Monash Home Automation

Student: Hong Minh Phan 27379914

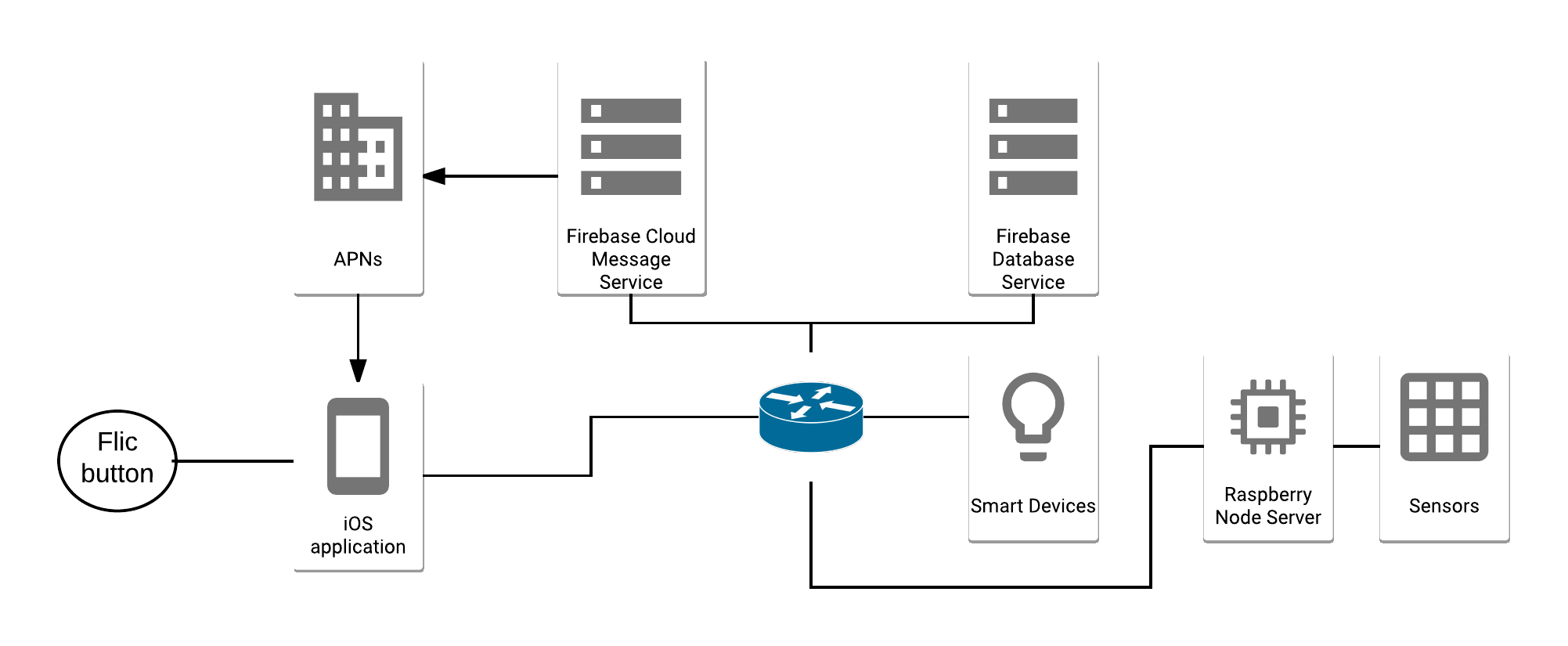
Weicheng Cheng 27906353

# Abstract

Monash Home Automation is a system created for transforming your normal home into smart home with combination of devices such as lights, power sockets, heaters … and sensors include gas/ smoke sensor and motion sensor to help you manage your home.

This document describes how the system is designed and implemented. How each component in the system can work individually and combine with others.

# Overall design



There are 6 main components has been included in this system:

* Raspberry Pi Model B: act as platform for running a backend server, which is constructed on NodeJS and many other libraries. This backend server duty is receiving requests and responding to iOS application client. It is also a bridge between all sensors (Gas sensor, Motion sensor and Temperature sensor) and backend server via GPIO or I2C connectors. Based on values which are captured from sensors, the server will make decisions like turn off the heater or push a notification to user
* Sensors: there are 3 different types of sensors include in the system. Gas/ smoke sensor and Motion sensor will be used for monitoring the home when the user away from it. The last sensor is temperature sensor will be used to monitoring the current temperature of the room, based on values captured from the sensor, the server will turn on or off heater to make sure the temperature will be kept in the defined range
* iOS application client: the application will be installed on any iOS devices and act as the end user program. Every single command will be made via the application. User can setup devices, turn on or off devices via application, define range temperature and many other functionalities
* Firebase cloud service: the purpose of using Firebase in the system is storing device tokens which be used for sending remote notifications. The iOS application will save data to the cloud and the backend server will query data from the cloud
* Devices: it means that all the devices in your house can be transformed to smart devices in many ways. In this project, by using smart devices which has been built by Xiaomi, a Chinese based company, is popular with smart things; for example: smart lights, smart power sockets, smart robot vacuums… They also provide the API to help developer can communicate with the devices via function calls.
* Flic button: it is the Bluetooth button, connect directly with the phone and will be used to control lights (turn on/ off).

# Libraries in use

* Control Mi Home devices project
  + Author: Andreas Holstenson
  + <https://github.com/aholstenson/miio>
  + Purpose: this library is using on the Node server to communicate with Xiaomi devices
* rpi-gpio Node library.
  + Author: James Barwell
  + <https://www.npmjs.com/package/rpi-gpio>
  + Purpose: this library is using on the Node server to communicate with sensors via GPIO connectors
* Johnny-five Node library
  + Author: Rick Waldron
  + <https://www.npmjs.com/package/johnny-five>
  + Purpose: this library is using to controlling Barometer sensor
* Firebase Cloud Message Library
  + Author: Firebase
  + <https://firebase.google.com/docs/cloud-messaging/admin/send-messages>
  + Purpose: this library support sending push notifications to iOS application from Node server
* Firebase Real-time Database
  + Author: Firebase
  + <https://firebase.google.com/docs/database/admin/save-data>
  + Purpose: this library support saving device tokens which will be used to identify which device receive remote notifications
* Ios-boilerplate-swift library
  + Author: Anton Meier
  + <https://github.com/50ButtonsEach/ios-boilerplate-swift>
  + Purpose: this library is used on iOS application to implement flic button
* Node-raspicam library
  + Author: Linda Nichols
  + <https://github.com/troyth/node-raspicam>
  + Purpose: this library is used on Node server to control Raspberry Pi’s camera
* Paper-onboarding ios
  + Author: Ramotion
  + <https://github.com/Ramotion/paper-onboarding>
  + Purpose: this library is used to create Launch Screen