# beAWARE

Enhancing decision support and management services in extreme weather climate events
700475

# D7.7

# User applications

| | |
|---|---|
| **Dissemination level:** | Public |
| **Contractual date of delivery:** | Month 33, 30 September 2019 |
| **Actual date of delivery:** | Month 33, 25 September 2019 |
| **Workpackage:** | WP7: System development, integration and evaluation |
| **Task:** | T7.5 - End-users applications |
| **Type:** | Report |
| **Approval Status:** | Final |
| **Version:** | V0.6 |
| **Number of pages:** | 36 |
| **Filename:** | D7.7_beAWARE_User_applications_v0.6.docx |

**Abstract**

This deliverable presents the results of the developed end user applications. They are used by citizens as well as first responders before, and during an emergency event. The applications are the main interaction channel of citizens and first responders with the authorities. This allows citizens to send multimodal reports about the in-field situation and to receive alerts from the authorities, whereas first responders can receive and report on assigned tasks. This deliverable firstly summarises the user requirements that have been gathered to design and develop the applications. It presents the results and developed functionalities. The integration into the beAWARE platform is described. Finally, the evaluation of operating of the applications during the pilots is presented.

The information in this document reflects only the author's views and the European Community is not liable for any use that may be made of the information contained therein. The information in this document is provided as is and no guarantee or warranty is given that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability.

Co-funded by the European Union

# History

| Version | Date | Reason | Revised by |
|---|---|---|---|
| 0.1 | 09/07/2019 | Deliverable outline | Philipp Hertweck (IOSB) |
| 0.2 | 26/07/2019 | Deliverable content added | Philipp Hertweck (IOSB) |
| 0.3 | 13/08/2019 | Revised content. Executive Summary. | Tobias Hellmund (IOSB) |
| 0.4 | 09/09/2019 | 1st internal review | Ilias Koulalis(CERTH) |
| 0.5 | 17/09/2019 | 2nd internal review<br>3rd internal review | F. Lombardo (AAWA)<br>I. Koren (MSIL) |
| 0.6 | 24/09/2019 | Incorporated reviews | Tobias Hellmund (IOSB) |

# Author List

| Organisation | Name | Contact Information |
|---|---|---|
| IOSB | Philipp Hertweck | philipp.hertweck@iosb.fraunhofer.de |
| IOSB | Tobias Hellmund | Tobias.hellmund@iosb.fraunhofer.de |

# Reviewer list

| Organisation | Name | Contact Information |
|---|---|---|
| AAWA | Francesca Lombardo | francesca.lombardo@distrettoalpiorientali.it |
| MSIL | Itay Koren | itay.koren@motorolasolutions.com |

# Executive Summary

This document constitutes the **Deliverable 7.7 "User applications"**. It encompasses the tasks executed for the beAWARE mobile applications. These mobile applications, one version for citizens and one version for first responders (FR), are the main point of interaction with the beAWARE platform for each user group.

The application establishes a bilateral communication channel in between app users and the commanding authorities. In one direction, the software is used to pass multi-modal data, such as pictures, videos, text or voice reports from the public to the beAWARE platform. In the first responder version, users can send data, such as about their task status, to the authorities. First responders have a more specialised version to carry out more strategic tasks and can send more complex data. FRs will have received a special training with the app, enabling them to send specific information based on the type of incident encountered. In the return direction, authorities decide, whether they want to contact civilians or first responders; information sent to the former group could be warnings not to enter a specific location or suggestions for precautions. The latter receive information that manage their tasks, such as what and where mitigative or preventive actions need to be executed. Also, it can be referred to a previously sent report about which more information is necessary. The data sent through the mobile app is structured through a dedicated ontology. This ontology operates separately from the beAWARE Ontology (see D4.2 – Semantic Representation and Reasoning) and stores information about reports, their content and their metadata. Metadata contain information as the time and date or the location, from when and where the report was issued.

An issued report from the mobile app is sent to the message bus of the beAWARE platform. Depending on the content or attachment of a report (picture, speech, video, …), the corresponding analysis tool analyses the data and feeds its results back to the platform. Reports can be accessed from the Public Safety Answering Point (PSAP), or from the Knowledge Base.

The application development is based on the user requirements and uses cases, summarized in D2.1 – Use cases and Initial User Requirements, due in May 2017. The revised version of the user requirements was documented in D2.10 – Final Uses cases and requirements in December 2018 and contains adjustments and gained experience of the users and developers. The necessary capabilities of the app were categorized into the following modules:

- **Multimodal incident reporting**: Citizens and first responders can send reports to the authorities. Multimodal input is sent to the message bus from where consuming components, such as the command and control centre, can receive the data.
- **Alerting:** Authorities can send alerts from the PSAP to mobile devices with warnings or recommended actions; the alert can be assigned to a location for which it contains relevant information.

- **Team and task management:** The app should have a distinct view for first responders and authorities, which offers enhanced functionalities compared to the version for civilians. The distinct view offers a more sophisticated possibility to submit data. Further, it contains methods for the authorities for the task management of first responders. This view is locked for unauthenticated users.

For all capabilities different views in the app were realized.

One goal for the developed app was to be *platform independent*, to facilitate the installation on devices of different manufactures. This was achieved by following a web-based approach: the application was developed as web app in a first step, consisting of HTML, JavaScript and CSS. In the second step, the mobile application was deployed using the Cordova mobile framework, giving the ability to target multiple platforms with one code base.

The framework delivers software, that can access the different sensors of the device it operates on, such as camera, microphone or GPS-positioning sensor, offering a wide range of sensors and information. The resulting application should operate on different platforms; yet, through software and development restrictions, the application can only be run on the Android platform.

The application has been evaluated during two pilots. In the first pilot simulated in Thessaloniki from the November 19th through November 22nd of 2018. The second pilot demonstrated handling a flood situation in Vicenza from March 4th through March 8th of 2019. During the pilots, minor technical issues were identified that could be fixed. The collected feedback complimented empowering civilians to send information to the authorities and further the establishment of a direct communication channel to them. Feedback from First Responders verified the usefulness of the app, yet they would not completely substitute legacy technology through the new development. D2.4 and D2.6 refer to feedback and issues collected during the two pilots.

Further development in preparation of the last pilot in Valencia (November 2019) is going on. In the pilot, the beAWARE platform will be evaluated at hand of a fire event. For this scenario, functionalities for authorities to contact distinct groups of first responders, such as firefighters, police or technical first responders will be implemented. Furthermore, the collected feedback from the first two pilots, as well as the review are converted into concrete implementation steps.

# Abbreviations and Acronyms

The following abbreviations have been used in this document:

| | |
|---|---|
| **API** | Application Programming Interface |
| **APK** | Android Package |
| **CDR** | Central Data Repository |
| **CSS** | Cascading Style Sheets |
| **FR** | First Responder |
| **GPS** | Global Positioning System |
| **HTML** | Hypertext Markup Language |
| **HTML5** | fifth version of Hypertext Markup Language |
| **JS** | JavaScript |
| **PSAP** | Public-safety answering point |
| **UI** | User Interface |
| **VHF** | Very High Frequency |
| **WP** | Work Package |

# Table of Contents

TABLE OF CONTENTS ........ 7

LIST OF FIGURES ........ 7

1 INTRODUCTION AND RATIONALE ........ 9

1.1 What to expect from this document? ........ 9

1.2 Relation to other Work Packages ........ 9

1.3 Relation to other WP7 Tasks ........ 9

1.4 Document Structure ........ 9

2 REQUIREMENTS ........ 11

2.1 User Requirements ........ 11

2.2 Chapter Summary ........ 15

3 MOBILE APPLICATIONS ........ 17

3.1 The beAWARE mobile applications ........ 17

3.2 Structure of the user interface ........ 18

3.3 Multimodal incident reporting ........ 20

3.4 Alerting ........ 21

3.5 Team- and task management ........ 23

4 IMPLEMENTATION AND INTEGRATION ........ 25

4.1 State of the Art technology ........ 25

4.2 Architecture of the end-user application ........ 26

4.3 Integration into the beAWARE platform ........ 27

5 EVALUATION ........ 33

5.1 Results of the 1st pilot ........ 33

5.2 Results of the 2nd pilot ........ 34

5.3 Chapter Summary ........ 35

6 CONCLUSIONS AND NEXT STEPS ........ 36

6.1 Conclusions ........ 36

6.2 Next Steps ........ 36

# List of Figures

Figure 1 Signing in as first responder to enable advanced features. ........ 17

Figure 2 Main screen map-based view ........ 18

Figure 3 List view of the app ........ 18

Figure 4 Navigation options in the app ........ 19

Figure 5 Main screen for creating an incident report ........ 20

Figure 6 Categories for incident reporting ........ 20
Figure 7 Categorization of a flood report ........ 21
Figure 8 The PSAP UI used to issue Alerts ........ 22
Figure 9 Detailed description of the alert ........ 22
Figure 10 Alert with affected area ........ 22
Figure 11 Notification on the mobile device ........ 23
Figure 12 Notification on the PC ........ 23
Figure 13 Updating the team status ........ 23
Figure 14 Overview of the task an its location ........ 24
Figure 15 Updating the status of the task ........ 24
Figure 16 The mobile app ontology ........ 28
Figure 17 High-level view of the beAWARE architecture ........ 30
Figure 18 The informational flow of an issued alert ........ 31

## List of Tables

Table 1 User Requirements ........ 11
Table 2 Topics on message bus, used by enduser applications ........ 29

# 1 Introduction and Rationale

## 1.1 What to expect from this document?

This deliverable presents the end user applications, which are used by citizens and first responders to interact with the beAWARE platform. Therefore, this document summarises the development that was done in Task 7.5 – End-users applications.

## 1.2 Relation to other Work Packages

The requirements of the user applications, developed in this work package, have been collected in collaboration with the end-users in work package (WP) 2. Therefore, the results of WP2, especially D2.1 and D2.10 form the basis of the development of this task. In addition, the evaluation of the end-user applications was done on the one hand by continuously involving them into the current development and, on the other hand, by the extensive usage of the applications during the execution of the pilots. Therefore, the detailed evaluation can be found in the deliverables of WP2, foremost D2.4 and D2.6.

## 1.3 Relation to other WP7 Tasks

WP7 handles the system development, integration and evaluation. It contains the development of an overall system architecture, integration and testing of the beAWARE platform. The mobile application is one of the components of the platform. Therefore, this component needs to be seamlessly integrated into the system architecture (defined in T7.1). It is deployed into and makes use of the infrastructure, provided by Task 7.4. The mobile application is also a crucial part of the overall technical testing of the beAWARE platform (T7.3), since it is the main user interface (UI) for citizens and first responders to interact with the other components.

## 1.4 Document Structure

This document is structured as follows:

- **Chapter** 2 summarizes the user requirements concerning the end-user applications, which form the basis for development.
- **Chapter** 3 presents in detail the realized applications. The design and realization is explained by describing the functionality in depth.
- **Chapter** 4 presents the integration into the beAWARE platform. It is shown how data exchange is realized between the different components and how the interact with the end-user applications.
- **Chapter** 5 summarizes the results from evaluating the applications during their usage in the Heatwave and Flood pilot.

- Finally, **Chapter** 6 concludes with closings remarks and directions for further improvements.

Every chapter starts with an introduction and ends with a recapitulation of its content.

# 2 Requirements

In this section, we want to recap the user requirements. Those have been collected initially in “D2.1 Use Cases and Initial User Requirements” in May 2017. The updated version “D2.10 Final use cases and requirements” was released in December 2018.

## 2.1 User Requirements

The basis for developing the user applications are the requirements collected, together with the end-users. The results of this process are available in the previously mentioned deliverables D2.1 and D2.10. For a more detailed explanation and the mapping of the requirement number to the use case, we refer to those documents.

In the following table, we are picking up the user requirements defined in “D2.10 – beAWARE final use cases and requirements”. We are listing all the requirements that involve the interaction with the mobile application and give a short explanation, how the end-user applications are fulfilling the corresponding user requirement.

Table 1 User Requirements

| UR | Requirement Name | Final description provided (D2.10) | Implementation in the application |
|---|---|---|---|
| UR_104<br>UR_327 | Send/receive emergency reports | Allow citizens to send text, images, audio and video messages from their mobile phone (for the different operative systems) […]. | Multimodal incident reporting |
| UR_105<br>UR_328 | Send task reports | Allow First Responders to send reports about their assignments from their mobile phone to local authorities. | Team- and task management |
| UR_107<br>UR_330 | Localize video, audio and images | Provide authorities with the ability to localize videos, audio and images sent by citizens from their mobile phones. | Multimodal incident reporting |
| UR_108<br>UR_331 | Localize task status | Provide authorities with the ability to localize first responders reports regarding the status of their assigned tasks. | Team- and task management |

| UR | Requirement Name | Final description provided (D2.10) | Implementation in the application |
|---|---|---|---|
| UR_110<br>UR_333 | Localize calls | Provide authorities with the ability to localize Phone Calls (mobile application) to an emergency number concerning a flood event. | Multimodal incident reporting |
| UR_116 | Warning people approaching flood areas | Provide authorities with the ability to warn people in danger with warning messages, once they are approaching a flooded area. | Alerting |
| UR_117<br>UR_334 | Manage assignments in case of new emergencies | Provide authorities with the ability to manage first responder assignments. | Team- and task management |
| UR_119 | Manage assignments based on river level overtopping | Provide authorities the ability to assign task to first responder teams related to the overtopping of predefined river level thresholds. | Team- and task management |
| UR_120 | Map of rescue teams and task evaluation | Display to authorities the location in time of first responder teams in all the municipality and provide the ability to evaluate in real time the execution of the assigned tasks with a global visualization of the activities performed. | Team- and task management |
| UR_125<br>UR_212<br>UR_336 | Traffic warnings | Provide authorities with the ability to send warnings to citizens in order to avoid interferences inside the area involved by civil protection activities. | Alerting |

| UR | Requirement Name | Final description provided (D2.10) | Implementation in the application |
|---|---|---|---|
| UR_133 | Send water level estimation from mobile app | Provide the Citizen and first responders with the ability to estimate roughly the river water level by choosing a pre-defined water level category from a specific list in the mobile app. | Multimodal incident reporting |
| UR_134 | Send specific type of incident reports | Provide to the Citizen and the first responders the ability to use their mobile applications so as to specify the type of incident report from a pre-defined list of incidents. | Multimodal incident reporting |
| UR_135<br>UR_227 | Specific mobile app for first responder and citizen | Provide different versions of the mobile app for citizen and first responders based on their different roles. | Login functionality for first responders that offers different views inside the app |
| UR_210 | Mobile application | Provide citizens to communicate a fire alert, detected neglects or other risk situations and even send visual data through a mobile application. | Multimodal incident reporting |
| UR_211<br>UR_337 | Location of personnel involved | Display authorities/first responders to visualize GPS location and/or real time footage of personnel on the incident site […]. | Team- and task management |
| UR_213 | Recommendations | Sending recommendations to citizens. | Alerting |
| UR_214<br>UR_338 | Warnings | Sending warnings of pre-emergency alerts to citizens by authorities. | Alerting |
| UR_215<br>UR_339 | Evacuation orders | Ordering evacuations of citizens at risk. | Alerting |

| UR | Requirement Name | Final description provided (D2.10) | Implementation in the application |
|---|---|---|---|
| UR_216<br>UR_340 | Internal sharing of information | Sharing data (images, videos, geolocation, reports) regarding the forest fire among authorities & first responders. | Multimodal incident reporting<br>Team- and task management |
| UR_219<br>UR_342 | Coordination and communication between different resources | Provide communication between authorities and first responders, in order to improve their coordination. | Team- and task management |
| UR_224 | Automatic translation from a foreigner applicant through mobile app | Make easy the communication between PSAP operator and people with different languages. | The language of the application can be changed. The application can receive messages in multiple languages. |
| UR_312 | Warning citizens | Provide to citizens warnings through the beAWARE app, of an imminent heatwave and a list of proactive measures and how to reduce its effects. | Alerting |
| UR_313 | First responder status | Provide to the authorities the current status and location of all first responders when they are performing their tasks. | Team- and task management |
| UR_314 | Assign tasks to first responders | Allow authorities to assign additional tasks to those first responders who are available or even instruct those who are able to assist other responders. | Team- and task management |
| UR_325 | Suggested places for relief | Provide citizens with information regarding the suggested places for relief through an app. | Alerting |

## 2.2 Chapter Summary

To summarize the user requirements, they can be roughly grouped into three categories:

- **Multimodal incident reporting**: Citizens and first responders should be able to send reports to the authorities. It should be possible to add a textual description as well as multimedia files to the report. In addition, it should be configurable what type of event is being reported. Metadata, such as coordinates or time need to be added to a report, to make it localizable.
- **Alerting:** Authorities are able to send alerts to the mobile devices. The recipients of the alert can be restricted, so it is possible to send information only to first responders. Furthermore, the recipients can be selected by their location. The alerting mechanism can be used to distribute various kind of information, e.g. recommendations for actions, warnings or evacuation orders.
- **Team and task management:** The app should have a distinct view for first responders and authorities to coordinate their activities. This view is unavailable for civilians and only available for registered users. After logging in, authorized first responders can enable advanced functionalities like team- and task management features. Information about available forces and task status of first responders are collected by the application provide an overview for authorities. . It should be possible to assign tasks to first responders and they should be able to report back the current status.

# 3 Mobile Applications

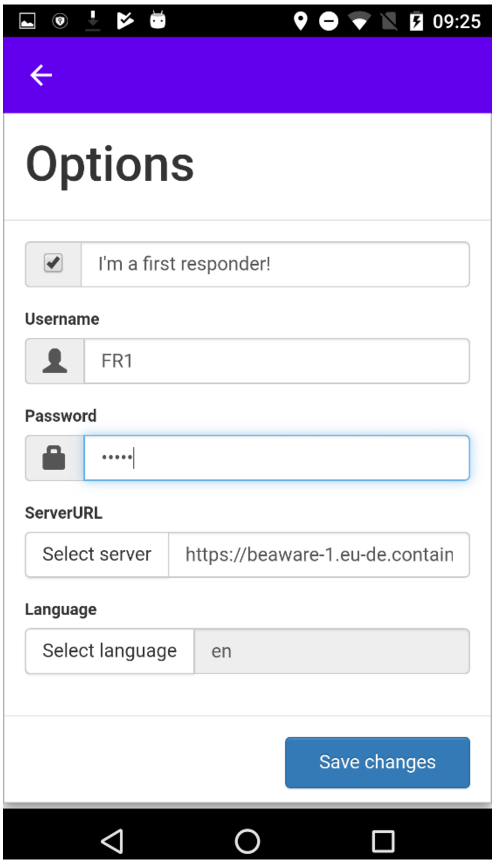
The development of the mobile applications are based upon the collected end-user requirements (see last chapter). This chapter presents the capabilities of the apps in detail and how they map to the user's requirements.

## 3.1 The beAWARE mobile applications

As already mentioned, the users of the applications can be categorized into two groups: citizens and first responders. To coordinate the crisis response and to prevent damage, citizens should be notified in case of an event. Notifications could contain warnings to take precautions or, by geofencing an endangered area, warnings to not enter a specific location can be issued. Furthermore, civilians should be able to send multimodal reports to the authorities through the application.

This basic functionality is the same for citizens and first responders. Therefore, it was decided to have a common application for both user groups. To satisfy the needs of advanced functionalities for first responders, a login form inside the application is provided to unlock these advanced features (see **Error! Reference source not found.**). The benefits are obvious: the distribution of the application is simplified, since there is only one version valid for all users. User can switch their role from citizen by logging in and become a first responder, if previously set by authorities. This way, the basic functionality stays available in both versions.

During the development and testing progress of the beAWARE platform, new, non-functional requirements arose, which have not been considered yet. In a weekly telephone conference, newly developed functionalities were presented and the integration with the other components was continuously validated. To visualize and embed all the partners, we used the screen-sharing functionality of the conference system to allow everybody to follow. Since sharing the display of a mobile device is not trivial, we decided to follow a so-called cross-platform approach: the developed application is not only runnable on mobile devices, but also in a standard web browser, which is available on each desktop PC. The mobile application was deployed, together with the other beAWARE components, to the IBM Kubernetes cluster (see e.g. D7.3 for more information). By following this approach, the latest version of the mobile application was easily and directly available for every partner. Therefore, we



Options

☑ I'm a first responder!

Username
FR1

Password
•••••

ServerURL
Select server https://beaware-1.eu-de.contain

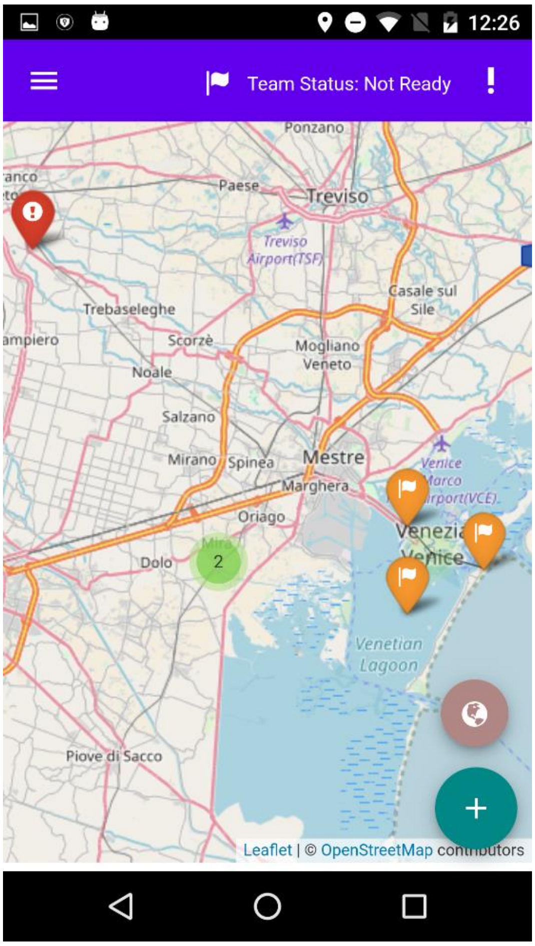
Language
Select language en

Figure 1 Signing in as first responder to enable advanced features.

could seamlessly integrate the app in our testing and validation sessions.

The following part of this chapter will present in detail the functionality of the end-user application.

## 3.2 Structure of the user interface

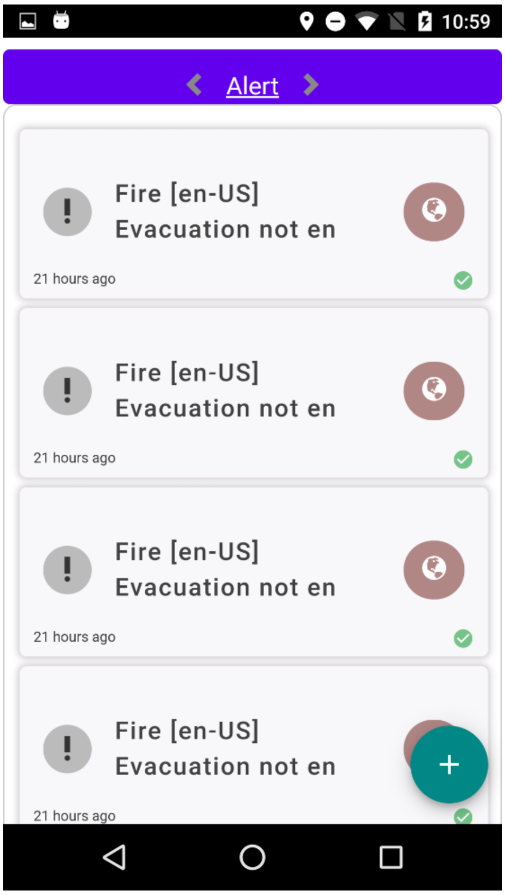


12:26

Team Status: Not Ready

Figure 2 Main screen map-based view

The starting point of the app is a map-based view, which shows all relevant information and provides a quick access to the main functionalities. Figure 2 shows an example situation, with the main elements: a public alert (red spot), multiple tasks (orange spots) and a cluster of two reports (green spot). The top bar indicates the status: in this example, the team is currently not ready to accept a new task. Clicking the “world”-bottom-right button will zoom to the own location, showing also the uncertainty of the current position, which depends on the GPS signal strength of the device. The “plus”-button, opens the form to submit an incident report (see section 3.3 ).



10:59

Alert

Fire [en-US]
Evacuation not en
21 hours ago

Fire [en-US]
Evacuation not en
21 hours ago

Fire [en-US]
Evacuation not en
21 hours ago

Fire [en-US]
Evacuation not en
21 hours ago

Figure 3 List view of the app

In addition to the map, there is a table, listing all elements (see Figure 3) with the most important information. This contains, e.g. the title and the time, when the element was sent. Taping on one item will show all available details.

There is a tight integration to the map: clicking on 🌐 will jump to the map, showing the affected area of the public alert or in case of a task, the location where the task should be executed.

Figure 4 shows the operable buttons within the app. Clicking on the buttons activates the corresponding views.

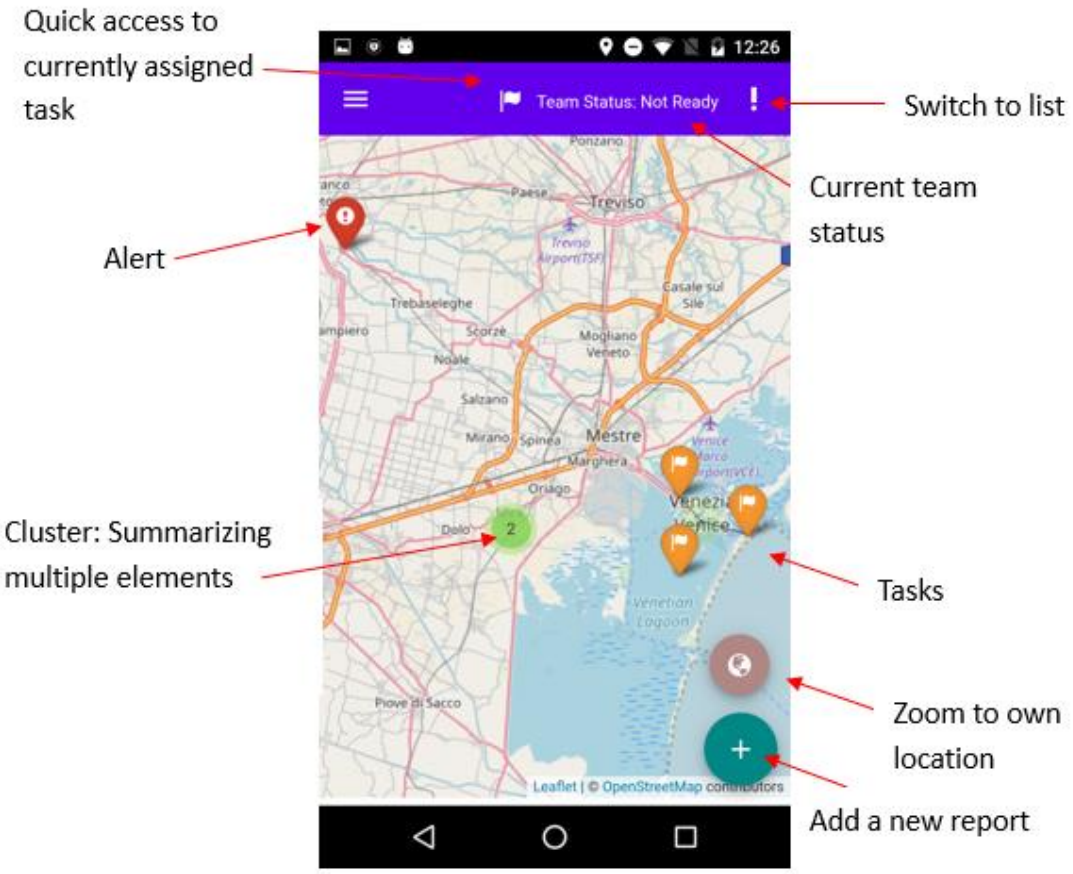
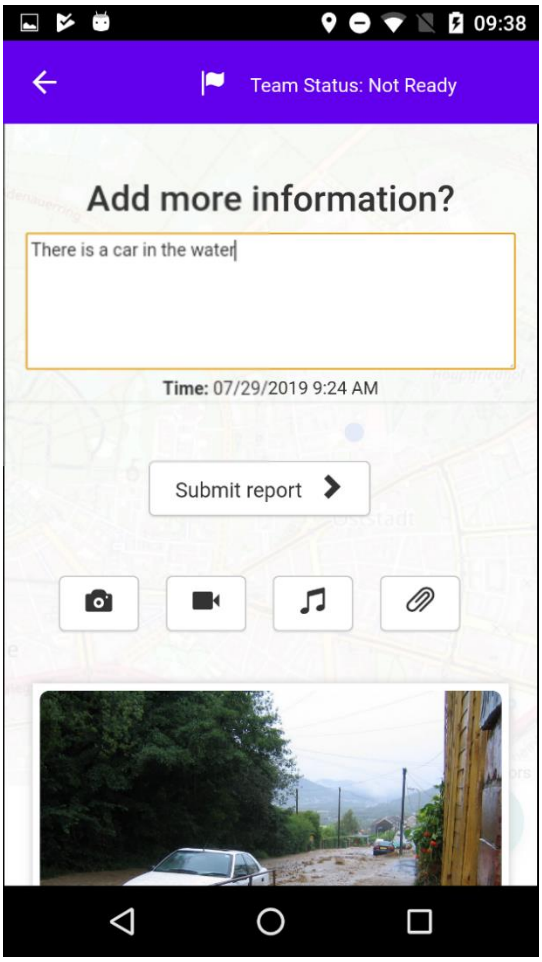


Figure 4 Navigation options in the app

## 3.3 Multimodal incident reporting

The end-user applications are used by citizens as well as first responders to provide up-to-date in-field information to the authorities. The data provided by the users of the mobile app are fed into the beAWARE platform, where the analytic modules process the input and the results are integrated into the overall picture of the situation. See section **Error! Reference source not found.**and the other WP7 deliverables for more information.

Figure 5 shows the main screen for sending a message (“incident report“) from citizens or first responders to the authorities. An incident report can consist of multiple elements: time, position, categorization, textual information and multimedia files, while only time and position attributes are mandatory. To simplify the completion of the report, these are filled in automatically: the time is taken by the current system time, while the position of the report is the user’s position determined by GPS. Sometimes, the position of the event is not the position of the user (e.g. the user moved to a save location to send the report, the incident has been observed from a distant position or simply for testing purposes). In this case, the position for the report be specified by double tapping on the desired location on the map.



There is a car in the water

Time: 07/29/2019 9:24 AM

Figure 5 Main screen for creating an incident report

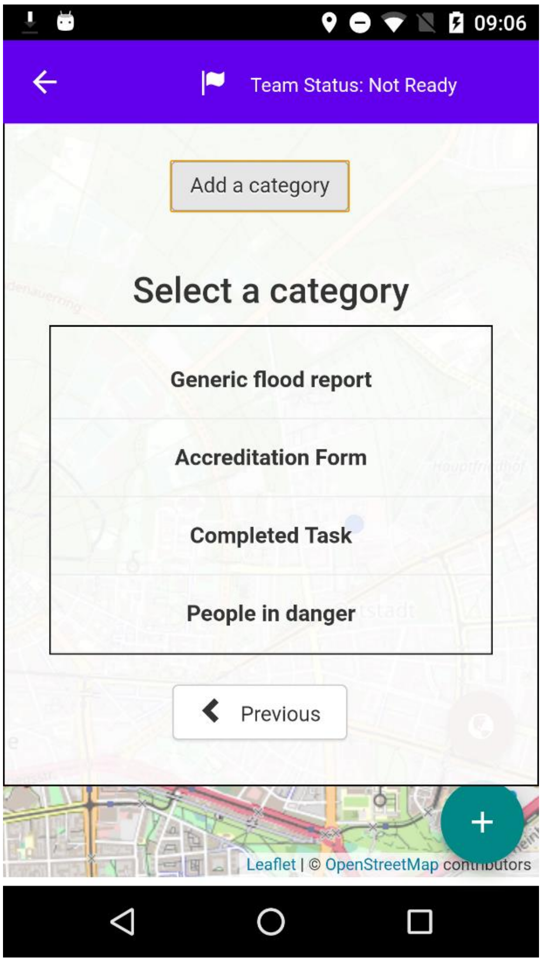
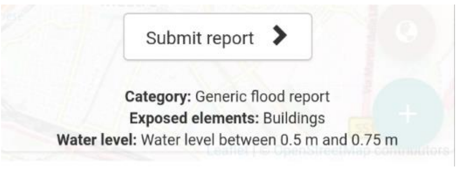


Figure 6 Categories for incident reporting

In alignment with the crisis classification module (see Deliverable 3.1), a categorization scheme for the incident reports was developed. The civilian user can select from a predefined list of categories (Figure 6), which summarises the content of the report supporting the upcoming processing of the report. A more sophisticated scheme was developed for first responders. Being trained and experienced in crisis management, FRs are able to deliver more detailed information.

Figure 7 shows a flood report before sending. The text summarises the information, which are going to be submitted. Here, a generic flood report will be issued with the exposed element *Buildings* and an estimated water level of *0.5m – 0.75m*.



**Category:** Generic flood report
**Exposed elements:** Buildings
**Water level:** Water level between 0.5 m and 0.75 m

Figure 7 Categorization of a flood report

Additionally, a user can further describe the event by using the text field. In the further processing steps of the beAWARE platform this context is evaluated by the text analysis module and the results are integrated into the knowledge base.

Often, it is not easy or even possible to describe the situation in words. Therefore, the enduser application offers the possibility to attach multimedia files, such as images, videos or audio files. These can be directly recorded through the app using the camera or microphone of the mobile devices. If the multimedia file was recorded previously, existing files can be choosen from the device. After submission, images and videos are analysed by the beAWARE platform to detect vulnerable object (people, cars, …) and possible threats (fire, water, …) to determine the content and especially estimate the importance of the report. The audio recording can be seen as alternative to the text field: the beAWARE platform automatically recognizes speech whereupon the same analysis components, as for plain text analysis, are used.

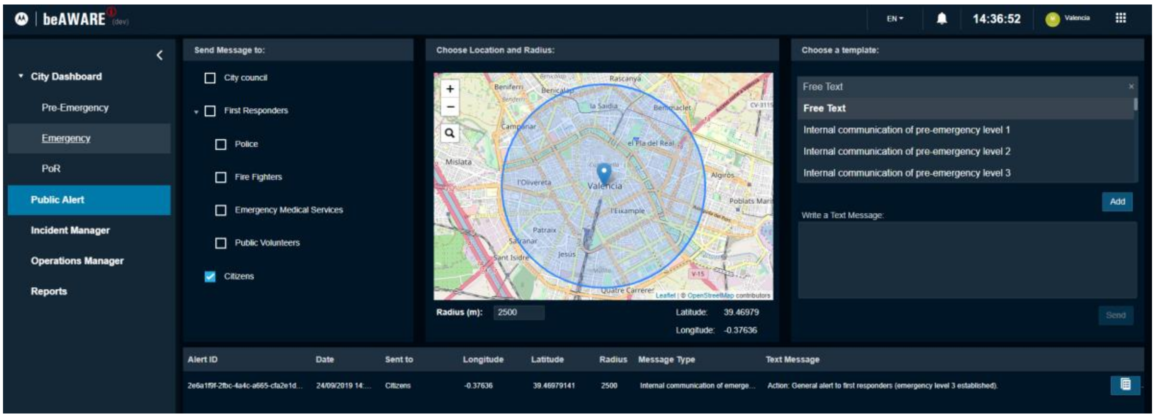
Since all the elements are optional, it is up to the user to decide which modality or combination of modalities he/she thinks is better suited to describe an event.

## 3.4 Alerting

An adequate response is a key factor in limiting extent and damage of a crisis event. Therefore, there is the need to inform citizens about the current situation, as well as providing recommendations for precautions. In case of a fire, this might be avoiding a dangerous area or in case of a flood, preparing the house with sand packs.

Alerts can be sent from the PSAP to distinct addressee groups: they can be issued to the public (all app users), to first responders (citizens will not see those alerts) In the final version it will be possible to send alerts only to first responder groups (police, firefighters, …). This allows broadcasting alerts to specific user groups. This prevents first responders from informational overflow and they only receive information about the current situation they are assigned to.

Figure 8 shows the PSAP menu for issuing public alerts. The location and radius, for which the issue is valid can be set and also the addressee groups can be selected. The informational content, that shall be sent, can either be chosen from templates or can be customized. After issuing the alert, the mobile applications will receive the message and alert the user.



Send Message to:

- □ City council
- □ First Responders
  - □ Police
  - □ Fire Fighters
  - □ Emergency Medical Services
  - □ Public Volunteers
- ☑ Citizens

Choose Location and Radius:

Radius (m): 2500
Latitude: 39.46979
Longitude: -0.37636

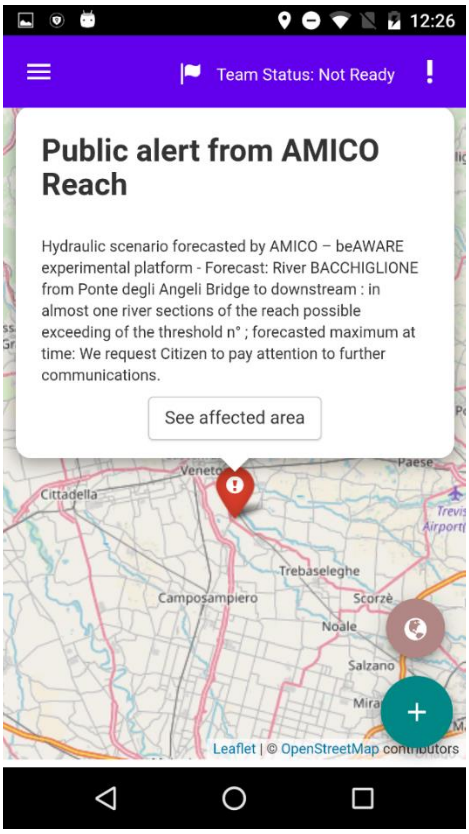
Free Text
Free Text
Internal communication of pre-emergency level 1
Internal communication of pre-emergency level 2
Internal communication of pre-emergency level 3

Write a Text Message:

| Alert ID | Date | Sent to | Longitude | Latitude | Radius | Message Type | Text Message |
|---|---|---|---|---|---|---|---|
| 2e6a1f9f-2fbc-4a4c-a665-cfa2e1d... | 24/09/2019 14:... | Citizens | -0.37636 | 39.46979141 | 2500 | Internal communication of emerge... | Action: General alert to first responders (emergency level 3 established). |

Figure 8 The PSAP UI used to issue Alerts

Figure 9 shows a sample alert with a detailed description. This can be the information that the flood forecasting system predicts a critical value, as shown, or other things like evacuation orders or recommendations for actions. We decided to hide the affected area, by default to prevent the map from getting packed and overcrowded. Anyhow, it is possible to display the affected area. Clicking the button *See affected area* in the alert will display the affected area, as shown in Figure 10.



Public alert from AMICO Reach

Hydraulic scenario forecasted by AMICO – beAWARE experimental platform - Forecast: River BACCHIGLIONE from Ponte degli Angeli Bridge to downstream : in almost one river sections of the reach possible exceeding of the threshold n° ; forecasted maximum at time: We request Citizen to pay attention to further communications.

See affected area

Figure 9 Detailed description of the alert

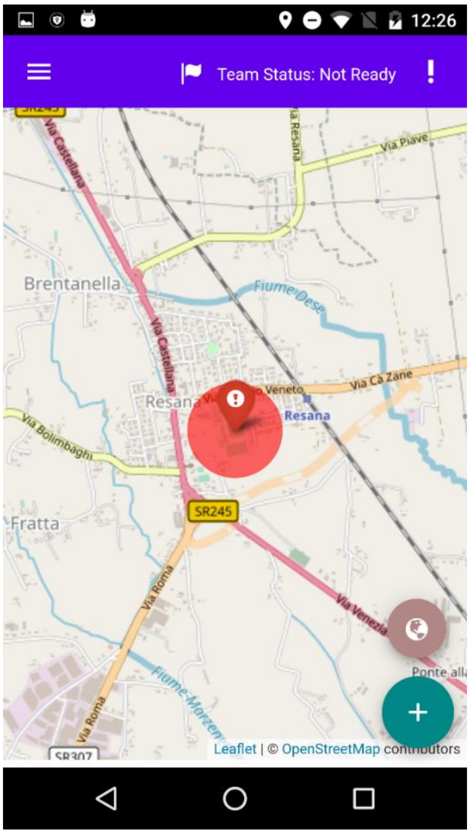
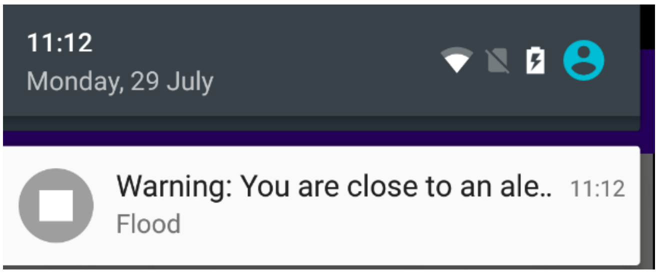


Figure 10 Alert with affected area

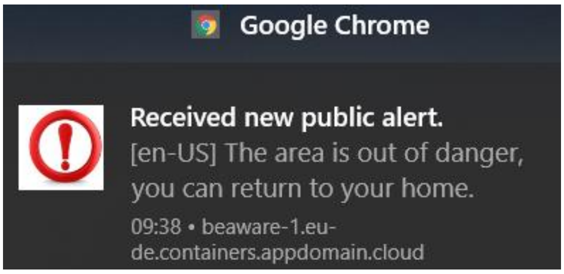
Often alerts contain information that should be quickly recognized by the user. Therefore, we make use of the notification capabilities, provided by the device. Figure 11 shows the reception of a new alert on an android device. In addition to the visual notification, the mobile vibrates and plays a notification sound, if enabled. We implemented the same functionality in the web-based version. Figure 12 shows a notification on Windows, where the application was used in the browser.



11:12
Monday, 29 July

Warning: You are close to an ale.. 11:12
Flood

Figure 11 Notification on the mobile device



Google Chrome

Received new public alert.
[en-US] The area is out of danger, you can return to your home.
09:38 • beaware-1.eu-de.containers.appdomain.cloud

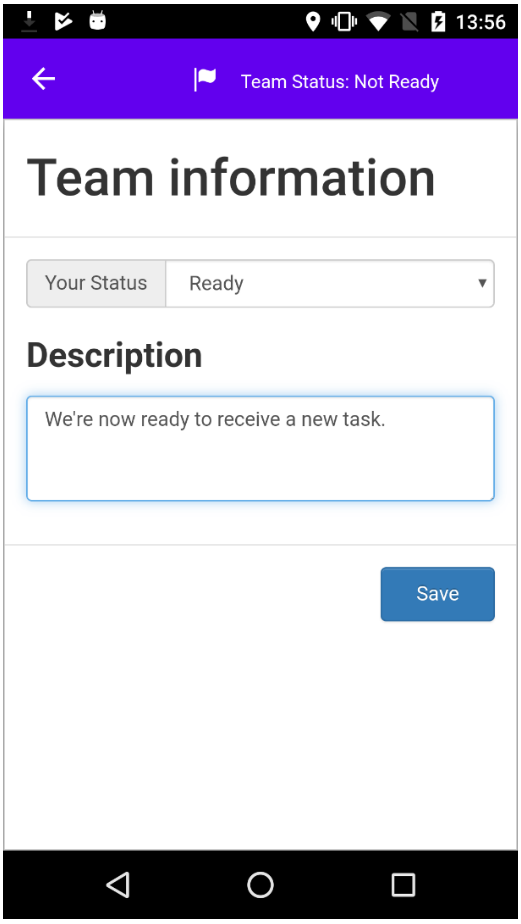
Figure 12 Notification on the PC

## 3.5 Team- and task management

The mobile application is not only used by citizens. It is also used by first responders, like volunteers or recue forces, police or firefighters. These user groups have access to advanced features, which are enabled by signing in with user-specific credentials. For each first responder team, the current position is automatically reported to the beAWARE platform and visible in the PSAP. In addition, each team can indicate its status (see Figure 13), e.g.

- i) not ready to receive a new task,
- ii) on the way to execute the task or
- iii) working on a task.

The position, together with the information is periodically (in our case every 30 seconds, proved as reasonable interval) updated and reported to the system. Once a team has declared its availability, decision makers in the PSAP can assign new tasks. In that case, the device of the first responders shows a notification (similar to Figure 9 and Figure 10Figure 11), starts to vibrate and plays a notification sound. To have a clear picture about the ongoing situation, a message is sent to the PSAP to indicate that the task was actually received by the mobile device.



13:56

Team Status: Not Ready

Team information

Your Status: Ready

Description

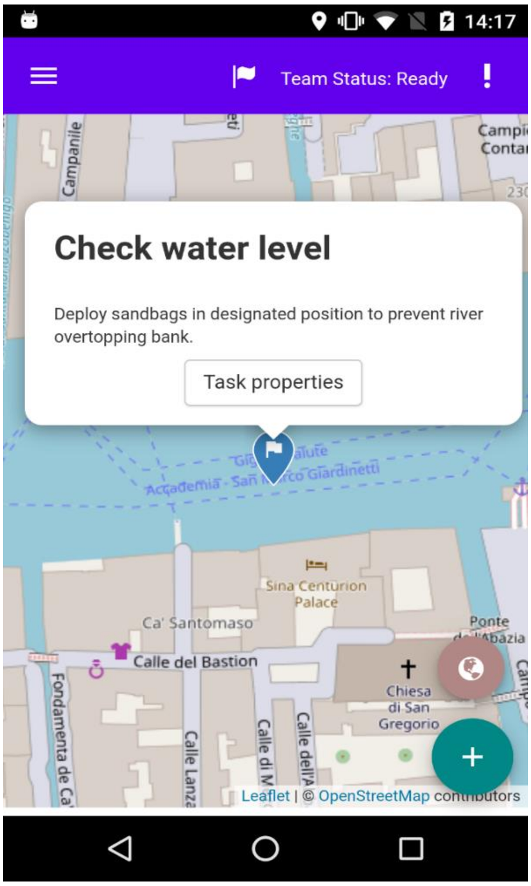
We're now ready to receive a new task.

Figure 13 Updating the team status

Implementing this mechanism, we avoid that decision makers mistakenly believe a task was assigned though the team was not able to receive it, due to bad network connectivity or low battery. An overview of the task and the position is visualized for the first responder, as shown in Figure 14. Expanding the task properties (Button *Task properties)* will show all information assigned to the task, e.g. the description and instructions what exactly to do. This is shown in Figure 15. The same window is used by first responders to update the task status (e.g. OTW – the team is on the way to the tasks position). FRs can choose a task status from the following list:

- i) Received
- ii) Accepted
- iii) Rejected
- iv) OTW (on the way)
- v) Working on the task
- vi) Completed task.

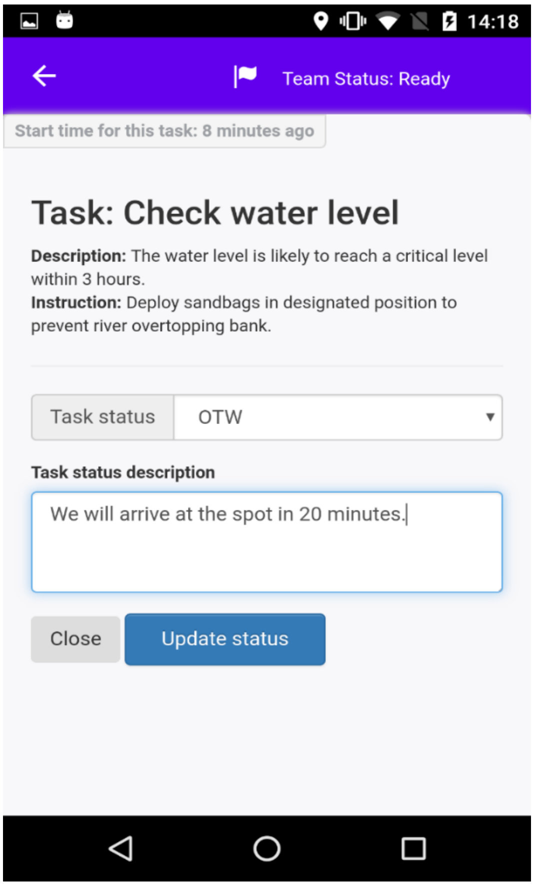
It also offers the possibility to add a textual description about the status, to provide more detailed information to the decision makers.



Check water level

Deploy sandbags in designated position to prevent river overtopping bank.

Figure 14 Overview of the task an its location



Start time for this task: 8 minutes ago

Task: Check water level

**Description:** The water level is likely to reach a critical level within 3 hours.
**Instruction:** Deploy sandbags in designated position to prevent river overtopping bank.

Task status: OTW

**Task status description**

We will arrive at the spot in 20 minutes.

Figure 15 Updating the status of the task

# 4 Implementation and Integration

This section describes the technology used implementing the mobile application. The end-user application is part of the external layer of the beAWARE platform (see D7.2 for further explanation). It is the main interface and interaction point for citizens and first responders in the field with the beAWARE platform. The following subsection will describe firstly state of the art technology for implementing a mobile application. Secondly, we will describe the internal architecture for the mobile app, together with the internal semantic data representation. Finally, the integration into the beAWARE platform is explained.

## 4.1 State of the Art technology

Nowadays, multiple technologies exist for implementing mobile applications. Each smart phone platform provides a dedicated ecosystem with a programming language, development-tools, services and distribution platforms (known as Play-/App-Store). For example, Apple provides the Swift programming language, together with the development tool XCode[1] to create applications for iPhones. Google, the company behind the Android platform makes use of the Java-Programming language and provides dedicated development tools[2]. Apps developed for a dedicated platform are called native applications. Specifically, designed apps for the corresponding smart phone platform offer benefits so that the applications can take advantage of all functionalities, services and sensors available on the platform. It is also a prerequisite to publish the application the Play[3]- or App-Store[4].

Those native applications are in general more performant if a lot of data needs to be handled or intensive visualizations (e.g. in games) are needed. On the other side, they can only be used on the platform they are developed for. Also, they are not usable as a web-based version. Since web-based development is, like already discussed, very helpful during the iterative development and validation in the beAWARE project, we decided not to develop a native app.

To overcome the issues with platform specific applications, so-called “Cross-Platform” frameworks were developed. The most prominent framework in this area is Cordova[5], maintained by the Apache Software foundation. React-native[6], developed by Facebook is a newer alternative. The idea is to develop a common source code, which then can be run on

[1] https://developer.apple.com/develop/

[2] https://developer.android.com/

[3] https://play.google.com/store/apps

[4] https://www.apple.com/de/ios/app-store/

[5] https://cordova.apache.org/

[6] https://facebook.github.io/react-native/

multiple platforms. This idea is often realized by using web technologies like HTML[7], CSS[8] and JavaScript[9]. These technologies are used every day by every user on the internet.

With the release of HTML5[10] in 2014 the development of web-based applications gained more and more momentum. It had been recognized that modern applications need the ability to use the devices' sensors and contextual information (e.g. current location) and require access to the devices camera or to files stored on the device. To meet those requirements browser vendors like Microsoft (Edge, Internet Explorer), Mozilla (Firefox), Google (Chrome) are continuously developing and integrating new APIs[11], which makes it possible for developers accessing the device's functionalities. To overcome the issue with functionality that is not available for web-based applications the Cordova project has developed proxy-functions, allowing to access the native functions in a web-based environment.

As introduced, we decided to use JavaScript, HTML and CSS for the development of the app; with the Cordova framework we translated the web-application into a mobile application. On the one hand, this offered flexibility to run the application either on an Android device or in the browser, which is beneficial by itself. On the other hand, Cordova offers the possibility to integrate access to the device's functionality needed to meet the user requirements. Moreover, it generates an application package (apk) for Android devices, so that the beAWARE mobile application can be installed on the device like a native application and the used technology is transparent for the user.

Since there is a lot of dynamic in the development of the HTML5 APIs, we are continuously evaluating which ones can be used and where we need the proxy functionality of Cordova. As future technology, Progressive Web Apps[12] (PWA) look very promising and might have the potential to realize applications, that are indistinguishable from native applications, but are running on different platforms. This technology might substitute Cordova in the future.

## 4.2 Architecture of the end-user application

For the internal architecture of the end-user application, we followed the common client-server architecture pattern. The mobile application itself, which is running in the browser or on the mobile phone, is a client interacting with a server component running in the IBM cloud environment. The client part is responsible for the interaction with the end user, which consists of:

[7] https://developer.mozilla.org/en-US/docs/Web/HTML

[8] https://developer.mozilla.org/en-US/docs/Web/CSS

[9] https://developer.mozilla.org/en-US/docs/Web/CSS

[10] https://developer.mozilla.org/en-US/docs/Web/Guide/HTML/HTML5

[11] https://developer.mozilla.org/en-US/docs/Web/API

[12] https://developers.google.com/web/progressive-web-apps/

- **Displaying:** The information, available from the beAWARE platform should be presented to the user in a clear and structured way.
- **Capture the user input**: A form is provided to the user to send new data.
- **Interacting with devices sensors:** The sensors of the devices (GPS, camera, microphone) need to be accessed and added to the user input fields.
- **Notifying:** By using the devices capabilities (e.g. vibrating) the user needs to be notified about newly available data like public alert or a team task.

The server component integrates the client part into the overall platform, especially by providing:

- **Authentication and authorization:** Each user should only be able to access elements dedicated for him. E.g., it should not be possible for citizens to see tasks of first responders.
- **Data management:** Data coming from the end-user devices, as well as sent from the beAWARE platform to the devices needs to be buffered, pre-processed and validated.
- **Interacting with the other beAWARE components:** To enable a smooth integration with the other components of the beAWARE platform, a well-defined interface was specified (see D6.2 and D7.2). The server component implements this interface to receive from and sent data to the other component. For example, if an incident report is received from the client, the multimedia files need to be uploaded to the central data repository (CDR) and a message on the bus needs to be posted.

To internally represent the data in the server part, we decided to implement semantic technologies. Therefore, a dedicated ontology was developed to handle all data regarding the end-user application. This allowed the development of a flexible data model with the same technology, applied in WP4. Figure 16 shows the ontology internally used by the end-user application.

## 4.3 Integration into the beAWARE platform

Figure 17 shows a high-level view of the beAWARE architecture, which is described in more detail in D7.5. Like foreseen in the overall architecture, the integration of the end-user application is realized through the message bus. The occurrence of events (new report or public alert, changed position of team, …) is propagated to the involved components over the message bus. Its function is explained in detail in Deliverable D6.2. It also introduces the concept of topics, where each of them represents one type of information published on the bus. Other components, that are registered on these topics receive the information published and start their own processing with this bit of data. Table 2 summarizes the topics, used by the end-user applications to interact with the other components of the beAWARE platform. The messages published on the bus contain all information and metadata, except for multimedia files, sent through the end-user application. The multimedia files (photos, videos or audio) are stored in the central data repository (CDR). Only the identifier of the multimedia file is included in the message. This way, the receiving components can download the file from the CDR, if needed.

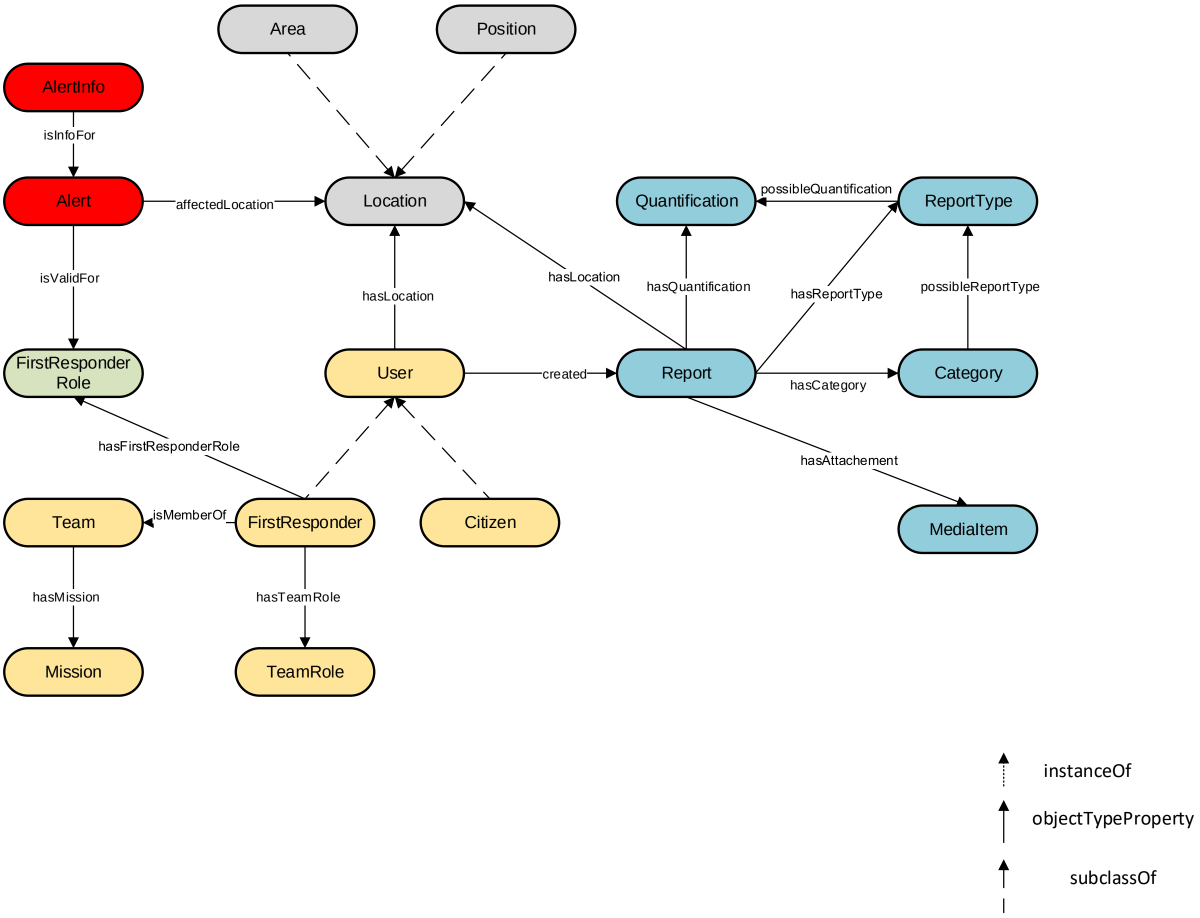


Figure 16 The mobile app ontology

Table 2 Topics on message bus, used by enduser applications

| Topic name | Description | Involved components |
|---|---|---|
| TOP021_INCIDENT_REPORT | New incident report, by either a citizen or a first responder. | End user application to Knowledge Base Service (KBS) |
| TOP022_PUBLIC_ALERT | New public alert created by authorities. | PSAP to end user application |
| TOP023_TASK_ASSIGNMENT | A new task was created by the authorities and was assigned to a team. | PSAP to end user application |
| TOP102_TEAM_REPORT | Periodically sent to update team status as well as the current team position. | End user application to PSAP |
| TOP103_TASK_REPORT | Status update (status or status description) for a task. | End user application to PSAP |

For a more detailed description, especially about the structure of the messages, sent over those topics on the message bus we refer to D6.2, respectively its confidential version.

Since the end-user application is the main interaction point for citizens and first responders, the inserted data is used by various components:

- **Knowledge Base Service (KBS)**: The KBS processes *TOP021_INCIDENT_REPORT* messages, which represent the event of a new incident report. It is responsible to populate the beAWARE ontology inside the Knowledge Base. The KBS also orchestrates the analysis of potentially attached media files and forwards the report to the PSAP.
- **Crisis classification**: In collaboration with the KBS and Knowledge Base, the Crisis Classification module watches for incoming incident reports, that cause change in the currently estimated crisis level.
- **Public Safety Answering Point (PSAP):** The PSAP is the interface for the authorities to interact with the beAWARE platform. Amongst various capabilities, it presents the incident reports (forwarded by the KBS) to the decision makers, issues public alerts and interacts with the team- and task-management functionality of the end-user applications.
- **Knowledge Base, Report Workbench:** Since all incident reports are persisted semantically, either in the *mobile app ontology* or in the *beAWARE ontology*, the report workbench (developed in WP4 to analyse Twitter messages) can also be used

to analyse reports, created by the end-user application, either during or after the crises event. More information about this can be found in the upcoming WP4 deliverables.

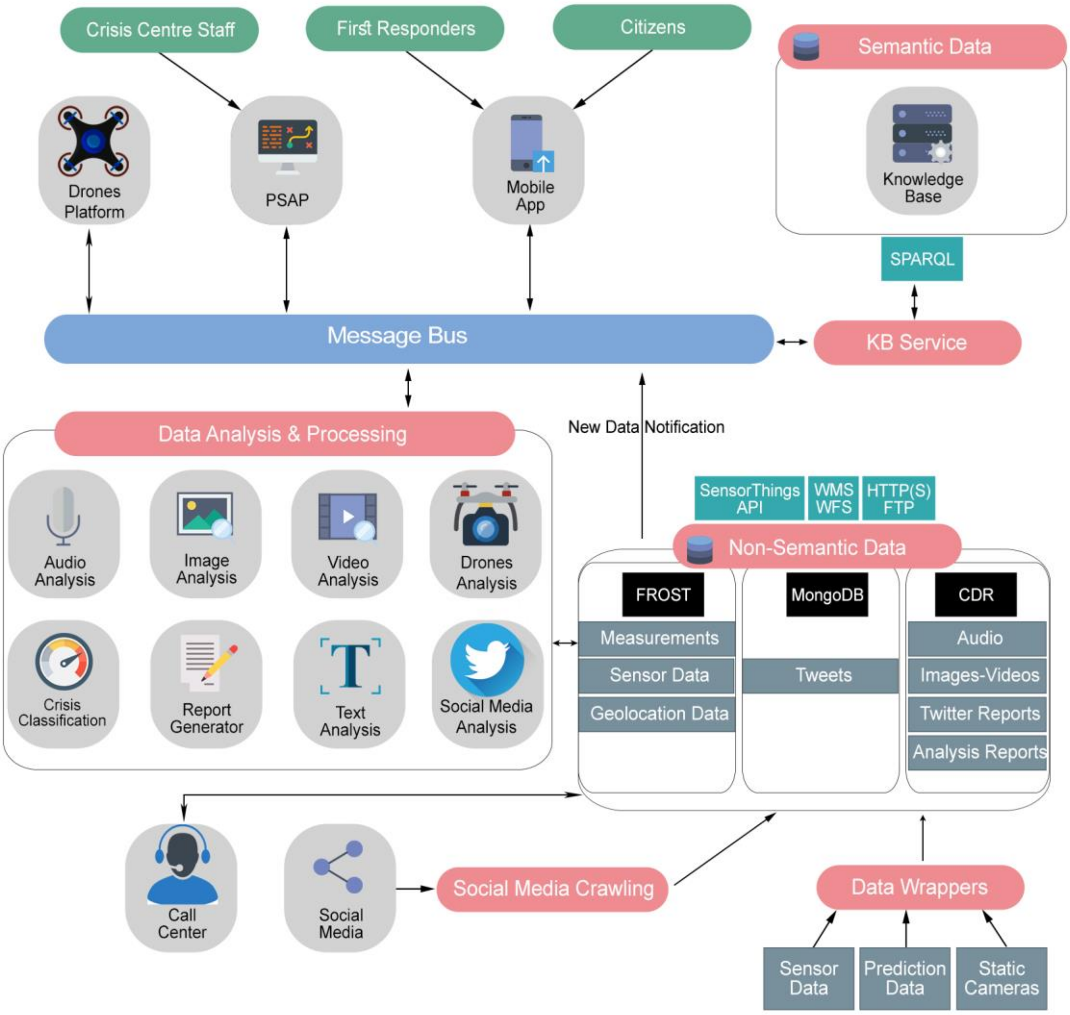


Figure 17 High-level view of the beAWARE architecture

The following pictures is representative for the information flow in the beAWARE platform. It shows the process from issuing an alert in the PSAP and the representation within the beAWARE mobile applications (Compare section **Error! Reference source not found.**. After the authorities send out a public warning, this is communicated from the PSAP to the message bus on Topic 22. The backend components, such as the KB, store this alert. After this, the applications retrieve the relevant information, such as description and position. Finally, based on the position of the device, the applications evaluate, if the alert is relevant for the user; if so, the user is alerted.

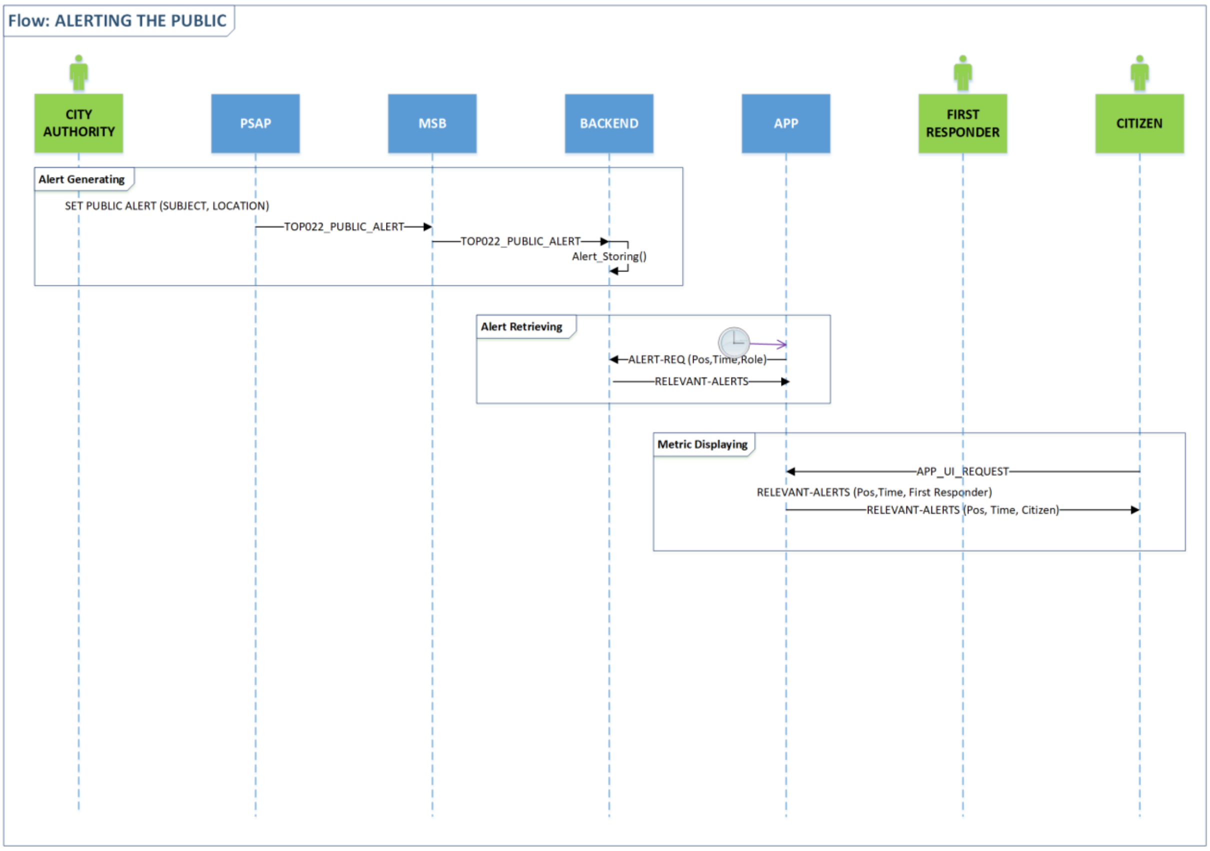


Figure 18 The informational flow of an issued alert

This chapter provided the technical background of the end-user applications. Firstly, by summarizing state of the art technologies and motivating our decision. Secondly, data representation was explained. Finally, we arranged the application in the overall architecture and explained the integration concepts with other components.

# 5 Evaluation

To guarantee the usefulness and applicability of the mobile applications, a continuous evaluation process was established. During our weekly telephone conferences within the beAWARE consortium, the current features have been presented and discussed. This allowed the end-users to review the current progress and to provide timely feedback. Those results allowed a quick adjustment of the development, to meet the user requirements.

A more thorough evaluation was performed during the execution of the pilots, where the mobile applications played a crucial role in the involvement of first responders and citizens. The setup of the pilots was described in D2.3 for the first and in D2.5 for the second pilot. A detailed evaluation of those can be found in D2.4, respectively D2.6. We shortly recap the results in the following sub-section. More details can be found in the corresponding deliverables.

## 5.1 Results of the 1st pilot

The first pilot took place on 19th to 20th of November 2018 in Thessaloniki (Greece). In general, the players (actively participating in the pilot) and the observers (monitoring the players and preparing documentations) confirmed that the mobile applications, together with the features of the beAWARE platform, supported the user in getting the needed information in real-time. Since localizing incoming data from legacy tools like radio (often referred as VHF) email or telephone is often an issue, evaluators think the location, automatically provided with every report from the mobile application is very helpful.

Smaller bugs have been detected during the pilot. For example, the parallel use of some social media apps (like Viber[13] or Messenger[14]) together with the beAWARE application caused some issues. In fact, the users reported freezing or restarting of the mobile application. The problems have been investigated and fixed for the second pilot. A bad internet connectivity, especially inside the shelters, caused delay in sending the reports (especially when multimedia files were attached). We took this opportunity and improved the offline functionality of the app. Nevertheless, the mobile app relies on mobile internet infrastructure to communicate with the beAWARE platform. Tackling this challenge would have required the deployment of an additional infrastructure, which is out of the scope of the beAWARE project.

In addition, it was noted that some of the participants needed more time to be properly trained in using the application, which was considered in the second pilot.

[13] www.viber.com

[14] www.messenger.com

## 5.2 Results of the 2nd pilot

The second pilot took place in Vicenza (Italy) on 7th of March 2019, where the 2nd beAWARE prototype was tested in a flood scenario.

As in the first pilot, the scenario was split into two sessions using only legacy systems (mainly VHF) and using the beAWARE platform. Since VHF is commonly used by volunteers and first responders to communicate with the authorities, many of the results contain a comparison between VHF and the mobile application. The feedback showed that participants in a crisis like to have a direct communication channel with the authorities: receiving feedback, when sending a report, updating the task status or the team position is considered useful.

Yet, we cannot offer functionality to answer all messages sent through the mobile application, since authorities would not be able to handle all information. Therefore, we explicitly omit a manual confirmation through the authorities. Anyhow, we display more clearly in the mobile application, if a report was received by the platform and if it is available to the authorities. Testers noted that the mobile applications are good and easy to use, but they see some benefits in using the VHF in some cases. The common suggestion was to not replace the radio, but to use the mobile application as an additional communication channel.

During the development, we tried to set the technical requirements as low as possible to support as many devices as possible. Therefore, it was unavoidable that special devices/configurations caused some issues (e.g. losing the geolocation, delays in uploading reports …). To overcome this, we provided instructions, what to do if the application does not behave like expected. The responses on the questionnaires showed, that following those recommendations solves the issues, so that everyone was able to use the mobile application.

The participants stated that the mobile application is also very helpful for citizens. Since citizens usually do not have a chance to communicate with the authority, the mobile application allows them to send real time data without the common problems related to call centres.

One requested functionality was seeing the reports, status and position of other first responder teams, since the VHF allows users to hear tasks assigned to other users. To keep the usage of the mobile application as clear as possible and to not overcrowd the display of the small devices, we decided to limit displaying only the own reports and tasks and did not follow this suggestion.

## 5.3 Chapter Summary

The execution of the pilots showed that the involved people in general like the mobile application. They identified smaller technical issues, mainly occurred due to the wide variety of used devices, configurations and other used applications. We used those reports to improve and stabilize the beAWARE mobile applications. In addition, the participants provided constructive feedback and suggestions, which were used to improve the development for the second prototype. Yet, some suggestions were omitted to keep the usability of the mobile application. Currently, the results of the second pilot are used to further improve the final version.

# 6 Conclusions and Next Steps

## 6.1 Conclusions

This deliverable presented the resulting mobile application, developed in Task 7.5. We have reintroduced all relevant user requirements and showed the functionality of the mobile app, implementing those requirements. We have presented the different features of the application, showed the resulting user interfaces and explained their usage. Currently available state-of-the-art technologies were described and we argued, why we followed a cross-platform approach using Cordova. The internal architecture of the end-user applications, together with the underlying semantic data model was described. We showed the integration into the overall beAWARE platform and finally recapped the overall positive evaluation results of the two pilots, already executed.

## 6.2 Next Steps

While writing this deliverable, the last development round for the third prototype of the beAWARE platform is going on. Therefore, we are still working on implementing the last functions. For example, the final version will allow first responders to indicate their profession and expertise. This information will be available to authorities through the PSAP to help authorities to assign the task directly to the correct team. Especially in the forest fire scenario, where several groups like fire brigade, police, ambulance and volunteers will be involved, this will be a valuable functionality. It will also be possible to restrict the recipients of public alerts to a specific group, to spread the information in targeted manner. We take into account the feedback gathered during the second pilot. Especially we are working on simplifying the usage of the mobile application and making functionality, that is often needed, easier accessible (e.g. reporting the status on the currently assigned task). Furthermore, there are still ongoing improvements in stabilizing the application and increasing the efficiency of the data exchange functionality. We observe the standardization of the HTML5-APIs and the cross-platform approach will be replaced by a Progressive Web App approach in the future, which will easily allow using the applications on other platforms/devices as well.