EC 544 Project

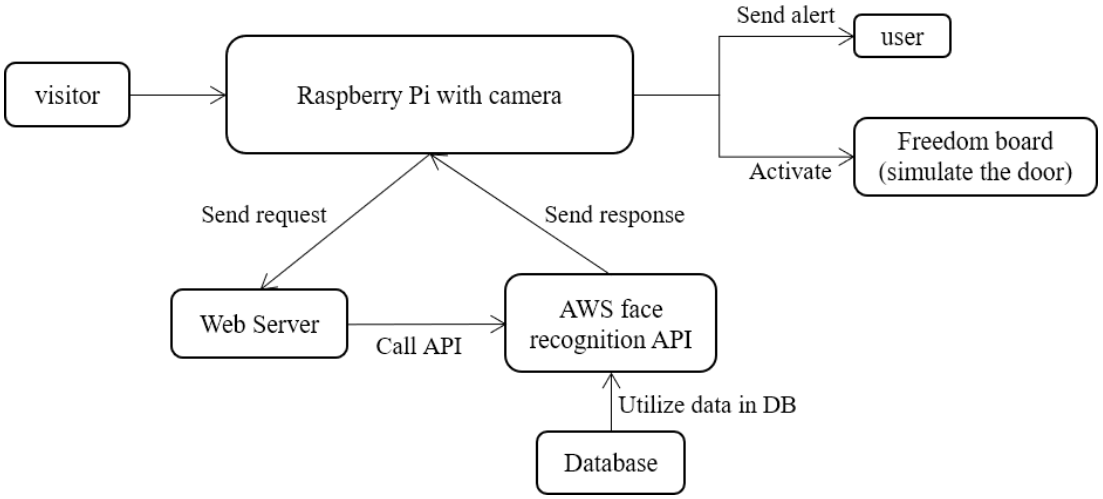
Smart Doorbell

Team Member	Email Address
Qianhao Liulin	linqianh@bu.edu
Ziyu Zhao	zepher@bu.edu

1. Introduction

In the context of the Internet of everything, the conception “smart home” has penetrated into all aspects of people's life. For example, in the field of Intelligent Security closely related to consumers, intelligent security devices represented by smart cameras have become an important support in family security activities. As a very important part of family security, intelligent doorbell undoubtedly plays the role of "frontier sentry", adding a security guarantee to the family door.

We want to build a smart doorbell for modern family which is capable of automatically open the door for friends and family members and send alerts to the user if their home is visited by strangers. Our idea is that use a embedded system to capture and preprocess the photo of visitor, then send the photo to a web server which will classify whether this visitor is permitted to enter, if so, activate the door, if not, record this visit and send alert to the user, and the user could proceed with further operations such as call the police or simply keep the door closed.



2. Resources

1) Hardware
   a) Raspberry Pi 2 Model B
   b) Raspberry Pi Camera Video Module
   c) FRDM k64F board(for door simulation)
2) Software
   a) Linux Ubuntu LTS 16.04
   b) Python 3.7
   c) C/C++
   d) Javascript
   e) CSS

f) HTML
g) Database
h) VSCode
i) PyCharm
j) MCUXpresso IDE

3) Web Server
a) AWS
b) Facebook Account
c) Heroku web app host service

3. Technical Risk Areas & Risk Management

1) Image caption and processing using camera and Pi board, fail to configure camera and Pi
2) Image quality is too low for classification
3) Image classification using AWS face recognition API, fail to effectively to classify input image
4) Fail to construct webserver to handle requests and send response to fulfill the requirement
5) Fail to accomplish communication between Pi, freedom board, and webserver

4. Technical Approach

Linqianhao: Hardware part and board programing

1) Basic configuration:
a) Connect PI board to the Internet
b) Confirm that the PI board can pass data to the server through the network and receive instructions from the server
c) Purchase a suitable camera to match the PI board. Here we choose raspberry PI camera video module 5 mega pixel 1080p Mini webcam sensor ov5647. Confirm that the camera can work normally and connect to the PI board
d) Connect PI board and freedom board to ensure that the latter can receive the former's instruction
e) Confirm that the PI board can send messages to the system owner's portable terminal via the network

2) Realization of function program:

The program of PI board is written by pycharm in Linux system to achieve the following goals：

a) Be able to record and transmit the recorded content to the server in real time
b) Be able to accept the instructions from the server and turn on or off the video recording function
c) Only uploaded videos within 20 seconds after the moving objects appeared
d) After receiving the server's return instruction:
i. If target is an acquaintance or friend, send an unlocking command to the freedom board, and send the verification message to the system owner's portable terminal
ii. If target is a stranger you don't know, send a warning instruction to the freedom board and return a warning to the system owner's portable terminal

3) Frdm k64f board is planned to be programmed by mcuxpresso ide to achieve the following goals:

a) 1 The default is to lock the door (replaced by LED red light display)
b) 2. After receiving the unlocking command, unlock and automatically return to the locked state after 10 seconds (replaced by LED green light display)
c) 3.After receiving the warning command, send a warning to the person outside the door (replaced by LED yellow light display), and recover after 5 seconds

Ziyu: Web Server Configuration

1) Functions of our webserver:
   a) Receive request from Pi board, which contain one image or multiple images of visitors
   b) Create, read, update and delete data in database
   c) Call AWS face recognition API to match image from request with images in database, analysis response from AWS API, then integrate information and send final response back to Pi, post visit record on the front-end website and send alert (email) to the user
2) Build web server and web application
   a) Use Python3 as server-side programming language
   b) Use HTML, CSS and Javascript to construct simple front-end website to post visit records for user to review

5. MileStones

1) Configure the camera and Pi to capture, preprocess, and store the images
2) Construct the webserver which is capable of handling requests from the Pi and process, classify received image
3) Build web app with front-end website which s capable of recording visit history
4) Make sure the server could send the response back to the Pi board
5) Make sure the server could send alert to the user through email
6) Pi board send activate signal to the door