gijon, Safi Port (Roro localization) and Assan Port (container localization) are updated. The other German business scenarios sections will be updated when the edition of the final deliverable must be submitted.

With respect to T5.2, "I^{2}PANEMA Security and Privacy Layer", the objective is to create a security module that will be a fundamental part of the I^{2}PANEMA platform, with its elements implemented in the various project's business scenarios (BSs). To do so, the first step is the analysis of the objectives of those BSs that need to be protected. This implies conducting a STRIDE analysis for them. In a second step potential security threats for all system entities are identified. Based on the DREAD methodology, the risk associated to the threats is assessed, e.g., how easy it is to reproduce vulnerability, how great can be the damage, etc for each BS. This process allows to develop a security prototype for the implementation of the respective BSs.

Regarding T5.3, "Operational Workflow Support Layer", it is intended to respond to the different workflows that occur in each BS. These workflows must be configurable at an administration level via, e.g., a Web-API (SOAP / REST web services). Workflows usually consist of typical order / task structures, like status information ("in progress", "confirmed", "finished", "critical changes", etc.) but can also have a specialized part where planned values and measurements indicate progress of a realistic order structure (e.g., "loading time", "ship arrival", "temperature checked", etc.).

In conclusion, in V2 this deliverable will describe the realised solutions at the Security and Workflow Layers of I^{2}PANEMA, where V1 represented a snapshot of the work in progress. These services must finally support the identified I^{2}PANEMA IoT Reference Architecture, for which the prototypes deployed as I^{2}PANEMA proof-of-concept pilots in the different BSs should serve as reference implementations of this architecture.

This version is the public version of the former submitted one without the confidential information on identified security threats (sections 1.n.3, n>3, removed).

Statement of originality

This document contains material, which is the copyright of certain I^{2}PANEMA consortium parties, and may not be reproduced or copied without permission. This deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation, or both.

The information contained in this document is the proprietary confidential information of the I^{2}PANEMA consortium and may not be disclosed except in accordance with the consortium agreement.

The commercial use of any information contained in this document may require a license from the proprietor of that information.

Neither the project consortium as a whole nor a certain party of the consortium warrant that the information contained in this document is capable of use, nor that use of the information is free from risk and accepts no liability for loss or damage suffered by any person using this information.

The information in this document is subject to change without notice.

Table of Contents

Table of Contents ... 4

List of Tables ... 6

List of Figures ... 7

List of Acronyms ... 8

1. I^{2}PANEMA Security and Privacy Layer ... 9

1.1. Introduction ... 9

1.1.1. Threat modelling and Security Analysis ... 9

1.1.2. STRIDE methodology ... 9

1.1.3. DREAD methodology ... 10

1.1.4. Threat resolution methodology ... 11

1.2. Solution - Generic architecture for I^{2}PANEMA Security Layer ... 11

1.3. Relation with I^{2}PANEMA objectives and business scenarios ... 13

1.4. Business Scenario Port of Gijón (Environmental Measurement) ... 13

1.4.1. Current status and next steps ... 13

1.4.2. Summary ... 13

1.4.3. Security Analysis ... 15

1.4.4. Implementation of the architecture in Gijón ... 15

1.5. Business Scenario TriCon CTT (Active Noise System) ... 23

1.5.1. Current status and next steps ... 23

1.5.2. Summary ... 23

1.5.3. Security Analysis ... 24

1.5.4. Implementation of the architecture in TriCon ... 24

1.6. Business Scenario Port of Dortmund (Intelligent Parking Management) ... 24

1.6.1. Current status and next steps ... 24

1.6.2. Summary ... 24

1.6.3. Security Analysis ... 25

1.6.4. Implementation of the architecture in Port of Dortmund ... 25

1.7. Business Scenario Port of Wesel (Active Noise Control) ... 25

1.8. Business Scenario cross country (Smart Port/Ship) ... 25

1.9. Business Scenario Port of Hamburg (Smart Ferry) ... 25

1.10. Business Scenario Port of Hamburg (Onshore Power Supply) ... 25

1.11. Business Scenario Safi port (RoRo localization) ... 26

1.11.1. Current and Next steps ... 26

1.11.2. Summary ... 26

1.11.3. Security Analysis ... 26

1.11.4. Implementation of the architecture in RoRo Localization ... 26

1.12. Business Scenario Assan port (Container Localization) ... 27

2. I^{2}PANEMA Operational Workflow support Layer ... 28

2.1. Business Scenario Port of Gijón (Environmental Measurement) 29
2.1.1. Workflow of execution of alerts 29
2.1.2. Implementation 29
2.2. Business Scenario TriCon CTT (Active Noise System) 30
2.3. Business Scenario Port of Dortmund (Intelligent Parking Management) 30
2.3.1. IT-architecture 30
2.3.2. User Story 34
2.4. Business Scenario Port of Wesel (Active Noise Control) 37
2.5. Business Scenario cross country (Smart Port/Ship) 37
2.6. Business Scenario Port of Hamburg (Smart Ferry) 37
2.7. Business Scenario Port of Hamburg (Onshore Power Supply) 37
2.8. Business Scenario Safi port (RoRo localization) 37
2.8.1. IT-architecture 37
2.8.2. Controller overview 39
2.8.3. Interfaces to externals 39
2.8.4. User-story 39
2.9. Business Scenario Assan port (Container Localization) 39
2.9.1. Gate In/Vessel Discharge – Container Place on the Field 40
2.9.2. Gate Out/Truck Load 40
2.9.3. Shifting Container 41
2.9.4. Stacker Status Check 41
3. Conclusion 42

List of Tables

Table 1: Threat classification ... 9
Table 2: Mapping of STRIDE threats to entities in STRIDE threat model diagram ... 10
Table 3: DREAD methodology with its description ... 10
Table 4: How to solve the threat attending to it value after apply DREAD methodology ... 11
Table 5: Threat applicable on P1 ... **Fehler! Textmarke nicht definiert.**
Table 6: DREAD Ratings for Threats for P1 as listed in Table 1Table 5 ... **Fehler! Textmarke nicht definiert.**
Table 7: Threats and DREAD rating applicable on E1 ... **Fehler! Textmarke nicht definiert.**
Table 8: Threats and DREAD rating applicable on E2 ... **Fehler! Textmarke nicht definiert.**
Table 9: Threats and DREAD rating applicable on DS1 ... **Fehler! Textmarke nicht definiert.**
Table 10: Threats and DREAD rating applicable on DS2 ... **Fehler! Textmarke nicht definiert.**
Table 11: Threats and DREAD rating applicable on P1 ... **Fehler! Textmarke nicht definiert.**
Table 12: Threats and DREAD rating applicable on P2 ... **Fehler! Textmarke nicht definiert.**
Table 13: Threats and DREAD rating applicable on P3 ... **Fehler! Textmarke nicht definiert.**
Table 14: Threats and DREAD rating applicable on DF1 ... **Fehler! Textmarke nicht definiert.**
Table 15: Threats and DREAD rating applicable on DF2 ... **Fehler! Textmarke nicht definiert.**
Table 16: Threats and DREAD rating applicable on DF3 ... **Fehler! Textmarke nicht definiert.**
Table 17: Threats and DREAD rating applicable on DF4 ... **Fehler! Textmarke nicht definiert.**
Table 18: Threats and DREAD rating applicable on DF5 ... **Fehler! Textmarke nicht definiert.**
Table 19: Threats and DREAD rating applicable on DF6 ... **Fehler! Textmarke nicht definiert.**
Table 20: Threats applicable on E1 ... **Fehler! Textmarke nicht definiert.**
Table 21: Threats applicable on E2 ... **Fehler! Textmarke nicht definiert.**
Table 22: Threats applicable on DS1 ... **Fehler! Textmarke nicht definiert.**
Table 23: Threats applicable on DS2 ... **Fehler! Textmarke nicht definiert.**
Table 24: Threats applicable on P1 ... **Fehler! Textmarke nicht definiert.**
Table 25: Threats applicable on P2 ... **Fehler! Textmarke nicht definiert.**
Table 26: Threats applicable on P3 ... **Fehler! Textmarke nicht definiert.**
Table 27: Threats applicable on P4 ... **Fehler! Textmarke nicht definiert.**
Table 28: Threats applicable on DF1 & DF6 ... **Fehler! Textmarke nicht definiert.**
Table 29: Threats applicable on DF2, 2, 5 ... **Fehler! Textmarke nicht definiert.**
Table 30: Threats applicable on DF4 & DF7 ... **Fehler! Textmarke nicht definiert.**
Table 31: List of identified threats, DREAD ratings and planned / implemented resolution**Fehler! Textmarke nicht definiert.**

List of Figures

Figure 1: Threat modelling ... 9
Figure 2: Initial I^{2}PANEMA Reference Architecture ... 12
Figure 3: Generic layered IoT system architecture with possible security solutions for respective layers. ... 12
Figure 4: Latest implementation of the architecture in Gijon ... 14
Figure 5: I2PANEMA Model Diagram (Gijon) ... **Fehler! Textmarke nicht definiert.**
Figure 6: STRIDE model diagram with threats ... **Fehler! Textmarke nicht definiert.**
Figure 7: Create and administrator type user ... 16
Figure 8: Creating a new user for the environmental department ... 16
Figure 9: Retrieve all users ... 17
Figure 10: Search user by role ... 17
Figure 11: Search user by ID ... 17
Figure 12: No results finding for this parameter (email) ... 18
Figure 13: Edit user properties ... 18
Figure 14: Delete a user and their response ... 19
Figure 15: Login a specific user ... 19
Figure 16: Example of token generated ... 20
Figure 17: Log Out and their response ... 20
Figure 18: Get Info method and their response ... 21
Figure 19: Example of correct execution of a method ... 21
Figure 20: Usermane already exists ... 21
Figure 21: User not authorized to perform this action ... 22
Figure 22: Flow of access to the different APIs ... 22
Figure 23: Node-RED access window ... 23
Figure 24: Illustration of the TriCon business scenario objectives ... 23
Figure 25: Basic architecture view of the TriCon Business scenario ... **Fehler! Textmarke nicht definiert.**
Figure 26: Composing the STRIDE threat model diagram of the TriCon Business scenario**Fehler! Textmarke nicht definiert.**
Figure 27: Finalized STRIDE threat model diagram of the TriCon Business scenario**Fehler! Textmarke nicht definiert.**
Figure 28: IT architecture overview of the port of Dortmund business scenario ... 25
Figure 29: Basic architecture view of the port of Dortmund business scenario**Fehler! Textmarke nicht definiert.**
Figure 30: Composing the STRIDE threat model diagram view of the port of Dortmund business scenario ... **Fehler! Textmarke nicht definiert.**
Figure 31: Finalized STRIDE threat model diagram view of the port of Dortmund business scenario ... **Fehler! Textmarke nicht definiert.**
Figure 32: Illustration of the architecture of RoRo business scenario ... 26
Figure 33: Basic Architecture overview RoRo BS ... **Fehler! Textmarke nicht definiert.**
Figure 34: STRIDE model diagram with threats ... **Fehler! Textmarke nicht definiert.**
Figure 35: STRIDE threat model diagram view of the container localization business scenario ... 27
Figure 36: OWL. Communication among actors, resources and devices ... 28
Figure 37: Status of a Workflow ... 28
Figure 38: Workflow of alerts execution ... 29
Figure 39: Alerts flow ... 30
Figure 40: Role management within the OWL ... 31
Figure 41: Overview platform IT-architecture BS Dortmund ... 32
Figure 42: User-story Admin ... 34
Figure 43: User-story Planner ... 35
Figure 44: User-story Endpoint User ... 36
Figure 45: IT Infrastructure of RoRo business scenario ... 38
Figure 46: Business Process ... 39

List of Acronyms

Acronym	Explanation
API	**A**pplication **P**rogramming **I**nterface
CRUD	**C**reate, **R**ead, **U**pdate and **D**elete
DREAD	**D**amage Potential, **R**eproducibility, **E**xploitability, **A**ffected users, **D**iscoverability
JWT	**J**SON **W**eb **T**oken
MVC	**M**odel – **V**iew **-** **C**ontroller
OW	**O**perational **W**orkflow
OWL	**O**perational **W**orkflow **L**ayer
PMS	**P**ort **M**anagement **S**ystem
REST	**Re**presentational **S**tate **T**ransfer
STRIDE	**S**poofing, **T**ampering, **R**epudiation, **I**nformation Disclosure, **D**enial of Service, **E**levation of Privilege
UI	User Interface
WP	**W**ork **P**ackage
XACML	e**X**tensible **A**ccess **C**ontrol **M**arkup **L**anguage

1. I^{2}PANEMA Security and Privacy Layer

The I^{2}PANEMA Security and Privacy Layer is the vertical layer responsible for ensuring proper access and control to the data or operations of the platform.

Some of the data collected by e.g. sensors or Port Management System (PMS) are sensitive data, to which access is restricted. Special attention should be given to prevent access to such data by unauthenticated or authorized persons. Considering privacy as a mechanism, a Policy Based Access Control Mechanism security system, based on the OASIS e**X**tensible **A**ccess **C**ontrol **M**arkup **L**anguage (XACML) is recommended to be adopted.

1.1. Introduction

This section gives an overview of the methodologies applied for the I2Panema security layer.

1.1.1. Threat modelling and Security Analysis

An iterative (model-based) process to find and address threats to any IT solution is illustrated below.

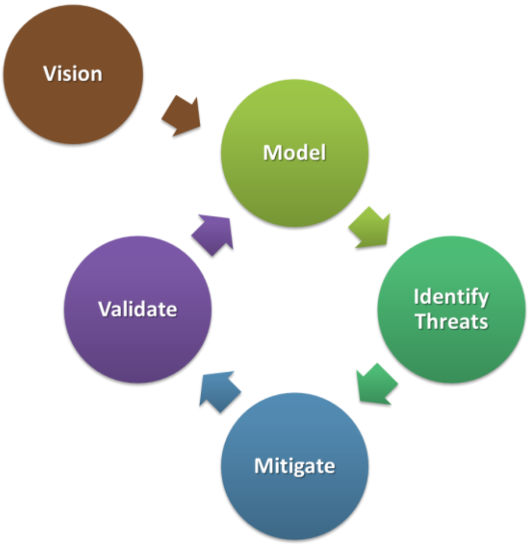


Figure 1: Threat modelling

Within this threat modelling, the STRIDE methodology has been proven to be useful. Therefore, we decided to apply it for the security analysis needs of the I^{2}PANEMA project in its business scenarios. It is explained in the following.

1.1.2. STRIDE methodology

STRIDE[1] is a model of threats developed by Praerit Garg and Loren Kohnfelder at Microsoft for identifying computer security threats. It provides a mnemonic for identifying security threats in six categories as illustrated in the table below:

Table 1: Threat classification

Threat	Security Property	Description
Spoofing	Authentication	Pretending to be something or someone other than yourself
Tampering	Integrity	Modifying something on disk, on a network or in a memory
Repudiation	Non-repudiation	Claiming that you didn't do something, or were not responsible
Information disclosure	Confidentiality	Providing information to someone not authorized to see it: **data leak** or **privacy break**
Denial of service	Availability	Absorbing resources needed to provide service

[1] https://en.wikipedia.org/wiki/STRIDE_(security)

Elevation of privilege	Authorization	Allowing someone to do something they are not authorized to do

These threat categories are mapped towards the system architectures entities according to the following mapping. Not all threats apply to all entities.

Table 2: Mapping of STRIDE threats to entities in STRIDE threat model diagram

Element	Threat					
	Spoofing	Tampering	Repudiation	Information disclosure	Denial of Service	Elevation of privilege
External Entity	✓		✓			
Process	✓	✓	✓	✓	✓	✓
Data Store		✓	✓ (if logged)	✓	✓	
DF (Data Flow)		✓		✓	✓	

1.1.3. DREAD methodology

In addition to STRIDE threat modelling, the DREAD methodology is used. With this methodology each threat is divided into 5 characteristics that can be easily rated from 1 to 3 points. An overall DREAD rating to serve as a guide towards the severity of the threat the range of 1 (uncritical) to 3 (severe) is obtained with the following formula:

$$minimum\left(\boldsymbol{D}amage; \frac{\boldsymbol{D}amage + \boldsymbol{R}eproducibility + \boldsymbol{E}xploitablity + \boldsymbol{A}ffectedUsers + \boldsymbol{D}iscoverability}{5}\right)$$

Table 3: DREAD methodology with its description

Risk	Risk Property	Description
Damage potential	How great can be the damage?	1pt (low): Leaking trivial information 2pts (medium): Leaking sensitive information 3pts (high): Can subvert the security system
Reproducibility	How easy to reproduce?	1pt (low): Very difficult to reproduce, even with knowledge of the security hole 2pts (medium): Can be reproduced, but only with a timing window and a particular situation 3pts (high): Can be reproduced every time and doesn’t require any particular situation
Exploitability	How easy to realize this threat?	1pt (low): Requires an extremely skilled person and in-depth knowledge every time to exploit 2pts (medium): A skilled programmer could make the attack, then repeat the steps

		3pts (high): A novice programmer could make the attack in a short time
Affected users	How many users are affected (%)?	1pt (low): Very small (%) of users, obscure feature; affects anonymous users 2pts (medium): Some users, non-default configuration 3pts (high): All users, default configuration, key customer
Discoverability	How easy to find this vulnerability?	1pt (low): The bug is obscure, and it's unlikely that users will work out damage potential 2pts (medium): Located in a seldom-used part, and only a few users should come across it 3pts (high): The vulnerability is located in the most commonly feature and is very noticeable

1.1.4. Threat resolution methodology

After having assigned the threat severity or DREAD rating for each threat, the decision are taken on how to deal with each of the identified threats. The following table lists the possible resolution options. The identified threat severity via DREAD can serve as a guidance in order to decide which on threats to focus the efforts and which threats might even be acceptable in the context.

Table 4: How to solve the threat attending to it value after apply DREAD methodology

Threat	Resolution
Accept risk	Doing nothing Works best when it's your risk Be really careful about accepting risk for customer!
Transfer risk	Pass risk to an externality (license agreements, term of service …) If happening silently can lead to unhappy customers
Avoid risk	Remove feature/component that causes the risk
Mitigate risk	Add/use technology to prevent attacks Developers and sysadmins have different toolkits for mitigating problems Use standard available approaches which have been tested & worked through if you need a custom solution, ask an expert

1.2. Solution - Generic architecture for I^{2}PANEMA Security Layer

The (draft / basic) reference architecture for I^{2}PANEMA can be seen below[2].

[2] This draft architecture has been defined early in the beginning of the project as baseline for further discussions within the project and is to be used to assist in driving the project's initial developments. A consolidated and more elaborated version of this reference architecture is planned to be documented in V2 of D1.3/D1.4.

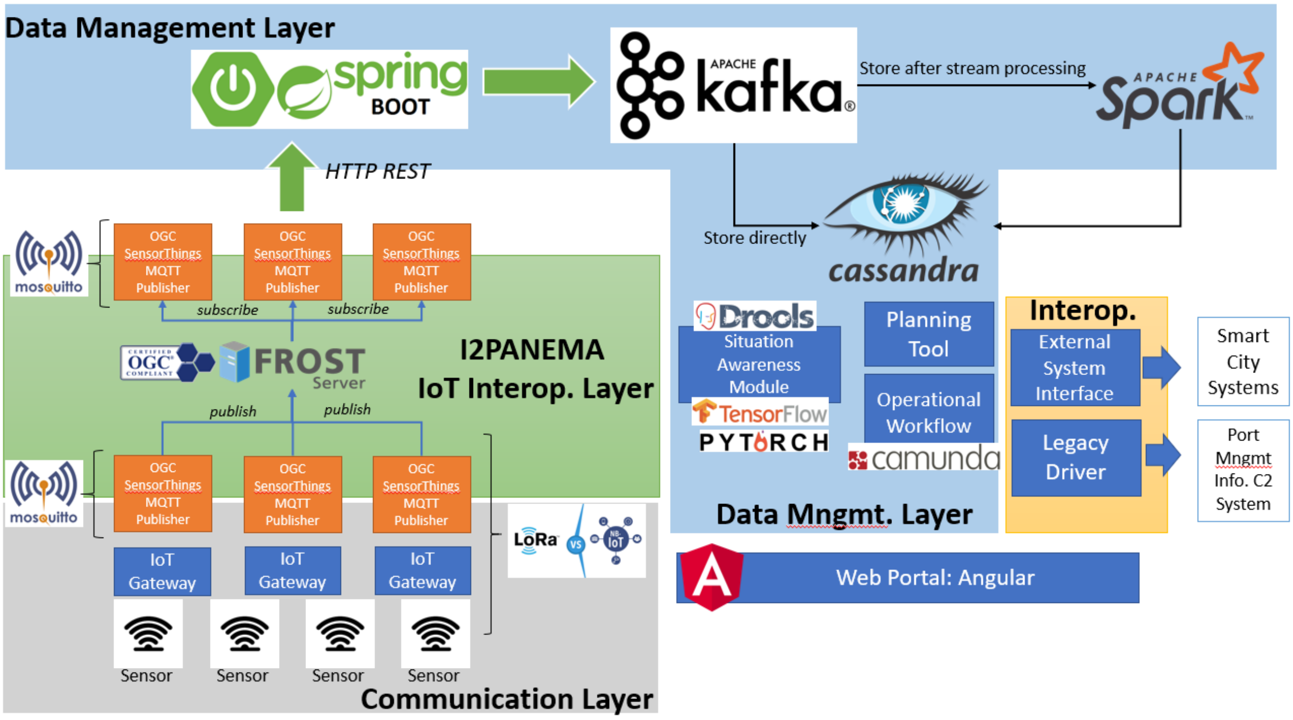


Figure 2: Initial I^{2}PANEMA Reference Architecture

Security is an inherent feature for any architecture. Basically, security for a specific architecture approach can be defined. But to define security as a general feature would be a difficult task as there is no one-size-fits-all solution. Hence, each business scenario has to be reviewed individually in this respect in order to determine its threats and flaws. Each scenario brings its own unique situations and risks. These risks need to be evaluated on a case-by-case basis and the security be tailored around these risks.

When defining the architecture of any system, one encounters many different layers of the system, for example, hardware, operating system, application, etc. One should, however, understand that security of any system cannot be defined on one such horizontal layer. Each layer will have components that require security in its own form. Hence, one must not expect a clear separation of a horizontal security layer in the definition of architecture of a system. However, one can identify particular security features that can be implemented for every horizontal layer and then be portrayed as a vertical layer within this system architecture.

As discussed, in detail, in I^{2}PANEMA deliverable D1.5, Security and Privacy will be implemented as a vertical layer. The vertical security layer can be visualized as in Figure 3, which also features selected security mitigations.

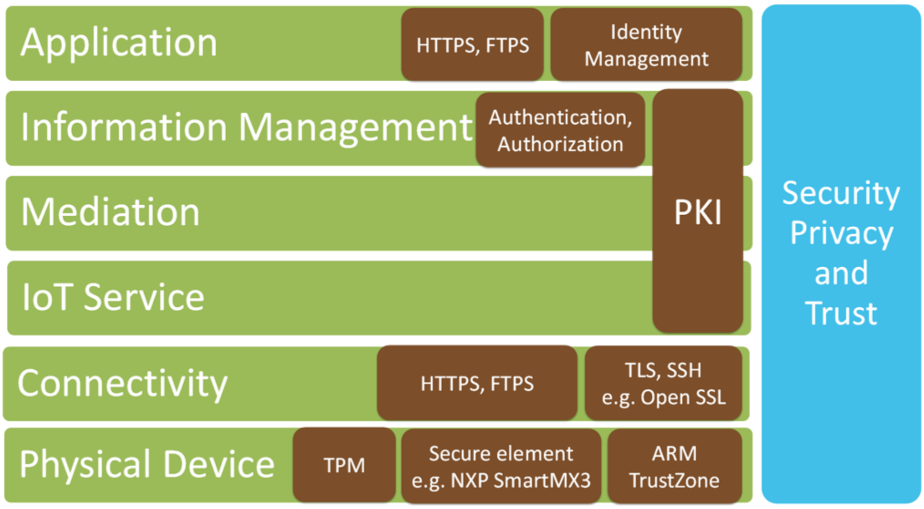


Figure 3: Generic layered IoT system architecture with possible security solutions for respective layers.

1.3. Relation with I²PANEMA objectives and business scenarios

Currently, recent IT innovations have not yet fully arrived at the world of ports. I²PANEMA is therefore aiming to deploy the full power of IoT to improve port operations in order to make them more efficient and to contribute to make them more sustainable. To reach this objective, business scenarios with different concepts, e.g. environmental measurement or an intelligent parking management system, have been developed.

The following sections of chapter 1 list the descriptions of the security layer for each business scenario respectively with the underlying performed security analysis, a description of the resolutions for the identified security threats and implementation in the business scenario and the current status.

1.4. Business Scenario Port of Gijón (Environmental Measurement)

1.4.1. Current status and next steps

The Gijon business scenario describes the development of a platform for the measurement of PM10 particles in and around the environment of the Port of Gijon. A more detailed description on the use cases considered in the business scenario for port of Gijon can be found in the first version of this deliverable. The STRIDE security analysis that was in progress when this first version of this deliverable has been submitted has been completed in the meantime. Based on the results and in alignment with the Port of Gijon decisions have been taken on how to resolve the identified security threats. The following sections give an overview of the business scenarios architecture, a brief recap of the steps of the security analysis performed in v1 of this deliverable. They are followed by an overview of the chosen resolutions and their implementation.

With this all threats planned to be resolved in the scope of the project are done. Consequently, there will be no further actions on the business scenarios security layer in the scope of the I2Panema project.

1.4.2. Summary

I2Panema Architecture

The following picture depicts the latest version of the I2PANEMA architecture for Gijon pilot.

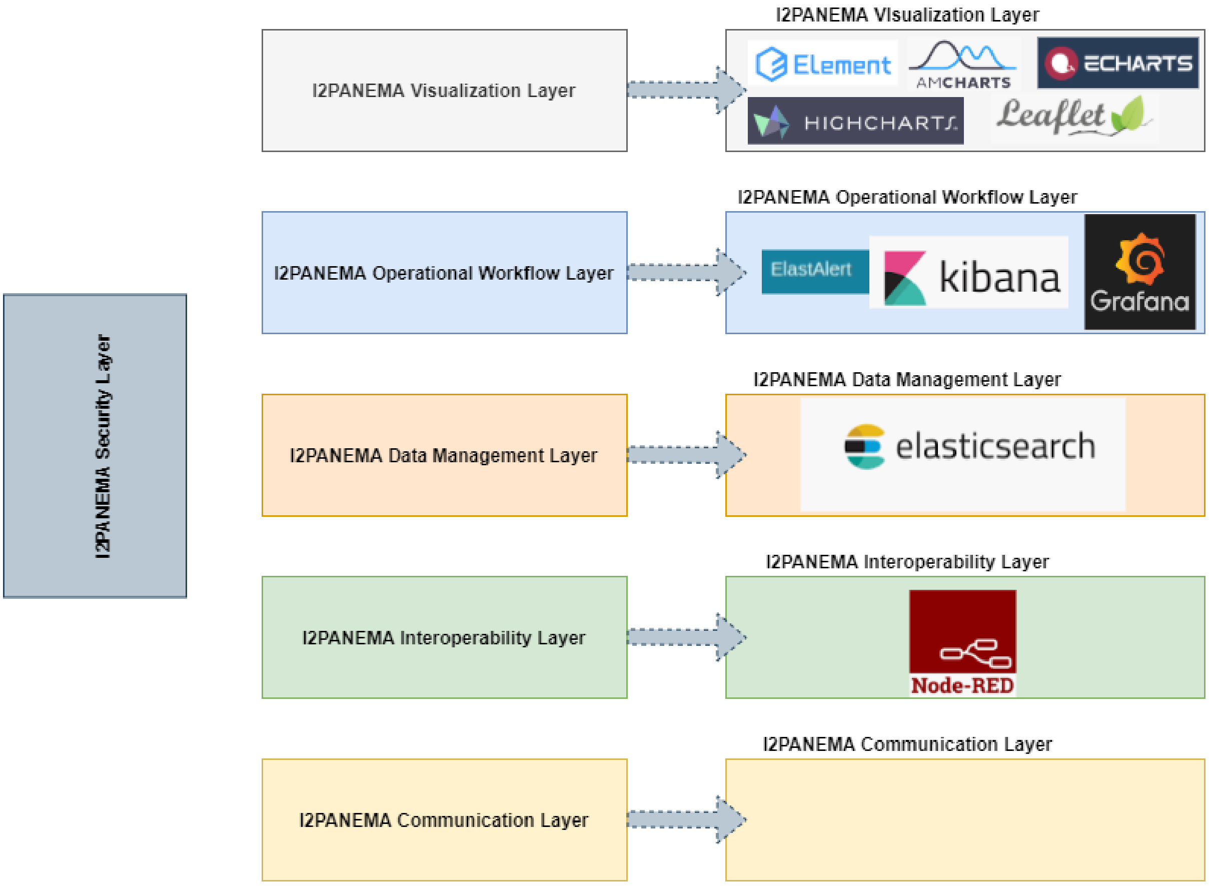

I2PANEMA Security Layer
I2PANEMA Visualization Layer
I2PANEMA VIsualization Layer
Element
AMCHARTS
ECHARTS
HIGHCHARTS
Leaflet
I2PANEMA Operational Workflow Layer
I2PANEMA Operational Workflow Layer
ElastAlert
kibana
Grafana
I2PANEMA Data Management Layer
I2PANEMA Data Management Layer
elasticsearch
I2PANEMA Interoperability Layer
I2PANEMA Interoperability Layer
Node-RED
I2PANEMA Communication Layer
I2PANEMA Communication Layer

Figure 4: Latest implementation of the architecture in Gijon

This version includes some differences with respect to the implementation that was in v1 of the deliverable. This version is the final version and therefore will not undergo any changes in the last months of the project.

The next sections will focus on the changes compared to the previous version with respect to the implementation of the security layer. It describes the current implementation of this layer and the changes implemented as a result of the STRIDE security analysis presented in the first version of the deliverable.

Differences with previous version of I2PANEMA Architecture

Finally, no broker has been included in the Interoperability layer. Data are retrieved directly and with flows defined in **Node-RED**[3] are inserted directly in the Data Management layer. The discarded option was **Kafka**[4]. One of the main reasons is that not having message routing would require the inclusion of a new component in the architecture prior to the insertion of data in the Data Management layer. This new element would be **Logstash**[5]. Therefore, for simplicity, and because our platform does not handle such a large volume of data, the use of a broker has been discarded.

Regarding the use of rule execution engines for the definition of alerts, their use has been discarded due to the specificity of our alerts. Therefore, no use will be made of **elastAlert**[6] or the alert definition functionality provided by **Grafana**[7]. In the Data Management layer, only **elasticsearch**[8] will be the system responsible for storing the data.

[3] https://nodered.org/
[4] https://kafka.apache.org/
[5] https://www.elastic.co/logstash/
[6] https://github.com/Yelp/elastalert
[7] https://grafana.com/docs/grafana/latest/alerting/
[8] https://www.elastic.co/

1.4.3. Security Analysis

This section has been removed from the public version.

1.4.4. Implementation of the architecture in Gijón

However, although the port of Gijon accepts the risks, security measures have been developed or taken to cover some of the identified risks. These measures include the following:

- Development of a role-based platform access API in order to cope with the identified spoofing threats
- Management of data access from a single point: Node-RED in order to cope with the identified tampering and information disclosure threats
- Securing access to Node-RED in order to cope with the identified tampering and information disclosure threats

The development of these measures will be described in the upcoming sections.

Development of a role-based platform access API

This role-based authentication system was developed after several discussions with the port of Gijon and the following roles were identified:

- Admin
- Environmental Department
- IT Department
- Security Department

Depending on the access role to the platform, some menu options will be enabled or others.

This REST API it's a **CRUD API** that follows the **MVC** pattern. It was developed in **Node.JS** and implements the **JWT** protocol. It works by storing the information in **MongoDB** (No-SQL DB).

I2PANEMA will not have a section (UI) to manage users. That is, the one where we can register, delete or update data about them. All this will be managed through this API and its different methods.

An easy and visual way to manage user management is through tools such as **POSTMAN**[9].

POSTMAN is a client API frequently used for developers to create, share, test and document their APIs. It will be used in next section to explain the different methods of the API.

Methods available

The methods of the API have been divided into two main blocks:

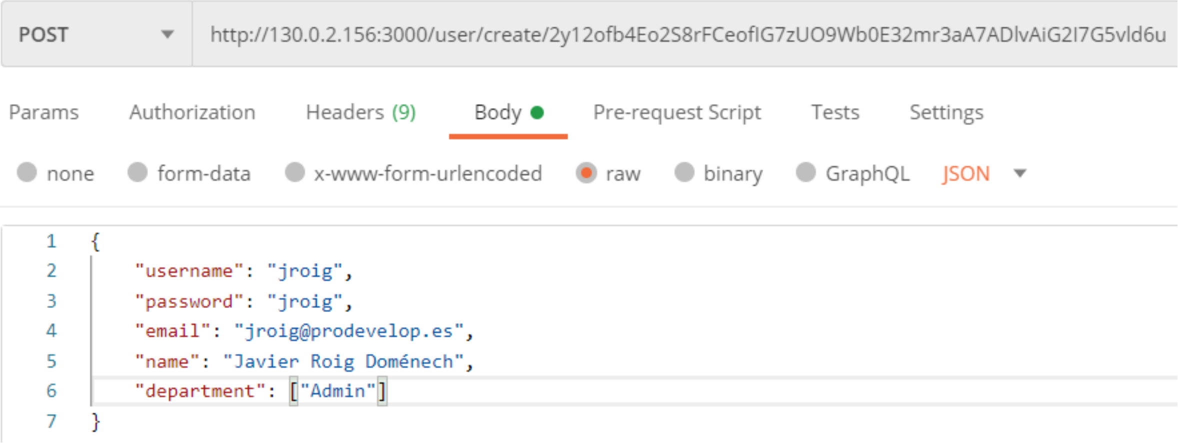
- **USER**. In this section there are the methods responsible for managing users: creating, deleting, updating.
- **AUTH**. This section contains the methods that validate the access to the platform: login, logout, etc.

USER

- *Create user as an administrator*.
  - **Method**: POST
  - **Body**: It will contain the needed properties as is depicted in the picture.
  - **Header**: "Content-Type: application/json"
  - **Endpoint**: /user
  - **Additional considerations**: A special code is needed to allow the insertion of the record in the database. This will only be necessary when the first record is inserted in the database. This code is called **admin code**.

[9] https://www.postman.com/

- **Admin code**: *2y12ofb4Eo2S8rFCeofIG7zUO9Wb0E32mr3aA7ADlvAiG2I7G5vld6u*



http://130.0.2.156:3000/user/create/2y12ofb4Eo2S8rFCeoflG7zUO9Wb0E32mr3aA7ADlvAiG2I7G5vld6u

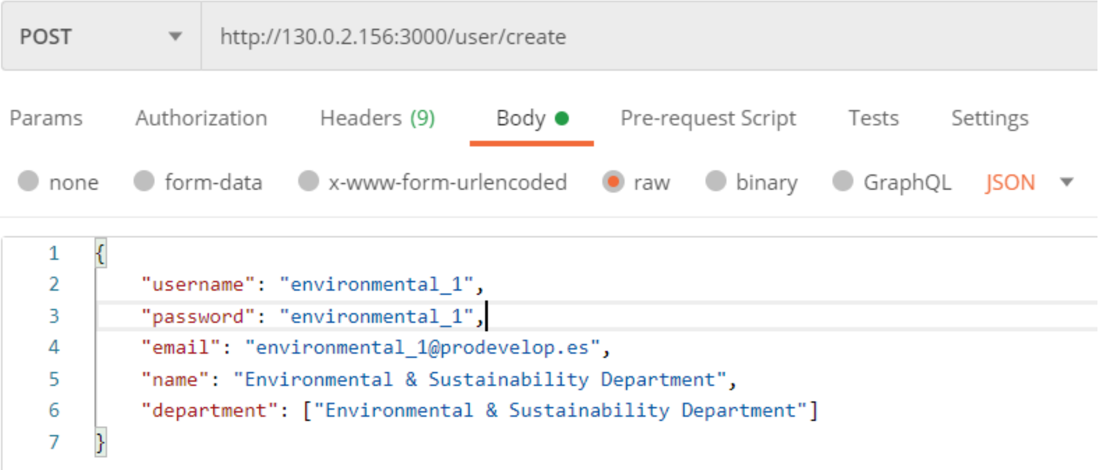
```
{
    "username": "jroig",
    "password": "jroig",
    "email": "jroig@prodevelop.es",
    "name": "Javier Roig Doménech",
    "department": ["Admin"]
}
```

Figure 5: Create and administrator type user

- *Create a user*.
  - **Method**: POST
  - **Body**: It will contain the needed properties (such as role) as is depicted in the picture.
  - **Header**: Two headers must be included:
    - "Content-Type: application/json"
    - "x-token": "value"

    The value of the header x-token is obtained by logging in with a user in the platform.
  - **Endpoint**: /user

  The only department values accepted in the platform are "Admin", "Environmental & Sustainability Department", "IT Department", "Security Department"



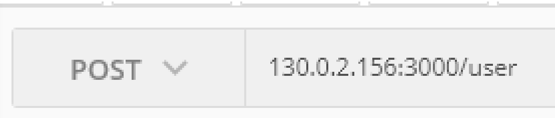
http://130.0.2.156:3000/user/create

```
{
    "username": "environmental_1",
    "password": "environmental_1",
    "email": "environmental_1@prodevelop.es",
    "name": "Environmental & Sustainability Department",
    "department": ["Environmental & Sustainability Department"]
}
```

Figure 6: Creating a new user for the environmental department

- *Retrieve all users registered on the platform*.
  - **Method**: POST
  - **Body**: Not necessary. No filtering by any parameter.
  - **Header**: Two headers must be included:
    - "Content-Type: application/json"
    - "x-token": "value"

    The value of the header x-token is obtained by logging in with a user in the platform.
  - **Endpoint**: /user

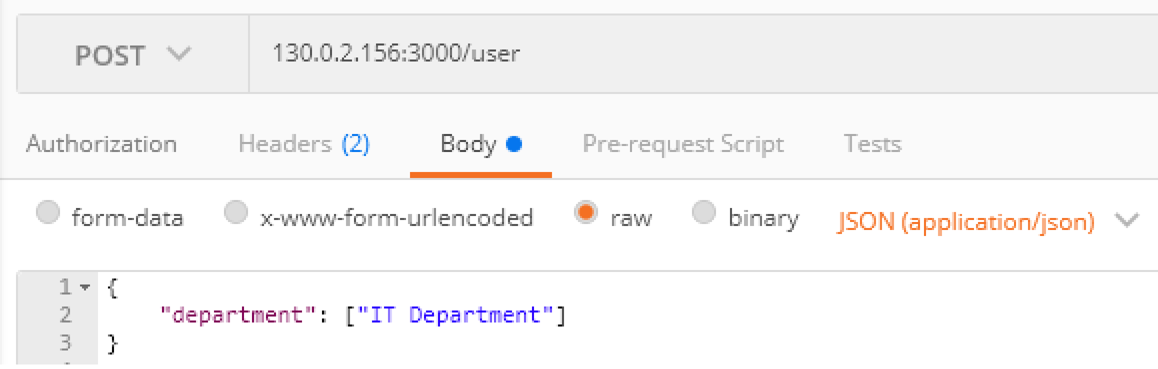


POST 130.0.2.156:3000/user

Figure 7: Retrieve all users

- *Search users (array of users) by role.*
    - **Method**: POST
    - **Body**: It will contain the property that we will filter through. In this case **department** (see the picture).
    - **Header**: Two headers must be included:
        - "Content-Type: application/json"
        - "x-token": "value"

      The value of the header x-token is obtained by logging in with a user in the platform.
    - **Endpoint**: /user



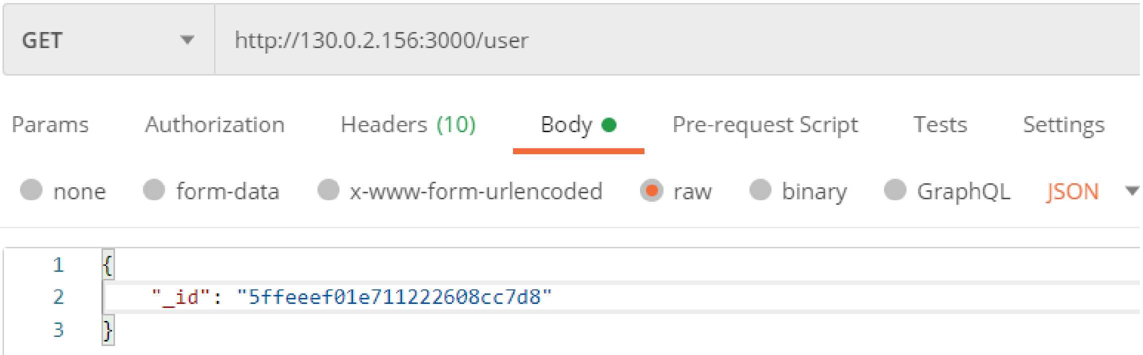
POST 130.0.2.156:3000/user

```
{
    "department": ["IT Department"]
}
```

Figure 8: Search user by role

- *Search user (only one) by ID.*
    - **Method**: GET
    - **Body**: It will contain the property we are going to filter through (see the picture).
    - **Header**: Two headers must be included:
        - "Content-Type: application/json"
        - "x-token": "value"

      The value of the header x-token is obtained by logging in with a user in the platform.
    - **Endpoint**: /user



GET http://130.0.2.156:3000/user

```
{
    "_id": "5ffeeef01e711222608cc7d8"
}
```

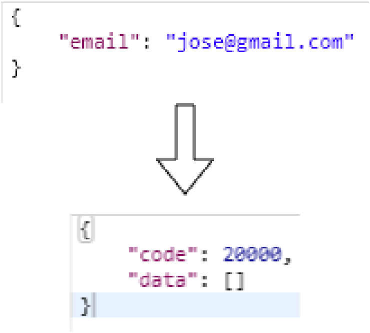
Figure 9: Search user by ID

- *Search users (array of users) by other properties than id.*

- **Method**: POST
- **Body**: It will contain the property we are going to filter through (see the picture).
- **Header**: Two headers must be included:
    - "Content-Type: application/json"
    - "x-token": "value"

  The value of the header x-token is obtained by logging in with a user in the platform.
- **Endpoint**: /user

In case no results are obtained by the search parameters. A json with code 20000 is obtained indicating that the request has been executed correctly. But whose answer will be empty.



```
{
    "email": "jose@gmail.com"
}
```

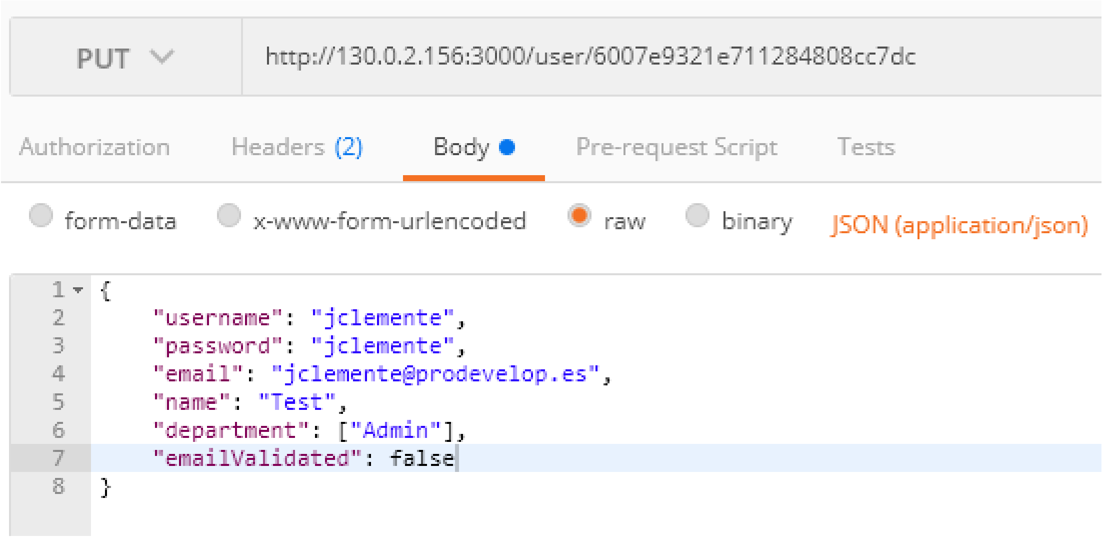
```
{
    "code": 20000,
    "data": []
}
```

Figure 10: No results finding for this parameter (email)

- *Edit user properties*.
    - **Method**: PUT
    - **Body**: It will contain all the properties. Even those we are to update (see the picture).
    - **Header**: Two headers must be included:
        - "Content-Type: application/json"
        - "x-token": "value"

      The value of the header x-token is obtained by logging in with a user in the platform.
    - **Endpoint**: /user/:id

  The id value could be recovered making the query that recovers all the users logged on the platform.



PUT http://130.0.2.156:3000/user/6007e9321e711284808cc7dc

○ form-data ○ x-www-form-urlencoded ● raw ○ binary JSON (application/json)

```
{
    "username": "jclemente",
    "password": "jclemente",
    "email": "jclemente@prodevelop.es",
    "name": "Test",
    "department": ["Admin"],
    "emailValidated": false
}
```

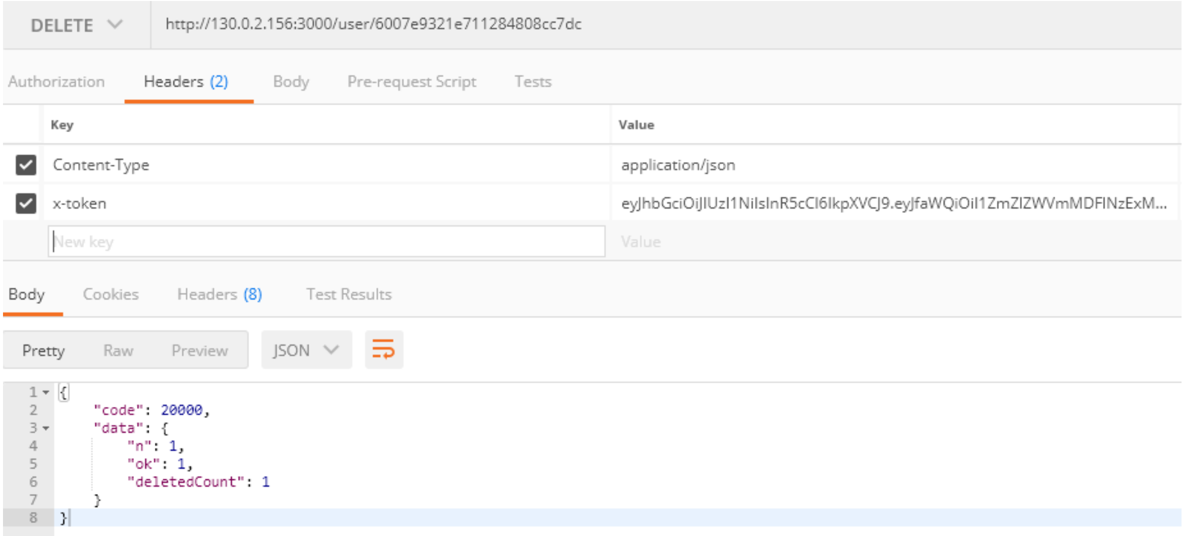
Figure 11: Edit user properties

- *Delete a user*.
    - **Method**: DELETE

- **Body**: Not necessary.
- **Header**: Two headers must be included:
  - "Content-Type: application/json"
  - "x-token": "value"

  The value of the header x-token is obtained by logging in with a user in the platform.
- **Endpoint**: /user/:id

The id value could be recovered making the query that recovers all the users logged on the platform.



DELETE http://130.0.2.156:3000/user/6007e9321e711284808cc7dc

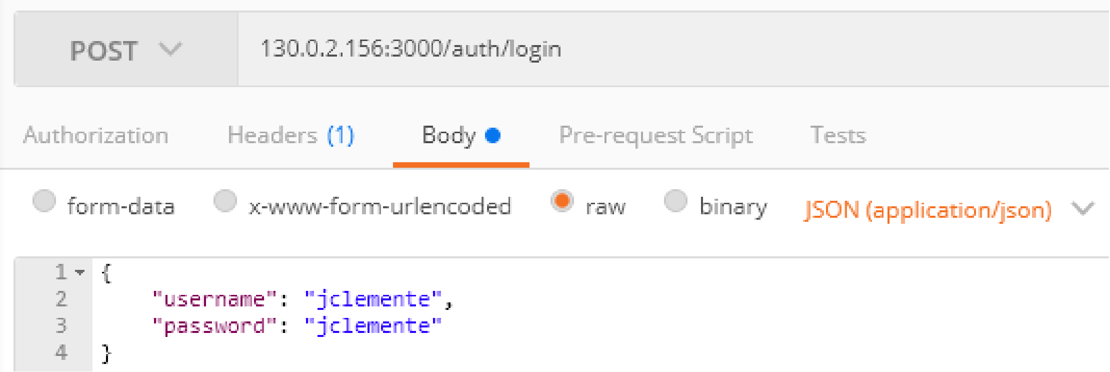
	Key	Value
☑	Content-Type	application/json
☑	x-token	eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJfaWQiOiI1ZmZlZWVmMDFlNzExM...
	New key	Value

```
{
    "code": 20000,
    "data": {
        "n": 1,
        "ok": 1,
        "deletedCount": 1
    }
}
```

Figure 12: Delete a user and their response

AUTH

- *Login a user*.
  - **Method**: POST
  - **Body**: It will contain the needed properties as is depicted in the picture.
  - **Header**: "Content-Type: application/json"
  - **Endpoint**: /auth/login



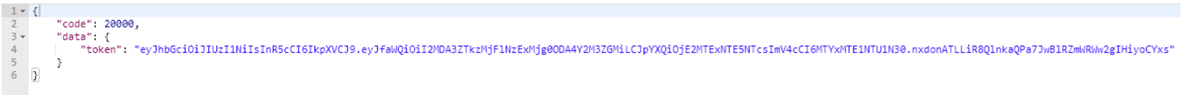
POST 130.0.2.156:3000/auth/login

○ form-data ○ x-www-form-urlencoded ● raw ○ binary JSON (application/json)

```
{
    "username": "jclemente",
    "password": "jclemente"
}
```

Figure 13: Login a specific user

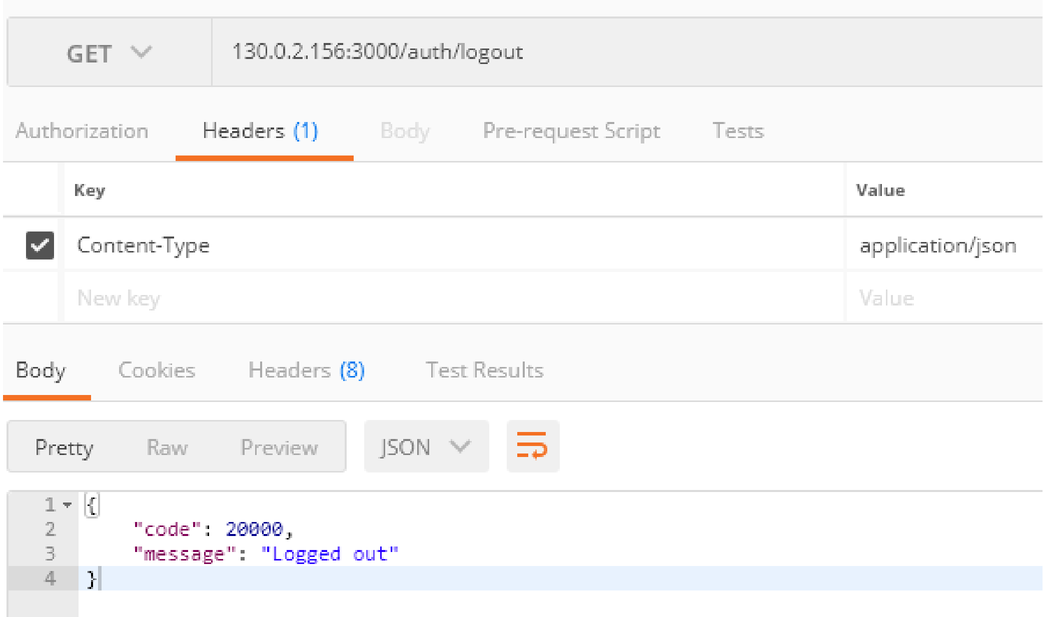
The answer to this method is a json with the code 20000 if it went well. In addition to the token generated. This token is valid for 1 hour and we can use it to create users identified as the last user we registered with.



```
{
    "code": 20000,
    "data": {
        "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJfaWQiOiI2MDA3ZTkzMjFlNzExMjg0ODA4Y2M3ZGMiLCJpYXQiOjE2MTExNTE5NTcsImV4cCI6MTYxMTE1NTU1N30.nxdonATLLiR8QlnkaQPa7JwBlRZmWRWw2gIHiyoCYxs"
    }
}
```

Figure 14: Example of token generated

- *Log Out*.
    - **Method**: GET
    - **Body**: Not necessary.
    - **Header**: “Content-Type: application/json”
    - **Endpoint**: /auth/logout



	Key	Value
☑	Content-Type	application/json
	New key	Value

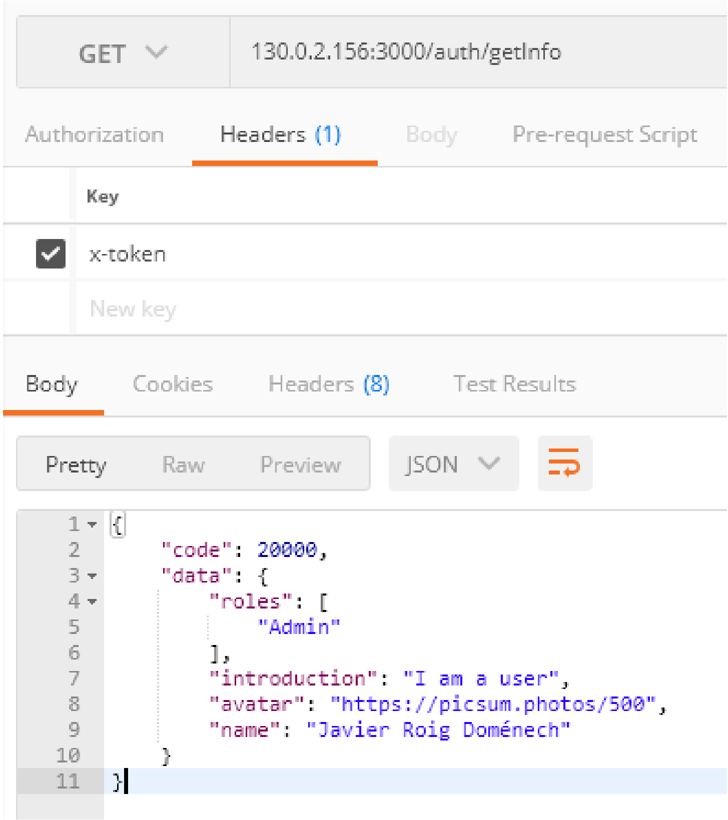
```
{
    "code": 20000,
    "message": "Logged out"
}
```

Figure 15: Log Out and their response

- *Get Info*.
    - **Method**: GET
    - **Body**: Not necessary.
    - **Header**:
        - “x-token”: “value”

        The value of the header x-token is obtained by logging in with a user in the platform.
    - **Endpoint**: /auth/getInfo

    This endpoint gets the information that is needed from the client application.



130.0.2.156:3000/auth/getInfo

☑ x-token

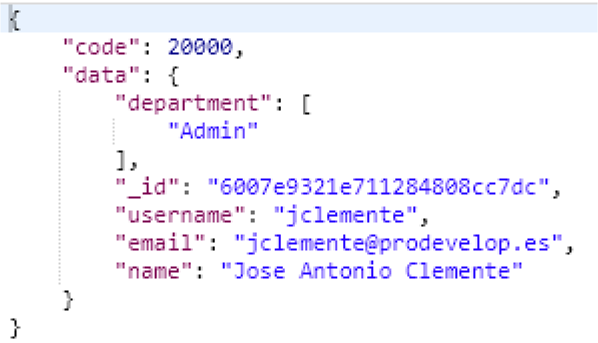
```
{
    "code": 20000,
    "data": {
        "roles": [
            "Admin"
        ],
        "introduction": "I am a user",
        "avatar": "https://picsum.photos/500",
        "name": "Javier Roig Doménech"
    }
}
```

Figure 16: Get Info method and their response

Issues & Solution

This section aims to give an insight into some issues & solutions that can happen when working with the API.

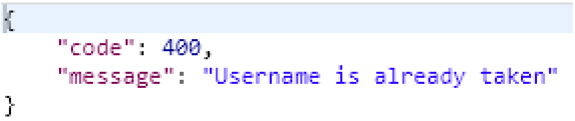
- **How to know if the execution of an API method has gone well?** If the execution of a method has gone well we will always get a json message whose will be **20000**. This json will also contain information regarding the execution of the method.



```
{
    "code": 20000,
    "data": {
        "department": [
            "Admin"
        ],
        "_id": "6007e9321e711284808cc7dc",
        "username": "jclemente",
        "email": "jclemente@prodevelop.es",
        "name": "Jose Antonio Clemente"
    }
}
```

Figure 17: Example of correct execution of a method

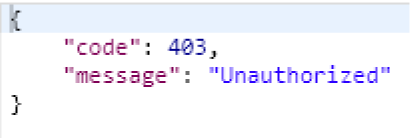
- **What messages we get if a user already exists in the application / platform?** If the user already exists on the platform we will get the following message:



```
{
    "code": 400,
    "message": "Username is already taken"
}
```

Figure 18: Usermane already exists

- **What happens if a user is created without passing the token?** A json message as the one shown in the figure will be obtained as a response. That message also indicates that we are not authorized to perform this action.



```
{
    "code": 403,
    "message": "Unauthorized"
}
```

Figure 19: User not authorized to perform this action

- **Where should the token be required to perform the actions be indicated?** The token has to be introduced as a header using the key "x-token".
- **How do we recover the token that must be included in the requests?** To recover the token the user has to login using the "auth/login" endpoint
- **How long is the token valid for?** The token is valid for 1 hour. Once that time has passed the access to the application is revoked and the user has to login again

Management of data access from a single point: Node-RED

The platform is divided into two components: client and server.

The server is an API developed in Node.JS from which requests are made to all the data integrated in the platform.

In order to control access to all data sources, requests to sensors or other data that are integrated in the platform are made from Node-RED. That is, the client would go to the server and from the server the requests are made to Node-RED. In this way, everything is centralized and in an isolated environment.

The following picture illustrates this flow:

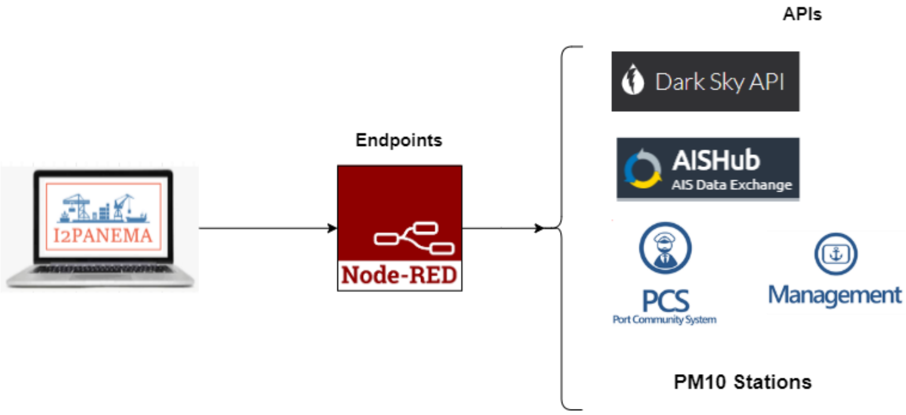


Figure 20: Flow of access to the different APIs

Securing access to Node-RED

Since all access management to sensors and other external data sources is done from Node-RED, it was essential to control access to the Node-RED url.

This is achieved from the Node-RED settings.js file. A password and user must be indicated in the file. The following image depicts the Node-RED access window when access is secured.

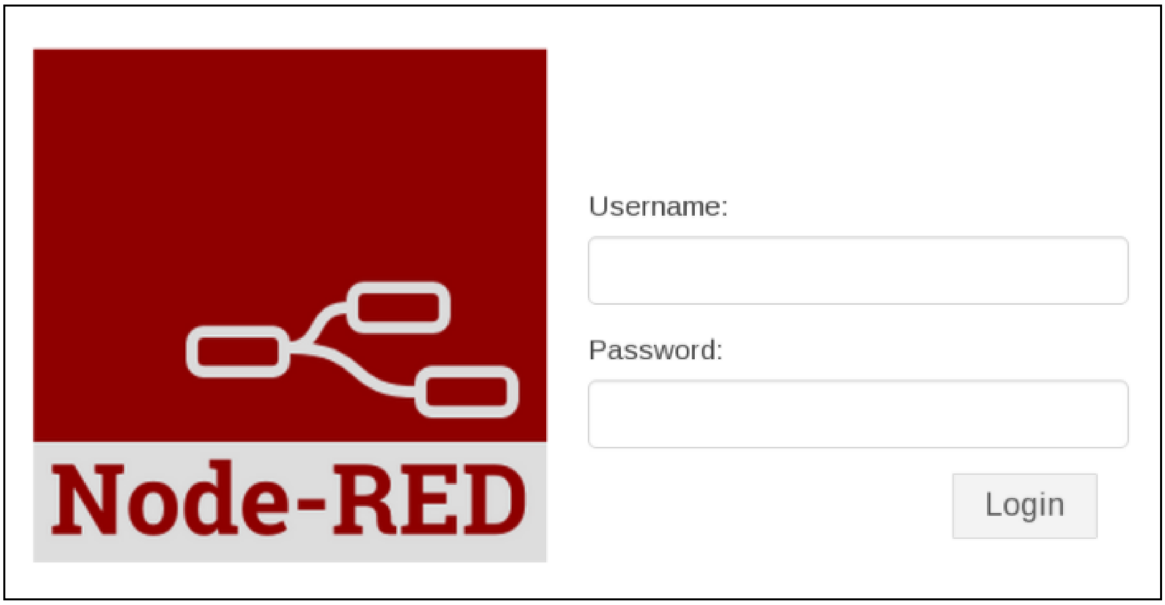


Figure 21: Node-RED access window

1.5. Business Scenario TriCon CTT (Active Noise System)

1.5.1. Current status and next steps

For the TriCon Business Scenario the initial STRIDE security analysis is also close to completion. All the scenario's assets have been identified. Based on the present architecture, the different potential security threats have been identified with the STRIDE methodology. All threats have been assigned a DREAD severity rating, which allows to prioritize the threats. Based on these potential threats, resolutions like mitigations have been identified and are currently being reviewed.

A review of the threat resolutions will be finalized in the upcoming weeks and will finalize the initial STRIDE security analysis. The chosen threat resolutions and recommendations then contribute to form the basis for the technical implementation of the business scenario.

1.5.2. Summary

This business scenario's objective is the reduction of noise emissions from handling operations in the Container Terminal Nuremberg by means of an active noise cancelling (ANC) system. Further goals are the improvement of the situation between port and residential areas as well as an increase of handling operations on the Combined Transport Terminals (CTTs).

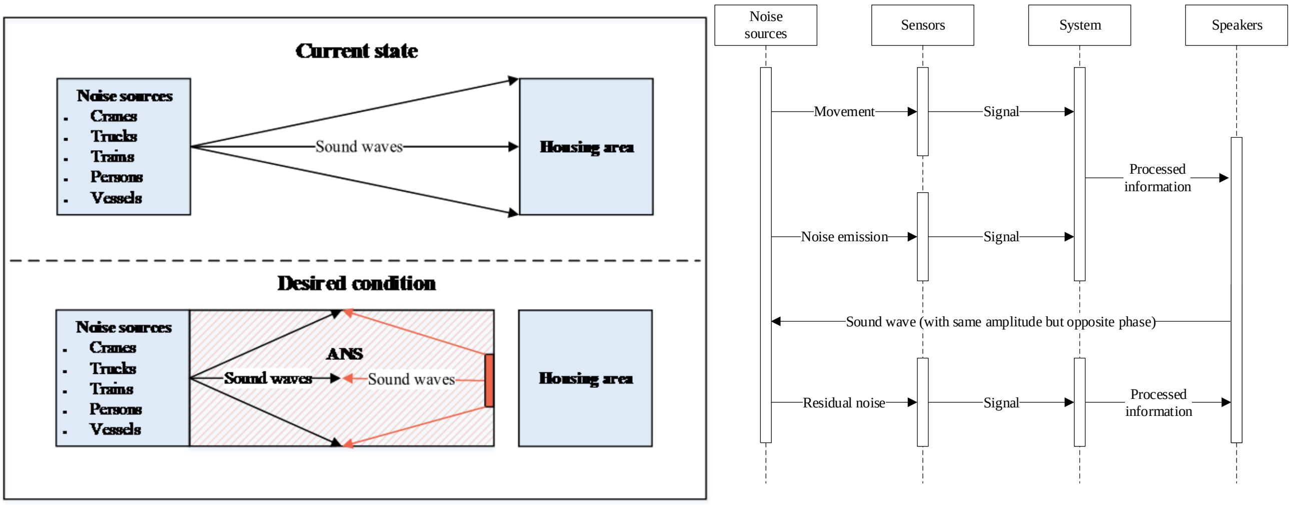


Figure 22: Illustration of the TriCon business scenario objectives

Description of ANC process in the business scenario

- User with physical access can switch IoT device on and off, modify wiring and hardware components (friendly/hostile)
- SBC (Single Board Computer) processes microphone data to generate loudspeaker signals
- Speaker (LS) emits sound to neutralize environmental noise
- IoT device (ANC system) provides filtered and compressed microphone data
- IoT device (ANC system) provides performance indicators
- Gateway transmits data to local server hosting the web application
- End user uses the web application connected via intra-/internet to view live/historical data and change settings
- Remote user can remotely connect to view live/historical data

1.5.3. Security Analysis

This section has been removed from the public version.

1.5.4. Implementation of the architecture in TriCon

Review of the threat resolutions will be finalized in the upcoming weeks and will finalize the initial STRIDE security analysis. The chosen threat resolutions and recommendations then form a base for the technical implementation of the business scenario.

1.6. Business Scenario Port of Dortmund (Intelligent Parking Management)

1.6.1. Current status and next steps

For the business scenario Port of Dortmund, the threat identification phase of the STRIDE analysis is almost completed. The systems assets have been identified and the architecture has been described. Based on this present architecture, the systems security threats have been identified with the STRIDE methodology and two iterations of feedback have been given on those from NXP side. The feedback is currently applied to the analysis document, which will conclude the threat identification phase. Also, the assignment of DREAD severity ratings is currently in progress.

The next steps comprise the review of the DREAD severity ratings and the definition of threat resolutions for the identified and rated threats. The chosen threat resolutions and recommendations then will form a base for the technical implementation of the business scenario from the security perspective.

1.6.2. Summary

This business scenario's objective is the reduction of waiting times for (container) trucks arriving at the area of the port of Dortmund and the reduction of congestions in the city of Dortmund, i.e., in the residents' quarters close to the port of Dortmund. They currently get blocked often, due to trucks parking in second row and/or along the streets.

For this purpose, an intelligent parking management system is developed in the context of this business scenario. A mobile app supplies the truck drivers with live information on the parking situation in the port area while they are still on their way to the port or on the highway. It can propose, reserve and route to free parking lots in the port area to avoid congestion due to trucks waiting in front of full parking lots. In case there is currently no available parking lot in the port area the system can recommend the truck driver to do his pause on a highway car park in order to shift the ETA to a point in time where there are again free parking lots in the port area. Among this, the system also provides an interface for companies and port authorities to manage processes and logistical data like order data connected to the truckloads.

The following figure gives an overview of the IT architecture in order to enable the described service, while further explanation on it can also be found in section 2.3.1 of this document.

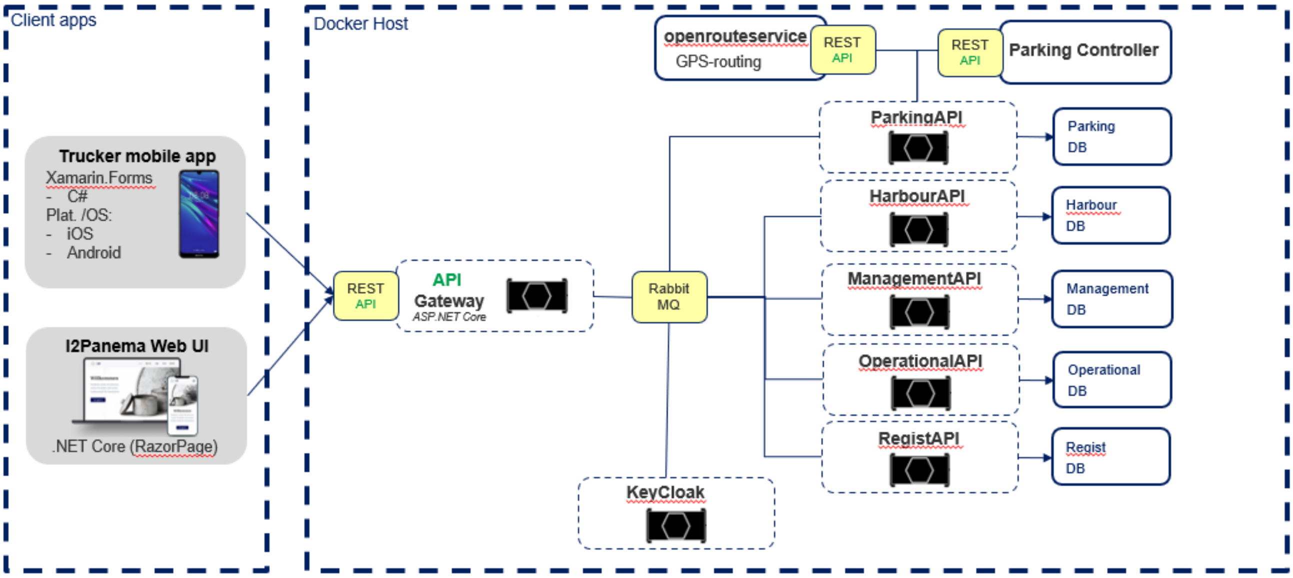


Figure 23: IT architecture overview of the port of Dortmund business scenario

1.6.3. Security Analysis

This section has been removed from the public version.

1.6.4. Implementation of the architecture in Port of Dortmund

Definition of threat resolutions and their implementation will be done after the DREAD ratings have been finalized.

1.7. Business Scenario Port of Wesel (Active Noise Control)

The STRIDE security analysis for the Port of Wesel business scenario is yet to begin. It will be conducted according to the same structure as used for Port of Gijon.

1.8. Business Scenario cross country (Smart Port/Ship)

The STRIDE security analysis for the Smart Port Ship business scenario is yet to begin. It will be conducted according to the same structure as used for Port of Gijon.

1.9. Business Scenario Port of Hamburg (Smart Ferry)

The STRIDE security analysis for the Port of Hamburg business scenario is yet to begin. It will be conducted according to the same structure as used for Port of Gijon.

1.10. Business Scenario Port of Hamburg (Onshore Power Supply)

The STRIDE security analysis for the Smart On Shore Power Supply business scenario is yet to begin. It will be conducted according to the same structure as used for Port of Gijon.

1.11. Business Scenario Safi port (RoRo localization)

1.11.1. Current and Next steps

This business scenario is about efficient discharge/load of cars from/to vessels in Safi port. A brief summary is given in chapter 1.11.2, which is complemented by further descriptions in chapter 2.8.

While being just started during submission of the first version of this deliverable, the security analysis for the RoRo scenario is completed by now. This involved consolidation of the architecture and transformation into the STRIDE threat model diagram, identification and severity rating of security threats, as well as identification of assets to protect and potential countermeasures. The process and results are described in chapter 1.11.3.

Current status is that the implementation of countermeasures is finished for most of the identified threats as can be seen from **Fehler! Verweisquelle konnte nicht gefunden werden.**, with further information listed in chapter 1.11.4.

For the remaining open threats an implementation of countermeasures is scheduled to be done during the planned commercialization of the system, which will likely go beyond the scope of the I2Panema project. With this all threats planned to be resolved in the scope of the project have been considered. Consequently, there will be no further actions on the business scenarios security layer in the scope of the I2Panema project.

1.11.2. Summary

This business scenario's objective is the optimization of the roll on roll off (RoRo) process in the Port of Safi, Turkey. For this purpose, an efficient system of loading/unloading of cars to/from ships, is developed in the scope of the business scenario, which allows remote localization of the individual cars during the process.

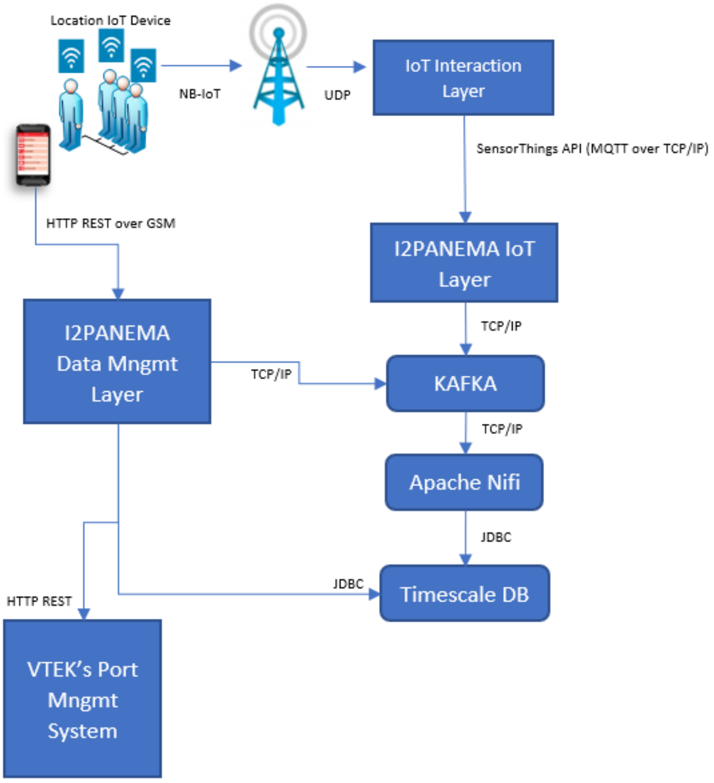


Figure 24: Illustration of the architecture of RoRo business scenario

1.11.3. Security Analysis

This section has been removed from the public version.

1.11.4. Implementation of the architecture in RoRo Localization

Further description of actions taken with regard to implementation of security measures:

- DF1: Mobile App – Data Management Layer: The connection is over GSM but it is encrypted through HTTPS.
- DF2 – DF3: The communication directly with the Location IoT device is on Lora network and cannot be accessed from outside.
- DF4: Field GW – IoT Layer: The connection over HTTPS and the connection is behind a firewall. Therefore, it cannot be accessed from outside.
- DF5 and DF7: Like in DF4, the connection over HTTPS and the connection is behind a firewall. Therefore, it cannot be accessed from outside.
- DF6: It is over JDBC. As it is behind firewall, it cannot be reached from outside.

1.12. Business Scenario Assan port (Container Localization)

The security implementation of the Container Localization business scenario is exactly the same as for the RoRo implementation. Due to the strong similarity to the RoRo scenario the security analysis results from the RoRo scenario are reused and no further dedicated STRIDE analysis was performed.

The only difference to the RoRo scenario (compare **Fehler! Verweisquelle konnte nicht gefunden werden.**) is that there is no Mobile app used in this scenario, and that the local users here are stacker drivers, instead of Tallymen who don’t interact with the IoT Localization devices directly, but instead receive SMS on their phone with order instructions on where to move which container. The STRIDE threat model diagram resulting from this architecture is depicted in Figure 25.

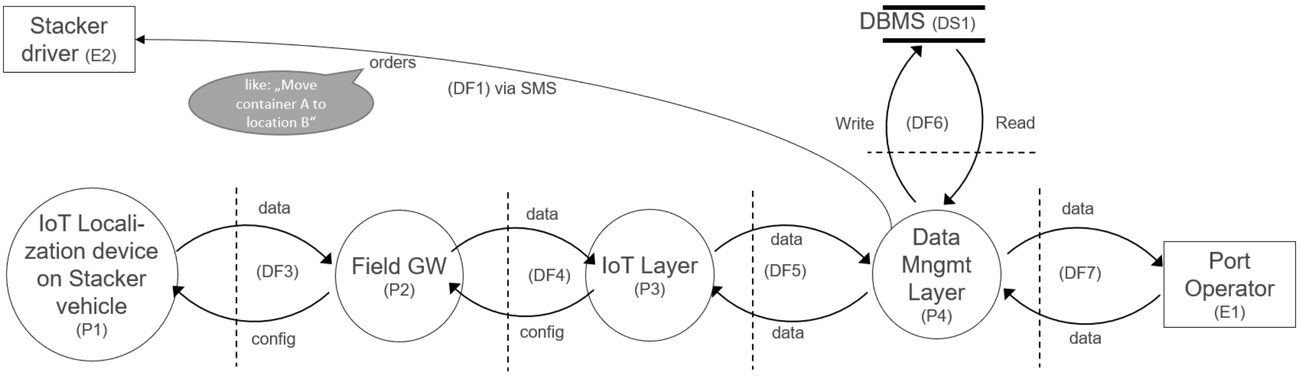


Figure 25: STRIDE threat model diagram view of the container localization business scenario

In terms of implementation of the security measures the following actions were taken:

- DF3: The communication directly with the Location IoT device is on NBIoT network and cannot be accessed from outside.
- DF4: Field GW – IoT Layer: The connection over HTTPS and the connection is behind a firewall. Therefore, it cannot be accessed from outside.
- DF5 and DF7: Like in DF4, the connection over HTTPS and the connection is behind a firewall. Therefore, it cannot be accessed from outside.
- DF6: It is over JDBC. As it is behind firewall, it cannot be reached from outside.

2. I^{2}PANEMA Operational Workflow support Layer

The I^{2}PANEMA Operational Workflow Layer (OWL) provides a set of methods that are meant to be placed in interaction with other business logic modules. This layer includes a persistence layer and data storage.

The OWL supports the processing of operational workflows in means of business services, i.e. services communicate between business partners, in typical customer – service provider relationships (see figure below).

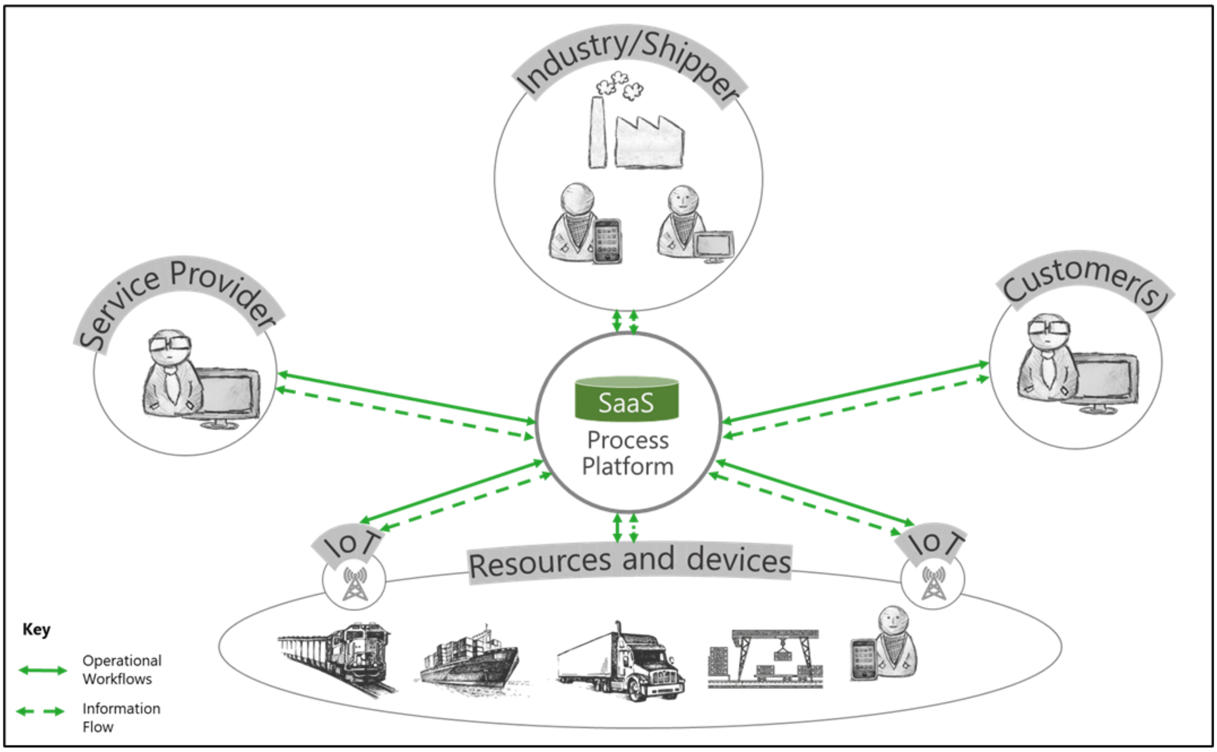


Figure 26: OWL. Communication among actors, resources and devices

A workflow consists of a flexible data structure, containing instructions of the business service.

In a logistics environment this could be a transport order including pick-up and receiver addresses as well as goods and load unit information. Furthermore, structured information can be reported by the receiver (example: the number of a container picked up during the transport process or a temperature measurement of a device inside of a refrigerated container).

The workflow passes through static status values, like “accepted, started, finished and cancelled”, etc.:

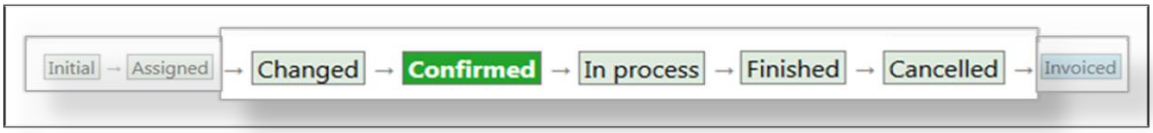

Initial → Assigned → Changed → Confirmed → In process → Finished → Cancelled → Invoiced

Figure 27: Status of a Workflow

A business service is usually communicated from one planning department (i.e. user group) of one company to another company (owner to contractor) or between user groups within the same company. At the contractor side, resources can be added to an instance of a workflow, referring to human resources or devices as an endpoint, i.e. a truck driver or a GPS device.

2.1. Business Scenario Port of Gijón (Environmental Measurement)

This layer is responsible for executing all workflows defined in the platform. The only workflow included in the pilot is the execution of alerts to generate notifications. Given that these alerts will be fixed, and users will not edit them, it was preferred to use a tool like Node-RED rather than having an alert execution engine. Node-RED, is a visual development tool based on programming flows. It was initially developed for

- connecting hardware devices
- connecting APIs and Services within the Internet of Things (IoT)

The alerts are defined according to the action protocol of the Port of Gijon.

2.1.1. Workflow of execution of alerts

The following image shows the alert execution workflow. This diagram is used for v1 of the predictive algorithm (**Prophet**[10]).

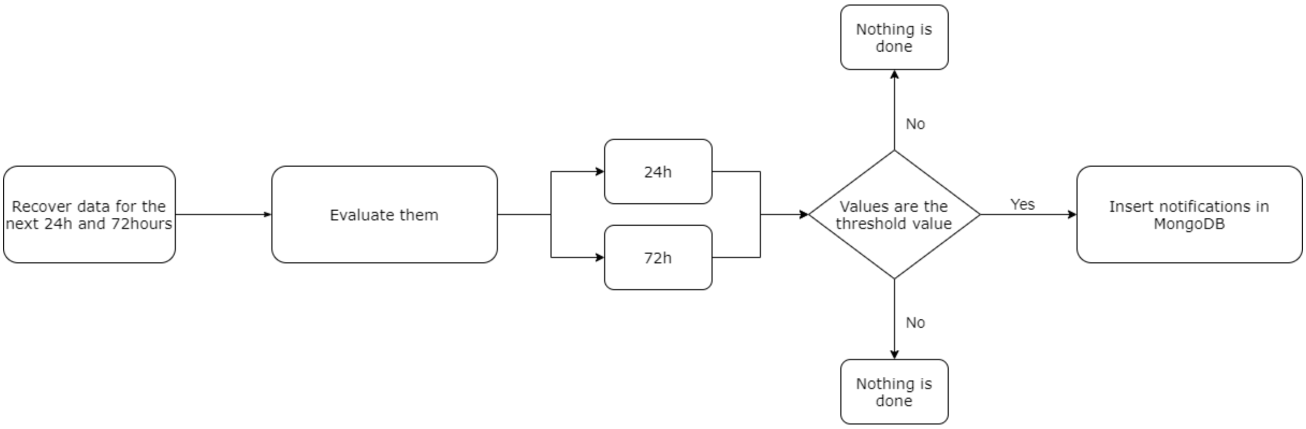

Nothing is done
No
Recover data for the next 24h and 72hours
Evaluate them
24h
72h
Values are the threshold value
Yes
Insert notifications in MongoDB
No
Nothing is done

Figure 28: Workflow of alerts execution

In the second version of the predictive algorithm (**TensorFlow**[11] is used), the diagram is practically identical. The only difference is that it only allows the generation of 24h alerts.

2.1.2. Implementation

The implementation of these flows, as already indicated in the previous version of the deliverable, have been done in Node-RED.

The choice of Node-RED is due to factors such as:

- Ease of use.
- Possibility to extend the flow with minimum effort
- Prior knowledge of the tool
- Chosen tool/component by the international consortium for the reference architecture

The alert generation flow has been explained in other deliverables within the project such as:

- D3.1 Innovation Report v2

Next picture depicts the appearance of the flow deployed in Node-RED. They are covering the execution of alerts.

[10] https://facebook.github.io/prophet/
[11] https://www.tensorflow.org/

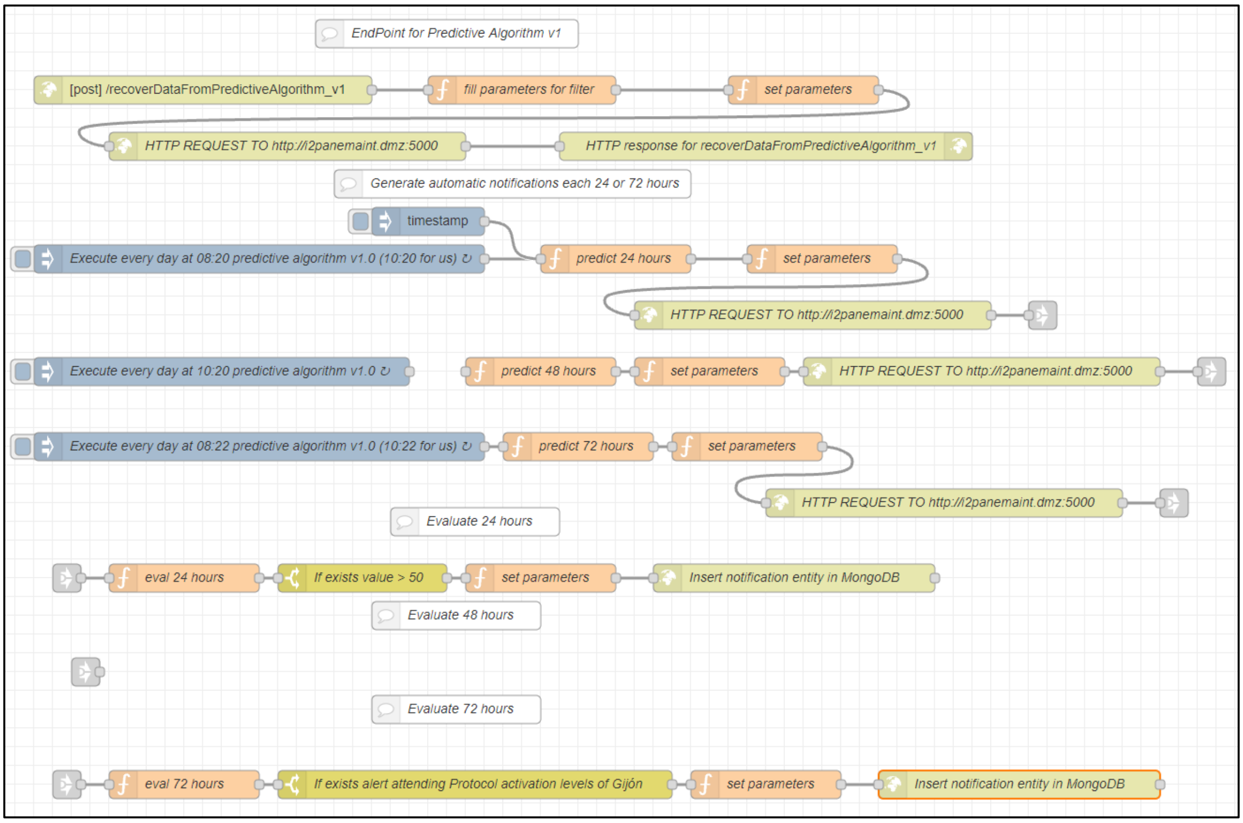

EndPoint for Predictive Algorithm v1
[post] /recoverDataFromPredictiveAlgorithm_v1
fill parameters for filter
set parameters
HTTP REQUEST TO http://i2panemaint.dmz:5000
HTTP response for recoverDataFromPredictiveAlgorithm_v1
Generate automatic notifications each 24 or 72 hours
timestamp
Execute every day at 08:20 predictive algorithm v1.0 (10:20 for us) ↻
predict 24 hours
set parameters
HTTP REQUEST TO http://i2panemaint.dmz:5000
Execute every day at 10:20 predictive algorithm v1.0 ↻
predict 48 hours
set parameters
HTTP REQUEST TO http://i2panemaint.dmz:5000
Execute every day at 08:22 predictive algorithm v1.0 (10:22 for us) ↻
predict 72 hours
set parameters
HTTP REQUEST TO http://i2panemaint.dmz:5000
Evaluate 24 hours
eval 24 hours
If exists value > 50
set parameters
Insert notification entity in MongoDB
Evaluate 48 hours
Evaluate 72 hours
eval 72 hours
If exists alert attending Protocol activation levels of Gijón
set parameters
Insert notification entity in MongoDB

Figure 29: Alerts flow

2.2. Business Scenario TriCon CTT (Active Noise System)

In this business scenario, an active noise control system as well as a real-time noise level information system will be developed. It is currently not planned to exchange information between actors or the organization. Thus, there will be no Operational Workflow in this Business Scenario.

2.3. Business Scenario Port of Dortmund (Intelligent Parking Management)

The objective of the Business Scenario Port of Dortmund is to implement an intelligent parking management system to reduce the waiting times of (container) trucks. Currently, trucks are parking along the streets and in the second row in the residents' quarters close to the port of Dortmund, when they cannot be handled in the port directly and the parking lots inside the port are occupied. Thus, congestion and traffic obstructions around the port area potentially can arise.

In the future, the truckers could be informed about the current situation of the parking lots inside the port. Therefore, they could get the information to adapt their estimated arrival times and look for a parking opportunity on their way to the port in advance, in case there is no parking lot reservation possible.

In the following, the IT architecture regarding to the Operational Workflow, will be described. First, the integrated actors and their kind of roles will be explained. The way of communication between the actors and the interfaces will be shown as well. After that, the necessary components of the IT architecture and their functions will be described one after the other. In addition, the interfaces to externals in the Business Scenario Dortmund will be mentioned. At least, there will be a user story, that describes the functions and the rights of the actors in a non-technical way. It is shown in a process chain and in different views of the possible users.

2.3.1. IT-architecture

Methods of the isolated OWL are divided in at least 2 main authorization levels:

The Administration and User Level are illustrated in next figure. User can be planners in a company department – either owner or contractor, mobile users that have access by the app (endpoint) or devices assigned to the workflow (endpoint).

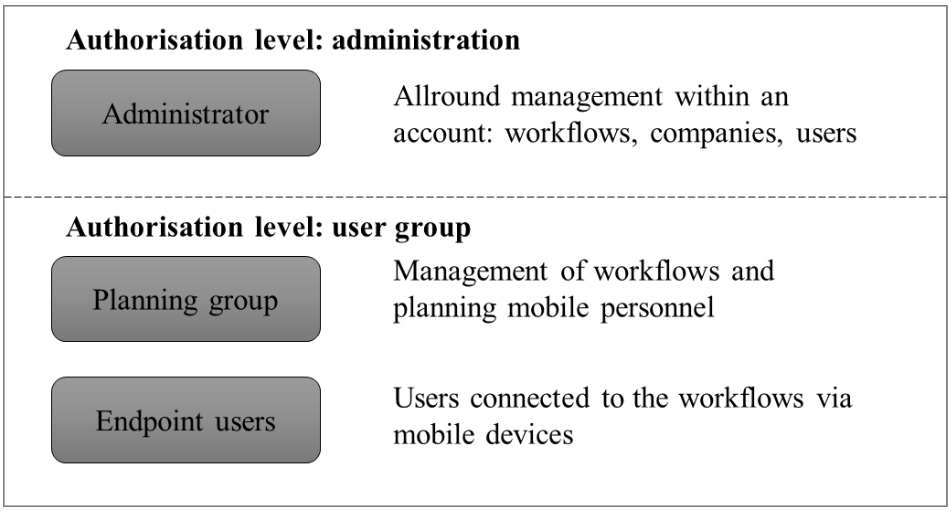

Authorisation level: administration
Administrator
Allround management within an account: workflows, companies, users
Authorisation level: user group
Planning group
Management of workflows and planning mobile personnel
Endpoint users
Users connected to the workflows via mobile devices

Figure 30: Role management within the OWL

As mentioned above, three role types are defined within the OWL as follows:

1. Administrator

The administrator is the only entity being responsible for carrying out administrative tasks in managing the port functionalities such as creating/editing port areas, parking lots and parking spaces and assigning groups, users, companies, and workflow types.

2. Planning

The planner can create, edit, and assign workflows based on the workflow types handled by the administrator. This group is also managing the loading dock queue for entering the company premises. Created workflows assigned from the planning user group will be planned and distributed to the endpoint users such as truckers.

3. Endpoint User

Endpoint users are connected to the workflow, representing the execution (i.e. mobile-user or a system device) at the location and reporting status information and measurements, like reserve, take, assigned parking lots. So, they have got the opportunity to register their truck. In comparison, the endpoint users have the least authorizations in the system.

The authorization of the user group level can be restricted, i.e. an endpoint user will only have access to a part of the user level methods. Also, the detailed access of the workflow owner users (disposers) differs from contractor users (recipients).

The way of communication inside the IT-architecture is described in the following picture.

In the box on the right side, the necessary components of the IT architecture and the interfaces to the externals are presented. The separate components will be described in the following subchapters. On the left side of the figure, the user interfaces are shown. There will be a mobile app for the end point users and an I^{2}PANEMA Web UI for all users.

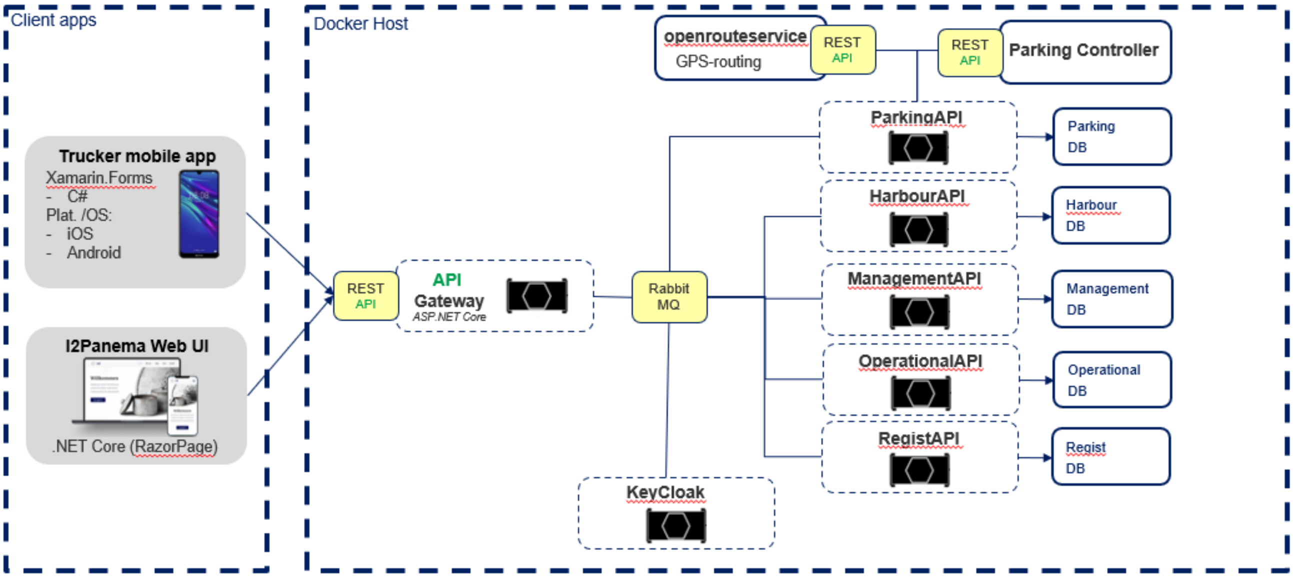


Figure 31: Overview platform IT-architecture BS Dortmund

This business scenario's architecture of the platform is based on a micro services approach. To ensure the platforms independent communication with the micro services, an API gateway is implemented. This one communicates exclusively with the message queue (RabbitMQ). The separate services, like parking API, subscribe to the respective channel from their own service regarding the reading and the writing part.

Every service works independent to the others to get self-sufficient data stocks. Via the Open Route service the routes can be obtained via an open source interface. The Parking API gets the information of the calculated ETA via a REST interface from the Open Route Service.

The separate API's, which are shown in the Figure 31, will be described with their functions one after the other in the following.

API Gateway:

The API Gateway is responsible for the communication between the user interface and the internal micro services. The functions of the API Gateway are the REST interface for Web UI and mobile app, ASP.Net backend and .NET Core (MVC) frontend.

Parking API:

The Parking API is responsible for the parking management. It creates and manages parking lots. Moreover, it reserves parking lots and connect/disconnect truckers to these parking lots.

Harbour API:

The Harbour API is responsible for the port management and it creates, manages and assigns ports and loading ramps.

Management API:

The Management API creates, manages and assigns users, groups, cities and companies and connects/disconnects users to companies.

Regist API:

The Regist API is responsible for the management of registration. It creates, manages and assigns registration codes. In addition, it authenticates users via these registration codes.

Operational API:

The Operational API describes an Operational Workflow Controller. It creates and edits different workflow types and operational workflows. To that end, it assigns and handles operational workflows. At least, it retrieves information.

Moreover, there are three interfaces to externals in this business scenario, i.e., to Keycloak, to the parking controller and to an Open Route service.

Keycloak is a service for user administration. It includes registration, login and save user specific information. In role management, the roles of the users are configurated and assigned.

The user management configures temporary users as well and a listing of orders and output of order details. The communication management retrieves user data and configures new sessions in the I^{2}PANEMA system. At least, the identity provider performs user authentication.

Via a REST interface to the **Parking Controller**, the intelligent parking management system gets the information of the current parking situation at the parking lots.

The **Open Route** service constitutes an open source GPS-solution with truck connection and is required to calculate a route from position A to position B and ETA's considering factors of the vehicle like size and weight. The restrictions of the API for free aref 2.000 requests a day or 40 in a minute and a maximum distance of 6.000 km and 1.000 km for a section. Because of these restrictions, a continuous ETA calculation of every tour is not possible. So, there is a limitation to calculate the ETA at the beginning of the tour and another triggered calculation in a defined time distance of the destination.

2.3.2. User Story

The user story is illustrated in three parts (administrator, planner and endpoint user).

The role of the **administrator**:

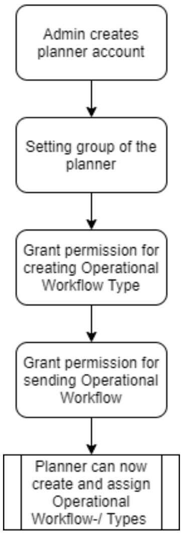

Admin creates planner account
Setting group of the planner
Grant permission for creating Operational Workflow Type
Grant permission for sending Operational Workflow
Planner can now create and assign Operational Workflow-/ Types

Figure 32: User-story Admin

The administrator is able to configure users (like planners) and workflow types and to distribute the rights of the different users in the system. The admin can create the establishment of ports and parking lots as well. So, the admin forms the basic structure, in which the other users work in.

The **planner** inside the system:

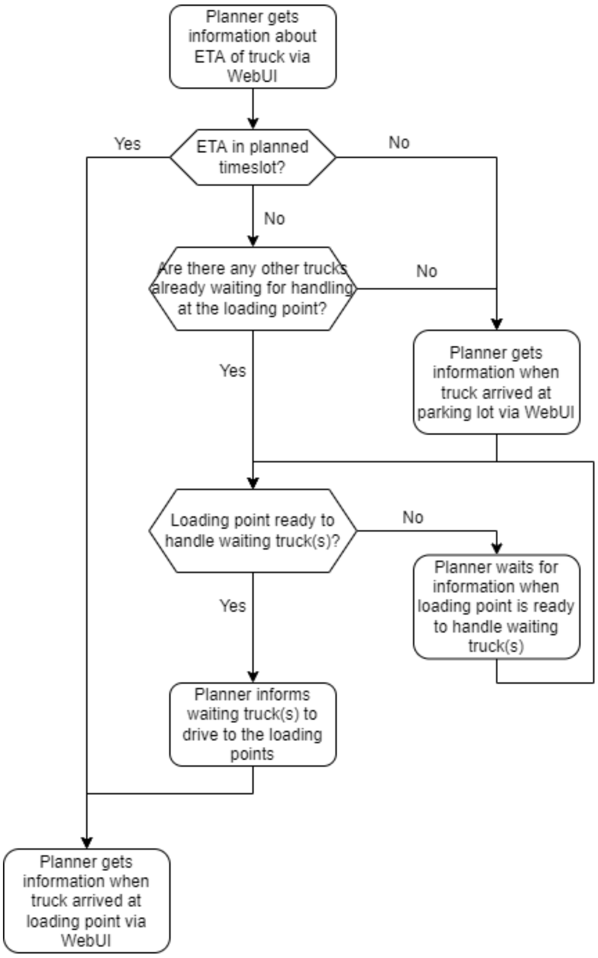

Planner gets information about ETA of truck via WebUI
ETA in planned timeslot?
Yes
No
No
Are there any other trucks already waiting for handling at the loading point?
No
Yes
Planner gets information when truck arrived at parking lot via WebUI
Loading point ready to handle waiting truck(s)?
No
Yes
Planner waits for information when loading point is ready to handle waiting truck(s)
Planner informs waiting truck(s) to drive to the loading points
Planner gets information when truck arrived at loading point via WebUI

Figure 33: User-story Planner

The planner is responsible for the creation of new users (like different and temporary endpoint users). It requires an independent registration and activation of an admin.

In the business scenario Dortmund, the planner is a dispatcher and interacts to the system via a web app. He gets the information about the calculated ETA of every truck. If a truck will arrive outside his planned timeslot, the dispatcher has to organize the other waiting trucks and call them, when the loading ramp has got capacities to handle the trucks. Moreover gets the information, when a truck arrives at his reserved parking lot or ends his ride because of his arrival at the loading point. In addition, he gets an overview about the current parking situation via the web app.

Endpoint user:

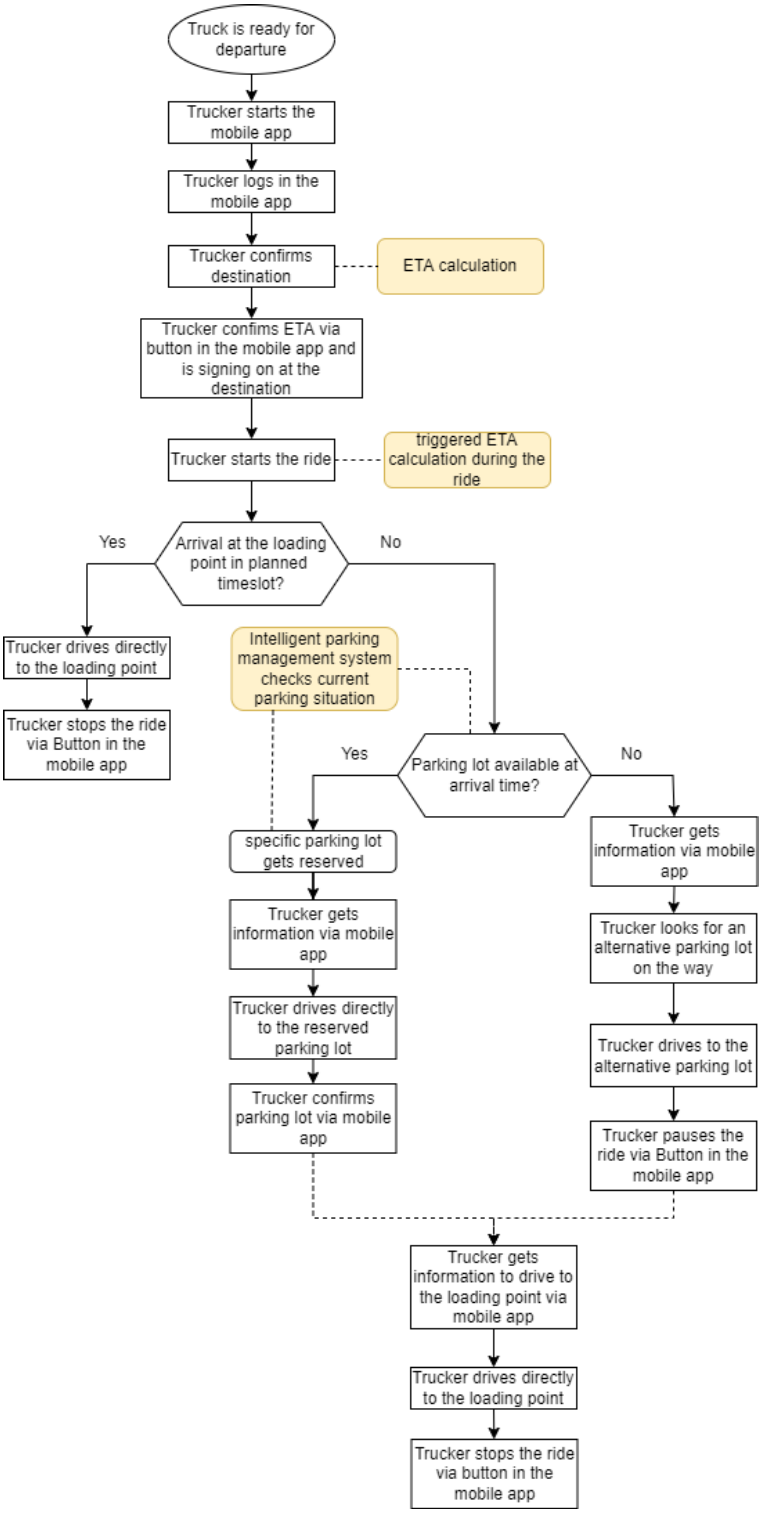

Truck is ready for departure
Trucker starts the mobile app
Trucker logs in the mobile app
Trucker confirms destination
ETA calculation
Trucker confims ETA via button in the mobile app and is signing on at the destination
Trucker starts the ride
triggered ETA calculation during the ride
Yes
Arrival at the loading point in planned timeslot?
No
Trucker drives directly to the loading point
Trucker stops the ride via Button in the mobile app
Intelligent parking management system checks current parking situation
Yes
Parking lot available at arrival time?
No
specific parking lot gets reserved
Trucker gets information via mobile app
Trucker drives directly to the reserved parking lot
Trucker confirms parking lot via mobile app
Trucker gets information via mobile app
Trucker looks for an alternative parking lot on the way
Trucker drives to the alternative parking lot
Trucker pauses the ride via Button in the mobile app
Trucker gets information to drive to the loading point via mobile app
Trucker drives directly to the loading point
Trucker stops the ride via button in the mobile app

Figure 34: User-story Endpoint User

In the business scenario Dortmund, the endpoint user is a trucker and interacts to the system via a mobile app. The trucker chooses his destination and the ETA calculation for his ride starts automatically. When the trucker confirms his calculated ETA, his ride is official announced to the dispatcher of the port. During his ride, the ETA will be calculated again, if the trucker is in a defined period before arrival according to the ETA calculation. If the trucker arrives at his destination in his planned timeslot, he can drive directly to the loading point. If he will arrive outside his planned timeslot and the ETA calculates this, the parking management system checks if there will be a parking lot available. If yes, the system reserves a specific one for the trucker and he will be informed, so he can drive directly to this. After that, he confirms his position in the parking lot. If there will be no parking lot available at his time of arrival, the trucker gets also informed about this and gets the request to look for an alternative parking lot on his way. When he arrives at his alternative parking lot and he can take a break, he pauses his ride via the mobile app. In both scenarios the trucker waits for the information by the dispatcher to drive to the loading point. When he finally arrives at the loading point, he always has to confirm his arrival via the mobile app to end his ride.

2.4. Business Scenario Port of Wesel (Active Noise Control)

In this business scenario an active noise control system will be developed.

It is currently not planned to exchange information between actors or the organization. Thus, there will be no Operational Workflow in this Business Scenario (like business scenario TriCon CTT).

2.5. Business Scenario cross country (Smart Port/Ship)

In this business scenario a standardized interface to exchange data and control activities between ships and shore partners should be developed.

An overall picture of this standard was drafted, taking into account the research conducted on other standards to also avoid duplication in regulation work. Currently, this draft is being discussed in ISO 4891 and presented to I^{2}PANEMA partners to consider more requirements.

The implementation of this business scenario has not started, yet. As a consequence, no technical progress in terms of OWL could be reported in this document´s first version.

2.6. Business Scenario Port of Hamburg (Smart Ferry)

Input for version two of this document is currently in progress. There was no input given in this document's first version.

2.7. Business Scenario Port of Hamburg (Onshore Power Supply)

The business scenario related to onshore power supply aims to improve the usage of such facilities by container ships. This scenario is currently being detailed. Accordingly, no information on the technical innovation could be reported in this document's first version.

2.8. Business Scenario Safi port (RoRo localization)

This business scenario deals with efficient discharge/load of cars from/to vessels.

2.8.1. IT-architecture

The IT infrastructure is shown below.

Location IoT-Device: This IoT-Device is responsible for detecting a car's current location and reporting it. In the RoRo business scenario, each driver will be assigned a Location IoT-Device. A different Location IoT-Device may be assigned to a driver at different times. Reporting can be implemented as periodic reporting, request-response, notification as the location changes more than a threshold, or as some other method. Location detection technology (e.g. GPS, etc.) is not finalized yet.

IoT-Gateway: The IoT-Gateway is responsible for communicating with the IoT devices, which may have different data models and/or communication protocols, (Location IoT-Devices in this case) and reporting the data to upper layer.

I^{2}PANEMA IoT Layer: This layer is responsible for storing the collected sensor data using a standard data model and serving it with a standard protocol. In the RoRo business scenario, Sensor Things API will be used as the data model and communication interface standard, as suggested by the I^{2}PANEMA reference architecture.

More detailed information about the technical architecture can be found in Deliverable D1.3 system architecture.

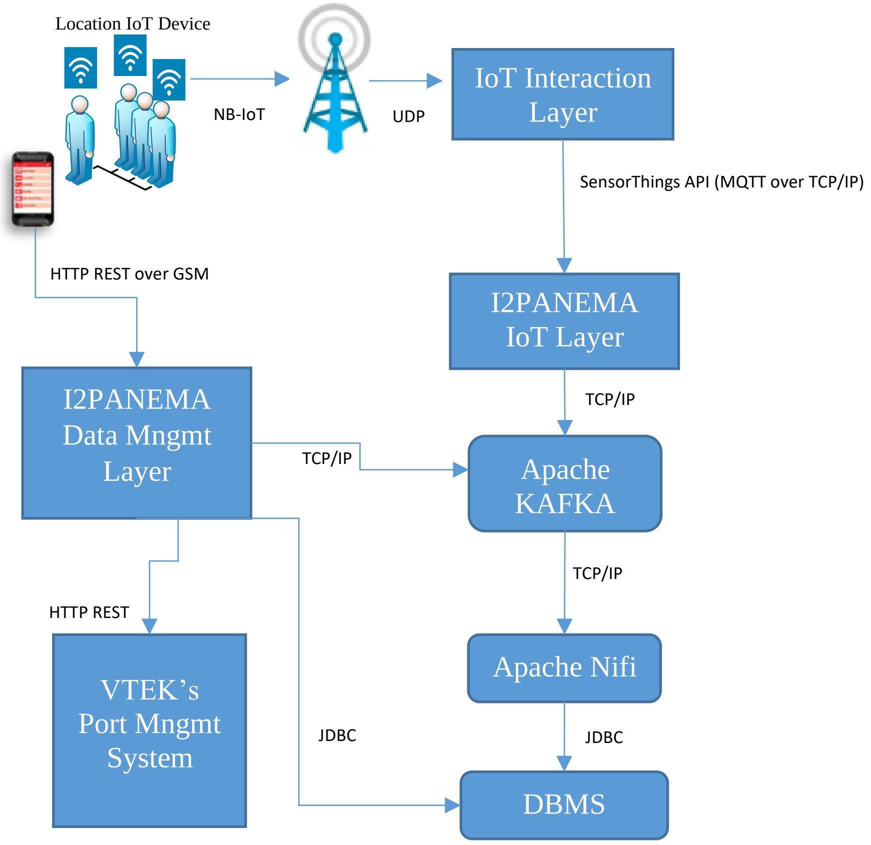


Figure 35: IT Infrastructure of RoRo business scenario

On the other hand, the business process from operational perspective is displayed below. The Customer Relationship Management Department (CRM Department) sends a "Gate In Request" to Planning Department in the port. The Planning Department makes a plan for the drivers and optimum yard settlement.

The plan can be:

- Loading of cars from the trucks to the parking lots, i.e., when the vessel comes to the port, they can be loaded to the vessel quickly and efficiently.
- Discharging of cars from the vessel to the parking lots.

The plan is sent to a Tallyman (who is manager of the drivers on the field). After that, the Tallyman sends the driving order (showing which car should be parked to which parking lot) to the drivers, who actually drive the cars to the assigned parking lot.

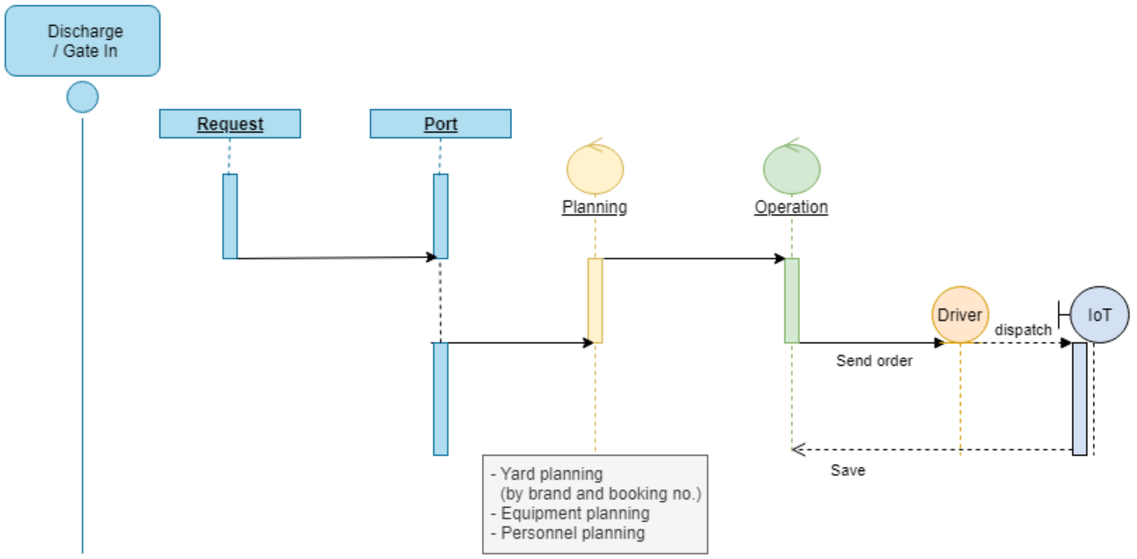


Figure 36: Business Process

In the business process the roles and their responsibilities are as follows:

- Customer Relationship Management Department: Makes the “Gate In Request”
- Planning Department: Plans the yard settlement, equipment (IoT Device) planning and personnel planning.
- Tallyman: Driver management.
- Driver: Driving the car to correct parking lot.

The interfaces of the users will be on the mobile device for the users on the field. On the other hand, the users in CRM and Planning Department will use their own Port Management System and I^{2}PANEMA Visualization Dashboards for real time monitoring.

2.8.2. Controller overview

The IT Infrastructure is explained in D1.3 I^{2}PANEMA Software Design deliverable.

2.8.3. Interfaces to externals

There is no interface to external systems. The only interface is to the VTEK’s Port Management System.

2.8.4. User-story

Explained above in detail.

2.9. Business Scenario Assan port (Container Localization)

The aim of this business scenario is to better track the containers in the port area through stacker vessels.

There are mainly 4 business scenarios in this business scenario:

1. Gate In/Vessel Discharge – Container Place on the Field
2. Gate Out/Truck Load
3. Shifting Container
4. Stacker Status Check

2.9.1. Gate In/Vessel Discharge – Container Place on the Field

This business process is for placing the container to the port area.

Input: Order to stacker

Output: The slot information where the container is placed

Business Process Description

- Door Pointer or Undership Pointer allows containers to enter the port in a controlled manner. During the entrance, the site information where the container will be placed by the system is told to the driver by pressing the entrance ticket at the entrance. The port vehicle driver gets the information at the ship's discharge, or it is sent by the TOS as information to the on-board terminals.
- At the same time, a work order is sent to the Stacker crane via the TOS.
- When the driver goes to the field where he will leave the container, the Stacker operator activates the work order by selecting the Plate No and/or verbally stating it. The field information to be placed in the work order appears automatically on the VMT (on-board terminal), which is located on the vehicle.
- The stacker operator handles the container on the truck and leaves it on the field. On release, it requests the I^{2}PANEMA application for the position of the Stacker boom. While recording site placement in the system, it saves the domain name and GEO Location information from I^{2}PANEMA to the system.
- If there is a difference between the area where it will be located and the actual area, the TOS gives a warning via the VMT on-board terminals. The operator checks again.

2.9.2. Gate Out/Truck Load

This business process is for placing the container to the truck.

Input: Order to truck and truck information

Output: The confirmation that the container is loaded to the truck

Business Process Description

- Door Checker after checking the truck with a door entry request and the container to be taken, it ensures that the empty truck enters the port.
- During the entrance, the site information where the container will be taken by the system is given to the driver by pressing the entrance ticket.
- At the same time, a work order is sent to the Stacker crane via the VPORT-TOS.
- When the driver goes to the field where he will receive the container, the Stacker operator activates the work order by selecting the Plate and/or verbally stating it. The field information to be received in the work order appears automatically.
- When the Stacker operator locks the container to retrieve the container in the field, it requests the Stacker boom location from the I^{2}PANEMA application. While registering the truck loading to the system, it saves the domain name and GEO Location information from I^{2}PANEMA to the system.
- If the locked container is in a different location than the work order, the TOS VMT gives a warning via the on-board terminals. The operator checks again.

2.9.3. Shifting Container

This business process is for changing the locations of the containers on the port area.

Input: Shifting order, current location of the container and target location of the container

Output: The confirmation that the container is transferred to the target location

Business Process Description

- Container relocation work orders, created by the Operations Planning department are sent to the VMTs (onboard terminals) on the Stackers.
- The stacker operator sees the work order coming from the TOS system and sees the container and its location.
- First, when it locks the container to receive the container to be relocated, it requests the I^{2}PANEMA application for the position of the Stacker boom. If the locked container is in a different location than the work order, the TOS VMT gives a warning via the on-board terminals. If the correct location is on the system, it records the shifting has started and makes the location information to be placed clear. The stacker operator drops the container to its location on the field. On release, it requests the I^{2}PANEMA application for the position of the Stacker boom. While recording site placement in the system, it saves the domain name and GEO Location information from I^{2}PANEMA to the system.
- If there is a difference between the area where it will be located and the actual area, the TOS VMT gives a warning via on-board terminals. The operator checks again.

2.9.4. Stacker Status Check

This business process is for checking the status of the containers on the port area.

Input: IoT device status

Output: Stacker status, location and speed information

Business Process Description

- Operations planning department will monitor the status of Stacker equipment through the TOS system. The information that the stacker equipment is active or passive is instantly queried from the I^{2}PANEMA system. It integrates with Equipment ID and Location information.
- Stacker crane information that does not move and does not work for a certain period of time is integrated with Equipment ID and Location information from the I^{2}PANEMA system to the VPORT TOS system.
- Stacker equipment will have sensors. The routes of the stackers, the roads they pass and the time they spend in the areas will be monitored. At the same time, the speed of the vehicle is calculated from the GPS data and if it is going faster than 10 km/h, a warning mail will be sent.

3. Conclusion

This deliverable describes the results of the I^{2}PANEMA security layer in chapter 1 and those of the operational workflow layer in chapter 2. This edition for the Spanish and the Turkish subprojects contains the updates to D5.2 v2 for the respective Spanish and Turkish business scenarios. The sections for the German business scenarios are kept at their status from v1 of this deliverable and will be updated as clarified in the abstract.

The security analysis for the Gijon business scenario as well as for the Turkish business scenarios have been finalized and its results and the corresponding security implementation described. Due to the ongoing COVID 19 situation the other business scenarios had different levels of maturity at the stage of compilation of v1 of this deliverable. Hence, some business scenarios were not at a development stage in which a clear IT architecture was defined yet. This basic definitions of the dataflows and communicating entities in the system represent the basis of the STRIDE security analysis, which is conducted to implement the I²PANEMA security and privacy layer. The corresponding sections for these delayed business scenarios in chapter 1 thus denote that these sections are still work in progress. These remaining sections will be populated in the upcoming final edition of this v2 deliverable.

Similarly, the description of the Operational Workflow is currently only finalized by the Spanish and Turkish business scenarios. There is also a small update of the Dortmund business scenario with the current status, but the final description will only be available at the end of the German subproject. In addition, in the business scenarios of TriCon and Wesel no Operational Workflow exists, because the relevant data will not be exchanged between actors or the organization.

In the Gijon business scenario, the only workflow is the execution of alerts to generate notifications. This is a very small workflow and it is handled by a tool, named Node-RED. Using a tool like Node-RED is preferred, because the alerts will be fixed and users will not edit them.

Thus, the operational workflows of the business scenario Dortmund and the RoRo business scenario Safi port are described in more details in this current status of this document. Both architectures contain an authorisation level, where planners organise and coordinate the operative activities of the mobile users. Moreover, both systems integrate the mobile users and the planning personnel with different user interfaces like mobile app and web app. In each case, there will be an IoT-Gateway to communicate with the IoT-Devices and the message queue, how it is designed in the I^{2}PANEMA reference architecture. But every business scenario is different, so there are various IoT-Devices to generate data and various interfaces to externals. For example, the RoRo business scenario has got an interface to VTEK's Port Management System, whereas the Dortmund business scenario uses an Open Route service as an open source solution for the ETA-calculation. Whereas the platform of the RoRo business scenario has to handle the information of the GPS sensors, the platform of the business scenario Dortmund has to process the analysed information of the current parking situation. Both systems work with REST-interfaces to exchange data and information.

In the Assan port business scenario, the I^{2}PANEMA system is mainly responsible to efficiently track the locations of the containers on the port field. Without I^{2}PANEMA, the stacker drivers need to enter the coordinates of the containers manually; hence, the process is very error prone in hectic port environment. With the location sensors placed on the stacker boom, I^{2}PANEMA saves the GEO-location information of the containers through OGC SensorThings based standard interfaces to the VTEK's Terminal Operating System, automatically. All the operations can be monitored without stacker driver interventions.