PRO RESCUE SOLUTION
Deepak & Sharang
23rd Sept'18
Version 0.0.1

REVISION HISTORY			
DATE	VERSION	DESCRIPTION	AUTHOR
23-sept-2018	0.0.1	Technical Specifications	Deepak & Sharang

REVISION HISTORY			
DATE	VERSION	DESCRIPTION	AUTHOR

Table of Contents

INTRODUCTION

PURPOSE

This product can be used for disaster management and relief purpose. Pro Rescue project is based on an end to end solution to prevent and control the disaster situations.

INTENDED AUDIENCE AND PERTINENT SECTIONS

This product is useful for all the users across the globe for prevention from disasters like fire hazards, water bourne hazards and chemical and gas hazards. This project is also very useful for rescue operations team to monitor disasters and control its widespread effects.

PROJECT SCOPE

This project can be used in real time environment to prevent natural and man-made disasters. It provides a highly compatible and effective solution to limit the calamity caused by disasters.

REFERENCES

https://www.openimpulse.com/blog/products-page/product-category/flame-sensor-module-2/
https://robu.in/product/water-level-sensor-depth-detection-water-sensor-arduino/?gclid=CjwKCAjw85zdBRB6EiwAov3RiqbmyLqX73pKu2bF9lbXPdrMuihcvU6o_QvgF44oQIjOYhpmyGBT0RoCev0QAvD_BwE
http://wiki.seeedstudio.com/Grove-Gas_Sensor-MQ5/
https://www.cnx-software.com/2015/04/18/nodemcu-is-both-a-breadboard-friendly-esp8266-wi-fi-board-and-a-lua-based-firmware/

DESCRIPTION

PRODUCT PERSPECTIVE

The idea of this product arises from the fact that at the time of a disaster, a user should know details and location of nearby safe areas, rescue teams, people ready for help, as well as, a rescue team should know the location and contact details of people who need help or rescue and monitor the performance of their own teams.

FEATURES

Hardware:

Fire Flame Sensor:

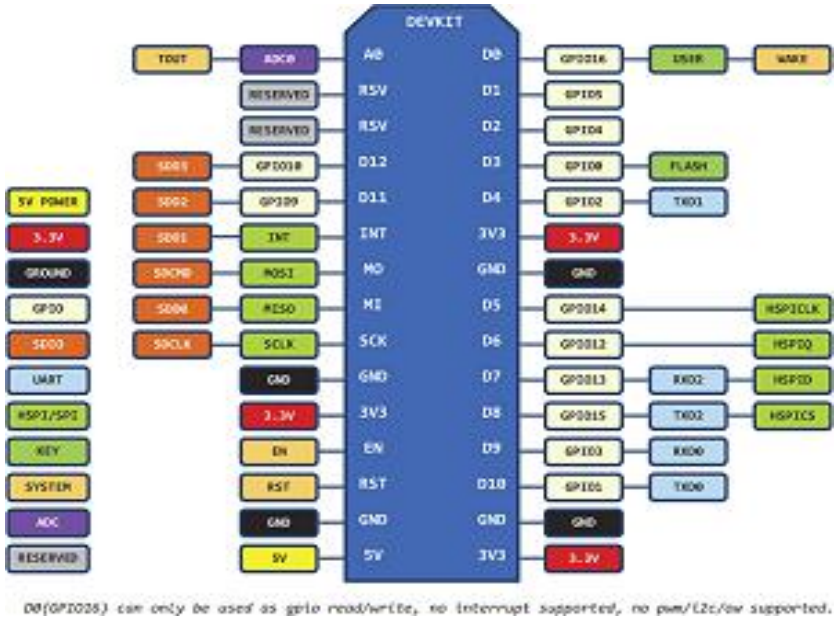
- Sensitive to flame spectrum.
- Features wide range voltage comparator LM393
- Adjustable sensitivity
- Signal output indicator
- Spectrum range 760nm~1100nm
- Detection angle 0-60°
- Power 3.3V-5.3V
- Operating temperature: -25°Cto85°C
- Can be used in fire detection, fire fighting robot and fire alarm.

Water Sensor:

- Working voltage: 5V
- Working current: <20mA
- Interface: Analog
- Low power consumption
- High sensitivity
- Working temperature: 10°C to 30°C
- Rainfall detection
- Tank overflow detection
- Liquid leakage

Gas Sensor:

- High quality dual panel design
- Onboard LED indicator
- Dual signal output: Analog and TTL level
- Reliable stability and long life
- Adjustable sensitivity via potentiometer
- Input voltage: 5V
- Current: 150mA
- Detects liquified petroleum gases



NodeMCU:

- 240MHz dual core Tensilica LX6 microcontroller with 600 DMIPS
- SRAM: 520kB
- Integrated 802.11BGN HT40 WiFi transceiver
- Flash memory: 16MB
- Operating voltage: 2.2V to 3.6V
- Onboard PCB antenna/IPEX connector
- Operating temperature: -40°C to 125°C
- Oscillator: 32kHz crystal
- 32x GPIOs, 3x UARTs, 12x ADC, 2x DAC and 3x SPI
- Supports WEP/WPA/WPA2 PSK/Enterprise
- Max data rate: 150Mbps
- Deep sleep current: 2.5A

Software:

Backend:

APIs:

- POST /postSensor HTTP/1.1
  Host: disastermgm.eu-gb.mybluemix.net
  Content-Type:text/plain

- POST /getSensor HTTP/1.1
  Host: disastermgm.eu-gb.mybluemix.net
  Content-Type: application/json

- POST /signup HTTP/1.1
  Host: disastermgm.eu-gb.mybluemix.net
  Content-Type: application/json

- POST /api/authenticate HTTP/1.1
  Host: disastermgm.eu-gb.mybluemix.net
  Content-Type: application/json

- POST /userHelp HTTP/1.1
  Host: disastermgm.eu-gb.mybluemix.net
  Content-Type: application/json

NodeJs server on IBM Bluemix:

- http://disastermgm.eu-gb.mybluemix.net

MongoDB:
- mongodb://<dbuser>:<dbpassword>@ds161102.mlab.com:61102/disastermgm

Frontend UI:
- HTML,CSS: For building UI
- Angular6: Frontend logic
- Charts.js: Library for charts and graph
- Agm maps: Library for google maps, with markers and directions

CODE OVERVIEW

Pro Rescue App:
- App-component: Menu bar and page navigation comes from app component
- Pages:
  - Home: login and user signup page
  - Mock-disaster: To show fire spread disaster alert based on mock data from sensor
  - Need-help-details & need-help list: It shows the types of help a user can ask for at the time of disaster or post-disaster like food, accomodation, medical treatment, police, firefighter, relief materials etc.
  - Overview: This is the landing page after login. It shows the safe assembly areas near the user based on location and the direction to the safe place using Haversine Algorithm, mock-disaster alert, need help, rescue me and help others. All the data when filled is posted to Bluemix server which in turn gets stored in MongoDB.
  - Provide-help: It shows a form where a user who is willing to help others can select the types of help he/she is willing to provide. This data is posted to API endpoint with the user data which can be accessed by other users to find the necessary relief materials.
  - Rescue-me: This page is most important at the time of disasters where a user can select rescue me in emergency situations. It would then ask the user if he/she is alone, if anyone is injured and if there are old, children or disabled with user, which will then show the nearest rescue team location along with contact. User can also opt for instant rescue where nearest rescue team is found on the map and direction is shown to the nearest rescue team.
  - Rescue-monitor: This page has the mock-details of rescue team which will be shown upon rescue me.
  - Chatbot-page: This page has chatbot support based on machine learning and uses designflow token and algorithm for training datasets.
- Components: Login and Signup components call /authenticate and /signup APIs of NodeJs Bluemix server to post details of signup and fetch details of login for authentication. Message-form, message-item and message-list are the components for providing chat support which is based on Machine Learning whose data is getting trained from designflow.

- Providers: user-handler is a service class which is used to get and set the data which is usable in different components and pages of the code.

Admin Portal:
- Dashboard: This page has live disaster updates based on mock-data, rescue stats, nearby rescue team locations and notifications/updates for the rescue team.
- Analysis: It has plots of the sensor data values coming from on-field sensors for monitoring fire, water and gas sensors graph data.
- Maps: This page shows the maps of locations of rescue requests, fire affected areas, water submerged areas and gas explosion or leakage affected areas.

OPERATING ENVIRONMENT

Hardware setup:

- Connect NodeMCU A0 to sensor A0.
- MCU Vcc <-> sensor Vcc
- MCU GND <-> sensor GND
- Using microUSB connect NodeMCU with PC
- In Arduino IDE run unzipped code from NodeMCU code.zip
- Now reset the NodeMCU and open serial monitor to see the output of data being posted.

Download the project zip file. Open the terminal and go to the project folder:
->unzip ProRescueSolution.zip

Open the project location:
->cd ProRescueSolution

- To run Admin portal:
  ->unzip AdminPortal.zip
  ->cd AdminPortal
  To run angular projects one needs to install node.js: Get node binaries and .exe windows executable file from : https://nodejs.org/en/download/
  Install node.js by running .exe file.
  Download node modules by:
  ->npm install
  Install angular-cli:
  ->npm install -g @angular/cli
  Now run the Portal using angular/cli:
  ->ng serve
  Open the browser and hit link: localhost:4200
- To run Prorescue App:

1. In development mode:
   ->unzip ProrescueApp.zip
   ->cd ProrescueApp
   To run ionic projects one needs to install node.js: Get node binaries and .exe windows executable file from : https://nodejs.org/en/download/
   Install node.js by running .exe file.
   Download node modules by:
   ->npm install
   Install ionic:
   ->npm install -g ionic cordova
   Now run the Portal using ionic:
   ->ionic serve
   Open browser and hit the link: localhost:8100
2. Run apk:
   There is an apk of the app attached with the project.
   Install the app in an android phone or tablet.