Appendix F – User Manual

1 Introduction

This document describes how to run and test the developed product. The project consists of two standalone applications (web app and mobile app). Thus, it is required to run two applications simultaneously on the same device (See Appendix E for Source code). The source code directory contains two folders (FirstAidMobile and FirstAidWeb). One holds the source code for mobile application and the second has source code for web application.

2 Requirements

To test the project, it is required to have two devices. A computer that supports modern web browsers and a smartphone device (Android or iOS).

The computer needs to have Node.js installed which can be downloaded here.

The smartphone requires Expo Client application. The application for iOS can be downloaded from here. The application for Android can be downloaded here.

3 Running the applications

(See Appendix E for Source code)

To run the mobile application, open the terminal or command line and navigate to */FirstAidMobile* directory.

Run the following command:
```
Juris-MacBook-Pro:FirstAidMobile juriilmjarv$ npm install
```

This will install all required libraries and packages required for this project.

Run the following command to start the project:
```
Juris-MacBook-Pro:FirstAidMobile juriilmjarv$ npm start
```

This will automatically open a new window in the browser that should look like **Figure 1.** When the window appears, take the smartphone device and scan the QR code using the camera. This will automatically load up the project within the Expo Client mobile application. Next step is running the web application.
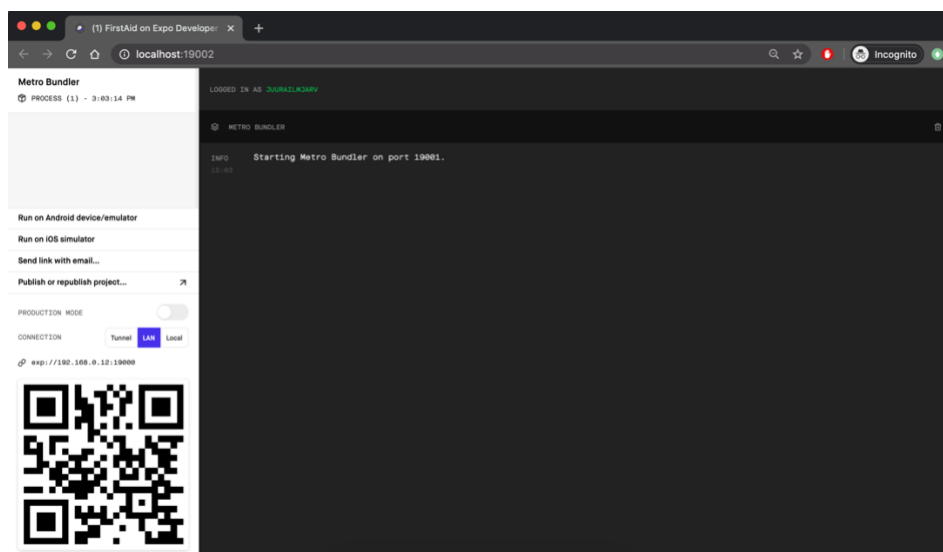

Figure 1: Starting point of mobile application

To run the web application, open new terminal or command line window (don't use the same one) and navigate to */FirstAidWeb* directory.

Run the following command:

```
Juris-MacBook-Pro:FirstAidWeb juriilmjarv$ npm install
```

Similarly, this will install all required packages for the web application.

Navigate into */FirstAidWeb/functions* directory and run the following command again:

```
Juris-MacBook-Pro:functions juriilmjarv$ npm install
```

This will install all necessary packages for the Cloud Functions. Once all the packages are installed, the project is ready to run.

Go to the root directory of the project and run the following command:

```
Juris-MacBook-Pro:FirstAidWeb juriilmjarv$ npm start
```

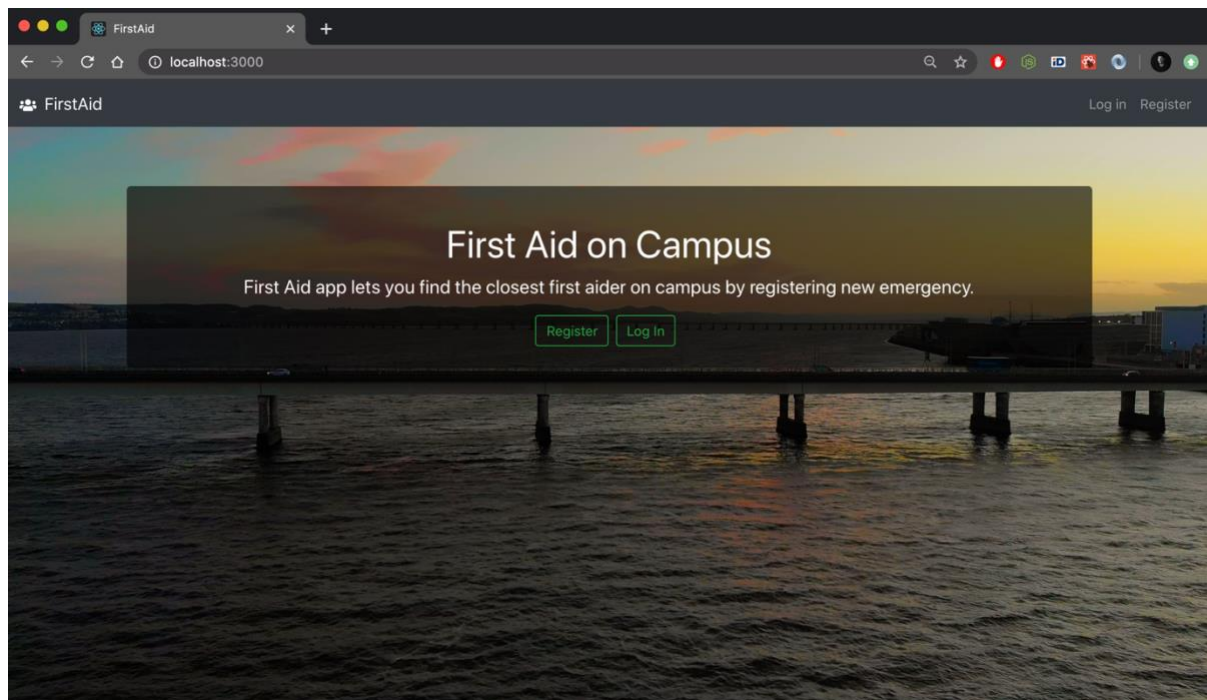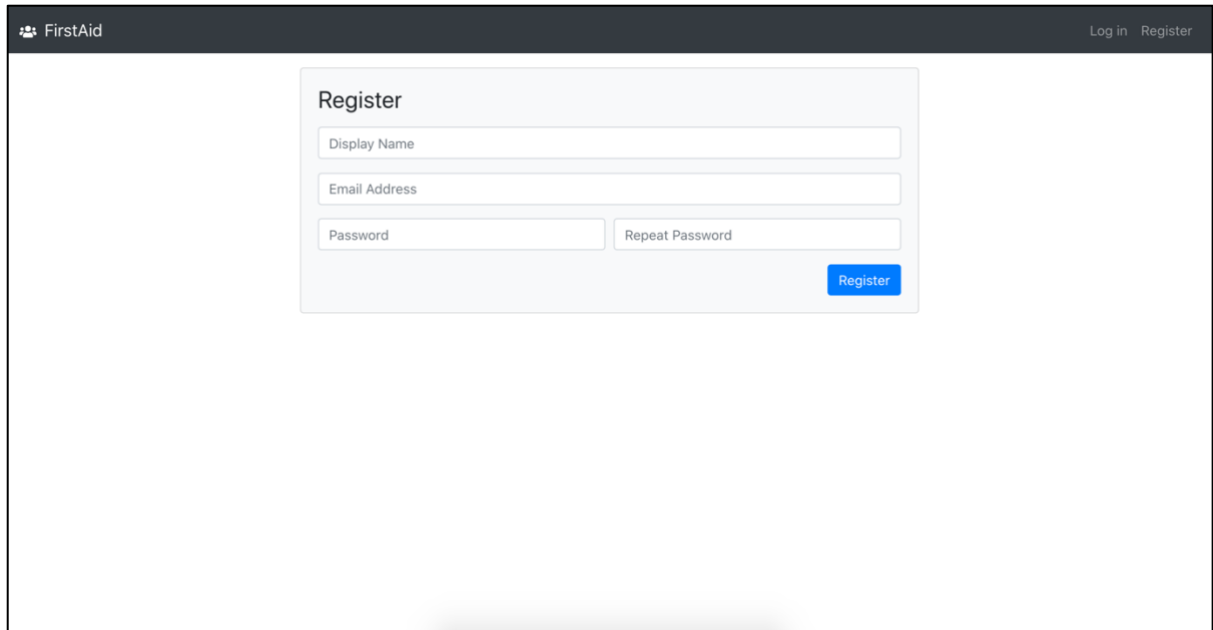The project should automatically start and open the landing page inside the default browser (**Figure 2**).

Figure 2: Web application landing page

4 User Manual

This section demonstrates the contents of the whole system and guides on how new emergencies can be registered.
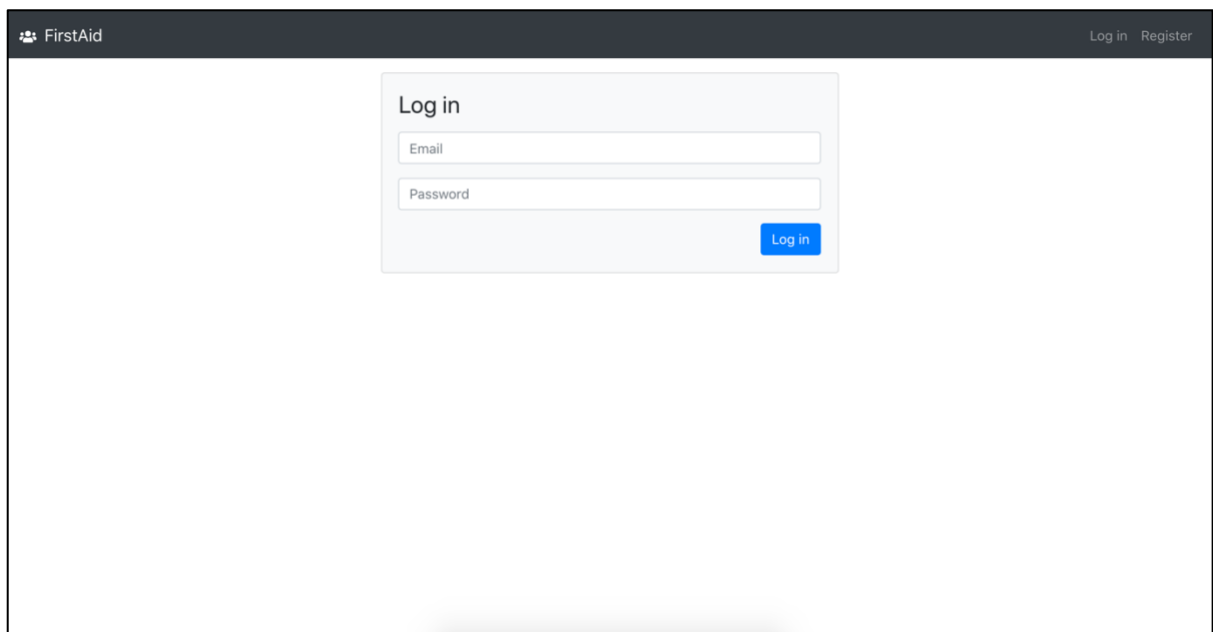
4.1 Web application

On the web application, the landing page **(Figure 2)** has options for logging in or registering. To register new account, click 'Register' button and it will navigate to registration page **(Figure 3).** The registration page requires the user to insert name, email address and password that needs to be at least 6 characters long.

Figure 3: Web app registration page

Once the registration is completed, navigate to the log in page and insert email and password that was used during registration process **(Figure 4).**

Figure 4: Web application log in page

After logging in, the web application will navigate to the new emergency page and reveal all the other pages in the navigation bar **(Figure 5).**

Figure 5: Web application new emergency page

Try navigating to different pages. The aiders page shows the information about every first aider who is registered and their availability status **(Figure 6).** The live tasks page shows the current events that are being executed by the first aiders **(Figure 7).** The completed tasks page shows the list of the tasks that have been completed **(Figure 8).** Try to interact with the completed tasks table.

Figure 6: Web application aiders page

Figure 7: Web application current events page

Figure 8: Web application completed tasks page

4.2 Mobile application

Open the mobile application and register new account. Click on 'Sign In With Google' button and choose your preferred Google account **(Figure 9).**
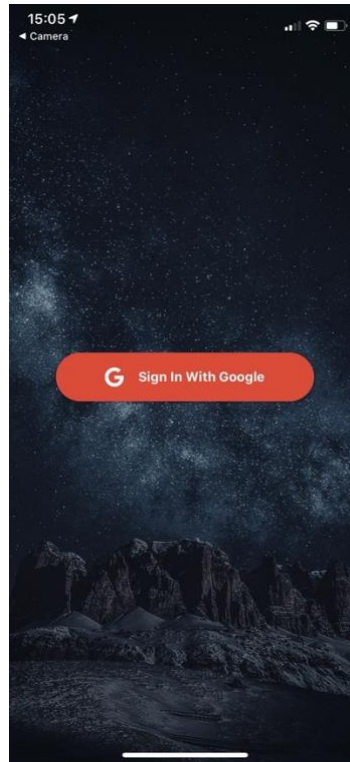
Figure 9: Mobile application sign in page

If the registration is successful, the app will navigate to the dashboard screen where the first aiders can set their availability (**Figure 10**). Set yourself as available so that the system can start tracking your location. At this point the app will ask the user for permissions to send notifications and track the location. Both of those requests have to be allowed. Also, try to navigate to aiders page on the security staff web application **(Figure 6).** It should display the account that was just created.

Figure 10: Mobile application dashboard screen

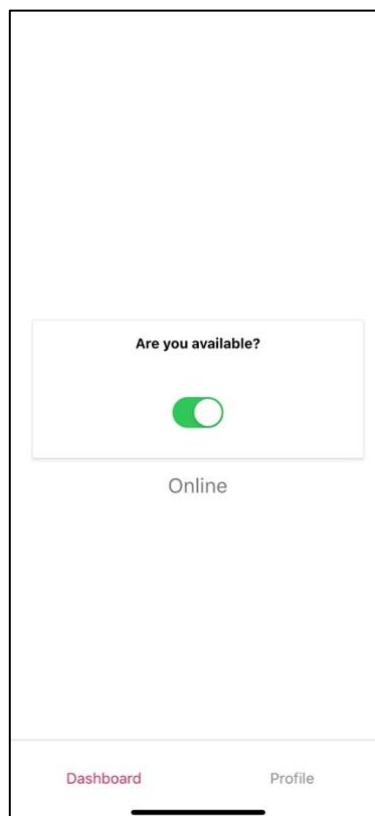Try navigating to profile page by clicking 'Profile' button on the navigation bar that is situated at the bottom of the screen. The profile page shows the information about the registered account and allows the first aider to set their telephone number **(Figure 11).** Try to submit new telephone number. This page is also used for signing out of the app.
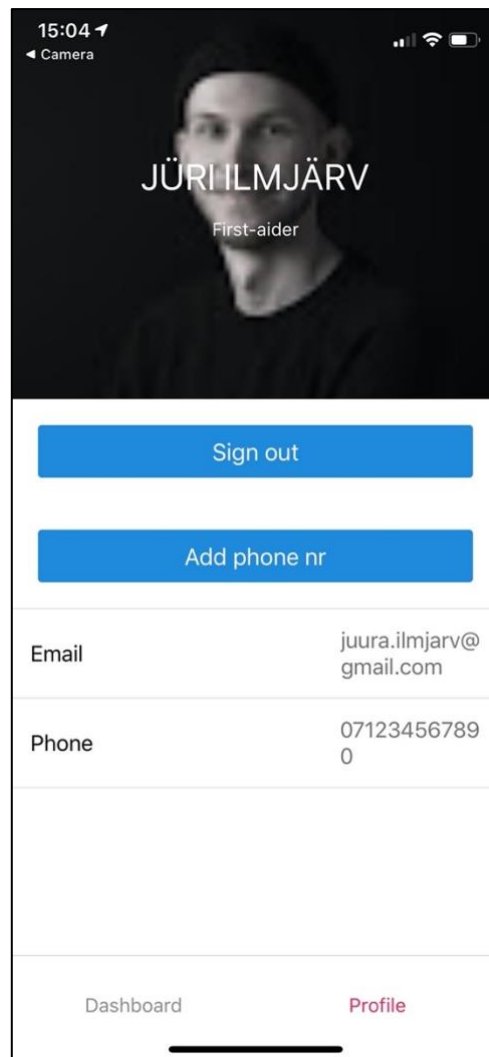
Figure 11: Mobile application profile page

4.3 Registering new emergency

Once both applications are running and the accounts have been registered, we can try and register new emergency. Make sure that your availability status on mobile application is set to online. On the web application navigate to the new emergency page and insert all the necessary information for submitting new emergency **(For example like in Figure 12).** If you are situated in Dundee city, try to set the location address within that city.

Figure 12: Web application new emergency page with sample input data

Once you click the 'Submit' button, the web app will navigate to live tasks page and display current registered emergency **(Figure 13).** If you click on 'Actions' button, it will give you multiple options to manually manipulate the event. You can choose to send notification to the same aider again, find the second closest first aider or complete the task yourself. There is no point in trying to find second closest first aider if you only have one mobile device running. If you wish to test this feature, please run the same app on the second device as well.

Figure 13: Web application live tasks page showing new registered emergency

After the new emergency is registered on the web app, the smartphone should get a push notification **(Figure 14).**
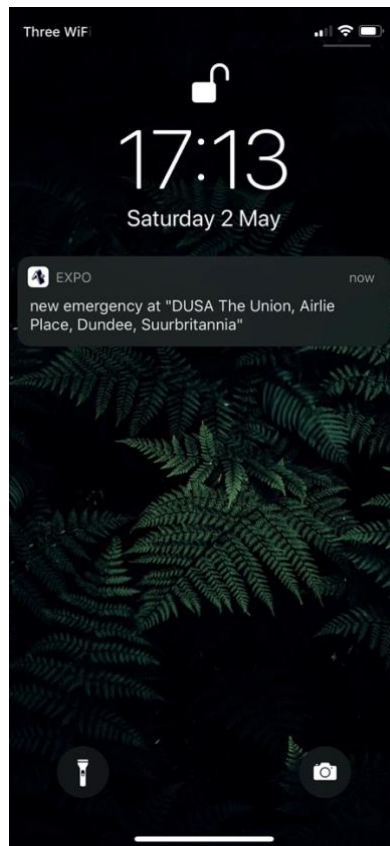
Figure 14: Incoming notification about new emergency

Once the notification is opened the mobile application will display the screen with all necessary information about the emergency **(Figure 15).**
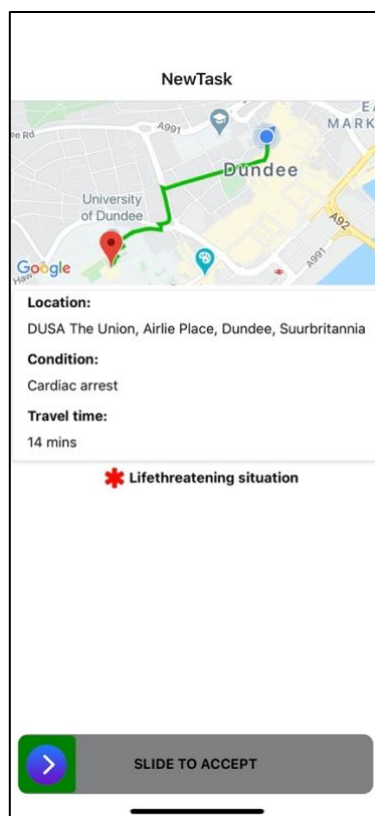

Figure 15: Mobile application displaying information about new emergency

Try to accept the task by sliding the slider on the bottom of the screen. The status on the web application live tasks page will be updated and show the details of first aider who has accepted the task **(Figure 16).** The mobile application will be redirected to the next page where more information about emergency will be shown **(Figure 17).** Try to press 'Call person on site' button. If the 'Open in maps' button is pressed, the smartphone will open default map application and automatically show directions to the emergency location.

Figure 16: Web application live tasks page showing the successful allocation of the emergency

Figure 17: Mobile application accepted task page

When you slide the 'slide to complete task' slider on the mobile app, the application will be navigated back to the dashboard where availability can be set. The web application will move the emergency information from live tasks page into the completed tasks page. Try to go into the completed tasks page and find the emergency that you just completed by clicking on the 'Full info' button **(Figure 18).**
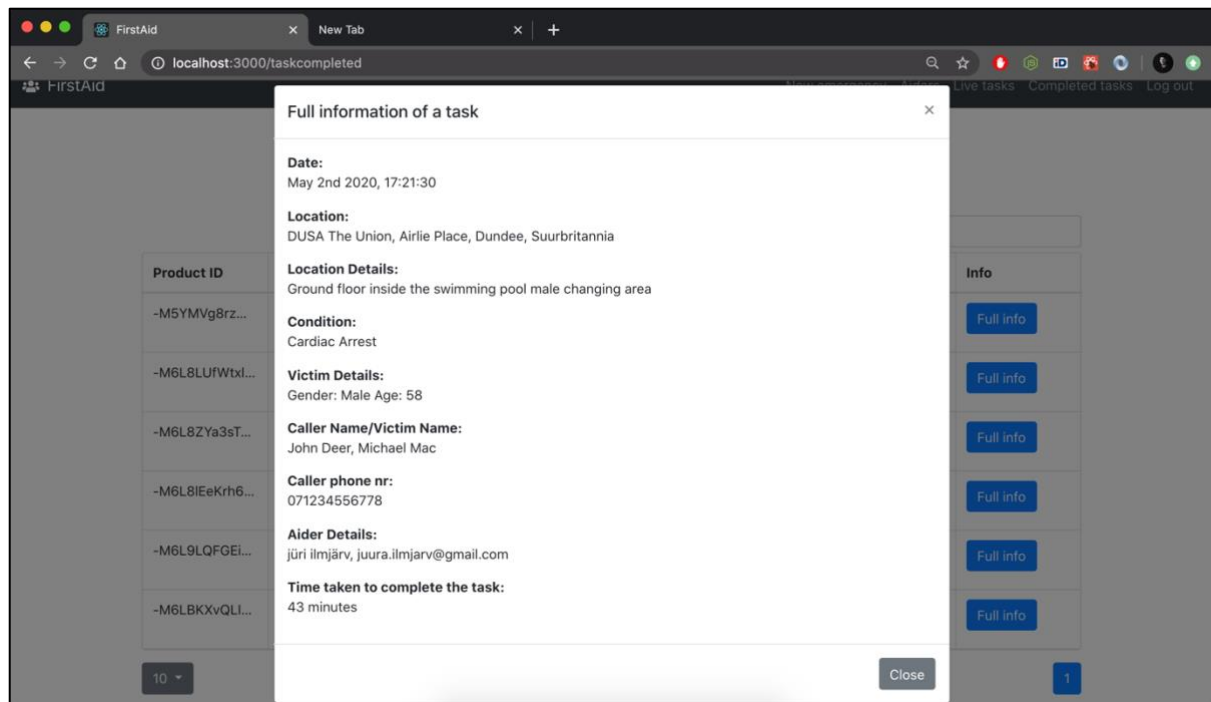
Figure 18: Completed task information on the web application

This tutorial has demonstrated how create accounts inside both applications and how to assign a new task to a first aider. There are many more use cases that could be tested. For example, if you have multiple mobile devices, try and register several tasks. Feel free to try different combinations.

5 Configuration files and APIs

Various Application Programming Interfaces (APIs) have been used throughout the project and require specific API keys for using them. This section specifies the APIs that have been used and their locations inside the source code. Also, each application holds special configuration file that is used for accessing Firebase services like database.

Currently, every required API key has been included in the source code and made active. Thus, the submitted source code is ready for running on any devices. Once the project is marked, every key will be destroyed.

5.1 Mobile application APIs

(See Appendix E for FirstAidMobile directory)

The Firebase configuration file is used for accessing the database. Those credentials can be obtained when new Firebase project is created. The configuration file is stored in */FirstAidMobile/config.js*.

For user authentication two unique OAuth 2.0 client IDs have to be created using Google Cloud platform. One is used for iOS users and the other for Android users. They can be found inside */FirstAidMobile/src/screens/LoginScreen.js* on lines 68 and 70.

When the first aider receives a notification and new task screen pops up, they are shown a map with suggested route and travel time. This is made possible with Google Directions API. The API key was created using Google Cloud platform. The key can be found inside */FirstAidMobile/src/screens/NewTask.js* on lines 77 and 210.

5.2 Web Application APIs

(See Appendix E for FirstAidWeb directory)

Similarly, the Firebase configuration file is stored inside */FirstAidWeb/src/firebase.config.js*.

11

When security staff members register new emergencies, they can choose the address from a list of suggestions that they are given. This automatically generated list is using 4 Google APIs at once (Places API, Maps JavaScript API, Geocoding API and Geolocation API). The key is specified inside */FirstAidWeb/public/index.html* on line 24.