

VNCS: A VOLUNTEER ONLINE SOCIAL NETWORK FOR IMMEDIATE CITIZEN AWARENESS AND ACTION IN CRITICAL SITUATIONS

Merkouria Dimopoulou¹, Evi Papaioannou^{2*}, Christos Kaklamanis²

¹University of Patras, Greece, vardaki@ceid.upatras.gr

²University of Patras and CTI "Diophantus, Greece, papaioan@ceid.upatras.gr,
kakl@ceid.upatras.gr

*Corresponding author

Abstract

Exploitation of modern Information and Communications Technology can raise awareness and efficiently support citizen initiatives and actions at a local level for facing emergency situations before they spread.

Motivated by the recent destructive wildfires in Attica, Greece, we suggest Volunteer Network for Critical Situations (VNCS), an online collaborative system for immediate channeling and efficient diffusion of information about emergency situations based exclusively on the contribution of people / citizens located within a small range around the area where an incident occurs.

Our objective is two-fold. On the one hand, we wish to demonstrate how advances in technology and communications can be exploited in order to strengthen social solidarity and social welfare. On the other hand, we wish to provide a practical solution, an online social service, for supporting citizens to turn awareness into action during critical situations prior to the intervention of special government agencies and mechanisms.

Keywords: internet-based online collaborative system, smart devices, dissemination of information, emergency situations, online social network of volunteers, social solidarity and welfare.

1 INTRODUCTION

In emergency situations (e.g., fires, floods, earthquakes, accidents, etc.), information about urgent needs for assistance - before or during the degraded situation - is usually concentrated in a small area around the point of incident. For example, when a fire burnout occurs or when an accident occurs, people in the surrounding area get informed about the incident in very short time, e.g., through simple observation. Furthermore, it is very likely that many of them also have knowledge of important geographical and geotopological details of the specific area (e.g., shortcuts, obstacles, works in progress, one-way streets, etc); such details can be of very crucial nature for immediate reaction when emergency situations occur.

Nowadays, due to rapid advances in communications and digital technology, the vast majority of the population are increasingly connected to the world of digital information while "on the go" via smartphones and other mobile devices. In addition, social networks, exploiting computer-mediated technologies, facilitate

the creation and sharing of information, ideas, and other forms of expression via virtual communities and networks (Hermkens and Kietzmann, 2011). As a result, the vast majority of the population can directly diffuse information, via short messages, posts in social networks, etc, which can then spread extremely fast (Favre, Abdelkader Zighed, Guille, Hacid, 2013, Olinsky, Quinn, Quinn, Quinn, 2016). Furthermore, the emergence of special-purpose social networks and services can certainly provide ability and motivate willingness for special forms of information to be shared, thus, translating awareness to action.

Despite the existence and operation of special civil protection services, volunteering, individual contributions and / or the contribution of small groups of individuals remain important and critical factors for the immediate response to emergency situations. For example, if a person detects a small fire spot and has a way to instantly alert his neighbors, it is highly likely that a small group of people / volunteers will prevent it from spreading or contribute effectively to informing relevant services and providing on-site assistance to people in need.

With these in mind, and motivated by the tragic consequences of the recent destructive wildfires in Attica, Greece (e.g., <https://www.bbc.com/news/av/world-europe-44947962/greece-wildfires-aerial-view-of-destruction>), we designed and implemented the Volunteer Network for Critical Situations (VNCS), an online collaborative system for direct channeling and efficient dissemination of information about emergency situations based exclusively on the contribution of people / citizens located within a small range around an area where an emergency incident occurs.

People owning a smart device connected to the Internet can register to VNCS essentially creating an online social network of volunteers. VNCS keeps track of the location of each connected user and indicates user positions on a map. Whenever a user submits a critical event, the system automatically informs all connected users within a small range around the incident area for the type of event, its start time, the level of criticality and the range of the area from which immediate assistance can be offered.

The aim of this special-purpose social network is to efficiently manage and exploit initiatives and voluntary contributions of individuals or small groups of people for facing emergency situations on-site in order to avoid their spread. Furthermore, our objective is to exploit advances in technology and communications in order to strengthen social solidarity and social welfare via offering a practical approach for motivating and self-organizing citizens in order to turn awareness to action during critical situations prior to the intervention of special government agencies and mechanisms.

2 EMERGENCY SERVICES AND SOFTWARE SOLUTIONS

In this section, we provide an overview of indicative existing frameworks for supporting the management of emergency situations. However, we must stress that these systems mainly focus on receiving incident information, simulating emergency situations or transmitting information in the form of protection guidelines from state civil protection agencies to people in an emergency area. Unlike VNCS, these systems are not intended to make participatory platforms or services aiming to mobilize individuals or groups of individuals for active participation in facing emergency incidents.

2.1 The Single European Emergency Number 112

According to the Decision 91/396/EEC repealed by Directive 2002/21/EC regarding the introduction of a single European emergency call number, EU Member States were required to ensure that the number 112 was introduced in public telephone networks as the single European emergency call number by 31 December 1992. People in distress can call the 112 emergency number from anywhere in the EU and free of charge. 112 was introduced in parallel with any other existing national emergency call numbers. It is essentially an emergency communications service using the single European emergency call number, 112, which is enhanced with location information of the calling user. As an “emergency service”, 112 is intended to provide immediate and rapid assistance in situations where there is a direct risk to life or limb, individual or public health or safety, to private or public property, or the environment but not necessarily limited to these situations. It utilizes “location information”, i.e., data indicating the geographic position of a user’s mobile terminal or data about the physical address of the termination point in a public fixed network, which is appropriately forwarded to public safety answering points, i.e., physical locations where emergency calls are received under the responsibility of a public authority.

According to the 2018 Report on the implementation of the European emergency number 112 based on data collected in 2018 throughout the EU Member States, calls to “112” represented 48% of the calls placed in case of an emergency. Furthermore, as stated by Mariya Gabriel, the Commissioner for the Digital Economy and Society, “112 is a life-saving number for all users. The Commission services are engaging actively with

relevant Member States authorities in order to address all outstanding issues and to secure swift and concrete solutions that ensure equivalent access to emergency services for everybody. The recently adopted European Communications Code and the political agreement on the European Accessibility Act reinforce our Union's commitment to ensure an inclusive society."

2.2 Wildfire Evacuation Trigger Buffers for Sensitive Areas (EVITA)

The Project EVITA "Wildfire Evacuation Trigger Buffers for Sensitive Areas" (ECHO/SUB/2012/640929) was funded by the Directorate-General for European Civil Protection and Humanitarian Aid Operations (DG ECHO) during the period 2013-2014 (<http://evita.eu-project-sites.com/>).

EVITA suggested a framework for simulating fire incidents for the whole southern part of Europe, which can be easily extended to provide Pan-European and even global coverage.

The main objectives of EVITA included the promotion of a common understanding of trigger buffer zones as a novel means to assess the time until an active wildfire approaches a sensitive site. Athens (Greece), Mallorca (Spain) and Sardinia (Italy) were used as case study areas.

In the context of EVITA, a web-based computational system was developed for simulating the propagation of wildfire incidents, according to several topographic and meteorological factors. Trigger Buffers were identified for user-selected sensitive locations. Moreover, "Landmark" locations, i.e., locations which if burnt, great risk would incur for more than two neighboring sensitive locations, were also identified for wide geographic regions.

2.3 ENGAGE IMS/CAD

ENGAGE IMS/CAD (<http://www.satways.net/products-sw/engage-ims-cad/>) is a commercial an integrated Call-Center solution for public and private safety organizations providing tools for call and incident management, computer aided dispatch, operational resource management and disparate crucial information data integration.

According to its description, ENGAGE allows seamless interconnection of control centers and the field by deploying mobile dispatch capabilities on smartphones, and tablets allowing field personnel to update event details, access databases, and exchange rich media messages. Using a distributed Event Driven architecture, ENGAGE supports comprehensive incident control and dispatching for Public Safety offering speed, reliability, and adaptiveness to highly complex communication environments.

3 VNCS TECHNICAL DESCRIPTION AND DEMONSTRATION

VNCS - Volunteer Network for Critical Situations - is an online collaborative system for direct channeling and efficient dissemination of information about emergency situations based exclusively on the contribution of people / citizens located within a small range around an area where an emergency incident occurs.

It is currently available at: <https://snf-864568.vm.okeanos.grnet.gr/>

For using VNCS, entities - individuals or groups - owning a smart device connected to the Internet must first register to VNCS. In this way, an online social network of volunteers is actually created. VNCS keeps track of the location of each connected user and indicates user positions on a map. Whenever a user submits a critical event, the system automatically informs all connected users within a small range around the incident area for the type of event, its start time, the level of criticality and the range of the area from which immediate assistance can be offered.

Below, we provide technical details for VNCS together with a comprehensive demonstration regarding its setup and use.

3.1 Database

The Relational Database Management System (RDBMS) used for the VNCS is a Microsoft SQL Server.

VNCS uses a simple database composed of 6 tables.

Table 'User' contains the users of the system and information such as username, password, email, status (online/offline), coordinates of their geographical location according to the geographic coordinate system, account creation date and date of the last login session.

Table 'Incident' contains critical events submitted by VNCS users. For each event the following details are

stored: description, location coordinates, status (active/expired), user who submitted it, level of criticality, the category it belongs and datetime of its submission.

Table 'Category' contains a typical description of the category each event belongs to, e.g. Fire, Flood, Missing Person, Earthquake etc..

Table 'Level' contains the levels of criticality; a user submitting an incident can define how critical this incident is by assigning it a criticality level.

Table 'IncidentRelUser' contains temporary information regarding users who should be notified for specific events, according to the VNCS notification algorithm. When the VNCS notification algorithm "decides" that a user must be informed about an incident, a record containing the ids of both the user and the incident is stored in table 'IncidentRelUser'.

Table 'Helper' contains the active users who accepted to assist after they were informed via notifications.

In addition, for purposes of integrity and reliability of VNCS, we defined the following conditions: every table which implies a relationship between an incident and a user contains a 'status' column indicating whether each record is valid or not. Thus, when a user declares an incident as expired all corresponding records in tables 'IncidentRelUser' and 'Helper' which refer to the specific incident also expire.

Furthermore, when the geographical location of a user or when a user logs off the system all corresponding records in tables 'IncidentRelUser' and 'Helper' which refer to the specific user expire, as user's previous location and status are outdated.

The Entity Relationship Diagram (ERD) of VNCS database is shown below.

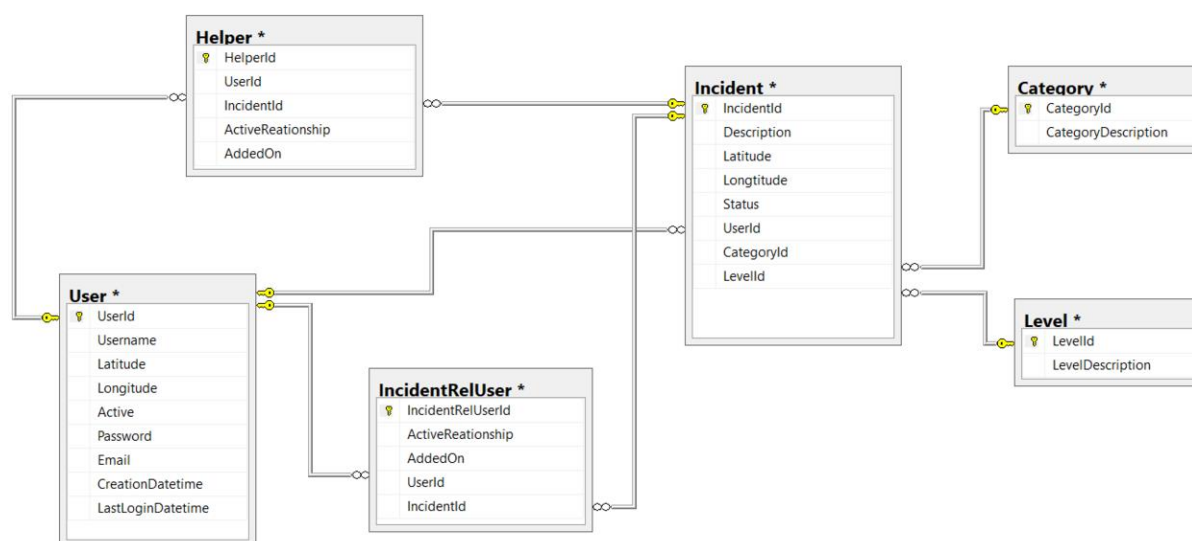


Fig. 1. The Entity Relationship Diagram (ERD) of VNCS database.

3.2 Server Side

For the implementation of the server-side part of VNCS, ASP.NET MVC framework (<https://dotnet.microsoft.com/apps/aspnet/mvc>) was used. ASP.NET Model View Controller (MVC) is a server-side web application framework which is based on ASP.NET (<https://dotnet.microsoft.com/apps/aspnet>) designed for web development to produce dynamic web pages. ASP.NET is a developer platform that extends the .NET developer platform with tools and libraries specifically for building web apps. MVC architectural pattern defines web applications with 3 logic layers: Model (business layer), View (display layer), Controller (input control). Below, we describe how an MVC pattern was implemented for the purposes of VNCS.

The Model represents the domain specific data and business logic or operations that should be performed by the application in MVC architecture. It maintains the data of the application. For VNCS, the layer Model is implemented via Microsoft Entity Framework (<https://docs.microsoft.com/en-us/ef/>), an Object Relational Mapping (O/RM) tool that sets a data access layer from a database automatically using .NET objects. Our

Model is generated according to the VNCS database and the logic is stored there.

MVC Controller is responsible for responding to requests to the ASP.NET MVC application. It handles user requests by exposing actions. A controller action is a method that gets invoked when a corresponding URL is accessed. Controllers handle interactions and update the model to reflect a change in the state of the application, and then render the appropriate view with the model data as a response. Briefly, a controller and its action methods handle incoming browser requests, retrieve necessary model data and return appropriate responses. In our application controllers constitute all VNCS functionalities, such as execution of the algorithm, submission of the critical events, displaying details for a selected incident, declaration of a user's availability to offer assistance, displaying the system live map, tracking the geographical location of a user, etc.

A View accepts necessary information from the controller and renders a user interface (UI) to display that information. View is a UI, it displays data from the model to the users and also allows users to modify the data. According to the VNCS architecture, UI constitutes the client side which is described below.

3.3 Client Side

VNCS User Interface is essentially a simple website, so that the whole system remains simple and functional. VNCS User Interface follows a responsive design and can adapt to displays of various widths. VNCS UI was developed using Bootstrap 3 (<https://getbootstrap.com/docs/3.3/>), a grid-based responsive framework used to develop responsive, mobile-centric projects. It is extensively used for building various mobile-friendly, robust websites that are compatible with a wide range of browsers as well as for developing web-based interfaces according to various mobile responsive designs. Furthermore, HTML and CSS were used - for custom parts of the VNCS website - together with additional JavaScript libraries.

One of the main functionalities of VNCS is to inform groups of users within a specific range (determined by the VNCS notification algorithm) around the area of an incident. VNCS immediately informs users about all incidents tracked near them in an efficient way. The notification functionality was implemented via real-time personalized notifications sent to eligible VNCS users. SignalR (<https://dotnet.microsoft.com/apps/aspnet/real-time>) was used for the development of the notification functionality, as VNCS was built on ASP.NET. SignalR is a software library for Microsoft ASP.NET that allows a server to send asynchronous notifications to client-side web applications.

Additionally, VNCS makes extensive use of maps, in order to display the geographical locations of users and incidents. We decided to use Mapbox GL (<https://docs.mapbox.com/help/glossary/mapbox-gl/>), a suite of open-source libraries for embedding highly customizable and responsive client-side maps. Mapbox is a large provider of custom online maps for web, mobile, and desktop applications. We rendered Mapbox interactive maps through Mapbox GL JS, a JavaScript library that uses WebGL (<https://get.webgl.org/>) for rendering interactive 2D and 3D graphics.

3.4 VNCS tracking algorithm

VNCS uses a Breadth-first search (BFS) algorithm for tracking active users who should be informed for each incident. BFS (Rivest, Stein, Cormen, and Leiserson, 2009) is a fast algorithm for traversing or searching tree or graph data structures. Starting at an arbitrary node of a graph v , also referred as a search key, it gradually builds a tree rooted at v by exploring all neighboring nodes at the present depth before moving on to the nodes at the next depth level. Via the use of appropriate data structures (i.e., adjacency lists), BFS can obtain a time complexity of $O(V+E)$ for a given graph $G=(V,E)$, where V is the set of nodes and E is the set of vertices in G .

In the context of VNCS, the underlying graph is composed of vertices corresponding to VNCS active users and edges created in each step according to the distance between active users. For each active incident, l_{active} , the following procedure takes place. Incident l_{active} becomes the root node of BFS search tree. During the first step of the VNCS tracking algorithm, a search for active users within a disk of predefined range takes place. Tracked users are added to the BFS tree as level-1 nodes (i.e., neighbor nodes to the root of the BFS tree) and are marked as visited. In next step, for each marked user, neighbor nodes not visited yet are explored. This procedure is repeated until a maximum number of steps is reached.

VNCS tracking algorithm pseudocode

Input: usersActiveList (list of users), incidentsActiveList (list of incidents), steps

Output: usersNearIncidentFinalList (list of users who will be informed about incidents)

1. Find all active incidents and store them in *incidentsActiveList*.
2. Find all active users and store them in *usersActiveList*.
3. For each active incident from *incidentsActiveList*:
 - a. For each active user in *usersActiveList*:
 - i. Calculate the distance from current incident
 - ii. If user is in the range of the defined disk, add user to the list *usersNearIncidentList*
 - b. Copy all users from *usersNearIncidentList* to final list *usersNearIncidentFinalList*
 - c. For each user of *usersNearIncidentList*:
 - i. For each active user from *usersActiveList*:
 1. Calculate the distance from current user
 2. If user is in the range of the defined disk and is not in the lists *usersNearIncidentList* or *usersNearIncidentFinalList* add this user to the list *usersNearIncidentListTemp*
 - ii. Copy all users from *usersNearIncidentListTemp* to final list *usersNearIncidentFinalList*
 - iii. Clear list *usersNearIncidentList* and move users from list *usersNearIncidentListTemp* to *usersNearIncidentList*
 - iv. Clear *usersNearIncidentListTemp* list
 - d. Repeat step 3.c until max number of steps is reached.

To maintain the reliability of VNCS due to continuous connection and disconnection of the users, the VNCS tracking algorithm runs at very regular intervals.

3.5 VNCS demonstration

VNCS is accessible via the URL <https://snf-864568.vm.oceanos.grnet.gr/>. Below, we provide a detailed description of the functionalities offered by VNCS.

When visiting VNCS, users are prompt to register or log in to the system (Fig. 1).

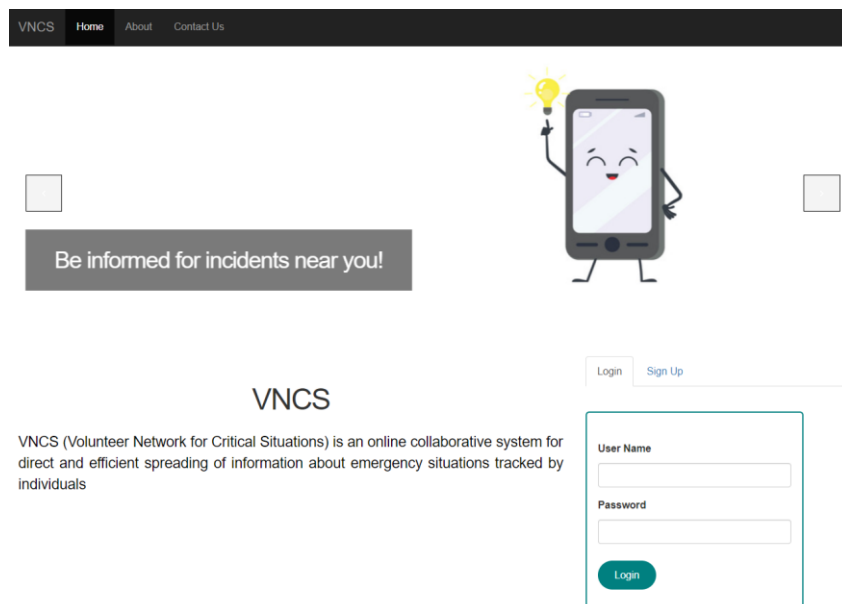


Fig. 1. VNCS homepage.

For registering, a user enters username, password and a valid email and clicks 'Sign up'. For signing in, a user is prompt to enter username and password. In case wrong credentials are provided, an appropriate message is displayed. When a successful login is confirmed, the user can have access to VNCS

functionalities. VNCS User interface is a simple responsive website composed of 6 webpages. Each one contains a header, a body and a footer.

The header contains the main menu which consists of the following options: 'Home', 'Track me!', 'Add Incident', 'My incidents', 'Live Map!', 'About', 'Contact' and 'Logout'. A notification icon is also placed on top corner of the webpage, which is triggered every time the system decides to inform a user about a critical event nearby according to the VNCS algorithm. The VNCS homepage (Fig. 2) contains two buttons, namely 'Track Me' and 'Add a new incident'.

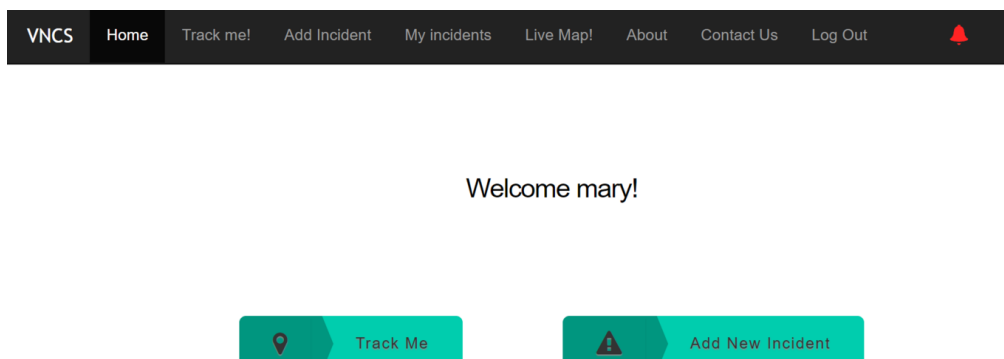


Fig. 2. VNCS user homepage.

By clicking 'Track Me', users declare their location in order to inform the system about their geographical coordinates. Then, users are redirected to the corresponding webpage where a map and a form are displayed. By clicking the symbol on the right top of the map, the system tracks the user location and a marker is placed on the map. The location is stored in the VNCS database by clicking 'Submit' and then the user is redirected to the VNCS homepage where a confirmation appears.

By clicking on 'Add a new incident', users can upload a critical event in order to spread the information and ask for help (Fig. 3). Again, users must specify the incident location and complete a form providing details about the detected incident. By clicking 'Submit' a record regarding the incident is stored in the VNCS database while the user is redirected to the VNCS homepage receiving a confirmation message.

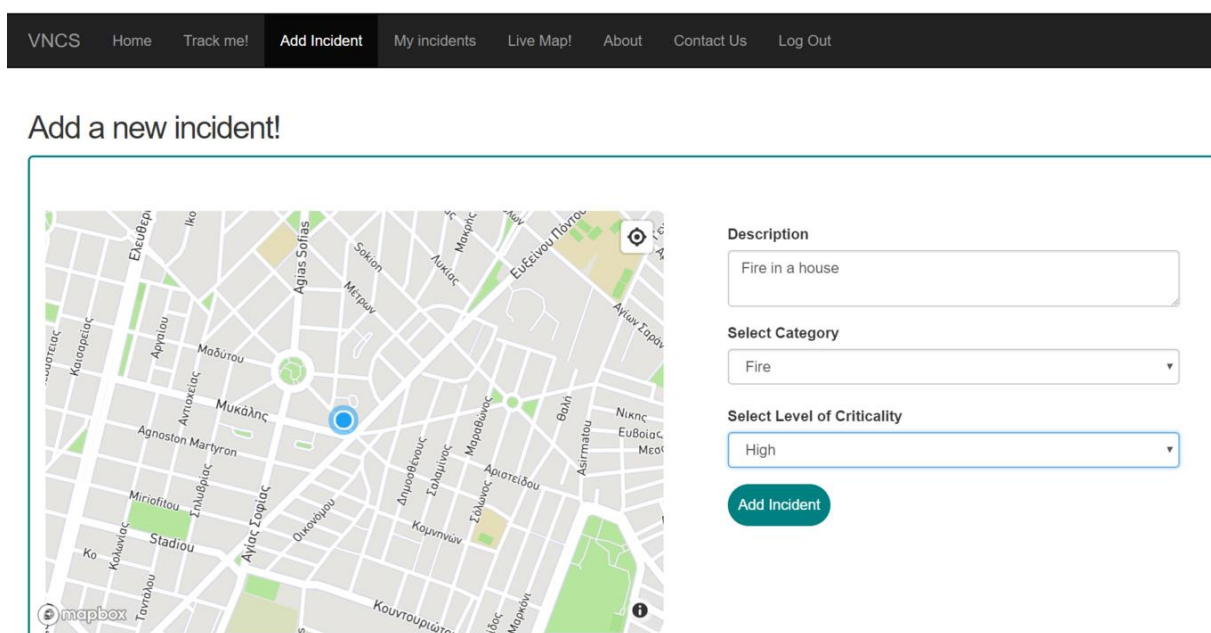


Fig. 3. Submitting a new incident.

When the VNCS notification algorithm indicates that a user must be notified about an incident within range, a notification is triggered. According to the number of events tracked within a user's range, a number appears at top of the notification icon. By clicking it, a list containing the description and the datetime for each incident appears. By clicking on a notification, users are redirected to a webpage where detailed information about the corresponding critical event is displayed (Fig. 4). The webpage contains a map where a marker indicates the location of the incident; additional details like description, category, level of criticality, starting date, submitting user and distance from the incident are also provided. Furthermore, users are asked to declare whether they can offer assistance or not. In case users have already informed the system about their actions regarding a particular incident, a corresponding message is also displayed.

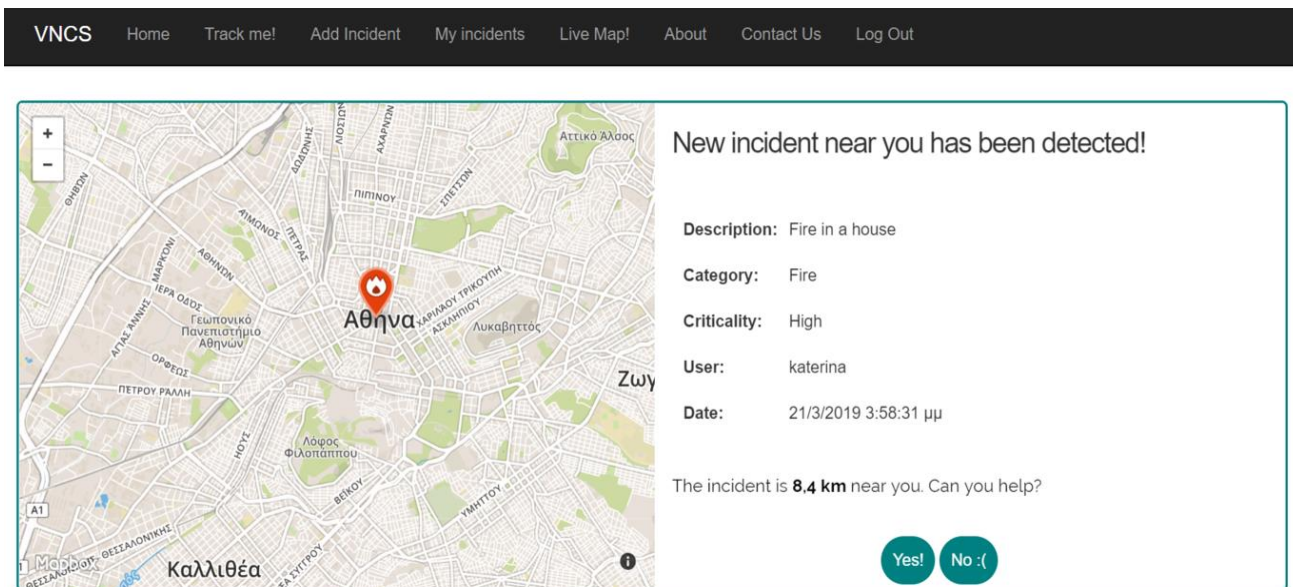


Fig. 4. By clicking on a notification, users are redirected to a webpage where detailed information about the corresponding critical event is displayed .

By clicking on the menu item 'My incidents', users can view or manage submitted events. A full list of event activity is displayed on the corresponding webpage. By clicking on the 'Show on map' button, users can track the progress of submitted incidents. Then, users are redirected to the incident webpage where markers on a map indicate the coordinates of the incident location, the locations of users who accepted to help and their distance from the incident site (Fig. 5).

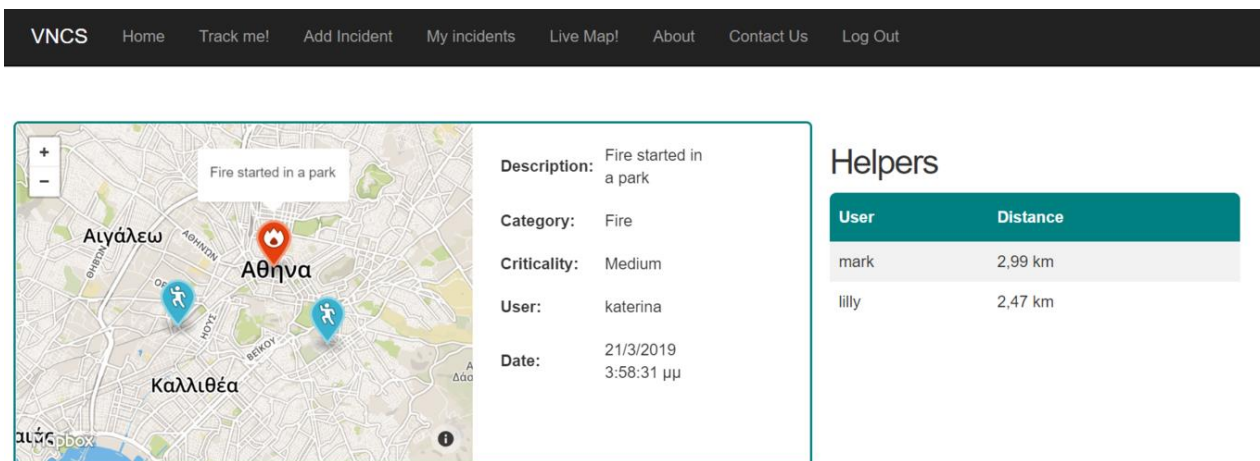


Fig. 5. Viewing users who offered to help.

In case no user has accepted to offer help yet, an appropriate message is displayed.

By clicking on 'Expire', users can declare an incident as "expired"; in such a case, the status of the incident in the VNCS database is immediately updated, as well.

An option 'Live Map!' is also available on the header menu. The corresponding webpage contains a map for visualizing the current state of VNCS. Locations of active incidents are indicated by red markers. Icons indicate the category of each active incident. Users can click on them, in order to view the description of each incident. Active users who accepted to offer help also appear on the map indicated by blue clickable markers containing their username (Fig. 6).

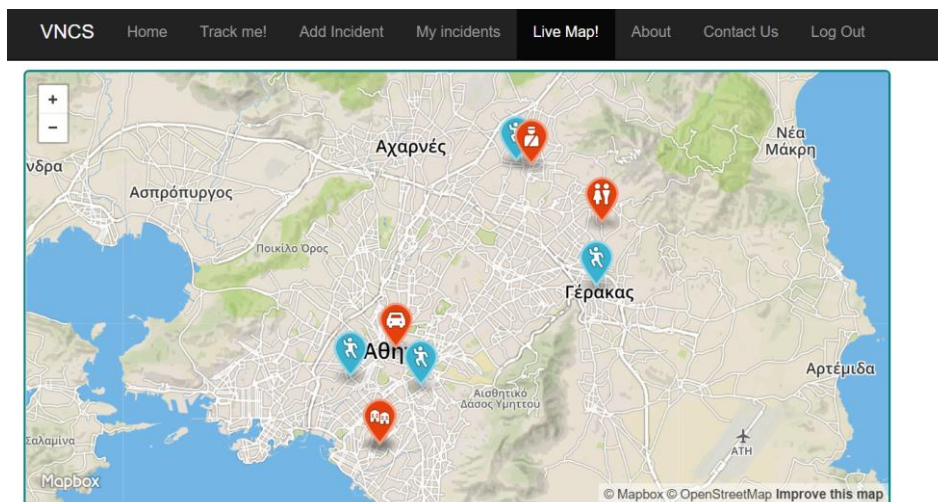


Fig. 6. VNCS Live Map snapshot.

Furthermore, via 'About' and 'Contact Us' options, VNCS users and visitors can be informed about the VNCS development team and contact VNCS administrators. By clicking 'Log out', users log off the system. Then user status is updated from 'online' to 'offline' in the VNCS database and the login webpage appears.

4 CONCLUSIONS - FUTURE PLANS

Motivated by the recent destructive wildfires in Attica, Greece, we suggest VNCS, a Volunteer Network for Critical Situations, which is an online collaborative system for immediate channeling and efficient diffusion of information about emergency situations based exclusively on the contribution of people / citizens located within a small range around the area where an incident occurs.

Our ongoing and future plans include testing VNCS in real-world situations and evaluating it on real data.

VNCS clearly demonstrate how advances in technology and communications can be exploited in order to strengthen social solidarity and social welfare. Furthermore, VNCS makes a practical solution, an online social service, for supporting citizens to turn awareness into action during critical situations prior to the intervention of special government agencies and mechanisms.

REFERENCE LIST

- Favre, C., Abdelkader Zighed, D., Guille, A., Hacid, H. (2013). Information Diffusion in Online Social Networks: A Survey. ACM SIGMOD Record, 42(2).
- Olinsky, A. D., Quinn, J. T., Quinn, C. J. and Quinn, M. J. (2016). How Information Spreads in Online Social Networks. Advances in Business and Management Forecasting, 11.
- Hermkens, K. and Kietzmann, J. H. (2011). Social media? Get serious! Understanding the functional building blocks of social media. Business Horizons, 54 (3).
- Rivest, R. L., Stein, C., Cormen, T. H. and Leiserson, C. E. (2009). Introduction to Algorithms. The MIT Press.