



PROJECT DELIVERABLE REPORT



Introducing advanced ICT
and Mass Evacuation Vessel design
to ship evacuation and rescue systems

D6.5 PALAEMON Incident Management Module (PIMM)

A holistic passenger ship evacuation and rescue ecosystem

MG-2-2-2018

Marine Accident Response

"This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 814962"



Document Information

Grant Agreement Number	814962	Acronym	PALAEMON	
Full Title	A holistic passenger ship evacuation and rescue ecosystem			
Topic	MG-2-2-2018: Marine Accident Response			
Funding scheme	RIA - Research and Innovation action			
Start Date	1 st JUNE 2019	Duration	36 months	
Project URL	https://palaemonproject.eu/			
EU Project Officer	Georgios CHARALAMPOUS			
Project Coordinator	AIRBUS DEFENCE AND SPACE SAS			
Deliverable	D6.5 PALAEMON Incident Management Module (PIMM)			
Work Package	WP6 – PALAEMON Back-End Infrastructure			
Date of Delivery	Contractual	M32	Actual	M34
Nature	R - Report	Dissemination Level	PU-PUBLIC	
Lead Beneficiary	KONNEKTABLE TECHNOLOGIES			
Responsible Author	Sotirios Moustogiannis	Email	smoustogiannis@konnektable.com	
		Phone	+30 694 391 6790	
Reviewer(s):	Irene Stylianou (ITML), Julia Ruiz (ATOS)			
Keywords	PALAEMON Incident Management Module, User Interface, Central Multi-layer Agent			

Authors List

Name	Organization
Kyriakos Giannakis	KT
Sotirios Moustogiannis	KT
Panagiotis Panagiotidis	KT
Elias Chatzidouros	ESI
Maria Belesioti	UAEG
Irene Stylianou	ITML
Alfredo Gardel	UAH
Ashraf Ragab	JU
Jens Hubel	JU
Julia Ruiz	ATOS
David Gomez	ATOS
Alexandros Mixelis	NTUA
Konstantinos Louzis	NTUA

Revision History

Version	Date	Responsible	Description/Remarks/Reason for changes
0.1	2021/10/05	KT	Report write-up
0.2	2021/12/17	KT	Request of partners' contributions
0.3	2022/02/09	KT	Inclusion of partners' contributions
0.4	2022/02/24	KT	Version ready for internal review
0.5	2022/02/25	ITML	Internal Review (IR)
0.54	2022/03/15	ATOS	Internal Review (IR)
0.5	2022/03/15	KT	Changes from IR's feedback
1.0	2022/03/22	KT	Review and Release

Disclaimer: Any dissemination of results reflects only the author's view and the European Commission is not responsible for any use that may be made of the information it contains.

© PALAEMON Consortium, 2022

This deliverable contains original unpublished work except where clearly indicated otherwise.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. Reproduction is authorised provided the source is acknowledged.



Table of Contents

1.	Introduction.....	1
1.1	Purpose and Scope.....	1
1.2	Structure of the Deliverable	2
1.3	Relation to other WPs and Tasks	2
2	System Overview.....	3
2.1	Scope.....	3
2.2	PALAEMON PIMM.....	3
3	Integrated Components on PIMM (Data Aggregation)	4
3.1	Overview	4
3.1.1	Ship Stability Toolkit.....	4
3.1.2	Weather Forecast Toolkit.....	4
3.1.3	Smart Risk Assessment Platform.....	6
3.1.4	PAMEAS.....	6
3.1.5	Smart Cameras.....	7
3.1.6	Decision Support System.....	7
3.1.7	Smart Safety System	8
3.1.8	Ship Structural Monitoring Ecosystem	8
3.1.9	Evacuation Coordinator	9
3.2	Data Description.....	11
3.2.1	Ship Stability Toolkit.....	11
3.2.2	Weather Forecast Toolkit.....	11
3.2.3	Smart Risk Assessment Platform.....	12
3.2.4	PAMEAS.....	14
3.2.5	Smart Cameras.....	31
3.2.6	Decision Support System.....	31
3.2.7	Smart Safety System	34
3.2.8	Ship Structural Monitoring Ecosystem	35
3.2.9	Evacuation Coordinator	36
3.3	Data Fusion Bus Integration	37
3.3.1	Description of the Data Fusion Bus.....	37
3.3.2	Integration Methodology	38

4	Functionality	39
4.1	Proposed Methodology.....	39
4.2	User Interfaces	42
4.2.1	User Interface Composition.....	42
4.2.1.1	Ship Stability Toolkit (T3.2).....	45
4.2.1.2	Weather Forecast Toolkit (T3.3)	46
4.2.1.3	Risk Assessment Platform (T3.5)	47
4.2.1.4	PAMEAS (T5.4).....	48
4.2.1.5	Smart Cameras (T5.5).....	49
4.2.1.6	Decision Support System (T6.4).....	50
4.2.1.7	Smart Safety System.....	50
4.2.1.8	Ship Structural Monitoring Ecosystem	51
4.2.1.9	Evacuation Coordinator	51
4.2.1.10	Early Fire Detection System	52
4.3	User Authentication and User Management	53
4.4	Containerization and Deployment.....	53
4.4.1	Docker advantages.....	54
4.4.2	Benefits of Docker Compose	54
4.4.3	Benefits of Container Deployment.....	55
5	Technical Specifications of the PIMM Module	55
5.1.1	Django Framework.....	55
5.1.2	Python	56
5.1.3	Apache Kafka	57
5.1.4	VueJS.....	57
5.1.5	PostgreSQL Database	58
5.1.6	ElasticSearch.....	59
6	Conclusions and Future Plans	60
6.1	Conclusions.....	60
6.2	Future Plans.....	60

List of Figures

Figure 1 PALAEMON WFT Information processing.....	5
Figure 2 Weather forecast data (almost) real-time workflow.....	5
Figure 3 Ship Evacuation Stages and triggers.....	9
Figure 4 Component's operation mode dependency with Ship Evacuation Status	10
Figure 5 Evacuation Coordinator feature present as part of PIMM's layout	10
Figure 6 Data fusion Bus.....	38
Figure 7 PIMM Layout - Decision Support System (1).....	43
Figure 8 PIMM Layout - Decision Support System (2).....	43
Figure 9 PIMM Layout - Video Streaming.....	44
Figure 10 PIMM Layout - Incident Assessment	44
Figure 11 PIMM Layout - Ship Stability Toolkit.....	45
Figure 12 PIMM Layout - Weather Forecast Toolkit	46
Figure 13 PIMM Layout - Danger in Ship	47
Figure 14 PIMM Layout - PaMEAS	48
Figure 15 PIMM Layout - Video Streaming.....	49
Figure 16 PIMM Layout - Decision Support System	50
Figure 17 PIMM Layout - Evacuation Coordinator.....	51
Figure 18 PIMM Layout - Early Fire Detection System	52

List of Tables

Table 1 The nodes from all the models of SRAP that will be provided to DSS.	13
Table 2 Personal User Data Model	15
Table 3 Health Specific Data Model	16
Table 4 Voyage Specific Data Model	17
Table 5 Associate Passengers Data Model.....	17
Table 6 Professional Information Data Model.....	18
Table 7 On Duty Data Model.....	20
Table 8 Network Access Data Model	23
Table 9 Location Access Data Model	25
Table 10 Geofence History Data Model	25
Table 11 Location History Data Model	26
Table 12 Minimal Location Data Model	29
Table 13 SHM Data	35

Abbreviations

AE	Acoustic Emission
AIS	Automatic Identification System
API	Application Programming Interface
DFB	Data Fusion Bus
DS	Data Streams
DSS	Decision Support System
EU	European Union
GA	General Alarm
ICT	Information Communication Technology
IMU	Inertial Measuring Unit
IoT	Internet of Things
ISM	International System Management
JSON	JavaScript Object Notation
MEV	Mass Evacuation Vessels
MVZ	Main Vertical Zone
OS	Operating System
PaMEAS	Passengers Mustering and Evacuation Process Automation System
PIMM	PALAEMON Incident Management Module
PMS	People Management System
SC	Smart Cameras
SEM	Smart Evacuation Management
SHM	Ship Health Monitoring
SOPEP	Shipboard Oil Pollution Emergency Plan
SRAP	Smart Risk Assessment Platform
SSS	Smart Safety System
SST	Ship Stability Toolkit
SWH	Significant Wave Height
UI	User Interface
VM	Virtual Machine
WFT	Weather Forecast Toolkit

1. Introduction

1.1 Purpose and Scope

PALAEMON's vision is to support the safety of passengers by providing a total solution for the evacuation of high-capacity passenger ships. PALAEMON aims to address the increased need for advanced passenger ship evacuation methodologies by defining a new ICT framework and radical rethink of mass evacuation systems with the introduction of MEVs with a wider scope than a sole technical proposition. Driven by the need to yield an end-to-end uniform approach, PALAEMON redefines the issue of passenger ship safety by taking into consideration issues pertaining to cost, complexity, vulnerability, societal acceptance, and ethics.

The intelligent ecosystem of PALAEMON incorporates innovative technologies for sensing, people monitoring and counting, and localization services as well as real-time data during accident time. These will be integrated into an independent, smart situation-awareness and guidance system for sustaining an active evacuation route for large crowds, making an emergency response in EU passenger ships more efficient. Continuous monitoring and permanent control will enhance the capacity to detect, prevent and mitigate any issue and potential harm arising from physical and/or man-made accidents and disasters.

To achieve PALAEMON's vision, the implementation of the PALAEMON Incident Management Module (PIMM) is necessary. The main object of the PIMM is to develop a central multi-layer agent that will be capable of:

- real-time prediction and prevention
- early detection
- multi-response
- risk mitigation

The system will support a multivariate input scheme, where raw data, features, and decisions can be accessed to provide the necessary outputs. Each layer of the Incident Management Module will be able to seamlessly exchange information, to coordinate the plan of action. It will be capable of running in an automated infinite loop manner, where it will continuously monitor everything and record the results. Embedded in the PIMM engine will be a reporting system, with results, alerts, statistics, localization, visualizations, and plans of action. Furthermore, the proposed PIMM engine will work closely with the corresponding disaster response module PALAEMON Decision Support System (T6.4).

Since maritime disasters in recent years are a stark reminder of the imperative need for timely and effective evacuation of large passenger ships during an emergency, this project aims to maximize the effectiveness of passenger evacuation, during an emergency and/or a serious incident. So, the PALAEMON Incident Management Module (PIMM) has been developed to be a part of the PALAEMON project for the achievement of this goal.

1.2 Structure of the Deliverable

The deliverable is organized as followed:

- Section 2 focuses on the System Overview of the PALAEMON Incident Management Module (PIMM). It describes the overall scope of the PIMM.
- Section 3 focuses on the data aggregation of the PALAEMON Incident Management Module from other PALAEMON components.
- Section 4 presents the Functionality of the PIMM and how it combines the PALAEMON Components between them.
- Section 5 focuses on the analysis of the technical specifications that are used to complete the PIMM's functionality.
- The deliverable concludes in Section 6 with final remarks and the forthcoming plans.

1.3 Relation to other WPs and Tasks

In this subsection, all the related Work Packages and Tasks of the PALAEMON Incident Management Module Component will be mentioned by name. The PALAEMON Incident Management Module belongs to Work Package 6, which is called PALAEMON Back-End Infrastructure, specifically to Task 6.5.

First of all, the 6.5 - PALAEMON Incident Management Module Task is related to other tasks in the same WP.

- Task 6.2 - PALAEMON Data Fusion Middleware
- Task 6.4 - PALAEMON On-Board Decision Support System (DSS) has an immediate connection with the PIMM, as the DSS is one of the components that are integrated with the PIMM.

Also, the WP3 - PALAEMON Intelligence Framework - AI Services and Algorithms has many tasks that are associated with the T6.5. The tasks are as follows:

- Task 3.2 - Ship Stability Toolkit
- Task 3.3 - Weather Forecasting toolkit
- Task 3.5 - Risk Assessment Platform Development

Lastly, from WP5 - PALAEMON on-board mustering tools and services

- Task 5.4 - Passengers Mustering and Evacuation Process Automation System (PaMEAS)
- Task 5.5 - Smart Cameras



2 System Overview

2.1 Scope

The PALAEMON project objective is to support the safety of passengers by providing a total solution for the evacuation of high-capacity passenger ships. Different toolkits have been deployed to achieve the PALAEMON project vision. The toolkits that have been implemented to achieve the PALAEMON vision are integrated into the PALAEMON Incident Management Module (PIMM). The main object of the PIMM is to be the connector of the PALAEMON toolkits between them. The scope of the PIMM is to be the bridge that transfers data between the components. In this way, the PALAEMON components can communicate and exchange data. It is very important for the components to exchange data in order to improve their functionalities as it leads to a better whole solution for the Master/Bridge in case of an incident. Also, the PIMM is responsible to display to the Master the outputs of the components providing solutions in order to avoid the adverse consequences of an incident.

2.2 PALAEMON PIMM

The PALAEMON PIMM aims to connect all the components the PALAEMON project consists of. The scope of the PALAEMON project is to provide a total solution for the evacuation of high-capacity passenger ships. The reason for the implementation of the PIMM platform is to achieve the purpose of the PALAEMON project. So, the PALAEMON PIMM integrates the components of the other partners of the PALAEMON project in order to provide a complete solution to the Master/Bridge and crew members. The communication between the components becomes through the PIMM. The PIMM is the brain of the PALAEMON project and the toolkits that are used are the body of the PIMM. The data is gathered from the components into the PIMM and exchanged information and data through PIMM. The Master is able to observe continuously the current situation of the ship as the components are updated every time the inputs data change.

In case of a grounding, fire/explosion, collision incident the Evacuation Coordinator sends a new message of the stage that the ship is. For example, the Evacuation Coordinator from the Normal Operation stage changes the stage to the Situation Assessment stage. The PIMM is updated and triggers components that are activated whenever an incident occurs, such as the Decision Support System, Smart Risk Assessment Platform components.

All the components of the PALAEMON project have been created in order to achieve PALAEMON's vision. When these elements exchange information with each other and are interconnected, then they contribute to the achievement of the main purpose of the PALAEMON project with more positive results. So, the PIMM operates by connecting the components to different stages to act in a way to help the Master to have a better understanding of the situation and avoid the adverse consequences in case of an incident.



3 Integrated Components on PIMM (Data Aggregation)

3.1 Overview

3.1.1 Ship Stability Toolkit

The “Ship Stability Toolkit (SST)” is a component of the PALAEMON project (D3.4 Development of Ship Stability Toolkit), which is designed to assist the Master and Bridge Command Team during an incident with ship motion predictions. These predictions are of major interest for the evacuation process, as the survival crafts on board might be limited in their possibility to board and launch, while the vessel's movement exceeds certain limits. Another important point for the ships motion prediction by the Ship Stability Toolkit is the fact, that some passengers might be handicapped in their ability to walk, especially on an unstable underground like a rolling vessel, which might have a huge impact on the time needed until all passengers arrived at their muster stations and therefore delay the evacuation process. The predictions of the SST will be delivered to the Decision Support System according to the ship's actual navigational, stability and weather data. Based on this information the Bridge Command Team can decide when it is the right time to muster the passengers or to start abandoning the vessel. But the toolkit will also assist by determining if there are ways to reduce the motions and ease the evacuation process.

3.1.2 Weather Forecast Toolkit

The main purpose of the Weather Forecast Toolkit is to identify the most appropriate set of actions that meet the needs of the corresponding incident case considering the weather conditions and utilising past incident cases. Moreover, the WFT identifies potential risks for human life using historical data to raise the awareness of the crew members (D3.6: PALAEMON Weather Forecast Tool (v2)).

As can be seen in Figure 1, T3.3 Weather Forecasting Toolkit is strongly related to:

- WP3: PALAEMON Intelligence Framework - AI Services and Algorithms,
- WP6: PALAEMON Back-End Infrastructure and
- WP7: PALAEMON Integrated System and Technology Validation Trials.

Several components may use the output produced by the WFT. Specifically, T3.3: Weather Forecasting toolkit is also related to T6.2: PALAEMON Data Fusion Middleware, T6.4: Development of PALAEMON On-Board Decision Support System, T6.5: PALAEMON Incident Management Module and T7.4: VDES Deployment. The WFT will retrieve the data (e.g., weather conditions) from the Data Fusion Bus (DFB) to provide its output. The VDES (VHF-Data Exchange System) will provide the weather conditions at the incident time through the DFB. Furthermore, the PALAEMON Incident Management Module (PIMM) dashboard will display the WFT output. Moreover, the WFT will deliver to the PALAEMON DSS the courses of action for further analysis.

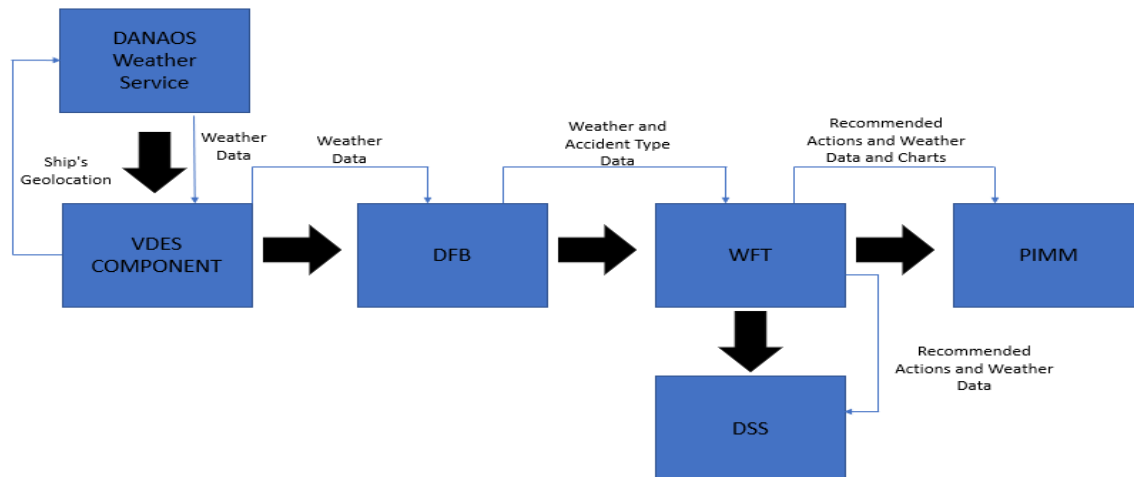


Figure 1 PALAEMON WFT Information processing

In Figure 1, several components intervene in the pipeline, distinguishing up to three different tiers: on the one hand, the ship infrastructure establishes a VDES link with the shore to exchange the data. Moreover, at shore level, we use a regular Internet connection to get the data from a public service interface carried out by DANAOS.

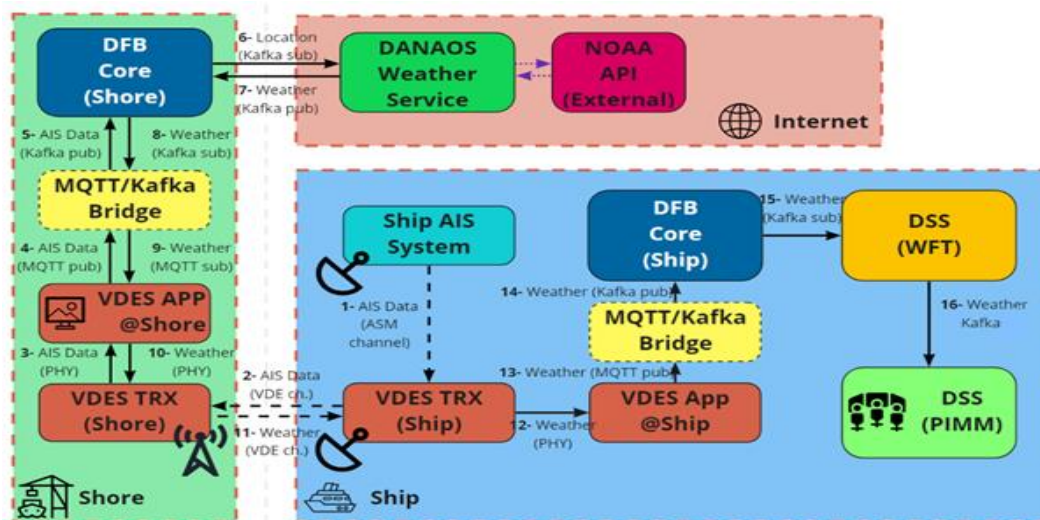


Figure 2 Weather forecast data (almost) real-time workflow

The WFT also provides the weather conditions at the incident time (e.g., wind speed, wind direction, Wind Waves significant wave height (SWH), etc.) displayed at the PIMM's dashboard. Furthermore, the WFT aggregates all the incidents in the dataset based on the wind speed and the visibility conditions at the incident time and provides charts that show the expected percentages of injuries and deaths for each incident. To aggregate the cases based on the wind speed, we use the following categorization. The first category includes accidents with light or moderate wind speed conditions, i.e., (0, 34] Knots, whereas the second con-

tains cases with strong wind speed conditions, i.e., (34, 130] Knots. Regarding the visibility conditions, we use the following categorization, good (more than 5 nautical miles), moderate (between 2 and 5 nautical miles), poor (between 1 and 2 nautical miles), and very poor or foggy visibility conditions (less than 1 nautical mile) (For more detailed information refer to deliverable D3.6: PALAEMON Weather Forecast Tool (v2)).

3.1.3 Smart Risk Assessment Platform

The main goal of the SRAP is to enhance the situation awareness of the crew members and to assist in the incident/accident management lifecycle (D3.10 Development of Risk Assessment Platform (v2)), by providing decision-making support during various phases of the evacuation process. Therefore, SRAP consists of three different models that are developed to maximize the situation awareness of the Master and Bridge Command Team for the three stages of evacuation, i.e., situation assessment, mustering assessment and pre-abandonment assessment. The indication on PIMM of each model is analysed below:

- **Situation Assessment**

The situation assessment model will indicate decision support to the Master whether to raise the General Alarm (GA) or not.

- **Mustering Effectiveness**

The mustering model will provide its outcome on a color-coded scale (according to the level of risk of delay) for all areas of the ship that will be divided per Main Vertical Zone (MVZ) and the specific ship deck under examination. The outcome of the assessment will assist crew members to identify the area of the ship that may need to take mitigative and/or corrective actions during the mustering.

- **Pre-abandonment**

The Pre-abandonment model will indicate decision support to the Master whether to give the order to abandon ship or not.

3.1.4 PAMEAS

Passengers Mustering and Evacuation Process Automation System (PaMEAS) component of the PALAEMON project, implements an IT-enabled functionality that can streamline and support in real-time the evacuation process. Specifically, PaMEAS develops a model for “Smart Evacuation Management” (SEM) in cruise and RoPax ships, which brings together IT and process management technology, human factor, ship’s safety processes and (regulated) operating procedures to improve “evacuability”.

PaMEAS essential objective is to optimise the evacuation process via an “augmented”, technology-aided process redesign. PaMEAS is the essential evacuation management component of the SEM system and the main automation engine of the evacuation process. It provides the means of execution of the Standard Evacuation Functions:

- a) tracking the status & location of passengers,

- b) marking the Evacuation Path,
- c) engaging the Notification process and providing directions to and through the Evacuation Paths and
- d) tracking the status and location of resources (crew support, etc.) and reassessing response plans.

Furthermore, it performs Incident Management Functions, i.e. the real-time management of incidents, e.g. injury of a passenger or crew member, if they occur during the evacuation process. PaMEAS establishes a spanning service layer that covers the whole evacuation process from the activation of the Evacuation Plan until the use of the Mass Evacuation Vessel (MEV) and the clarity of the ship.

3.1.5 Smart Cameras

The Smart Cameras (SC) gather visual information about people in different areas (D5.18 Smart Cameras component (v2)). An emergency example could be one where there are people confined in a room. The video feed will confirm if there is an emergency through an automatic scene analysis. It will mean that people need to be evacuated. The information from both systems might be fused at a later stage by the decision support systems (PIMM, DSS). Each SC node will capture the visual information from a surveillance area established in the system's installation stage. The device captures the video information, processes the data providing the number of people in the area, and tracks them to detect anomalous behaviour such as stampedes or trapped/fallen people.

3.1.6 Decision Support System

The PALAEMON Decision Support System (DSS) is a system that helps the Master to take the proper actions after a grounding, fire or collision incident of the ship, in order to avoid the adverse consequences of the incident and avert the problem that may drive to an evacuation of the ship (D6.4 Development of PALAEMON On-Board Decision Support System). The DSS is based on the International System Management (ISM) Code. The ISM Code ¹provides an international standard for the safe management and operation of ships and pollution prevention. Also, it takes notice of the Shipboard Oil Pollution Emergency Plan, known as SOPEP². The purpose of the SOPEP is to guide the Master and Bridge Command Team on board the ship concerning the steps to be taken when an oil pollution incident has occurred or is likely to occur. So, the actions that the DSS displays to the Master of the ship have been checked to comply with the protocol of the protection of the environment. The

¹ [International Maritime Organization. 1997. International safety management code \(ISM code\) and guidelines on the implementation of the ISM code. London: International Maritime Organization](#)

² [IMO Shipboard Marine Pollution Emergency Plans](#)

DSS will enrich the suggested actions with real-time data provided by other components. There is data from other components that are integrated into the DSS and are associated with the suggested actions. So, the DSS not only provides actions to the Master but also combines the proper data in a way that is useful to the Master. Also, the PALAEMON DSS provides the Master the knowledge of the fire sensor real-time values so that they can respond promptly.

3.1.7 Smart Safety System

The "Smart Safety System (SSS)" component of the PALAEMON project is intended to create an assisting safety system during evacuation. The system aims to improve the safety evacuation system on board passenger ships and to provide information on the condition and all relevant information during the evacuation process. All crew members on board should have access to this information. This also reduces the burden on VHF communication and provides a detailed overview of the situation on board. This Model of the SSS was built on the adapted similar System which is used at coordinating firefighters when they are proceeding with building evacuation. The needs and systematic changes are done and adapted within the system to meet the objective of T3.1 and the whole process of WP3.

This module gives an overview of the evacuation status and the current status of the evacuation progress. It also provides an interface for the connection between the Master and the evacuation team. The information is provided as a gadget in the PALAEMON Decision Support System (DSS). Officers and crew members provide information and can rely on sensors (e.g., wristbands) and also on communication devices (e.g., tablet or Bluetooth receiver).

3.1.8 Ship Structural Monitoring Ecosystem

The SHM (Ship's Health Monitoring) and AE (Acoustic Emission) systems are used for the monitoring of the Ship's response in normal but also adverse, e.g. accident, conditions (D6.1 Ship Structural Monitoring Ecosystem). The SHM is comprised of IMUs (Inertial Measuring Unit) which are equispaced on one of the decks of the ship, from aft to fore. The numbers of the IMUs depend on the length of the ship. The data from the IMUs are collected from a central processing unit, e.g., laptop, with the SHM software, translated to useful data (angles, accelerations and displacements). The SHM software records the data, transmits alarms when a parameter has exceeded a critical value and also sends a status report at prearranged time intervals. The AE is placed at critical areas of the ship, where defects might form but also where damages are likely to occur. The AE systems also record real-time data and can transmit alarms when a parameter exceeds a critical value. These values are hit counts and/or acoustic emission energy. These values indicate when a defect is forming or a defect propagating (getting bigger). The IMU and SHM software are used for ship global stability and structural strength monitoring, while the AE is used for local condition monitoring, to detect defects and damages while they are being formed or propagating.

3.1.9 Evacuation Coordinator

At this point, it is worth recalling that PALAEMON hinges around the concept of Maritime Evacuation Process. As illustrated in Figure 3, an incident may occur during a voyage, thus unleashing the whole evacuation process represented in the schema. Nonetheless, we must state that, if the situation gets solved (e.g., the fire extinguisher team puts out a fire and has the situation under control), the chain of events may be interrupted, getting back to the normal situation. As we can also appreciate in the figure, the actual elements that trigger the status changes are humans (i.e., officers/crew, Master), so we can consider the process as “analog”.

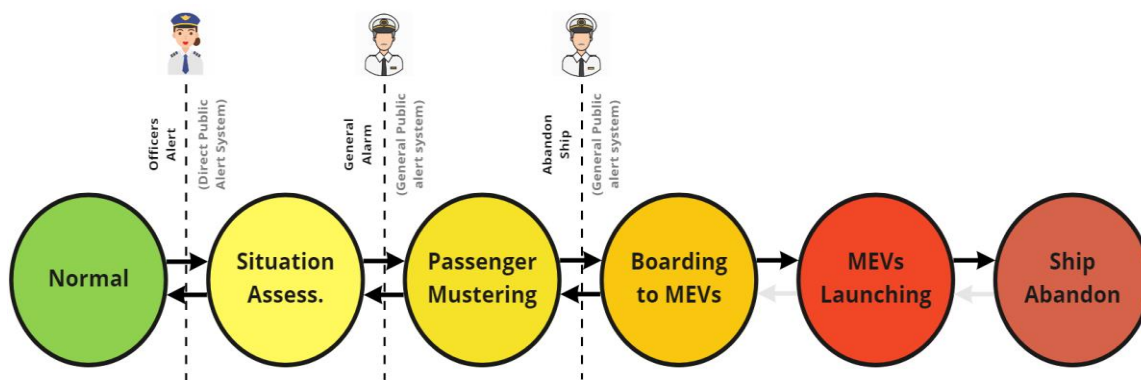


Figure 3 Ship Evacuation Stages and triggers

With the Evacuation coordinator, the idea is to translate these human-based commands to (simple) digital signals that are propagated across the PALAEMON system, to warn all software components (i.e., the ones that depend on the ship evacuation status) that something has occurred and they must react accordingly. Some components (not all) do modify their operation mode as a function of the ship evacuation status, as shown in Figure 3. It is out of the scope of this deliverable to analyze the individual behavior of these components upon this status change notification.

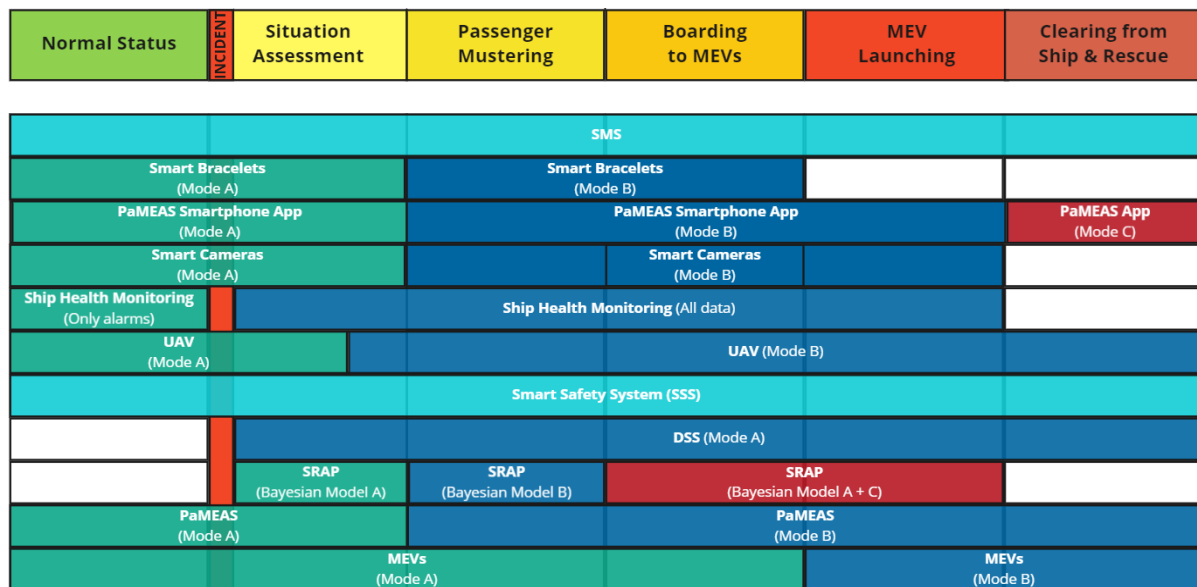


Figure 4 Component's operation mode dependency with Ship Evacuation Status

Technically speaking, the process goes as follows:

1. The PIMM displayed in the bridge permits the Master and Bridge Command Team to manually click on the upcoming status (as shown in Figure 4).
2. PIMM sends a message (i.e., HTTP) to the PALAEMON Evacuation Coordinator.
3. The PALAEMON Evacuation Coordinator broadcasts (i.e., via Kafka) a notification of the status shift to all (subscribed/interested) components. More information about the format of this message is in Section 3.2.6.
4. All affected components reply with an acknowledgment message back to the PALAEMON Evacuation Coordinator, notifying in the operation mode modification. As a side note, those without a direct dependency can directly ignore the message.

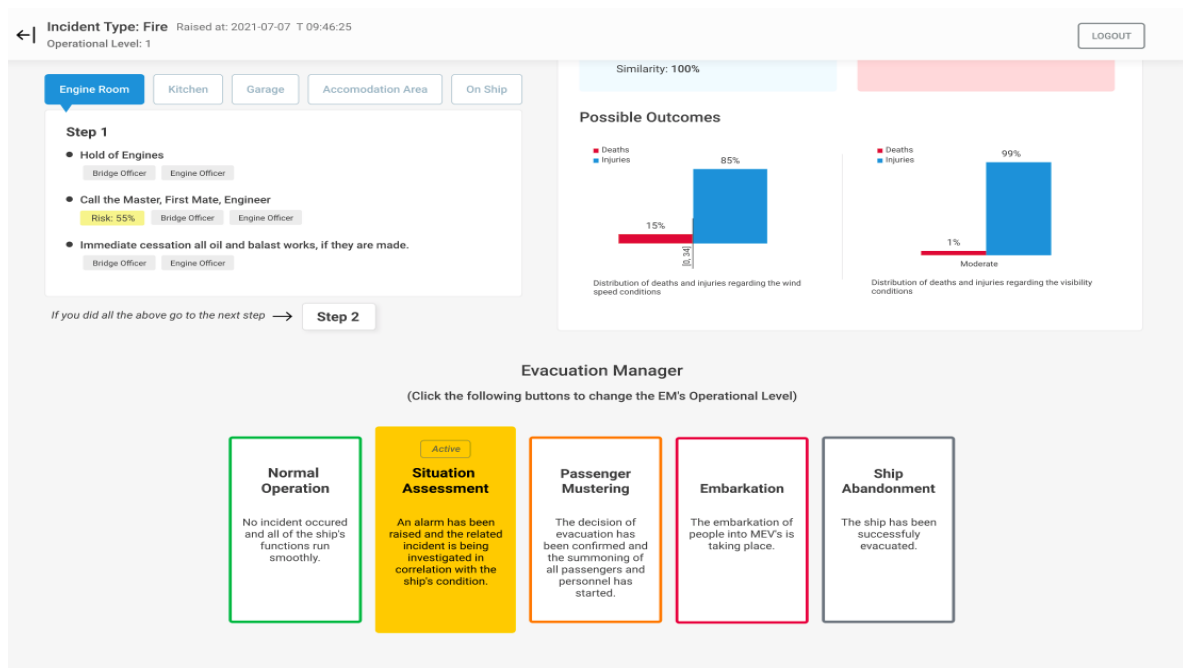


Figure 5 Evacuation Coordinator feature present as part of PIMM's layout



Under the hood, the PALAEMON Evacuation coordinator keeps track of all the status updates, whose traces are stored in the Data Fusion Bus for further (e.g., post-incident) analysis. The reader might refer to D7.1 (PALAEMON Communications Platform) to get more information about the process.

3.2 Data Description

3.2.1 Ship Stability Toolkit

- **Short Description of the data being passed:**

The Ship Stability Toolkit relies on inputs that are gathered either manually or automatically from the ship's AIS data, the Weather Forecast Toolkit and the Ship Health Monitoring Ecosystem. Based on the provided information the SST will output the motion prediction data via Kafka to the PALAEMON System.

- **Dispatch interval:**

On major changes or latest every 60s.

- **Data description:**

A JSON form:

```
{
  "state": "request",
  "fn": "0",
  "t": "12",
  "Hs": "8",
  "L0": "225",
  "beta": "120",
  "head": "180",
  "Surge": "3.087589",
  "Sway": "5.388004",
  "Heave": "7.44123",
  "Roll": "32.45254",
  "Pitch": "7.415234",
  "Yaw": "3.53242"
}
```

3.2.2 Weather Forecast Toolkit

- **Short Description of the data being passed:**

The WFT provides three types of data displayed to the PIMM's dashboard. First, the WFT provides weather data based on the ship's course. Also, WFT provides the effective and the ineffective actions that the crew implemented during the incident for a similar accident based on the weather conditions. Finally, WFT aggregates all the incidents in the dataset based on the wind speed and the visibility conditions at the incident time and provides charts (to

PIMM's dashboard) that show the expected percentages of injuries and deaths for each incident.

- **Dispatch interval:**

Every time that the WFT receives data from the DANAOS API the component will update its output to the PIMM's dashboard.

- **Data description:**

In JSON Format

```
{
  "Timestamp": "2019-10-02T09:00:00",
  "Event": "Fire",
  "windSpeedKnots": 4.80129589524,
  "windDirection": 114.97,
  "wavesPeriod": 2.07,
  "wavesSWH": 0.16,
  "wavesDirection": 116.64,
  "swellPeriod": 7.52,
  "swellSWH": 1.22,
  "swellDirection": 170.99,
  "combPeriod": 7.52,
  "combSWH": 1.23,
  "combDirection": 170.18,
  "currentsSpeedKnots": 0.51,
  "currentsDirection": 2.2,
  "vis": 23.867,
  "msl": 1012.43,
  "iceCover": 0.0,
  "temp": 11.36,
  "hum": 89.4,
  "Suggestions": {
    "Suggested Actions": "The actions of the crew to extinguish the fire were effective",
    "Risks": "The fire was spreading quickly"
  }
}
```

3.2.3 Smart Risk Assessment Platform

- **Short Description of the data being passed:**

SRAP will provide its main output to PIMM every time that SRAP will be provided with revised input data. All nodes along with a short description and the respective variable type are present in Table 3-1 (the detailed description of the nodes is provided in D3.10). The data are provided at an asynchronous rate. The output rate of SRAP is depending on the rate of the data input. In particular, every time SRAP identifies input data will produce a revised input to DSS according to the new data.

- **Dispatch interval:**



The data are provided at an asynchronous rate. The output rate of SRAP is depending on the rate of the data input. In particular, every time SRAP identifies input data will produce a revised input to DSS.

- **Data description:**

Table 1 The nodes from all the models of SRAP that will be provided to DSS.

#	Node			Variable	
	Name	States	Description	Name	Type
1	Effectiveness of mitigation measures on-site	Effective Not effective	Likelihood of mitigation measures being effective.	effectivenessOfMitig	Double
2	Passengers proximity to hazards	Low Medium High	The probability of passengers being close to danger determines the probability of being directly exposed to and affected by hazards.	passengerProxToHaz	Double
3	Status of passive containment	Effective Not Effective	Indicates whether passive containment of the incident is effective given the status of the detectors and the containment doors.	statusOfPassiveCont	Double
4	Spreading	Contained Not Contained	Probability of incident spreading to other parts of the ship other than the incident location.	spreading	Double
5	Structural Integrity	Not Compromised Compromised	Probability of ship structural integrity being compromised.	structuralInt	Double
6	Stability	Sufficient Not sufficient	Probability of ship stability being sufficient.	stability	Double
7	Hull status	Safe Unsafe	Probability of hull being in a safe or unsafe state.	hullStatus	Double
8	Ability to communicate	Fully operational Degraded Not operational	Indicates the ship's ability to communicate, within the ship and with the outside world.	abilityToCom	Double
9	Critical systems status	Fully operational Degraded Not operational	Probability of critical systems being operational.	criticalSystemsStatus	Double

10	Vessel status	Safe Unsafe	Probability of vessel being in a safe or unsafe state.	vesselStatus	Double
11	Incident severity	Low Moderate High	Probability of the incident resulting in adverse/ unacceptable consequences for the crew and passengers.	incidentSeverity	Double
12	Sound the GA	Sound GA NO Sound GA	This is an indication to assist the Master to decide to sound the GA or not	sound	String
13	Risk of Delay	Low Medium High	The risk of delay will be provided for all the areas of the vessel per MVZ and deck	riskOfDelay	Double
14	Situation Severity	Low Medium High	Probability of the severity regarding the condition of ship given the abandonment readiness level, the risk of the abandonment and the vessel status.	situationSeverity	Double
15	Abandon vessel	Abandon ship Do Not Abandon ship	This is an indication to assist the Master to take the decision to abandon ship or not	abandonVessel	String

3.2.4 PAMEAS

- **Short Description of the data being passed:**

To enable the tracking and monitoring of the passengers and crew position within the ship, the PaMEAS system components generate various data streams. Specifically, PaMEAS primarily implements Data Flows with each data flow consisting of a set of Data Streams (DS) as described below:

1. The Registration Process Data Flow
 - a. Personal User Data Stream (DS-P)
 - b. Network Access Data Stream (DS-N)
2. The Evacuation Management Process Data Flow
 - a. Location Data Stream (DS-L)
 - b. Minimal Location Data Stream (DS-mL)



Each such data stream outputs a different Data Model. These Data Models are described below.

Personal User Data Stream

This Data Stream transfers the bulk of the information that constitutes a user's profile (Passenger or Crew). Specifically, this DS provides to the PALAEMON back-end storage layer (Data Fusion Bus) the Personal Information Profile of the user. In detail, it contains and transfers:

- Personal User Data (as retrieved from the eID system integrated to PALAEMON);
- Health Specific Data, related to the users' capabilities of executing evacuation procedures (for Passengers);
- Voyage Specific Data, which includes Associate Passengers Data (for Passengers and Crew);
- Professional Information Data, which includes On Duty Data (for Crew).

- **Dispatch interval:**

This Data Stream outputs data once a new user is registered. The registration flow is implemented by the PaMEAS Registration Service - People Management System (PMS) - which implements the onboarding functionality of PaMEAS. Thus, this DS emits data once a user completes the registration flow via the PMS and as a result, a specific interval cannot be defined. Finally, all Personal Identification information about the user (name, surname, etc.) is transmitted encrypted.

- **Data description:**

The data model for this stream is defined as follows:

Table 2 Personal User Data Model

	Attribute Name	Data Type	Notes
Personal User Data			
Attributes Retrieved from the connected eID system			
1	name	String	First name
2	surname	String	Last name
3	date_of_birth	Date	Date of birth formatted as yyyy-MM-dd
Self-attested attributes			
4	email	String	Users email address
5	phone_number	String	Users mobile phone number

6	emergency_contact_details	String	Phone number of user's emergency contact
7	country_of_residence	String	Code of the country of residence of the user, e.g. GR
8	preferred_language	String	Country code indicating the preferred communication language of the user

Table 3 Health Specific Data Model

	Attribute Name	Data Type/Values	Notes
Health Specific Data			
Mobility Personal information			
1	mobility_issues		
		none	Passengers with no mobility problems.
		assisted_gait	Passengers with assisted gait
		walking_disability	Passengers with a walking disability.
		severe_walking_disability	Passengers with a severe walking disability.
		unable_to_walk	Passengers who are unable to walk
		visually_impaired	Passenger with visual impairments
		hearing_impaired	Passenger with hearing or speech impairments
		cognitive_impaired	Passenger with cognitive or developmental impairments that might affect their ability to follow evacuation procedures
Medical Personal information			
2	medical_assistance		
		equip_required	Passenger that requires special medical equipment during evacuation

		stretcher	Passenger that requires a stretcher and carrying assistance during evacuation
		heavy_doses	Passenger receiving medication affecting his/her thinking skills that might affect their ability to follow evacuation procedures
Pregnancy Personal information			
3	pregnaency_data		
		complicated	The passenger in a complicated pregnancy (i.e., pregnancies with any form of increased risk of miscarriage),
		normal	Passenger with no complications of their pregnancy

Table 4 Voyage Specific Data Model

	Attribute Name	Data Type	Notes
Voyage Specific Data			
Attributes Retrieved from the connected Shipping company IT system			
1	associated_passengers	Associated Passengers Object	People they are travelling with
2	embarkation_port	String	Port of embarkation
3	disembarkation_port	String	Port of disembarkation
Attributes automatically generated by PMS			
4	embarkation time	Unix Timestamp	Time of embarkation of the user (This timestamp is generated once the user registers their device to the system).

Table 5 Associate Passengers Data Model

	Attribute Name	Data Type	Notes
Associated Passengers Data			
1	name	String	Passenger's name
2	surname	String	Passenger's surname
3	gender	String	Passenger's gender
4	age	String	Passenger's age

Table 6 Professional Information Data Model

	Attribute Name	Data Type/Values	Notes
Professional Information Data			
1	is_crew	Boolean	Denotes whether this person is part of the crew
2	role		
		captain	Denotes the Captain of the Vessel
		day_work_officers	Denotes that the person is a Day Work Officer
		watch_keepers	Denotes that the person is a Watch Keeper
		seamen	Denotes that the person is a seaman
		security_staff	Denotes thea the person is a Security Staff
		chief_engineer	Denotes that the person is the Chief Engineer
		electrical_engineer	Denotes that the person is the Electrical Engineer
		officer	Denotes that the person is an Officer
		technical_crew	Denotes that the person is part of the Technical Crew
		cadet	Denotes that the person is a Cadet

		hotel_officer	Denotes that the person is a Hotel Officer
		waiter	Denotes that the person is a Waiter
		bar_staff	Denotes that the person is part of the Bar staff
		chef	Denotes that the person is a Chef
		staff	Denotes that the person is part of the Hotel Staff
		back_of_house	Denotes that the person is part of the Back of House staff
		cabin_steward	Denotes that the person is a Cabin Steward
		entertainers	Denotes that the person is an Entertainer
		shop_staff	Denotes that the person is part of the Staff of a Shop
		beautician	Denotes that the person is a Beautician
		casino_staff	Denotes that the person is part of the Casino Staff
2	emergency_duty		
		command_team	Denotes that this person is part of the emergency command team
		first_response_unit	Denotes that this person is part of the first response unit team
		boat_preparation_unit	Denotes that this person is part of the boat preparation unit team
		passenger_mustering_unit	Denotes that the person is part of the passenger mustering unit team
		firefighting_unit	Denotes that the person is part of the firefighting unit team

		damage_control_unit	Denotes that the person is part of the damage control team
		first_aid_unit	Denotes that the person is part of the first aid unit team
		medical_unit	Denotes that the person is part of the medical unit team
		passenger_assistance_units	Denotes that the person is part of the passenger assistance team
		cabin_searching_unit	Denotes that the person is part of the cabin search unit team
3	on_duty	Object Array	Denotes the time units that this person is on duty according to the ship's schedule.
4	languages	String Array	List of languages the person is fluent in

Table 7 On Duty Data Model

	Attribute Name	Data Type/Values	Notes
On Duty			
1	date	String	Denotes the date this duty assignment refers to
2	start	Timestamp	Denotes the starting time of the assigned duty
3	end	Timestamp	Denotes the ending time of the assigned duty

The JSON Schema of the aforementioned data model is the following:

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "name": {
      "type": "string"
    },
    "surname": {
      "type": "string"
    }
  }
}
```

```
    },
    "date_of_birth": {
      "type": "string"
    },
    "associated_passengers": {
      "type": "array",
      "items": [{
        "type": "object",
        "properties": {
          "name": {
            "type": "string"
          },
          "surname": {
            "type": "string"
          },
          "gender": {
            "type": "string"
          },
          "age": {
            "type": "string"
          }
        }
      }],
      "required": [
        "name",
        "surname",
        "gender",
        "age"
      ]
    },
    "embarkation_port": {
      "type": "string"
    },
    "disembarkation_port": {
      "type": "string"
    },
    "email": {
      "type": "string"
    },
    "phone_number": {
      "type": "string"
    },
    "postal_address": {
      "type": "string"
    },
    "emergency_contact_details": {
      "type": "string"
    },
    "country_of_residence": {
```

```

        "type": "string"
    },
    "mobility_issues": {
        "type": "string"
    },
    "medical_assistance": {
        "type": "string"
    },
    "pregnency_data": {
        "type": "string"
    },
    "is_cew": {
        "type": "string"
    },
    "role": {
        "type": "string"
    },
    "emergency_duty": {
        "type": "string"
    },
    "On_duty": {
        "type": "array",
        "items": [{
            "type": "object",
            "properties": {
                "date": {
                    "type": "string"
                },
                "start": {
                    "type": "string"
                },
                "end": {
                    "type": "string"
                }
            }
        }],
        "required": [
            "date",
            "start",
            "end"
        ]
    }
},
"languages": {
    "type": "array",
    "items": [{
        "type": "string"
    }]
}
},

```

```

    "required": [
        "name",
        "surname",
        "date_of_birth",
        "associated_passengers",
        "embarkation_port",
        "disembarkation_port",
        "email",
        "phone_number",
        "postal_address",
        "emergency_contact_details",
        "country_of_residence",
        "mobility_issues",
        "medical_assistance",
        "pregnancy_data",
        "is_cew",
        "role",
        "emergency_duty",
        "On_duty",
        "languages"
    ]
}

```

Network Access Data Stream

This Data Stream transfers the hardware and software identification details of the devices that a user is in possession of. This information is incorporated into the user's profile and enables the devices of the user to act as their digital fingerprint enabling their proper location identification. Furthermore, it allows PaMEAS to create secure communication channels with the users that can be used by the overall evacuation services and procedures to instantly communicate with them.

- **Dispatch interval:**

This Data Stream outputs data once a new user has registered their device. The registration flow is implemented by the PaMEAS Registration Service - People Management System (PMS) - which implements the onboarding functionality of PaMEAS. Thus, this DS emits data once a user completes the registration flow via the PMS and as a result, a specific interval cannot be defined.

- **Data description:**

The data model for this stream is defined as follows:

Table 8 Network Access Data Model

	Attribute Name	Data Type	Notes
Network Access Data			
1	macAddress	String	Users device MAC address
2	imsi	String	Users device IMSI
3	msisdn	String	Users device MSISDN
4	imei	String	Users device IMEI
5	messagingAppClientId	String	Users device PALAEMON messaging app client id

The JSON Schema of the aforementioned data model is the following:

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "deviceInfoList": {
      "type": "array",
      "items": [{
        "type": "object",
        "properties": {
          "macAddress": {
            "type": "string"
          },
          "hashedMacAddress": {
            "type": "string"
          },
          "imsi": {
            "type": "string"
          },
          "msisdn": {
            "type": "string"
          },
          "imei": {
            "type": "string"
          }
        }
      }
    ],
    "required": [
      "macAddress",
      "hashedMacAddress",
      "imsi",
      "msisdn",
      "imei"
    ]
  }
}
```

```

    ]
  }]
},
"messagingAppClientId": {
  "type": "string"
}
},
"required": [
  "deviceInfoList",
  "messagingAppClientId"
]
}

```

Location Data Stream

This Data Stream transfers the Location and Geofence history data of the user to the PALAEMON back-end storage facility (Data Fusion Bus) and incorporates them to the user's profile enabling in this way a clear view of the Location History of the user.

- **Dispatch interval:**

This Data Stream outputs data in predefined time intervals. Specifically, the current location and geofence of the users are published via this stream every 30 seconds.

- **Data description:**

The data model for this stream is defined as follows:

Table 9 Location Access Data Model

	Attribute Name	Data Type	Notes
Location Access Data			
1	geofenceHistory	Geofence Object Array	List of geofences the user has been identified at
2	locationHistory	Location Object Array	List of location (x,y-coordinates) the user has been identified at

Table 10 Geofence History Data Model

	Attribute Name	Data Type	Notes
Geofence History Data			
1	gfEvent	String	Geofence eventid

2	gfId	String	Geofence id
3	gfName	String	Geofence name
4	macAddress	String	MAC address of the user
5	isAssociated	Boolean	Denotes whether or not this Geofence is associated or not
6	dwellTime	Timestamp	Denotes the time the user remained in the geofence
7	hashedMacAddress	String	Hash of the users MAC address
8	timestamp	Timestamp	Timestamp the user has been noted in the geofence

Table 11 Location History Data Model

	Attribute Name	Data Type	Notes
Location History Data			
1	xLocation	String	x-coordinate of the user
2	yLocation	String	y-coordinate of the user
3	errorLevel	String	Estimation of the error level
4	isAssociated	Boolean	Denotes whether or not this Geofence is associated or not
5	campusId	String	The Id of the overarching project
6	buildingId	String	The Id of the vessel
7	floorId	String	The Id of the floor the user belongs to
8	hashedMacAddress	String	The hashed MAC address of the user
9	geofenceId	String	The Id of the Geofence the user was noted in (based on

			x,y-coordinates)
10	geofenceNames	String	The name of the Geofence the user was noted in (based on x,y-coordinates)
11	timestamp	String	Timestamp the user has been noted in the aforementioned x,y-coordinates

The JSON Schema of the aforementioned data model is the following:

```

schema": "http://json-schema.org/draft-04/schema#",
"type": "object",
"properties": {
  "geofenceHistory": {
    "type": "array",
    "items": [{
      "type": "object",
      "properties": {
        "gfEvent": {
          "type": "string"
        },
        "gfId": {
          "type": "string"
        },
        "gfName": {
          "type": "string"
        },
        "macAddress": {
          "type": "string"
        },
        "isAssociated": {
          "type": "string"
        },
        "dwellTime": {
          "type": "string"
        },
        "hashedMacAddress": {
          "type": "string"
        },
        "timestamp": {
          "type": "string"
        }
      }
    }
  },
  "required": [
    "gfEvent",
    "gfId",

```

```
        "gfName",
        "macAddress",
        "isAssociated",
        "dwellTime",
        "hashedMacAddress",
        "timestamp"
    ]
}]]
},
"locationHistory": {
    "type": "array",
    "items": [{
        "type": "object",
        "properties": {
            "xLocation": {
                "type": "string"
            },
            "yLocation": {
                "type": "string"
            },
            "errorLevel": {
                "type": "string"
            },
            "isAssociated": {
                "type": "string"
            },
            "campusId": {
                "type": "string"
            },
            "buildingId": {
                "type": "string"
            },
            "floorId": {
                "type": "string"
            },
            "hashedMacAddress": {
                "type": "string"
            },
            "geofenceId": {
                "type": "string"
            },
            "geofenceNames": {
                "type": "array",
                "items": [{
                    "type": "string"
                }]
            },
            "timestamp": {
                "type": "string"
            }
        }
    ]
}
```

```

    }
  },
  "required": [
    "xLocation",
    "yLocation",
    "errorLevel",
    "isAssociated",
    "campusId",
    "buildingId",
    "floorId",
    "hashedMacAddress",
    "geofenceId",
    "geofenceNames",
    "timestamp"
  ]
}]
}
},
"required": [
  "geofenceHistory",
  "locationHistory"
]
}

```

Minimal Location Data Stream

This Data Stream transfers the minimal Location and Geofence history data of the user to the PALAEMON back end Storage Layer (Data Fusion Bus). This Data Stream essentially publishes a restricted anonymized view of the user's location for all authorised connected services, in such a way that no link can be made to their Personal Identification Information.

- **Dispatch interval:**

This Data Stream outputs data in predefined time intervals. Specifically, the current location and geofence of the users are published via this stream every 30 seconds.

- **Data description:**

The data model for this stream is defined as follows:

Table 12 Minimal Location Data Model

	Attribute Name	Data Type	Notes
Minimal Location Data			
1	xLocation	String	x-coordinate of the user

2	yLocation	String	y-coordinate of the user
8	hashedMacAddress	String	The hashed MAC address of the user
9	geofenceId	String	The Id of the Geofence the user was noted in (based on x,y-coordinates)
10	geofenceNames	String	The name of the Geofence the user was noted in (based on x,y-coordinates)
11	timestamp	String	Timestamp the user has been noted in the aforementioned x,y-coordinates

The JSON Schema of the aforementioned data model is the following:

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "gfId": {
      "type": "string"
    },
    "gfName": {
      "type": "string"
    },
    "hashedMacAddress": {
      "type": "string"
    },
    "timestamp": {
      "type": "string"
    },
    "xLocation": {
      "type": "string"
    },
    "yLocation": {
      "type": "string"
    }
  },
  "required": [
    "gfId",
    "gfName",
    "hashedMacAddress",
    "timestamp",
    "xLocation",
    "yLocation"
  ]
}
```

```
]
}
```

3.2.5 Smart Cameras

- **Short Description of the data being passed:**

Three different types of information will be obtained by each Smart Camera (SC) node in the surveillance area: i) people count, ii) stampede detection, and iii) trapped people detection. These data will be reported to the general services (e.g., PIMM) to use them adequately. SC will consider at least three different areas: long corridors, open indoor spaces, and deck stairs. SC will execute at least 2 different modes, regular and evacuation modes, having different data rates, performance, and image analysis. Besides, SC will inform about the abnormal flow of people.



- **Dispatch interval:**

Event detection: Asynchronous dispatch when an incident occurs.

Video: Asynchronous dispatch when an incident occurs; Ship status = 2 or higher (evacuation mode).

- **Data description:**

Event detection (JSON form)

Video - Smart Cameras will start sending their video feed so that the crew can better control the different localizations in the ship and report any blocked path or any other problem with the passengers.

3.2.6 Decision Support System

- **Short Description of the data being passed:**

The PALAEMON DSS provides to PIMM's dashboard the actions-suggestions that the Master has to make depending on the incident that has been occurred. So, lists of actions are displayed to the DSS through the PIMM. Also, the actions are accompanied by the target (Master, engineer, etc.) that every action is referred to and risk level indications for the as-

assessment of the situation. The indication is shown on the DSS to assist the Master to assess and monitor the condition of the vessel better as he knows the risk of taking a decision. Furthermore, the values of fire real-time sensors that are installed into the ship are provided from the DSS to PIMM in order for the Master to be able to observe the condition of the sensors.

- **Dispatch interval:**

The DSS is triggered every time an incident occurs. When the Situation Assessment state is enabled from the Evacuation Coordinator the DSS is triggered and started to assist the Master until to return to the Normal Operation state.

- **Data description:**

A JSON file that contains all the suggestions of the DSS:

```
defaultSuggestion: [{
  incident: "Grounding",
  actions: {
    suggestions: [{
      id: "GRD_001",
      sug: "A visual inspection must be performed. All fuel tanks, ballast tanks and hulls must be counted.",
    },
    {
      id: "GRD_002",
      sug: "All other apartments that are in contact with the sea should be inspected to ensure that they are intact.",
    },
    {
      id: "GRD_003",
      sug: "The measurements of tanks, fuel and ballast and hulls must be compared with the latest measurements of the same areas to detect any leaks.",
    },
    {
      id: "GRD_004",
      sug: "Every inclination of the ship must be noted and included in the report.",
    },
  ],
  final_text: "Did you do all of the above?",
  actions: [{
    action: "YES",
    first_text: "Master should consider the following:",
    suggestions: [{
      id: "GRD_005",
      sug: "The danger to the ship crew if the ship slips from the grounding position",
    },
  ],
  ],
}
```

```

    },
    {
      id: "GRD_006",
      sug: "The risk of ship breaking due to rough sea or
big waves.",
    },
    {
      id: "GRD_007",
      sug: "The risk to the health of the crew and the population of
the area from the escape of dangerous loads in dangerous quan-
tities.",
    },
    {
      id: "GRD_008",
      sug: "The prevention of fire due to the escape of flamma-
ble materials in combination with uncontrolled sources of igni-
tion.",
    },
  ],
  .....
}]

```

A JSON file that contains the data enrichment of the suggestions:

```

suggestionDataMap: {
  "GRD_001": {
    target: ["Master"],
  },
  "GRD_002": {
    target: ["Master"],
    info: [
      ["Risk", "this.$store.getters.getHullStatus"]
    ],
  },
  "GRD_003": {
    target: ["Master"],
  },
  "GRD_004": {
    target: ["Master"],
    info: [
      ["Risk", "this.
$store.getters.getStabilityLikelihood "], ["
Inclination ", "
this.$store.getters.getStabilityToolkit "],
    ]
  },
}

```

```

"GRD_005": {
  target: ["Master"],
  info: [
    ["Risk", "this.$store.getters.getIncidentSeverity"]
  ],
},
"GRD_006": {
  target: ["Master"],
  info: [
    ["Risk", "this.$store.getters.getStructuralIntegrity"],
  ],
},
"GRD_007": {
  target: ["Master"],
  info: [
    [
      "Risk referred to the crew only, and not for population of the ar-
ea",
      "this.$store.getters.getIncidentSeverity",
    ],
  ],
},
"GRD_008": {
  target: ["Master"],
},
.....
}

```

3.2.7 Smart Safety System

- **Short Description of the data being passed:**

The Smart Safety System will generate Kafka messages in JSON form, based on the events, which are entered manually by the Bridge Team and Crew.

- **Dispatch interval:**

On event generation.

- **Data description:**

A JSON form

```

{
  "Id": 1,
  "Type": "Fire",
  "Timestamp": "2020-11-09T10:42:58.790517+00:00",
  "Deck": 3,
  "Position": [{"x": 55, "y": 85}]
}

```

}

3.2.8 Ship Structural Monitoring Ecosystem

- **Short Description of the data being passed:**

The SHM and IMU systems generate data regarding angles and heave, which is translated to deflection of the ship. These data are recorded real-time on the SHM software database and the file that has the following form:

Table 13 SHM Data

Timestamp	Roll angle	Pitch angle	Yaw angle
Fri May 28 06:37:55 2021	-0.3	2.8	-133.5
Fri May 28 06:37:55 2021	-0.3	2.8	-133.5
Fri May 28 06:37:55 2021	-0.3	2.8	-133.5
Fri May 28 06:37:55 2021	-0.3	2.8	-133.5
Fri May 28 06:37:56 2021	-0.3	2.8	-133.4
Fri May 28 06:37:56 2021	-0.3	2.8	-133.4
Fri May 28 06:37:56 2021	-0.3	2.8	-133.4
Fri May 28 06:37:56 2021	-0.3	2.8	-133.4

- **Dispatch interval:**

The SHM and IMU system transmits an alarm when a parameter, e.g. roll or pitch angle or deflection, exceeds a limiting value. For instance, if we have an asset with a limiting value for roll angle the value 25 degrees when the SHM system records a roll angle of 26 degrees the SHM system automatically sends a file with all the values of the parameters to the PALAEMON core system. Furthermore, it keeps sending real-time data with a frequency of 1 Hz until the value of the critical parameter drops below the limiting value, i.e. roll angle below 25 degrees. Lastly, the SHM also sends a status report file at predetermined time intervals, e.g. every 10 minutes. The AE system also works by sending alarms when a parameter has reached a critical value.

- **Data description:**

The SHM is connected to the PALAEMON core through Kafka and transmits a JSON file as described above. In particular, it sends JSON files as alarms, when a critical event has occurred and as status reports at predetermined time intervals.

3.2.9 Evacuation Coordinator

- **Short Description of the data being passed:**

Messages forwarded by the PALAEMON Evacuation Coordinator are straightforward, as illustrated, which represents an arbitrary notification captured by the system.

```
{
  "current": 1,
  "previous": 0,
  "timestamp": "2020-11-09T10:42:58.790517+00:00",
  "originator": "PIMM",
  "incident_type": "fire / grounding"
}
```

Where:

- **Current:** New evacuation status
 - **Previous:** Past status (NOTE: if current status has a value of “N”, the previous one can only be either “N-1” or “N+1”. No other option is possible)
 - **Timestamp:** Exact moment where the signal was generated (to keep track of the whole evacuation process)
 - **Originator:** Software element that triggered the event. At the time of writing this proposal, the only actor capable of generating this is the PIMM; however, during the execution of the Application Field Trials (WP8) we may detect the need of new originators.
 - **Incident type:** With the help of additional components (e.g., Smart Safety System, Smart Risk Assessment Platform, etc.), the system may detect the actual source that has given rise to the incident (e.g., Fire, grounding, etc.). In case this happens, the PALAEMON Evacuation Coordinator will share this with the rest of the components.
- **Dispatch interval:**

These messages are completely asynchronous since they are generated upon human intervention (i.e., the Master manually pointing at the new status on the PIMM dashboard).)

- **Data description:**

```
{
  "current": 1,
  "previous": 0,
  "timestamp": "2020-11-09T10:42:58.790517+00:00",
  "originator": "PIMM",
  "incident_type": "fire / grounding"
}
```

3.3 Data Fusion Bus Integration

Within the PALAEMON project, a large number of components intercommunicate in order to detect assess and facilitate the ship evacuation process. The data fusion Middleware, which is called Data Fusion Bus (DFB), is expected to provide a common way of collecting and combining diverse real-time data from multiple components, in a secure way and provide access to both real-time and historical data.

Following PALAEMON architecture (D2.7 PALAEMON Architecture (v2), DFB is part of:

- Data access: DFB provides an event-driven unified way of intercommunication between components
- Platform Core: DFB is responsible for storing and providing data

The exact functional and non-functional requirements were presented in the above-mentioned deliverable (D2.7), along with the main features of the Data Fusion Bus.

3.3.1 Description of the Data Fusion Bus

Data Fusion Bus is an analytics platform developed by ITML. The goal of DFB is to enable developing, deploying, operating and managing a big data environment with emphasis on real-time applications. It combines the features and capabilities of several big data open-source applications and utilities within a single platform.

The key capabilities of DFB are:

- Smart Production Digitization and IoT
- Data aggregation from heterogeneous data sources and data stores.
- Real-time analytics offering ready-to-use Machine Learning algorithms for classification, clustering, regression, anomaly detection.
- An extendable and highly customizable User Interface for Data Analytics, manipulation and filtering. The UI also includes functionality for managing the platform. Web Services for exploiting the platform outputs for Decision Support.

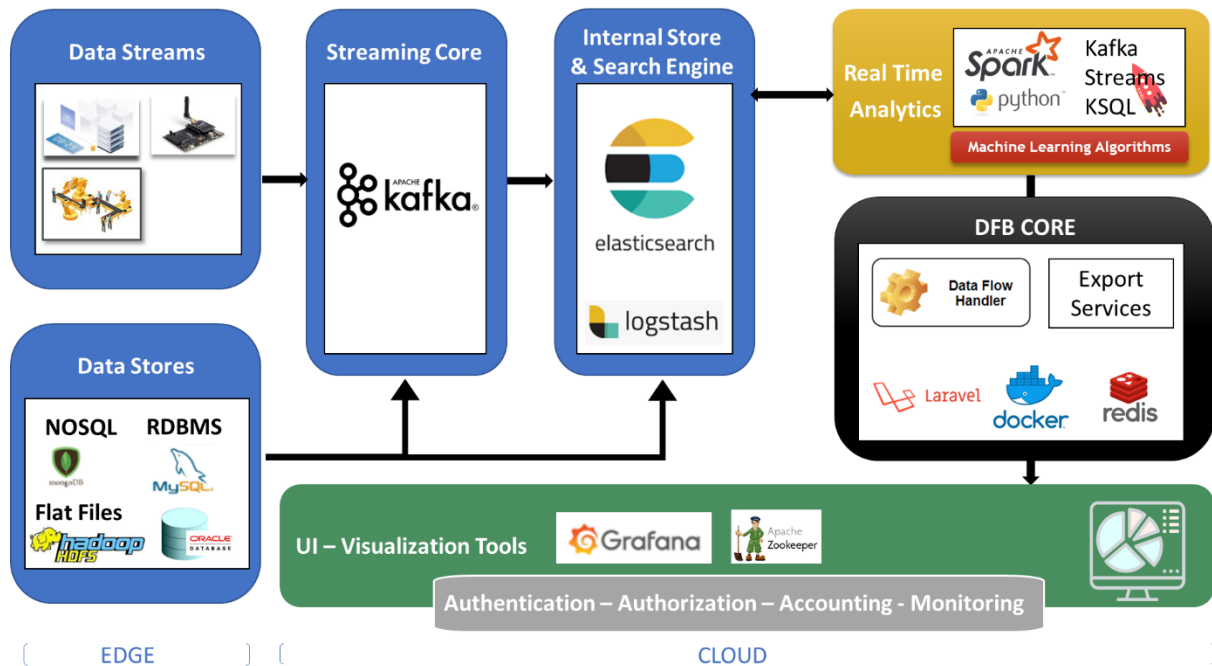


Figure 6 Data fusion Bus

The cornerstone technologies assembled in DFB, that are relevant to PALAEMON requirements are the following:

- Streaming Core
- Internal Store and Search Engine
- Monitoring stack

Within PALAEMON, parts of DFB have been adapted and reconfigured to meet the specific requirements of the project. Further depiction and information in Deliverable 6.2 PALAEMON Data fusion middleware.

3.3.2 Integration Methodology

The aim of the PALAEMON Incident Management Module (PIMM) is to interconnect the PALAEMON toolkits between them. The PIMM is the bridge that transfers data between the components. Therefore, the PALAEMON components can communicate and exchange data.

This section will be described the Integration Methodology of the PALAEMON components to the PIMM. First of all, we need to define what is Apache Kafka.

Apache Kafka is a distributed data store optimized for ingesting and processing streaming data in real-time. Streaming data is data that is continuously generated by thousands of data sources, which typically send the data records in simultaneously. A streaming platform needs to handle this constant influx of data, and process the data sequentially and incrementally.

Kafka provides three main functions to its users:

- Publish and subscribe to streams of records

- Effectively store streams of records in the order in which records were generated
- Process streams of records in real-time

Kafka is primarily used to build real-time streaming data pipelines and applications that adapt to the data streams. It combines messaging, storage, and stream processing to allow storage and analysis of both historical and real-time data.

It is important for the achievement of the PALAEMON project vision, the data that are entered into PIMM and are transferred between the PALAEMON Components to be very promptly responded to. When an incident occurs the PALAEMON components must communicate and exchange data rapidly.

Kafka is used to build real-time streaming data pipelines and real-time streaming applications. A data pipeline reliably processes and moves data from one system to another, and a streaming application is an application that consumes streams of data. Kafka combines two messaging models, queuing and publish-subscribe, to provide the key benefits of each to consumers. Queuing allows for data processing to be distributed across many consumer instances, making it highly scalable. The publish-subscribe approach is multi-subscriber, but because every message goes to every subscriber it cannot be used to distribute work across multiple worker processes. Kafka uses a partitioned log model to stitch together these two solutions. A log is an ordered sequence of records, and these logs are broken up into segments, or partitions, that correspond to different subscribers. This means that there can be multiple subscribers to the same topic and each is assigned a partition to allow for higher scalability. Finally, Kafka's model provides replayability, which allows multiple independent applications reading from data streams to work independently at their own rate.

It is necessary for the PALAEMON components to be interconnected fastly with reliability on high performance. Apache Kafka is one of the best solutions for integration on the PALAEMON project to achieve an excellent procedure to eliminate the problems that are created when an incident occurs.

4 Functionality

4.1 Proposed Methodology

Since maritime disasters in recent years are a stark reminder of the imperative need for timely and effective evacuation of large passenger ships during an emergency, the aim of this project (PALAEMON) is to maximize the effectiveness of passenger evacuation, during an emergency and/or a serious incident, from large Cruise and RoPax ships by combining the expertise of stakeholders from the field of cruise ship manufacturing, large cruise ship operators, classification societies, sensor, and technology organizations, with a multidisciplinary group of innovators. So, different systems have been implemented in order to achieve the above aim. For example, the Weather Forecast Toolkit (WFT) is a toolkit that provides the Master with current and predicted weather conditions data so he knows how the weather is and how it will evolve as it will help him how to act depending on the weather conditions. The Decision Support System (DSS) is a system that helps the Master take the proper ac-



tions after a grounding, fire, or collision incident of the ship, in order to avoid the adverse consequences of the incident and avert the problem that may drive to an evacuation of the ship. The Ship Stability Toolkit (SST) is a component of the PALAEMON project, which is designed to assist the Master and Bridge Command Team during an incident with ship motion predictions. The Smart Safety System (SSS) component of the PALAEMON project is intended to create an assisting safety system during evacuation.

The PIMM platform connects all the components of the PALAEMON Project. The importance of this is that provides to the Master and crew member information for the ship, knowledge of the current situation, and suggested actions to be taken in order to avoid adverse consequences. The PIMM Platform is the brain of the whole project that contains all the functionality of the different components.

Data Collection

The data of the different components that are able to assist when an incident occurs are collected and used on the PIMM Platform.

All the components that PIMM Platform uses are the followings:

- Ship Stability Toolkit
- Weather Forecast Toolkit
- Smart Risk Assessment Platform
- PAMEAS
- Smart Cameras
- Decision Support System
- Smart Safety System
- Ship Health Monitoring Ecosystem
- Evacuation Coordinator
- Data Fusion Bus Integration

Data Analysis

- Ship Stability Toolkit

The Ship Stability Toolkit relies on inputs that are gathered either manually or automatically from the ship's AIS data, the Weather Forecast Toolkit, and the Ship Health Monitoring Ecosystem.

- Weather Forecast Toolkit

The WFT provides three types of data displayed to the PIMM's dashboard. First, the WFT provides weather data based on the ship's course. Also, WFT provides the effective and the ineffective actions that the crew implemented during the incident for a similar accident based on the weather conditions. Finally, WFT aggregates all the incidents in the dataset based on the wind speed and the visibility conditions at the incident time and provides charts (to PIMM's dashboard) that show the expected percentages of injuries and deaths for each incident.

- Smart Risk Assessment Platform

SRAP provides to the PALAEMON DSS risk level indications for the assessment of the situation. These data points are separated into three categories: a. Situation Assessment, b. Mustering Assessment, c. Ship Abandonment.

- PAMEAS

Data that enable the tracking and monitoring of the passengers and crew position within the ship, the PaMEAS system components generate various data streams.

- Smart Cameras

Three different types of information will be obtained by each Smart Camera (SC) node in the surveillance area:

- i) people count,
- ii) stampede detection, and
- iii) trapped people detection.

- Decision Support System

The PALAEMON DSS provides to PIMM's dashboard the actions-suggestions that the Master has to make depending on the incident that has been occurred accompanied by the target (Master, engineer, etc.) that every action is referred to and risk level indications for the assessment of the situation.

- Smart Safety System

The Smart Safety System will generate Kafka messages in JSON form, based on the events, which are entered manually by the Bridge Team and Crew.

- Ship Health Monitoring Ecosystem

The SHM and IMU systems generate data regarding angles and heave, which is translated to deflection of the ship. These data are recorded in real-time on the SHM software database. The SHM and IMU system transmits an alarm when a parameter, e.g. roll or pitch angle or deflection, exceeds a limiting value.

- Evacuation Coordinator

Asynchronous messages are generated upon human intervention (i.e., the Master manually pointing at the new status on the PIMM dashboard). The evacuation Coordinator translates these human-based commands to (simple) digital signals that are propagated across the PALAEMON system, to warn all software components (i.e., the ones that depend on the ship evacuation status) that something has occurred and they must react accordingly. Some components do modify their operation mode as a function of the ship's evacuation status.

- Data Fusion Bus Integration



Data Fusion Bus is an analytics platform. The goal of DFB is to enable developing, deploying, operating, and managing a big data environment with an emphasis on real-time applications. It combines the features and capabilities of several big data opensource applications and utilities within a single platform.

Problems the PIMM Platform solves

The PIMM Platform gathers all the data of the PALAEMON components and exchanges information between the components. All the components are displayed on the PIMM screen. For this reason, the PIMM screen is separated into different parts to contain all the data used by the components. So, the Master is able to know the situation of the whole ship at every time.

So, when an incident occurs is so important for the Master to be afforded with the condition of the ship's components. In most incidents, confusion prevails because the severity of the situation makes it difficult for people to think. So, the Master is incapable of thinking calmly about the situation and this has the impact of making wrong decisions. Many times, the incident has taken adverse evolutions because the master decides that under normal circumstances, he would never make it. Therefore, as the PIMM screen provides all the proper information and suggested actions to the Master/ Bridge eliminates the confusion that is caused due to the incident.

Also, an equally important solution of the PIMM is that most components are updated automatically whenever the data changed. For example, the suggested actions that Weather Forecast Toolkit provides are displayed depending on the weather conditions. Providing this insurance of actions that the Master has to make can contribute to the successful handling of a maritime incident.

4.2 User Interfaces

4.2.1 User Interface Composition

This section will describe the PIMM User Interface. The PIMM screen is separated into three tabs. The first one is the Decision Support System, the second is the Videos Streaming Center and the third is the Incident Assessment.

The following image shows the Decision Support System tab that contains the Ship Stability Toolkit (SST), the Decision Support System (DSS), the Weather Forecast Toolkit (WFT), and the Evacuation Coordinator.



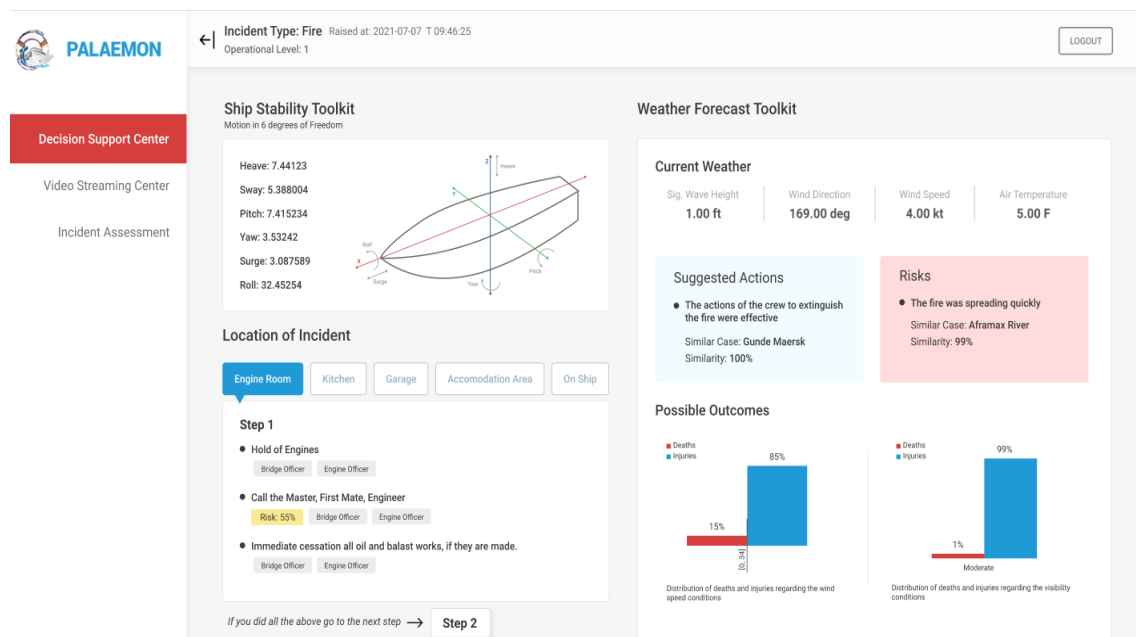


Figure 7 PIMM Layout - Decision Support System (1)

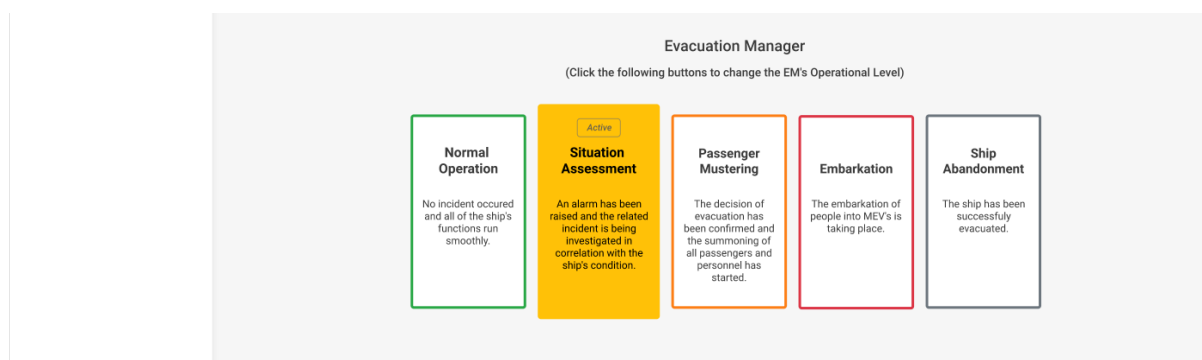


Figure 8 PIMM Layout - Decision Support System (2)

The Video Streaming Center tab contains the cameras from the Smart Cameras PALAEMON component.

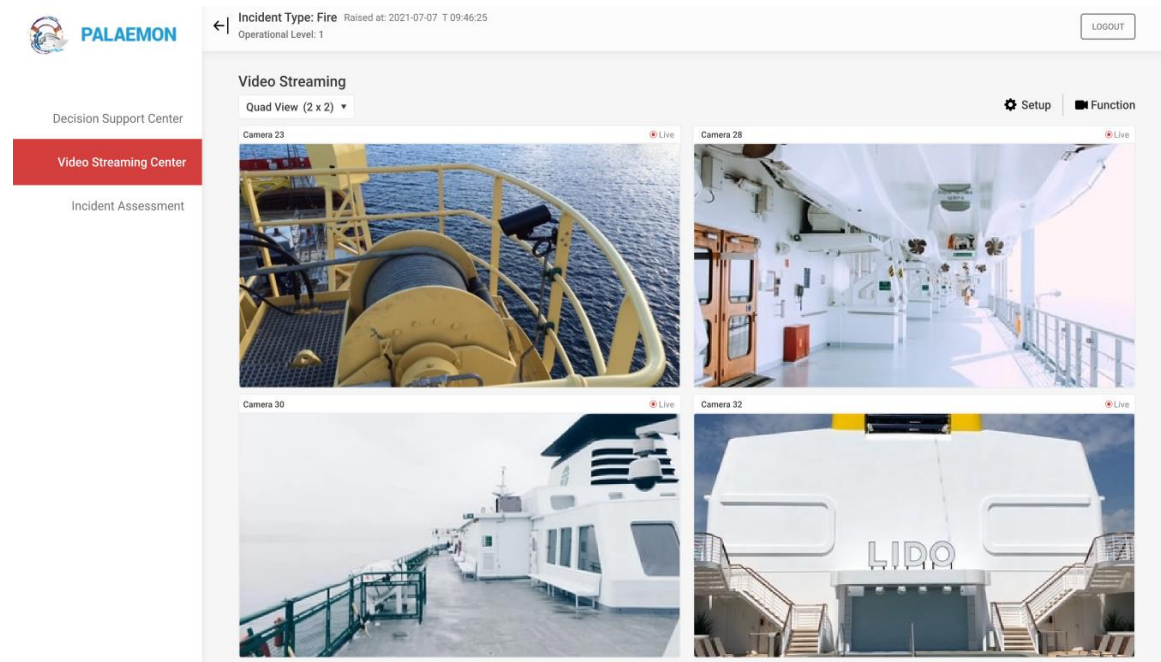


Figure 9 PIMM Layout - Video Streaming

The Incident Assessment tab contains the Diagram of Danger in Ship that is produced through the SRAP PALAEMON component, and the PaMEAS component.

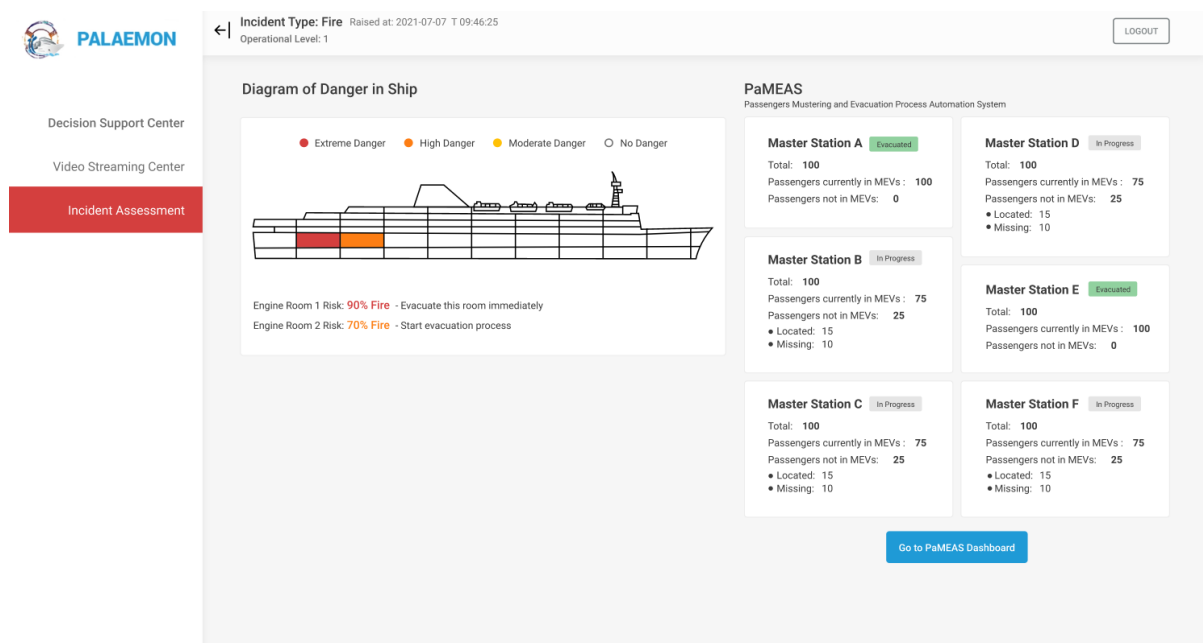


Figure 10 PIMM Layout - Incident Assessment

The reason that the PIMM User Interface is separated in this way is to gather the relative components together.

So, in the Decision Support System tab, there are all the components that provide suggested actions to the Master/ Bridge and the Evacuation Coordinator that the Master interacts with in order to change the state of the ship. The Cameras have been integrated into the Video Streaming Center in order for the Master to have absolute supervision. The Incident Assessment tab contains two PALAEMON components that indicate the state of every part of the ship when the evacuation process starts.

Also, there is a sensor button that is the Early Fire Detection System that will be described next.

4.2.1.1 Ship Stability Toolkit (T3.2)

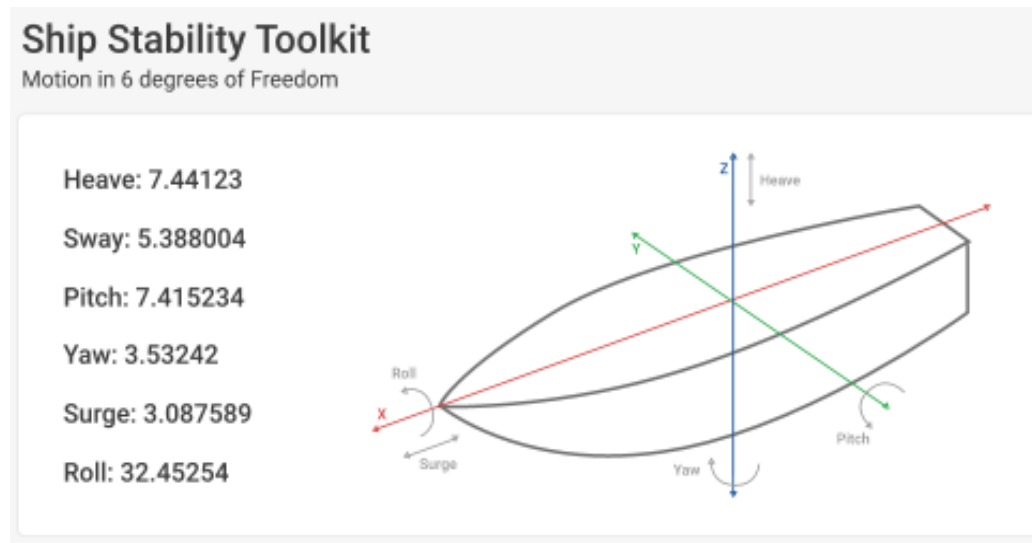


Figure 11 PIMM Layout - Ship Stability Toolkit

The Ship Stability Toolkit (SST) relies on inputs that are gathered either manually or automatically from the ship's AIS data, the Weather Forecast Toolkit, and the Ship Health Monitoring Ecosystem. Based on the provided information the SST will output the motion prediction data via Kafka to the PIMM.

The Ship Stability Toolkit indicates through PIMM the values of heave, sway, pitch, yaw, surge, and roll to the Master. Next to the above information, there is a diagram that shows how is the impact of them on the ship.

4.2.1.2 Weather Forecast Toolkit (T3.3)

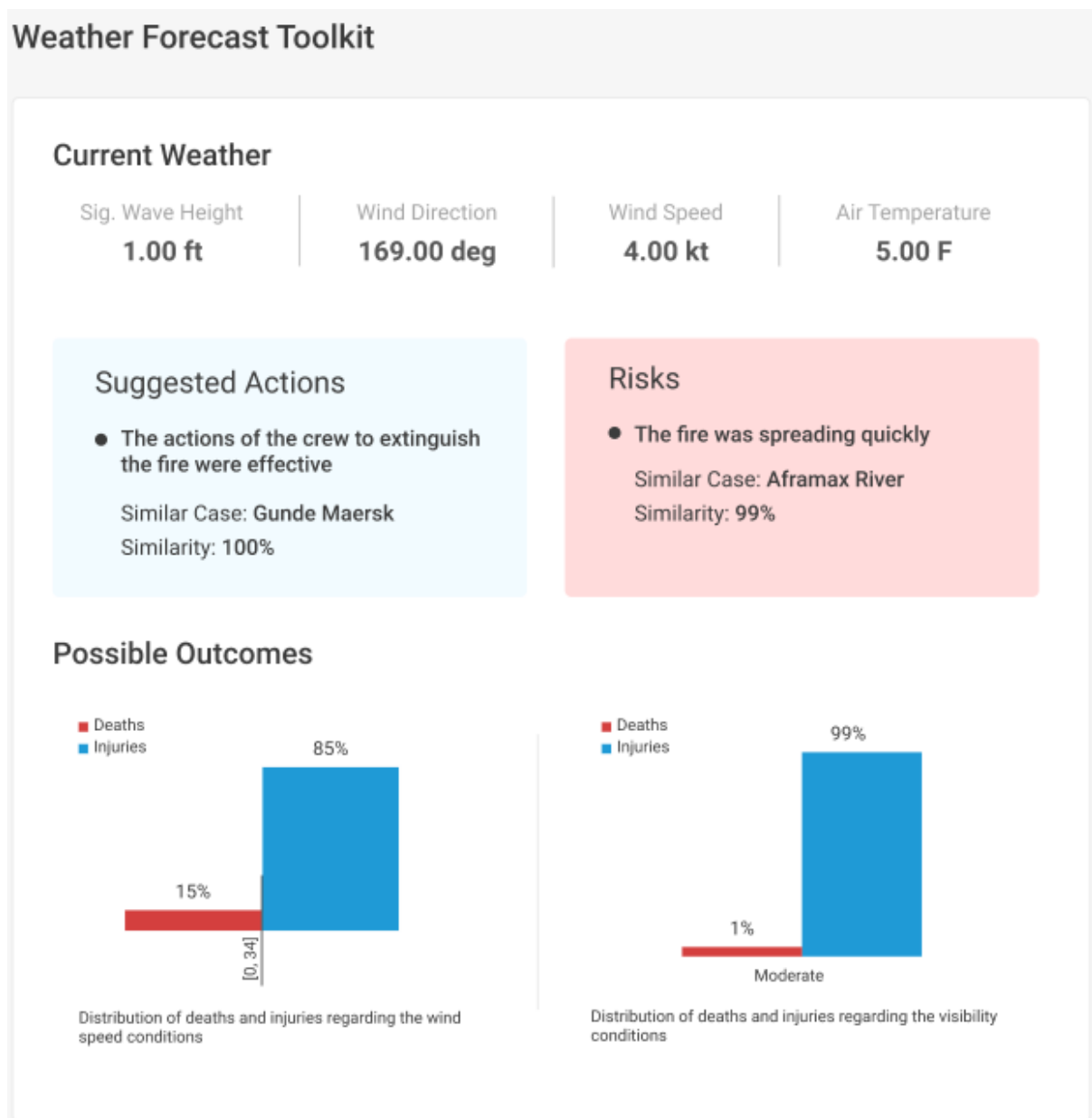


Figure 12 PIMM Layout - Weather Forecast Toolkit

The Weather Forecast Toolkit (WFT) provides weather data based on the ship's course. Also, WFT provides the effective and the ineffective actions that the crew implemented during the incident for a similar accident based on the weather conditions. Finally, WFT aggregates all the incidents in the dataset based on the wind speed and the visibility conditions at the incident time and provides charts (to PIMM's dashboard) that show the expected percentages of injuries and deaths for each incident.

4.2.1.3 Risk Assessment Platform (T3.5)

Diagram of Danger in Ship

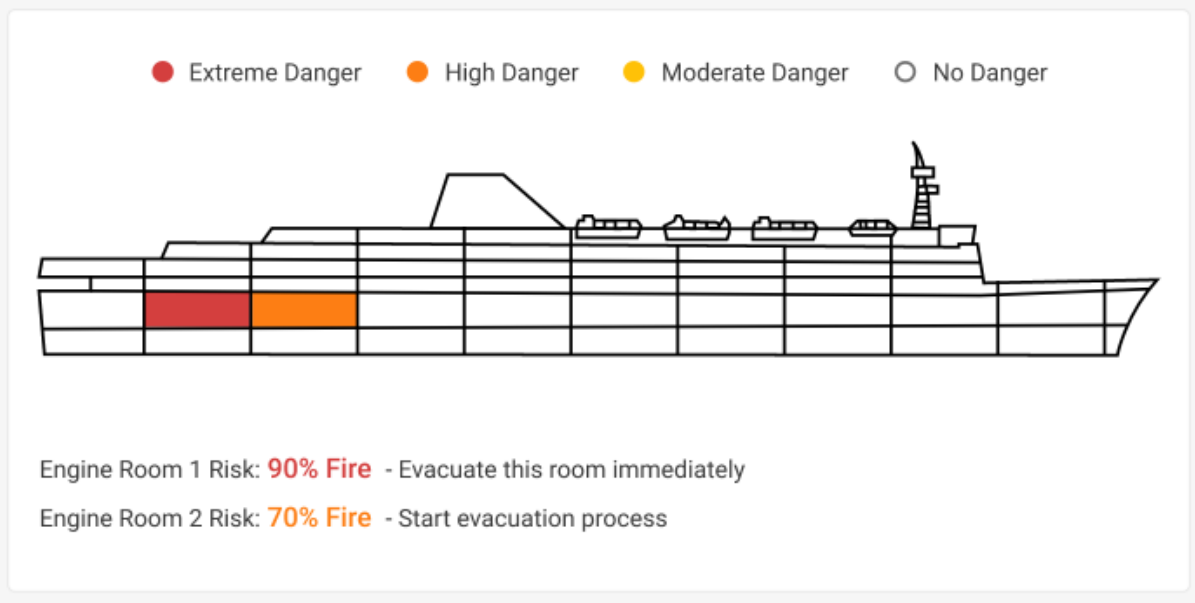


Figure 13 PIMM Layout - Danger in Ship

The main goal of the SRAP is to enhance the situation awareness of the crew members and to assist in the incident/accident management lifecycle. The SRAP provides its main output to PIMM every time that SRAP will be provided with revised input data.

The SRAP indicates the state of the parts of the ship. The parts that are colorized in red color have an extreme danger state. The parts that are colorized in red color have a high danger state. The parts with orange color have a moderate danger state and those with white color have no danger state.

4.2.1.4 PAMEAS (T5.4)

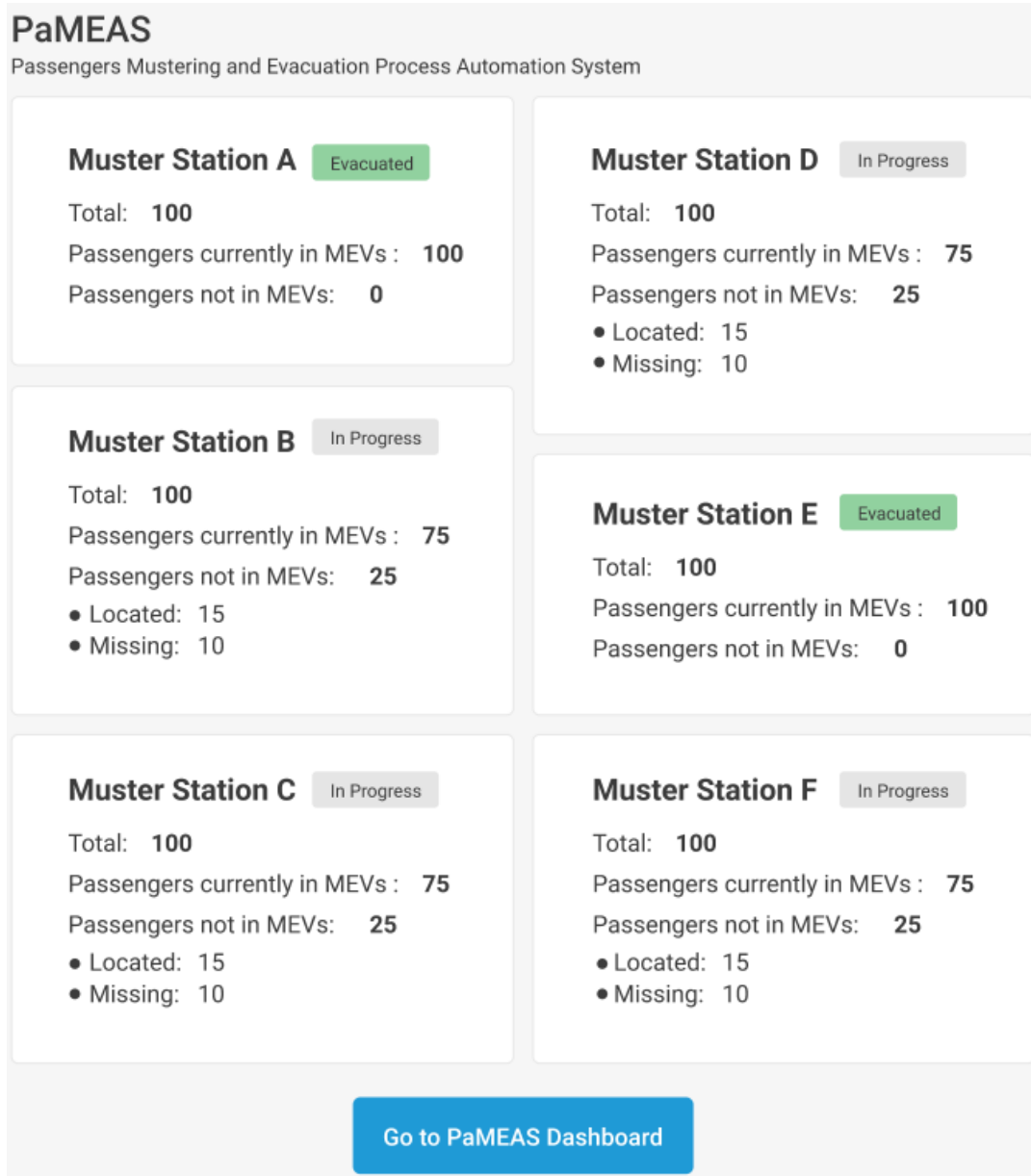


Figure 14 PIMM Layout - PaMEAS

Passengers Mustering and Evacuation Process Automation System (PaMEAS) component of the PALAEMON project, implements an IT-enabled functionality that can streamline and support in real-time the evacuation process. PaMEAS essential objective is to optimize the evacuation process via an “augmented”, technology-aided process redesign. The PaMEAS send data to PIMM such as the total number of passengers, how many of them are currently in MEVs and how many are not in MEVs depending on the Master Station. Also, it provides how many of them that are not in MEVs have been located and missed.

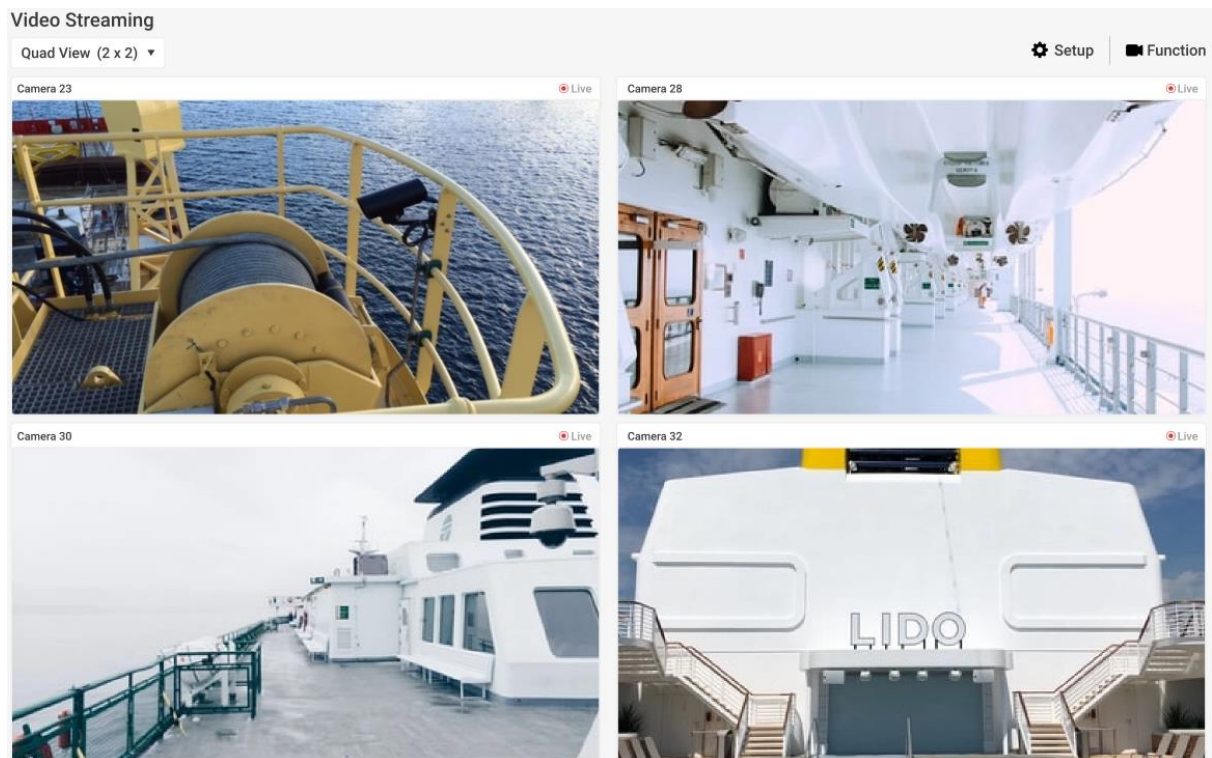
4.2.1.5 Smart Cameras (T5.5)

Figure 15 PIMM Layout - Video Streaming

The Smart Cameras PALAEMON component will provide the cameras of the ship to the PIMM. So, the Master will be able to observe the Cameras of the ship whenever he wants to have a better view of the situation of the incident.

4.2.1.6 Decision Support System (T6.4)

Location of Incident

Engine Room Kitchen Garage Accomodation Area On Ship

Step 1

- Hold of Engines
 - Bridge Officer Engine Officer
- Call the Master, First Mate, Engineer
 - Risk: 55% Bridge Officer Engine Officer
- Immediate cessation all oil and balast works, if they are made.
 - Bridge Officer Engine Officer

If you did all the above go to the next step → **Step 2**

Figure 16 PIMM Layout - Decision Support System

The PALAEMON DSS provides to PIMM's dashboard the actions-suggestions that the Master has to make depending on the incident that has been occurred. So, lists of actions are displayed to the DSS through the PIMM. Also, the actions are accompanied by the target (Master, engineer, etc.) that every action is referred to and risk level indications for the assessment of the situation. The indication is shown on the DSS to assist the Master to assess and monitor the condition of the vessel better as he knows the risk of taking a decision.

The above image is the screen of the PALAEMON DSS when a fire/explosion occurs that will be displayed to the Master/Bridge. The master has to choose where the location of the fire incident is and depending on that the DSS displays the suggested actions. Under the suggested actions are displayed the risk level indications from the SRAP toolkit of the PALAEMON project and to whom the action is addressed to.

4.2.1.7 Smart Safety System

The Smart Safety System (SSS) component of the PALAEMON project is intended to create an assisting safety system during evacuation. The aim of the system is to improve the safety evacuation system onboard passenger ships and to provide information on the condition and all relevant information during the evacuation process. This Model of the SSS was built on the adapted similar System which is used at coordinating firefighter, when they are proceeding with building evacuation.

This module gives an overview of the evacuation status and the current status of the evacuation progress. It also provides an interface for the connection between the master and the evacuation team. The information is provided as a gadget in the PALAEMON Decision Support System (DSS). The DSS detects if there is a fire incident through the SSS.

4.2.1.8 Ship Structural Monitoring Ecosystem

The SHM and IMU system transmits an alarm when a parameter, e.g. roll or pitch angle or deflection, exceeds a limiting value. For instance, if we have an asset with a limiting value for roll angle the value 25 degrees when the SHM system records a roll angle of 26 degrees the SHM system automatically sends a file with all the values of the parameters to the PALAEMON core system. Furthermore, it keeps sending real-time data with a frequency of 1 Hz until the value of the critical parameter drops below the limiting value, i.e. roll angle below 25 degrees. Lastly, the SHM also sends a status report file at predetermined time intervals, e.g. every 10 minutes. The AE system also works by sending alarms when a parameter has reached a critical value.

4.2.1.9 Evacuation Coordinator

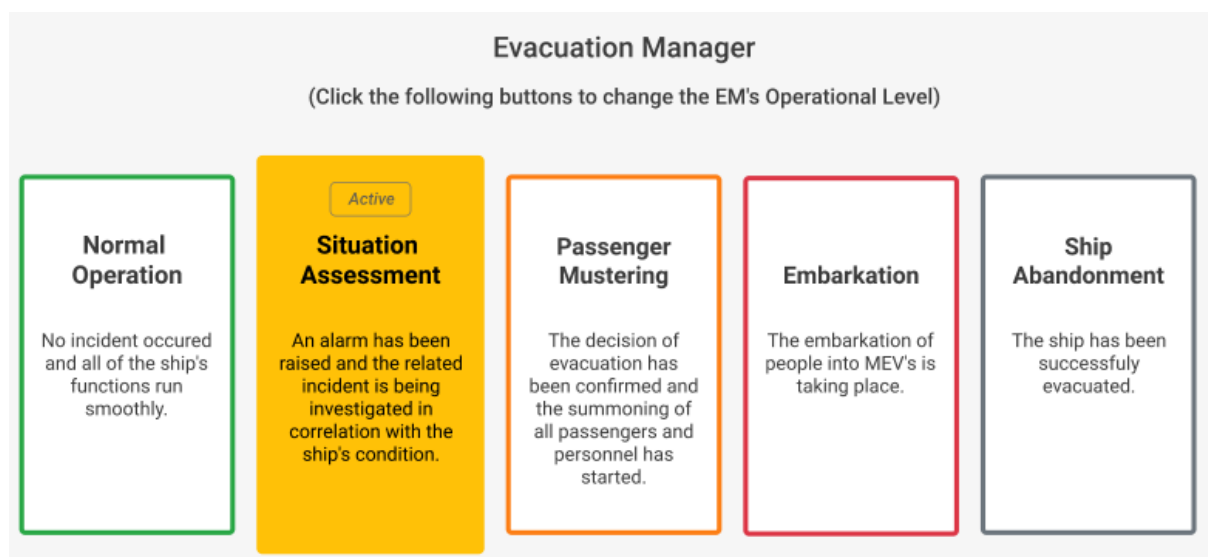


Figure 17 PIMM Layout - Evacuation Coordinator

The Evacuation coordinator manages to translate the above human-based commands to (simple) digital signals that are propagated across the PALAEMON system, in order to warn all software components (i.e., the ones that depend on the ship evacuation status) that something has occurred and they must react accordingly.

The Master is able to change the states of the evacuation manager by changing from normal operation state to the Situation Assessment, then next to the Passenger Mustering, etc. So, the Master updates the state that the ship is in order for the other PALAEMON components to be triggered and assist the situation to avoid the adverse consequences of the incident.

4.2.1.10 Early Fire Detection System

Fire Sensors			
Label	Value	Max	
Kitchen Fire Sensor	55	45	Ignore
Engine Room Fire Sensor	55	45	Ignore
Garage Fire Sensor	35	45	Ignore
Accommodation Area Fire Sensor	35	45	Ignore
Deck 1 Fire Sensor	35	45	Ignore
Deck 2 Fire Sensor	35	45	Ignore
Deck 3 Fire Sensor	35	45	Ignore
			<input type="button" value="Declare"/> <input type="button" value="Close"/>

Figure 18 PIMM Layout - Early Fire Detection System

The PALAEMON DSS provides an Early Fire Detection System that indicates the condition of the fire sensors of the ship. The Master can be notified when a value of a fire sensor exceeds the maximum threshold and is able to declare a fire incident in order for the DSS and other PALAEMON components to be activated.

When a value of a sensor surpasses the maximum threshold then it changes in red color in order for the Master to understand that there is a possibility of a fire incident. Then, the Master has the capability, for each fire sensor that has surpassed the maximum threshold, to declare a fire incident or ignore the situation after manual inspection of the sensor (in case of a malfunction). If the Master declares a fire incident then the DSS is triggered and displays the action lists depending on the place of fire that the incident happens.

4.3 User Authentication and User Management

The User Authentication and User Management have not been confirmed yet as of the time of writing, however, the PIMM will be integrated to the final Single-Sign-On (SSO) system as soon as the specifications are available.

4.4 Containerization and Deployment

Containerization is a form of virtualization where applications run in isolated user spaces, called containers while using the same shared operating system (OS). A container is essentially a fully packaged and portable computing environment

Everything an application needs to run—its binaries, libraries, configuration files, and dependencies are encapsulated and isolated in its container. The container itself is abstracted away from the host OS, with only limited access to underlying resources—much like a lightweight virtual machine (VM). As a result, the containerized application can be run on various types of infrastructure on bare metal, within VMs, and in the cloud without needing to refactor it for each environment.

With containerization, there's less overhead during start-up and no need to set up a separate guest OS for each application since they all share the same OS kernel. Because of this high efficiency, containerization is commonly used for packaging up the many individual micro-services that make up modern apps.

Here are the reasons to use containers:

- Containers have no guest OS and use the host's operating system. So, they share relevant libraries and resources as and when needed.
- Processing and execution of applications are very fast since applications specific binaries and libraries of containers run on the host kernel.
- Booting up a container takes only a fraction of a second, and also containers are lightweight and faster than Virtual Machines.

After the explanation of the containerization, Docker will be explained. Docker is an open platform for developing, shipping, and running applications. Docker enables you to separate your applications from your infrastructure so you can deliver software quickly. With Docker, you can manage your infrastructure in the same ways you manage your applications. By taking advantage of Docker's methodologies for shipping, testing, and deploying code quickly, you can significantly reduce the delay between writing code and running it in production.

Docker provides the ability to package and run an application in a loosely isolated environment called a container. The isolation and security allow you to run many containers simultaneously on a given host. Containers are lightweight and contain everything needed to run the application, so you do not need to rely on what is currently installed on the host. You can easily share containers while you work, and be sure that everyone you share with gets the same container that works in the same way.

4.4.1 Docker advantages

- Docker containers provide a way to build applications that are easier to assemble, maintain, and move around than previous methods allowed. That provides several advantages to software developers³.
- Docker containers are minimalistic and enable portability. Docker lets applications and their environments be kept clean and minimal by isolating them, which allows for more granular control and greater portability.
- Docker containers enable composability. Containers make it easier for developers to compose the building blocks of an application into a modular unit with easily interchangeable parts, which can speed up development cycles, feature releases, and bug fixes.

An important feature that the PALAEMON PIMM uses is the Docker Compose. Docker Compose is used for running multiple containers as a single service. Each of the containers here runs in isolation but can interact with each other when required. Docker Compose files are very easy to write in a scripting language called YAML⁴, which is an XML-based language that stands for Yet Another Markup Language. Another great thing about Docker Compose is that users can activate all the services (containers) using a single command.

4.4.2 Benefits of Docker Compose

- Single host deployment - This means you can run everything on a single piece of hardware
- Quick and easy configuration - Due to YAML scripts
- High productivity - Docker Compose reduces the time it takes to perform tasks
- Security - All the containers are isolated from each other, reducing the threat landscape⁵

Container deployment is the action of putting containers to use. The deployment of containers uses management software that simplifies the launch and updates of applications. Container deployment provides fast access to environments and speeds up development because secure containers can be quickly downloaded and put to use. Container deployment also minimizes errors because it reduces the number of moving parts in development.

Container deployments can replace many of the tasks previously handled by IT operations. When a tool like Docker deploys multiple containers, it places applications in virtual containers that run on the same operating system. This provides a benefit not offered by virtual ma-

³ Sahiti Kappagantula, (Oct. 29, 2019). Docker Explained – An Introductory Guide to Docker. Published at DZone with permission of Sahiti Kappagantula, DZone MVB.

⁴ [YAML specification](#)

⁵ [Simplilearn, \(Last updated on Feb 15, 2022\). What is Docker Compose: Example, Benefits, and Basic Commands. Lesson 8 of 12](#)



chines. Using a virtual machine requires running an entire guest operating system to deploy a single application.

This is costly and slow if deploying many applications. If you deploy a Docker container, each container has everything needed to run the app and can be easily spun up or down for testing. This is how container deployment saves resources like storage, memory, and processing power and speeds up the CI/CD pipeline.

4.4.3 Benefits of Container Deployment

- Run many individual applications on the same number of servers.
- Deliver ready-to-run applications in containers that hold all the codes, libraries, and dependencies any application needs.
- Deploy easier and more secure deployment and management of applications.
- Modify an application and instantly run it anywhere.

The PALAEMON PIMM is deployed through Gitlab CI/CD. GitLab CI/CD is the part of GitLab that you use for all of the continuous methods (Continuous Integration, Delivery, and Deployment). With GitLab CI/CD, you can test, build, and publish your software with no third-party application or integration needed.

5 Technical Specifications of the PIMM Module

In this section, the technical specifications that the PALAEMON Incident Management Module uses to achieve the implementation will be described.

For the backend, the PIMM is based on the Django Framework, Python, and Kafka.

5.1.1 Django Framework

Django is an open-source framework for backend web applications based on Python — one of the top web development languages. Its main goals are simplicity, flexibility, reliability, and scalability.

Django has its own naming system for all functions and components (e.g., HTTP responses are called “views”). It also has an admin panel, which is deemed easier to work with than in Lavarel or Yii, and other technical features, including:

- Simple syntax
- Its own web server
- MVC (Model-View-Controller) core architecture
- “Batteries included” (comes with all the essentials needed to solve solving common cases)
- An ORM (Object Relational Mapper)
- HTTP libraries
- Middleware support
- A Python unit test framework.



Additionally, Django provides a dynamic CRUD (create, read, update and delete) interface, configured with admin models and generated via introspection. CRUD is used to describe the basic database commands, which means the interface facilitates viewing, changing, and searching for information.

The reason why we use Django for the implementation of the PALAEMON Incident Management Module (PIMM) is that it is fast and simple. One of Django's main goals is to simplify work for developers. To do that, the Django framework uses:

- The principles of rapid development, which means developers can do more than one iteration at a time without starting the whole schedule from scratch
- DRY philosophy — Don't Repeat Yourself — which means developers can reuse existing code and focus on the unique one.

As a result, it takes a lot less time to get the project to market. Also, Django has one of the best out-of-the-box security systems out there, and it helps developers avoid common security issues. In addition, Django can tackle projects of any size and capacity, whether it's a simple website or a high-load web application.

- It's fully loaded with extras and scalable, so you can make applications that handle heavy traffic and large volumes of information
- It is cross-platform, meaning that your project can be based on Mac, Linux, or PC;
- It works with most major databases and allows using a database that is more suitable in a particular project, or even multiple databases at the same time.⁶

Lastly, Django is time- and crowd-tested. It has a big, supportive community accessed through numerous forums, channels, and dedicated websites. It's easy to find help when there's a problematic function in the code and to find developers if your company is looking to base the next project on Django. Django started off with great documentation, the best of any other open-source framework. And it's still maintained on a high level, updated along with the new functions and fixes, so you can easily adapt to changes. You can trust that any issues with the framework will be solved as soon as they arise. The software is constantly updated and new packages are released to make working with Django more convenient than it already is.

5.1.2 Python

The Django framework is based on the Python programming language. Python is a general-purpose and high-level programming language. You can use Python for developing desktop GUI applications, websites, and web applications. Also, Python, as a high-level programming language, allows you to focus on the core functionality of the application by taking care of common programming tasks. The simple syntax rules of the programming language further make it easier for you to keep the code base readable and the application maintainable. The

⁶ Julia Korsun, (Jan 25, 2016). [Why We Use Django Framework & What Is Django Used For.](#)

syntax rules of Python allow you to express concepts without writing additional code. At the same time, Python, unlike other programming languages, emphasizes code readability and allows you to use English keywords instead of punctuations.⁷

5.1.3 Apache Kafka

The PALAEMON Incident Management Module (PIMM) interconnects the PALAEMON toolkits between them. The PIMM is the bridge that transfers data between the components. Therefore, the PALAEMON components can communicate and exchange data. So, we use Apache Kafka.

Apache Kafka is a distributed data store optimized for ingesting and processing streaming data in real-time. Streaming data is data that is continuously generated by thousands of data sources, which typically send the data records in simultaneously. A streaming platform needs to handle this constant influx of data, and process the data sequentially and incrementally.

Kafka is primarily used to build real-time streaming data pipelines and applications that adapt to the data streams. It combines messaging, storage, and stream processing to allow storage and analysis of both historical and real-time data.

It is important for the achievement of the PALAEMON project vision, the data that are entered into PIMM and are transferred between the PALAEMON Components to be very promptly responded to. When an incident occurs the PALAEMON components must communicate and exchange data rapidly with reliability on high performance.

For the frontend, the PIMM is based on the VueJS framework.

5.1.4 VueJS

While JavaScript is its own beast, the ecosystem it exists within is more important than the programming language itself. Frameworks and libraries make the developer's life so much easier by providing a solid platform – culminating in fewer bumps in the development road.

VueJS is one such framework. Vue is an open-source progressive framework that is designed to be incrementally adoptable, as the core library is focused around the view layer only. That being said, it's more than capable of powering sophisticated single-page apps with the help of modern tools and if there are libraries to support them.⁸

Originally, VueJS was developed as a way to take the best parts of Angular, and build a custom tool around it. The data binding and data-driven way of dealing with an HTML-based Document Object Model or DOM were favoured in lieu of working with the DOM directly.

VueJS is really easy to pick up, and it's been this characteristic that has been the main driver behind its widespread and increasing adoption amongst programmers. Also, VueJS has ac-

⁷ Mindfire Solutions, (Oct 3, 2017). [Python: 7 Important Reasons Why You Should Use Python.](#)

⁸ KofiGroup. [7 Reasons why VueJS is so popular.](#)



cumulated a powerful set of tools for unit and end-to-end testing, as well as a plugin installation system.

As we touched on, a DOM is a representation of HTML pages with styles, elements, and page content shown as linked objects. This generates something that looks like an upside-down family tree, with the document branching off into its various parts, linked by lines that demonstrate the relations.

As the user interacts with the HTML-based web page, the browser has to update the information and render it to the user's screen as objects change their state. This process is normally cumbersome and slow, as the whole page has to refresh even if just one object changes. VueJS uses a virtual copy of the original DOM that figures out what elements require updating, without re-rendering the entire DOM – greatly improving app performance and speed. Even better than React JS and Angular.

Lastly, Because of this flexibility, VueJS is able to integrate with almost any existing application. It's JavaScript-based and doesn't really need any other tools to work. Switching from React or Angular to VueJS doesn't really cause any issues, as essentially Vue is a mash-up of the two and can talk to both Model View Controller and Component Based Architecture.

For the Database/Data Storage, the PIMM is based on PostgreSQL and ElasticSearch.

5.1.5 PostgreSQL Database

PostgreSQL is an enterprise-class open-source database management system⁹. It supports both SQL and JSON for relational and non-relational queries for extensibility and SQL compliance. PostgreSQL supports advanced data types and performance optimization features, which are only available in expensive commercial databases, like Oracle and SQL Server. PostgreSQL works on the most popular operating systems – almost all Linux and Unix distributions, Windows, Mac OS X. Its open-source nature makes it easy to upgrade or extend. In PostgreSQL, you can define your own data types, build custom functions, and even write code in another programming language (e.g., Python) without recompiling the database.

Below are the main advantages/benefits of PostgreSQL:

- PostgreSQL can run dynamic websites and web apps.
- PostgreSQL's write-ahead logging makes it a highly fault-tolerant database.
- PostgreSQL source code is freely available under an open-source license. This allows you the freedom to use, modify, and implement it as per your business needs.
- PostgreSQL supports geographic objects so you can use it for location-based services and geographic information systems.
- PostgreSQL supports geographic objects so it can be used as a geospatial data store for location-based services and geographic information systems.

⁹ Richard Peterson, (February 12, 2022). [What is PostgreSQL? Introduction, Advantages & Disadvantages.](#)

- Low maintenance and administration for both embedded and enterprise use of PostgreSQL.

5.1.6 ElasticSearch

Elasticsearch is a distributed, open-source search and analytics engine built on Apache Lucene and developed in Java. It started as a scalable version of the Lucene open-source search framework then added the ability to horizontally scale Lucene indices. Elasticsearch allows you to store, search, and analyze huge volumes of data quickly and in near real-time and give back answers in milliseconds. It's able to achieve fast search responses because instead of searching the text directly, it searches an index. It uses a structure based on documents instead of tables and schemas and comes with extensive REST APIs for storing and searching the data. At its core, you can think of Elasticsearch as a server that can process JSON requests and give you back JSON data.¹⁰

¹⁰ [Ralf Abueg, \(March 7, 2020\). Elasticsearch: What It Is, How It Works, And What It's Used For.](#)

6 Conclusions and Future Plans

6.1 Conclusions

The main aim of the PIMM is to connect all the provided data of the PALAEMON Project from the components and display them to the Master and crew members in order to assist them by observing the current situation and suggesting to them what steps they have to follow depending on the evacuation stage that the ship is. To achieve this the PALAEMON Incident Management Module (PIMM) connects the PALAEMON components, displaying their data to the PIMM screen. Through the PIMM screen, the Master is able to check the overall ship's condition. He can observe the situation of the ship when it is in the normal operation stage. The most important is that whenever an incident occurs, then components are being triggered depending on the Evacuation Coordinator stage. So, the PIMM platform starts displaying all the useful information of the components that Master and crew members need to avoid the bad situation of the incident. The PIMM platform eliminates the confusion that is caused when an incident occurs. The confusion that is caused affects the Master's way of thinking negatively. The implementation of the PIMM will help to decrease maritime accidents and their casualties. There are many maritime accidents every year and the ultimate goal is to reduce the number of them.

6.2 Future Plans

The PIMM develops a central multi-layer agent that supports a multivariate input scheme, where raw data, features, and decisions can be accessed to provide the necessary outputs. It is capable of running in an automated infinite loop manner, where it will continuously monitor everything and record the results. The upcoming plans of the PIMM are the followings:

First of all, PIMM has to be able to adjust to different types of passenger ships. The implementation of the PIMM aims to be adjusted with the ANEK LINES ships as it is a pilot of the PALAEMON Project. It would be useful for PIMM to be able to help other types of passenger ships just as efficiently.

Also, the PIMM is already exchanging data between the components. For example, when data from a component is changed then other components that are connected with it are being updated and changed the outputs that produce. So, finding dependencies between PALAEMON components will help to better correlate them and to contribute to a more complete solution.