DigLogs

User Manual and Test Cases PP4

5.2.2 Mobile Safety/Security Pilot

Responsible partner: UNITS			
Involved partners: All			
Version	Status	Date	Author
0	Draft	04/08/2021	UNITS
0.1	FINAL	05/08/2021	UNITS
Notes:			

DISCLAIMER

This document reflects the author's views; the Programme authorities are not liable for any use that may be made of the information contained therein.

Table of Contents

1. System components

The system runs using 4 physical components having the functionalities defined hereinafter:

1. **Server**: A Raspberry Pi (Ubuntu OS) where the Backend application runs, receives the user commands through a web browser, provides the sending beacons with the directions related to escape routes and stores in a MySQL database the log coming from receiving beacons related to the connected Bluetooth devices enabling their visualization through a web browser;

2. **Sending beacons**: receive from the server through WiFi connection the information that shall be exposed in the considered evacuation scenario. They transmit the information through Bluetooth to all the devices in the area covered by the beacon signal. For each message and each sending beacon, the signal strength can be selected to configure the system for the specific operative environment and evacuation scenario;

3. **Receiving beacons**: search for Bluetooth devices in their covered area. Transmit to the server the MAC address and signal strength of all the detected devices;

4. **Smartbands**: mobile wearable devices given to users. They are by default in a standby mode checking with a 10 seconds time interval the emergency signal exposed by sending beacons. Once an emergency signal is received, the smartbands switch to the emergency mode while vibrating. Then they show on the screen the information associated with the nearest sending beacon (if multiple signals are received, it is shown the one coming from the beacon having higher signal strength). Two information types are available (Fig. 1):

   a. Direction: an arrow indicating a fixed direction rotating accordingly as the user moves the smartband.

   b. Instruction: composed by an icon and a text

In the emergency mode, smartbands search for sending beacons with a 1-second interval. Besides, smartbands send their position to the nearest receiving beacon. The ID of the next beacons along the escape route can be added to the direction or instruction exposed by a sending beacon. Hence, if the smartband receives a new signal from a beacon not included in the foreseen escape route, the smartband vibrates and show the wrong direction instruction. Once the final destination beacon is reached, the smartband shows the related instruction and, if no other sending beacon is detected within 10 seconds, switches to the standby mode.

Symbols shown by the smartband:

 Direction (rotating)

 Turn left/right after the door

 Turn left or right, then come back

 Proceed upstairs/downstairs

 Destination reached

 Wrong direction, come back
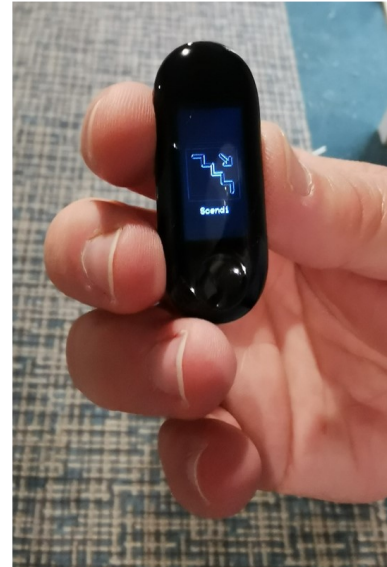WARNING! Wait the new signal

Figure 1. Directions and instructions provided by the smartbands

2. System installation

1. Formatting a micro SD card with the Raspberry Pi OS and the backend system
2. Execute the following commands to configure the system:

```
sudo -s
sudo apt-get update
sudo apt-get upgrade
sudo nano /boot/config.txt
decommentare hdmi_safe=1
sudo dpkg-reconfigure tzdata
sudo apt-get install ntp
apt-get install apache2
sudo apt install mariadb-server
sudo mysql_secure_installation
sudo apt-get -y install apt-transport-https lsb-release ca-
certificates
sudo wget -O /etc/apt/trusted.gpg.d/php.gpg
https://packages.sury.org/php/apt.gpg
sudo sh -c 'echo "deb https://packages.sury.org/php/
$(lsb_release -sc) main" > /etc/apt/sources.list.d/php.list'
sudo apt update
sudo apt-get install php7.4
sudo nano /etc/apache2/mods-enabled/dir.conf
sudo apt-get install php7.4-mysql
sudo apt install -y mosquitto mosquitto-clients
sudo systemctl enable mosquitto.service
sudo apt-get install php7.4-mbstring
sudo apt-get install phpmyadmin
sudo mysql -u root -p
CREATE USER 'etecminds'@'localhost' IDENTIFIED BY 'PaqMWFTvvN3h';
GRANT ALL PRIVILEGES ON *.* TO 'etecminds'@'localhost' WITH GRANT
OPTION;
FLUSH PRIVILEGES;
QUIT;
sudo service apache2 restart
sudo nano /etc/dhcpcd.conf
static ip_address=192.168.0.101/24
static routers=192.168.0.1
static domain_name_servers=192.168.0.1
```

3. Activate the SSH from the Raspberry Pi control panel. Then, save on /var/www/html the provided web server code.

4. Through the Phpmyadmin interface, load the provided database dump file.
5. Position the sending and receiving beacons in the desired location (according to system design).
6. Turn on the beacons.
7. Connect smartbands to pwer supply. When charged, disconnect them. Disconnected smart bands are fully operational.

3.   System testing

To check the proper functionality of the system, the following test shall be carried out in the provide order:

1. Activation of the WiFi router. Test WiFi network.
2. Turn on the server (Raspberry Pi). Check connection to the WiFi network.
3. Positioning and activation of beacons in the planned positions. Test the connection to the WiFi network.
4. If beacons are not connected to the WiFi network, position WiFi repeaters to improve the coverage. Then test again beacon connection to the WiFi network.
5. Check the proper functionality of the predefined evacuation scenarios (through a smartband). The signal power of the sending beacons can be tuned to obtain the proper behaviour of the system in each evacuation scenario.

4. System usage

The backend has 2 main functionalities: select the current evacuation scenario and visualize the data recorded during ship evacuations.

4.1. Selection of evacuation scenario

1. Turn up the Raspberry Pi and the WiFi network
2. Turn up the Sending and Receiving beacons
3. Disconnect from the power supply the smartbands (they automatically turn up in standby mode)
4. Through a web browser, connect to the IP address of the Raspberry Pi in the local network (the browser can run on the Raspberry Pi as well as on another device connected to the local network)
5. The backend will appear as in Figure 2. No escape route is selected by default and no information is exposed by sending beacons.

Figure 2. Backend in the initial status (no selected escape scenario)

6. From the drop-down menu, select the evacuation scenario among the ones available in the database (Fig. 3). As the evacuation scenario has been selected, the emergency signal will be sent. All the sending beacons start to expose the information associated with the active evacuation scenario. Smartbands switch to emergency mode. Receiving beacons start to track devices. Records are stored in the database.
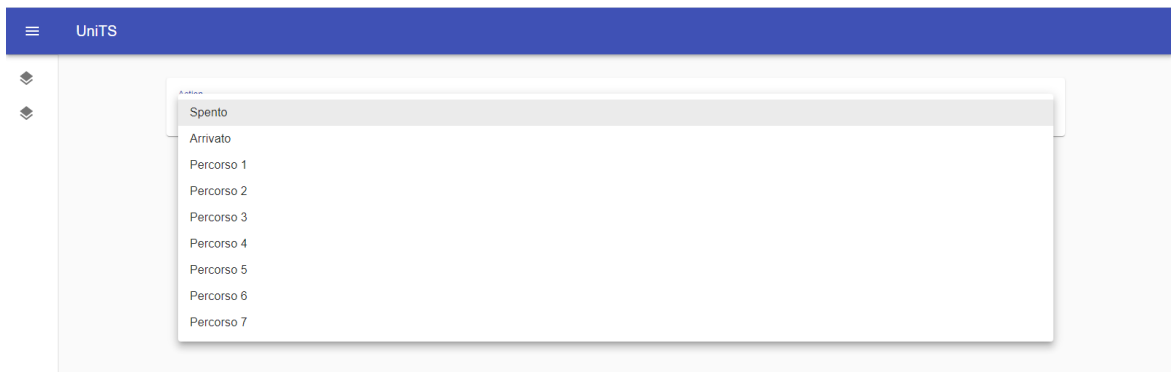
Figure 3. Backend in the initial status (no selected escape scenario)

7. After the completion of the evacuation procedure, the system can be again set on the initial status by selecting the option "Spento" (off in Italian language).

4.2. Visualize recorded data

1. Turn up the Raspberry Pi and the WiFi network
2. Turn up the Sending and Receiving beacons
3. Disconnect from the power supply the smartbands (they automatically turn up in standby mode)
4. Through a web browser, connect to the IP address of the Raspberry Pi in the local network (the browser can run on the Raspberry Pi as well as on another device connected to the local network)
5. The backend will appear as in Figure 2.
6. Through the hamburger menu (Fig. 4), select the option "History" to switch to the recorded data viewer.
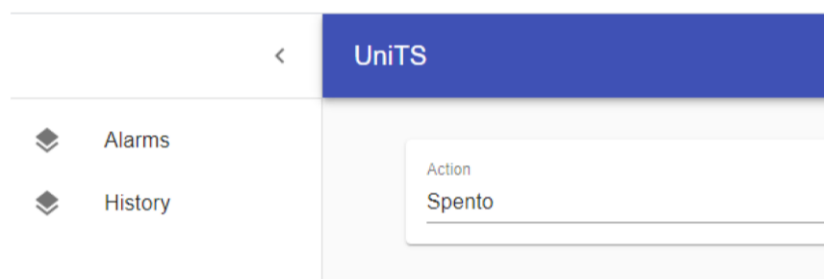

Figure 4. Backend in the initial status (no selected escape scenario)

From the dropdown menu, select the record to be shown among the ones stored in the database. The real-time record is shown on the ship general arrangement (Fig. 5). Each receiving beacon is highlighted by a blue circle, showing the number of connected smartbands.

Figure 5. Backend in the initial status (no selected escape scenario)