AIDERS

Deliverable 4.2 *Initial version of the incident mapping platform*

Coordinator Name: **Christos Panayiotou**

Coordinator Email: **christosp@ucy.ac.cy**

Project Name: **Real-time Artificial Intelligence for DEcision support via RPAS data analyticS**

Acronym: **AIDERS**

Grant Agreement: **873240**

Project Website: http://www.kios.ucy.ac.cy/aiders/

Version: **1.0**

Submission Date: **31/12/2020**

Dissemination Level: **Public**

The project has received funding from the European Union Civil Protection Call for proposals UCPM-2019-PP-AG for prevention and preparedness projects in the field of civil protection and marine pollution under grant agreement – 873240– AIDERS.

Funded by European Union Civil Protection

Contents

Executive Summary

The AIDERS project primarily aims at developing application-specific data analytics algorithms to harness the large volume of measurements that first responders are now able to collect through heterogeneous sensors (including visual, thermal and multispectral cameras, etc.). This deliverable reports on the initial design of the incident mapping platform by describing the core design decisions that are intended to guide the implementation of this platform and, therefore, offer a suitable ground for the further integration of advanced AI algorithms that can be executed in real-time to enable relevant, reliable, timely, and simple information extraction and representation. These specific AI algorithms to be integrated in the next version of the incident mapping platform are extensively detailed and studied in deliverable D4.1.

1. Introduction

This deliverable aims to report on the initial design and implementation of the AIDERS incident mapping platform, which is the software system that is intended to support the decisions and actions of first responders on field by delivering AI-enhanced analytics of the current status reported by drones.

This document, therefore, reports on the design decisions that have been taken by the members of the consortium to guide the development of a modular and extensible software solution. Indeed, while the AIDERS project will provide the proof-of-concept implementation of advanced AI-supported analytics, the consortium aims to offer a customizable solution that can be extended to support incident-specific analytics by incorporating a wide family of AI algorithms. This objective is particularly challenging as this requires us to reconcile the diversity of raw input streams with the diversity of reported analytics.

The remainder of this document covers the identification of key requirements for the incident mapping platform, the core design decisions and some elements of the current implementation.

2. Platform Requirements

The following table delivers a synthesis of the key requirements that have an impact on the implementation of the incident mapping platform:

Req. 1	The incident mapping platform should deal with a wide diversity of input information (discrete values, data streams)
Req. 2	The incident mapping platform should reason on both online and offline information
Req. 3	The incident mapping platform should be deployed on the field with limited Internet connectivity
Req. 4	The incident mapping platform should run on first responders' laptops
Req. 5	The incident mapping platform should cover a wide diversity of mission-critical scenarios
Req. 6	The incident mapping platform should be open to a wide diversity of AI algorithms
Req. 7	The incident mapping platform should leverage the collaboration of first responders
Req. 8	The incident mapping platform should be configured according to mission constraints

3. Platform Design

Our design of the incident mapping platform is guided by the objective of delivering a modular and extensible solution for first-responders. This calls for the selection and adoption of robust, yet innovating technologies, to implement a new generation of analytics platforms for emergency responses. In this section, we go through the list of requirements given in Section 2 and we describe the design choices we made to address each specific requirement.

To address Req. 1, we decided to leverage a NoSQL storage technology to adjust the storage capabilities of raw information streams forwarded by drone on the devices of first-responders. This choice is motivated by the fact that drones are intended to feed the incident mapping platform with a huge volume of raw metrics, being discrete or streams, and this calls for a solution that can store massive volumes of data. Furthermore, as the drones can be equipped with different sensing capabilities, one should propose a flexible solution that can accommodate and leverage the availability of specific data streams. Therefore, we decided to consider the adoption of graphs as a suitable formalism to structure raw input streams within domains of sensor information (cf. Figure 1). The graph offers a scalable structure to store raw data as nodes, and establish relationships among the information as edges between the related nodes. Following this approach, we consider that a drone and its onboard sensors can be modeled as a set of nodes in the graph, which are intended to be further processed by AI algorithms to produce the expected analytics. The design choice of storing the raw data is also motivated by the capability to analyze the mission *a posteriori*, thus requiring the availability of all the information. In Figure 1, we provide a sample graph that can be used to store mission-related information covering the area of the field as well as telemetry metrics reported by a group of drones.

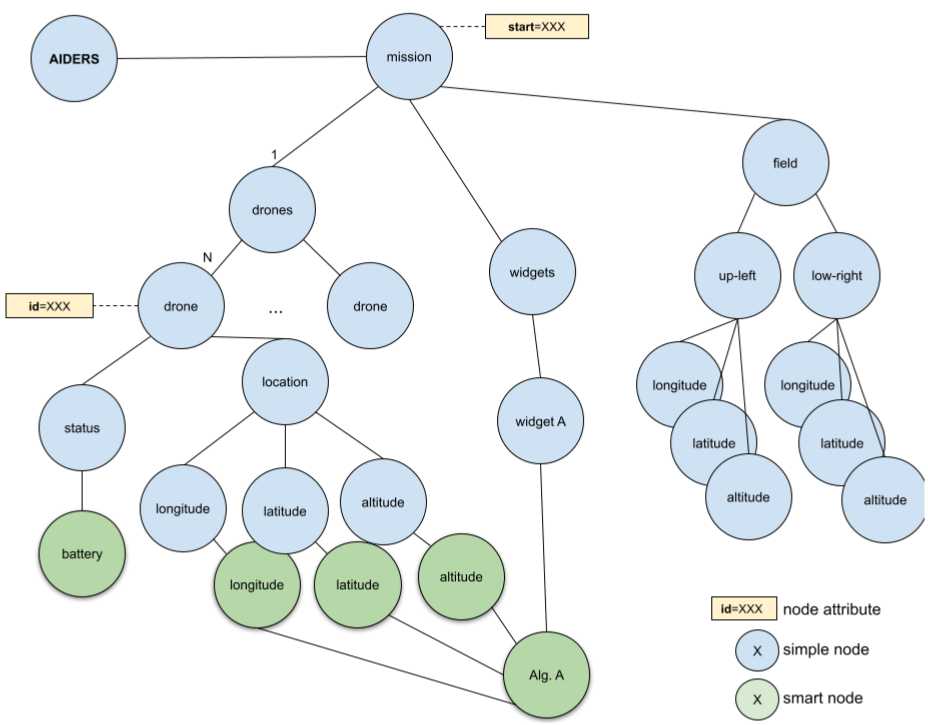

Figure 1. Graph representation of the mission.

To address Req. 2, we decided to consider temporal graphs to support the indexation of multiple time series. While standard time series databases (TSDB, like InfluxDB) do not support the storage of structured data, the adoption of temporal graphs offers a more promising alternative that covers both requirements. In particular, we adopted the GreyCat database (https://greycat.ai) as an appropriate technology to support the storage of raw input data streams as temporal graphs that can be structured upon the diversity of input streams and related sensors onboarded by drones. In Figure 1, the graph is associated with a time index so that both the nodes and the edges of the graph can be changed along with time. In the context of the incident mapping platform, this feature is thought to be particularly useful as it enables us to reason on the latest information made available by the drones. Given that some information can be added *a posteriori,* once the drones come back on the command center, the graph can be updated with the recorded media and the analytics can be updated accordingly. In addition to the time index, GreyCat also implements a geospatial index that can be used to process geotagged information, has it is the case in the case of drones. We, therefore, strongly believe that the GreyCat is well-fitting this specific requirement.

To address Req. 3, we decided to consider the use of offline maps that can be used to render analytics with no dependency requiring continuous Internet connectivity. In particular, while such maps can be downloaded before moving on-site, the core idea of this design choice is to support the enrichment of maps by incorporating information coming from drones. Concretely, we build on the Mapbox toolkit (https://www.mapbox.com), which is a widely acknowledged mapping library with a rich ecosystem. Mapbox offers both a customizable base map and a system of graphical overlays that can be leveraged by the incident mapping platform to render different analytics. As part of the invident mapping platform, we intend to include a custom base map provider that delivers an up-to-date ground information by composing pictures captured by drones. Atop this base map, the AIDERS analytics will be offered as an overlay that can be enabled to render added values to the base map. For example, heatmap overlays can be used to quickly visualise the field areas that have been covered by a fleet of drones during the last 15 minutes, thus enabling the first responders to react if a specific area remains uncovered. Alternatively. Ouragan maps can be used to predict the evolution of an ongoing disaster by taking into account the uncertainty associated to the event under monitoring.

To address Req. 4, we decided to build on the ElectronJS development platform (https://www.electronjs.org), which offers a cross-platform environment for developing desktop applications based on the HTML, CSS, JS standard of the W3C consortium. Combined with the React (https://reactjs.org) framework, this choice also facilitates the integration of the Mapbox library in a desktop application that can be easily and quickly started by first responders. By relying on web standards, this design choice also offers some flexibility to consider a more distributed deployment of the incident mapping platform to be used on multiple devices. By adopting the JavaScript environment, we implement a bridge between the Mapbox toolkit and the GreyCat database, which are the two key software components of our software solution. To propose an efficient solution, the ElectronJS is mostly intended to manage a set of modules that are intended to consume data from the drones and interact with the database and the user interface accordingly.

To address Req. 5, we decided to design a new plugin mechanism, based on the concept of *widget*. As illustrated below, a *widget* is a software component that can be loaded and activated in the incident mapping platform according to mission-critical scenarios. A *widget* is in charge of *i)* loading specific raw data streams from drones into the temporal graph managed by the platform, *ii)* injecting the artificial intelligence algorithm into the application to process the raw data streams to extract actionable insights for the first responders, and finally *iii)* providing appropriate visualizations through the Mapbox toolkit. Widgets are leveraging the ElectronJS and React JavaScript frameworks, so they are expected to perform a limited set of computations for performance reasons.

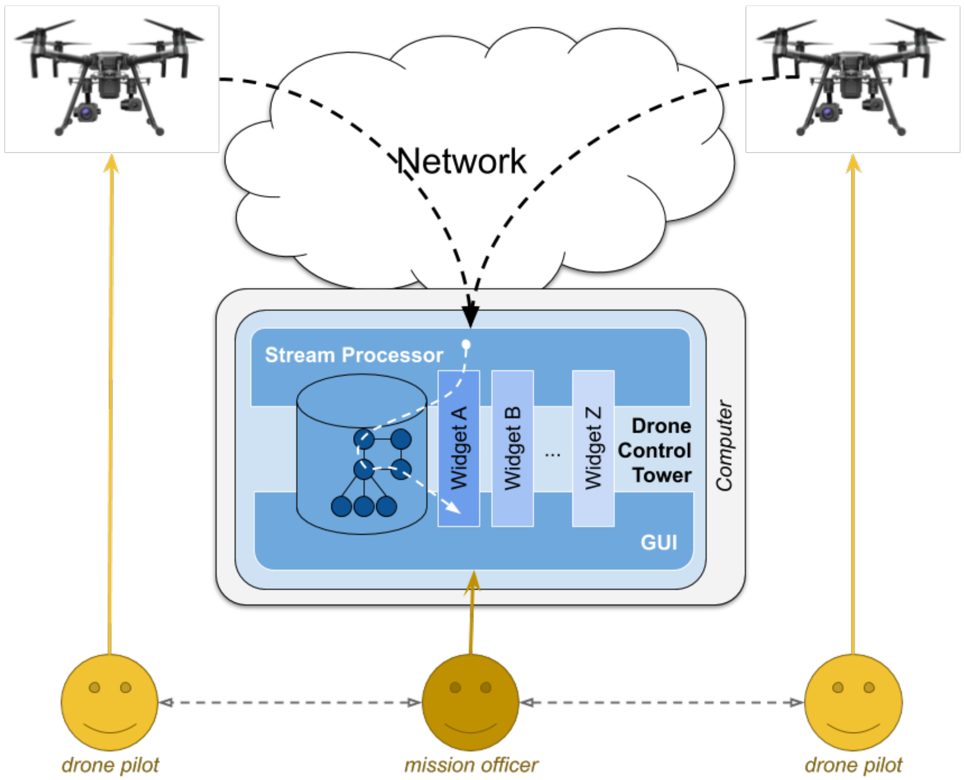

Figure 2. Architecture overview of the incident mapping platform.

To address Req. 6, we intend to build on the GreyCat database technology to incorporate state-of-the-art and novel AI algorithms within the temporal graph. By doing so, the processing of raw data streams is delegated to the database, which can adopt suitable strategies to produce the required analytics. In particular, GreyCat implements the principle of *tensors* and the base construct to implement advanced analytics. Depending on the class of information (telemetry, pictures, videos), we plan to apply domain-specific algorithms to infer insightful knowledge on the current mission. For example, the history of locations of a fleet of drones—encoded as specific nodes in the graph of Figure 1—can be processed by unsupervised machine learning techniques, like clustering, to build a heat map of the covered areas for the past minutes. The same nodes encoding the drones' locations can also be processed by supervised machine learning techniques, like polynomial regressions, to predict the trajectories of the fleet and to depict a hurricane map highlighting the potential risks of collisions. High-definition pictures will be stored as oriented nodes and aggregated to infer the tiles of the base map. Finally, video streams will be stored in the graph as BLOB (*Binary Large OBject*) nodes, which will be connected to companion nodes that will store the knowledge extracted from the streams. For example, companion nodes can encode the smoke areas detected by the smoke detection algorithm. Interestingly, these companion nodes can leverage the spatiotemporal indexes of GreyCat to render the inferred knowledge atop of the base map in a given widget.

To address Req. 7, the *widgets* can be composed according to the desires and needs of first responders in order to offer both bird view on the current situation and more focused views to cope with situations that require to be more carefully analyzed. As mentioned above, the incident mapping platform is also thought to be deployed in a standalone mode (using a single laptop device) or a distributed setting (using a server connected to lightweight client devices, including mobile devices). We believe that grouping widgets in the same window can enforce collaborations and drive more informed decisions from the first responders to quickly react to the evolution of the incident.

Finally, to address Req. 8, the *widgets* are associated with a specific mission. They can be configured *a priori* or during the mission, but their configuration will be stored together with the raw input data streams, as well as all inferred actionable analytics, to support *post mortem* mission replay and briefings of first responders. To achieve this capability, the configurations of *widgets* will be stored as nodes of the graph whose stored values can be used by the AI algorithms processing the mission data streams.

4. Platform Implementation

This section provides some screenshots of the initial version of the incident mapping platform that is based on the above reported design decisions. This initial version already includes the support for customizable widgets that can take the form of maps or video streams (cf. Figure 3).

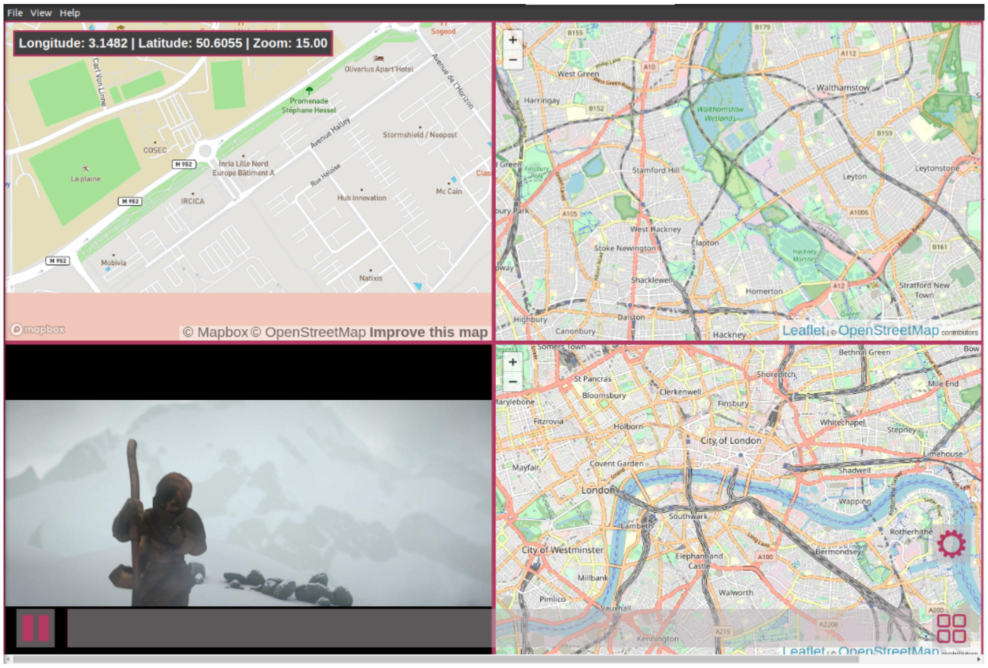

Figure 3. Bird view of the incident mapping platform.

The same instance of a *widget* can be rendered in real-time, no matter the incident mapping platform is configured in the bird view mode (as above) or the focused view (as below). The focused view provides a full screen rendering of any selected widget to let the first responders analyse the analytics reported by the platform and take appropriate decisions accordingly. Furthermore, the user interface also includes a timeline that can be used by the first responders to go back in time and replay a past situation to take more informed decisions by reasoning over previous events.

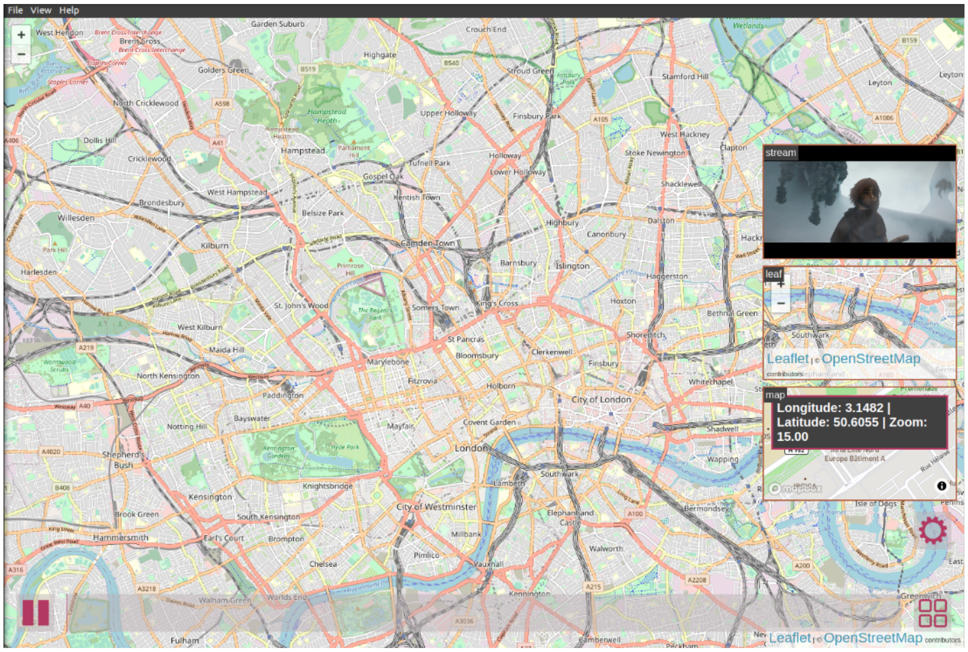

Figure 4. Focused view of the incident mapping platform.

Finally, Figure 5 describes the configuration panel that is automatically adjusted with the list of widget-specific settings.

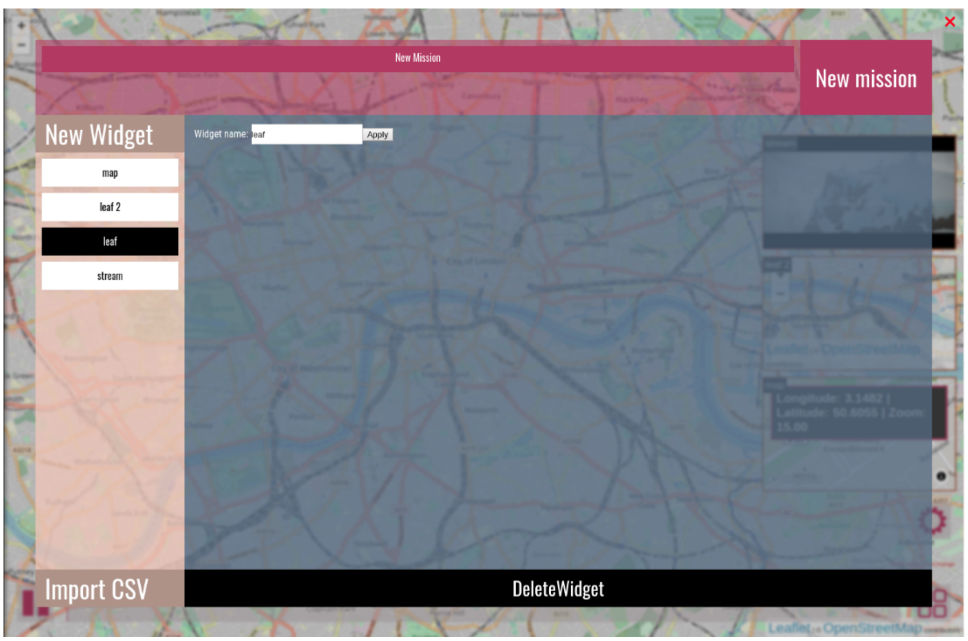

Figure 5. Configuration view of the incident mapping platform.

5. Perspectives

Since the ground has been set for the development of the incident mapping platform, our efforts are now focused on the development of new widgets that build on the AI algorithms identified in Deliverable D4.1 and integrate them into the GreyCat database as in-memory algorithms that are automatically triggered upon the forwarding of relevant sensor data streams. The resulting analytics are then consumed by the relevant widgets to be graphically rendered by the incident mapping platform UI. As the rendering of the graphical analytics night be complex to define, we intend to support the integration of renderings built from the Mapbox toolkit, as illustrated in Figure 6.

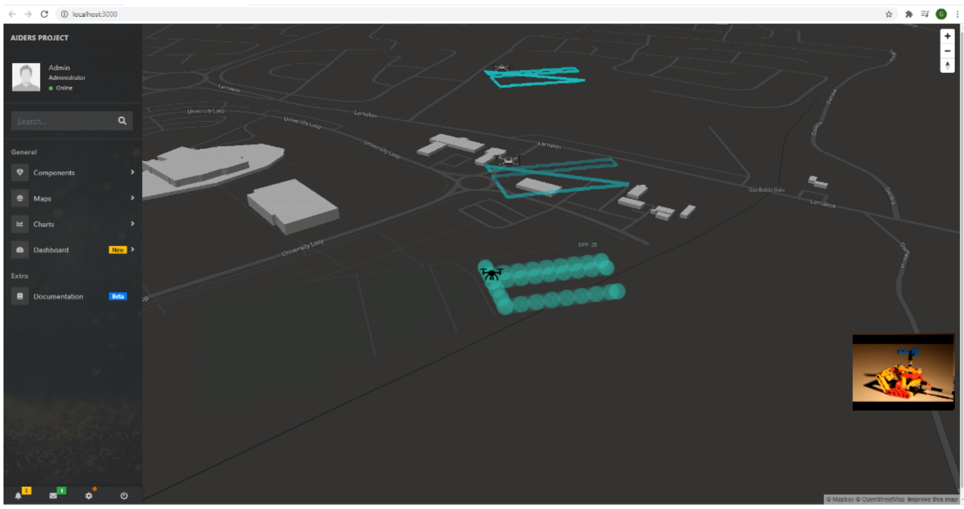

Figure 6. Screenshot of the Mapbox toolkit used to design graphical representation AI analytics.

6. Conclusion

This deliverable provides a summary of the activity conducted as part of WP4 to design and implement the incident mapping platform for the AIDERS project. This deliverable, therefore, focuses on the core design decisions that have been taken by the members of the consortium in order to offer a configurable platform that can be tuned according to mission-specific scenarios.

In particular, the concept of widget will allow the members of the consortium and interested ecosystems to integrate novel AI algorithms to support first responders and deal with novel type of input sources (drones, responders equipment, etc.).