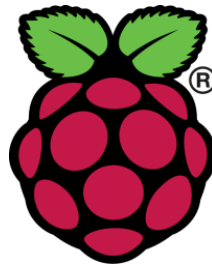


 **Institute of Technology Carlow**  
**Kilkenny Road,**  
**Carlow**  
**Bsc (Hons) Software Development**



**Home SecuriPi**

---

**Research Manual**

**Author: Karl Redmond**  
**Student Number: C00196815**  
**Submission Date: 18/04/2018**  
**Supervisor: Paul Barry**

## Abstract

*The purpose of this document is to provide an account of all the research that was carried out prior to the development of the Home SecuriPi project. The document will outline all the technologies investigated, including both hardware and software solutions, with the goal of gaining some insight into what is possible with the Raspberry Pi and similar devices.*

*Some of the material researched will not be included in the final project, but may have been researched to give an indication as to the feasibility of the project.*

# Table of Contents

---

<b>1. Introduction</b>	<b>4</b>
<b>2. Existing Systems</b>	<b>5</b>
2.1 Hobbyist Solutions	5
2.1.1 PrivateEyePi DIY Home Alarm System	5
2.1.2 DIY Smart Home Security System	7
2.2.3 Courses and Tutorials	8
2.1.4 Summary	9
2.2 Commercial Solutions	10
2.2.1 Danalock	10
2.2.2 Samsung SmartThings Home Monitoring Kit	11
2.2.3 Energenie	13
2.2.4 Bord Gais Hive	15
2.2.5 Nest	17
2.2.6 Summary	18
<b>3. Platforms</b>	<b>19</b>
3.1 Arduino	19
3.2 Raspberry Pi	20
3.3 Galileo	22
3.4 Which Platform?	23
<b>4. Pi Projects</b>	<b>24</b>
4.1 Doorjam	24
4.2 Magic Mirror	25
4.3 Electric Skateboard	25
4.4 Summary	26
<b>5. Development Environments</b>	<b>27</b>
5.1 C/C++	27
5.2 Java	29
5.3 Python	31
<b>7. Production Research</b>	<b>33</b>

7.1 Voice Communication Capability	33
7.1.1 Asterisk	33
7.1.2 Mumble	33
7.2 Dynamic DNS	34
7.3 NGINX	34
7.3 Apache	35
7.6 Twilio	36
7.7 TightVNC	38
7.8 Flask	39
<b>8. References</b>	<b>41</b>

# 1. Introduction

---

The following document will provide a detailed review of all the material that was researched for this 4th year project. The project aims to provide a home security solution, utilizing the Raspberry Pi. To embark on this journey, a clear definition of what the Raspberry Pi can achieve needed to be defined.

The document will begin by reviewing existing systems in production today, outlining a number of both hobbyist and commercial solutions.

Following this, a review of the platforms that were investigated which allow users to perform “Internet of Things” processes will be produced.

To provide the readers of this document with some insight into the potential of the Raspberry Pi, a review of three fun projects is then provided.

Finally, the languages and development environments available on the Raspberry Pi are outlined.

## 2. Existing Systems

---

Before undertaking the creation of a new application, it is important to investigate existing systems on the market. Investigation will provide insight into what our final product should provide as a minimum, and also where the current products are lacking in functionality. By carrying out this research we hope to bring something new to the home security field.

To start this research we began where most people nowadays do, Google. We will outline what the current trend is with both hobbyist and commercial implementations.

### 2.1 Hobbyist Solutions

There is a vast community of hobbyists involved with the Internet of Things, and this community is growing steadily with sales of the Raspberry Pi hitting 12.5 million in 2017[1]. Hobbyist home security solutions vary in their functionality, some have a simple sensor and camera implementation, while others can monitor door activity such as who enters and who leaves; and at what time and date this activity occurs. The sensor and camera set up seems to be the go-to project for home security hobbyists, and seems relatively easy to implement. The first project discovered was:

#### 2.1.1 PrivateEyePi DIY Home Alarm System

**Features:**

- Motion Sensors monitoring movement around doors and windows
- Receive email alerts when motion detection occurs
- Disable/Enable alarms (Fig. 1)
- View activity logs (Fig. 2)

**Skills Needed:**

- Basic low voltage electronics
- Basic Raspberry Pi and Linux

**Hardware:**

- Raspberry Pi
- Power Supply for Pi
- SD memory card

Screenshots:



Fig. 1 PrivateEyePi Home Screen

Alarms		
Date	Location	Type
11:07pm , Monday 1st April 2013	Front Door	Alarm
11:01pm , Monday 1st April 2013	Front Door	Alarm
10:18pm , Monday 1st April 2013	Front Door	Alarm
10:18pm , Monday 1st April 2013	Dining Room Windows	Alarm
10:17pm , Monday 1st April 2013	Front Door	Alarm

Logs		
Date	Location	Type
8:54pm , Tuesday 2nd April 2013	Front Door	Log
8:51pm , Tuesday 2nd April 2013	Front Door	Log
7:36pm , Tuesday 2nd April 2013	Front Door	Log
7:31pm , Tuesday 2nd April 2013	Front Door	Log
7:07pm , Tuesday 2nd April 2013	Front Door	Log
6:06pm , Tuesday 2nd April 2013	Front Door	Log
6:02pm , Tuesday 2nd April 2013	Front Door	Log
6:00pm , Tuesday 2nd April 2013	Front Door	Log
5:24pm , Tuesday 2nd April 2013	Front Door	Log
5:04pm , Tuesday 2nd April 2013	Front Door	Log

Fig. 2 Log Activity

Link: <http://www.projects.privateeyepi.com/home/home-alarm-system-project> <sup>2</sup>

While this system utilizes sensors to log activity of the motion sensors, improvements could be made by adding cameras to individual doors or windows. Initial investigation of this led to further investigation on RF controlled devices, and RF transmitter/receiver modules for the Raspberry Pi/Arduino. Also, gaining knowledge of electronic circuits became an evident requirement.

RF transmitter receivers are readily available from many online stores, and there are a number of online tutorials for intercepting and decoding RF signals. This will be a necessary component of our application if we are to allow remote entrance to the front door or perhaps the release of guard dogs. I enrolled in Lynda.com to gain knowledge of electronic circuits, and I undertook some courses as my current knowledge is null.

### 2.1.2 DIY Smart Home Security System

#### Features:

- Turn old smartphone into security camera
- Motion sensors activate camera
- On demand access to camera
- Control of camera remotely
- Remote triggering of alarm to intimidate intruders
- Push notifications

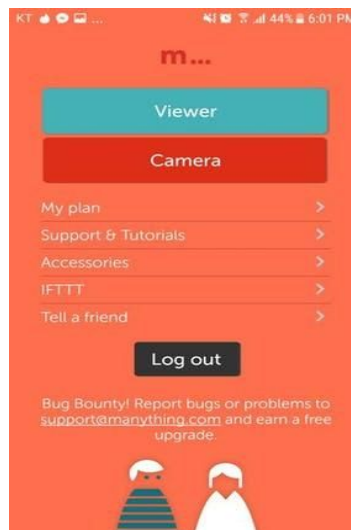
#### Skills Needed:

- Basic Tech Skills

#### Hardware:

- Raspberry Pi
- Wifi and Bluetooth dongles
- Webcam
- PIR sensor
- Old smartphone



**Screenshots:****Fig. 3** Manything App Interface

**Link:** <https://www.hackster.io/naran-inc/diy-smart-home-security-system-for-50767d> <sup>3</sup>

This tutorial shows the hobbyist how to either use an old smartphone to set up a security camera, or a Raspberry Pi with a webcam and PIR sensor to do the same. The application sends a notification to the user once motion is detected. They can then ask for a report, which sends pictures to the user. The user can also set off an alarm as a means to intimidate an intruder if one is present. All of this is controlled through a web application's interface, shown in Fig. 3, called Manything, which allows the cameras to start/stop recording and control other functions also. This system is similar to what we suggest to do, however improvements could be to add voice communication, which is a goal of this project.

### 2.2.3 Courses and Tutorials

Electronics Foundations: Fundamentals, available at <https://www.lynda.com/Development-Tools-tutorials/Electronics-Foundations-Fundamentals/197537-2.html>.

Electronic Foundations: Basic Circuits

<https://www.lynda.com/Software-Development-tutorials/Electronics-Foundations-Basic-Circuits/507570-2.html>.

Decoding and sending 433MHz RF Codes with Arduino and RC switch, available at <https://www.liwen.id.au/arduino-rf-codes/>.

## 2.1.4 Summary

There are a number of DIY Home Security projects using the Raspberry Pi. Most of the projects involve single operations such as notifying a user that a motion sensor has gone off, allowing remote access to a camera over IP connection, or setting off an alarm to intimidate intruders. While these are useful in themselves, we want to achieve a one-stop solution to these things, as well as making improvements to what is currently available for the DIY enthusiast. Some courses and tutorials have been enrolled in, to provide additional knowledge on electronic circuits and Decoding 433MHz RF codes.

---

## 2.2 Commercial Solutions

### 2.2.1 Danalock

#### Features:

- Easy to install Smart Lock
- Bluetooth and Smart Home connectivity
- Mobile App

#### Skills Needed:

- Basic DIY skills

#### Hardware:

- Danalock device

#### Screenshots:



**Danalock V3**  
 Smartest  
 Smallest  
 Safest  
 Fastest (4 x speed)  
 Strongest (2 x power)

Danalock V3 specifications:

Product Dimensions 2 1/3" x 2 1/3" 59 mm x 59 mm	Item Weight ex. Batteries 4.6 Ounces / 130 grams	Batteries 4 CR123A batteries	Material Anodized Aluminum / ABS	Encryption level: 256bit AES
Communication: Bluetooth 4.0 BLE, Z-Wave, Zigbee and Apple HomeKit*	Average Battery Life app. 9000 cycles	Finish Anodized Aluminum	Color Silver	

\*Comes in different versions with different radio specifications. Refer to distributor for info or product details when ordering.

**Fig. 4** Danalock Specifications

**Retail Price:** \$179

**Link:** <https://danalock.com/index.html> <sup>4</sup>

This is a popular device for renters and homeowners. Created by a Danish company with all the hardware built in-house.

Taking this device on its own, it is a trivial device. However connectivity to controllers, like the Z-wave, Nest and Harmony, make it a great addition to a person's smart home. It features an auto-lock mechanism, specified in Fig. 4, which can be timed from five seconds to three minutes, and also allows remote access through an application living on an old phone, over Wifi, to allow a guest into your home. It keeps a record of who has entered, and left through

the mobile application, however, it can only keep track of registered users and doesn't indicate any forced entry.

This device could be a useful addition to our project, as we could harness the remote entry of the door, with the additional benefit of being able to see who is at our home. Further investigation is needed at this point to figure out Bluetooth programming, and to figure out can we decode the signals to the device.

### 2.2.2 Samsung SmartThings Home Monitoring Kit

#### Features:

- Door/Window sensors (open/closed)
- Motion detection/notification
- Leak detection
- Camera streaming
- Electrical socket controllers
- Thermostats
- Mobile App

#### Hardware:

- Controller Hub(hardwired to router)
- Multi sensors
- Motion sensors
- Presence sensor
- Controllable power outlet

#### Screenshots:



**Fig. 5** SmartThings Starter Kit



Fig. 6 Door/Window sensor

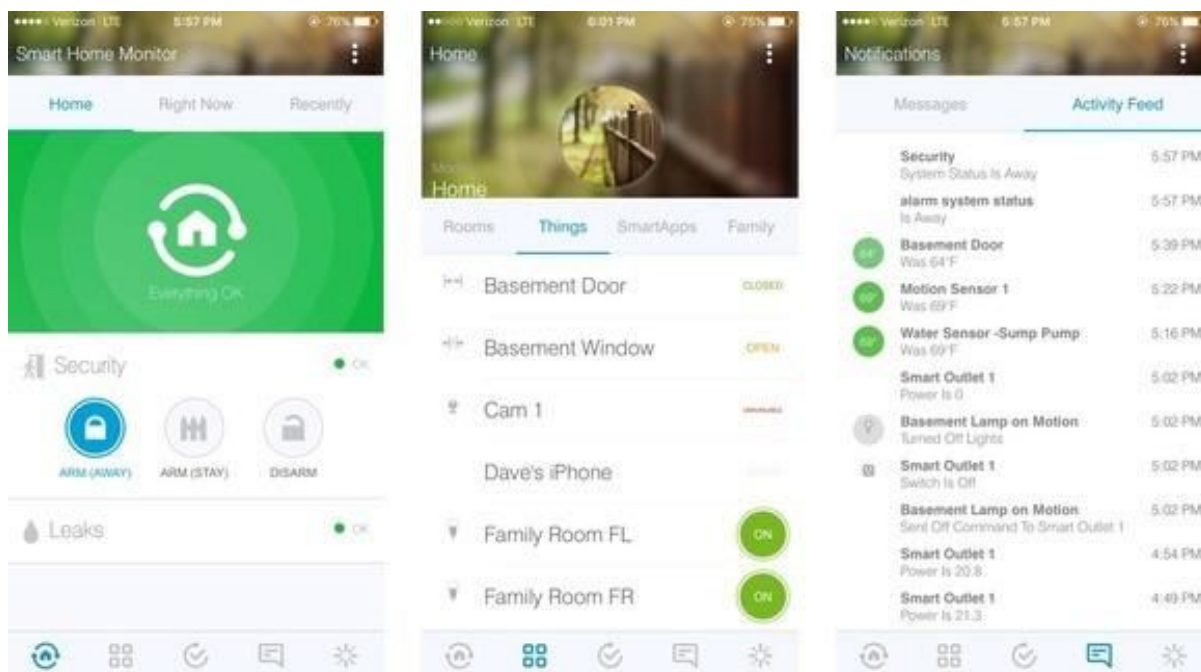


Fig. 7 Smart Home Monitor App <sup>5</sup>

**Price:** \$249

**Link:** <https://shop.smartthings.com/#!/kits/samsung-smarthings-home-monitoring-kit>

SmartThings, which was acquired by Samsung in 2014, is a home automation hub which tries to bring all of the automated parts of your home together, making them controllable from one device through a mobile application interface, displayed in Fig. 7. The device can communicate with almost every product currently available, that doesn't use a proprietary system.

The SmartThings Home Starter kit, displayed in Fig. 5, provides an excellent starter kit for the home monitoring enthusiast. While it retails at \$249, the kit includes some cool “things” to get you experimenting, including a smart socket, smart motion sensor, the controller hub and a door/window sensor, which is pictured in Fig. 6.

Samsung have really pushed the boat out, and have included an API for developers to include their own device handlers, such as camera's, moisture sensors and smoke detectors. This means that upon receiving a message from these sensors, we can have our devices interact through the use of the provided API.

This would be a great benchmark for our project, however, it may be slightly ambitious.

### 2.2.3 Energenie

#### Features:

- Motion detection
- Remotely controllable appliances
- Remotely controllable light switches
- Open sensors (doors/windows)
- Electricity usage
- Radiator control

#### Hardware:

- Central Hub
- Lighting Socket
- Relay(controlled switch)
- Power Socket
- Multi Plug
- Electricity Monitor (clamp)
- Thermostat

#### Screenshots:



Fig. 8 MiHome Accessories

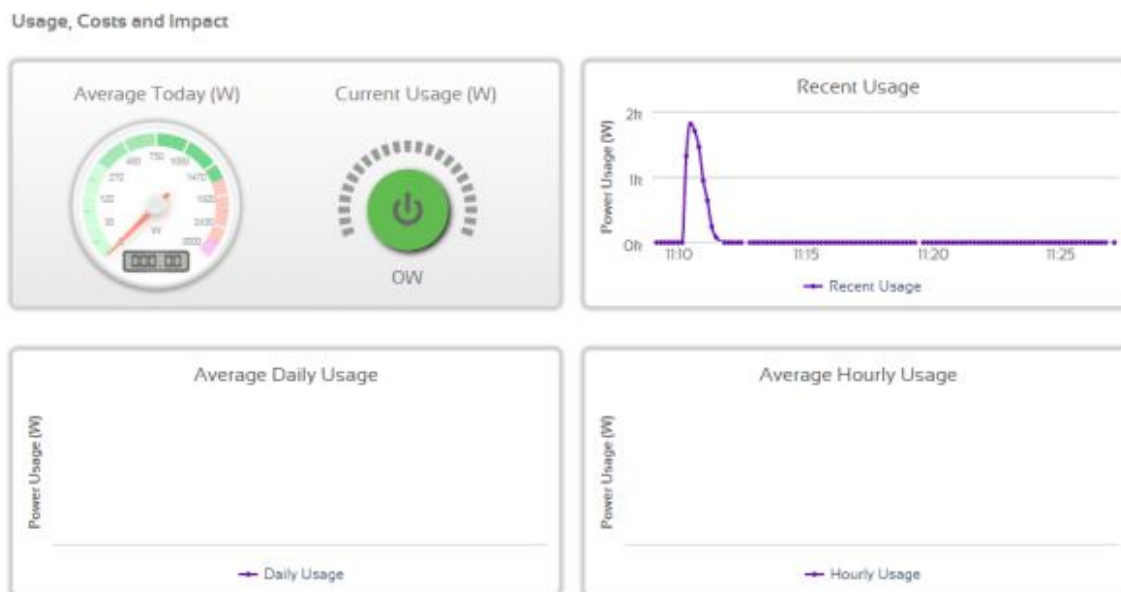


Fig. 9 MiHome Graphs

Link: <https://energenie4u.co.uk> <sup>6</sup>

**Retail Price:** Prices vary depending on what you want to do. If we are to include the controllable sockets and remote control, the price will be **\$80.00** with the hub included.

“Energenie provides an easy way of cutting the amount of energy used in homes, saving money on electricity bills, saving energy and reducing the risk of fires caused by electrical appliances” <sup>7</sup>.

Further investigation led to the discovery of the Energenie. This company have embraced the Internet of Things, and are creating Smart Home devices to automate devices and appliances around your house.

Their main goal is to reduce the amount of electricity being consumed by a household, and also to reduce the risk of fire. Energenie achieve this by providing a Monitor and Control screen on their application, which shows useful, easy to read graphs as can be seen in Fig. 9. It shows you items such as average hourly use, average daily use, recent usage, current usage and average today. When the user views these graphs, they can control a sockets output through radio frequency outputted from their MiHome central hub. The pictures shown in Fig. 8 give some insight into what these sockets look like. They have recently added some new additions to their range of products, including thermostats, motion sensors, radiator valves and LED high quality lighting.

All of these devices are controlled by a central hub which is connected to the internet over Ethernet cable. According to reviews from oxgadgets <sup>8</sup> and tech-reviews, the MiHome hub is very simple to set up, with a walkthrough provided when you register the device on their website.

We could incorporate the controllable sockets into our project, by possibly intercepting and decoding the radio frequency omitted from the remote control. This could act as a deterrent to an intruder, as turning on/off lights could mimic movement from inside the house.

## 2.2.4 Bord Gais Hive

### Features:

- Heating & Hot water scheduler
- Automatic frost protection
- Geolocation
- Holiday mode
- Child lock
- Text Control
- Temperature Graphs

### Hardware:

- Central Hub
- Smart Thermostat
- Motion Sensors
- Active Lights
- Smart Plugs
- Smart Cameras

### Screenshots:



**Fig. 10** Hive Welcome Home package <sup>9</sup>

**Retail Price:** \$299 (Hive Thermostat With Professional Installation in Ireland) <sup>10a</sup>



“Products can be linked together to make managing your home even easier, for instance you can set your smart plug linked to a kettle to turn on each morning as you walk past a sensor in your hall; or always arrive home to a warm welcome by setting your lights to switch on when you open the front door”<sup>11</sup>.

Bord Gais have recently entered the Smart Home industry, with a product from Centrica, named “Hive”. Originally intended as a way to monitor and control your central heating system in an effort to improve energy consumption efficiency, Hive has spawned into a system which now offers additional functionality such as turning on/off lights or appliances. They have also integrated geolocation capabilities on the app which can be setup to alert you when you move away from your home, to indicate that you have left the heating on.

The offering by Bord Gais of 299 euro for the thermostat, hub and mobile application may seem steep, but this comes with a professional installation, which is quoted to be completed in less than 60 minutes<sup>10b</sup>. Once this has been successfully installed, the additional features, such as plugs, lights and cameras, can be easily added to the hub via the application interface, which allows the user to simply scan for new devices, and place them on the dashboard in an arrangement to their liking.

The application also incorporates a child lock feature, which involves the requirement of a pin activation on changing your heating status. Should you wish your home to be nice and cosy upon returning from your holidays, the application allows you to set your holiday return dates to preempt the central heating system to come on.

A feature which hasn’t been previously noted in research is the ability to text the system, via SMS. Text commands include:

- HEAT STATUS - Sends you back the current mode, temperature and target temperature of your system.
- HEAT ON - Turns your heating on. Default temperature of 22 degrees, can be changed using “HEAT ON 18”. Heating will stay on until turn off manually.
- HEAT OFF - Turns your heating off.
- HEAT AUTO - Turns the automatic scheduler on.
- HEAT HELP - Returns a list of these commands.

This is a useful feature if perhaps you are out of data or internet coverage. This feature works by sending a text message to a specified number, which will update your system based on your account number.

Bord Gais claim that you can save a possible 120 euro from your heating bill each year and with an initial investment of 300 euro, it will take almost 3 years to reclaim your investment. For the frugal amongst us it should be noted that the hive can be installed easily by a reasonably tech savvy individual, for almost half the price.

## 2.2.5 Nest

### Features:

- Smart Thermostat
- Learning Algorithm
- Mobile Application

### Hardware:

- Smart Thermostat
- Heat Link

### Screenshots:



**Fig. 11** Nest Smart Thermostat <sup>12</sup>

**Retail Price:** €299.99 <sup>13</sup>

The Nest is first and foremost a heating control unit. Consisting of a smart thermostat displayed in Fig. 11, and a heat link which connects directly to your boiler that communicates wirelessly with the thermostat, the Nest aims to provide a means to reduce your heating bill from 10-13% per annum. Nest provides this functionality by adopting a learning algorithm, which learns your preferred heating settings based on past experience. It combines this with a GPS tracker on your phone, which is utilised to turn on/off your heating with respect to the distance you are from your property. For example, if you leave your house and forget to turn off your heating, Nest will turn off your heating automatically for you. Vice versa, if you are nearing your home, or are coming home from work at a given time, Nest will turn on your heating to have your house warm upon arrival.

Nest provides the functionality of setting a weekly heating schedule which can be done with the thermostat itself, or more easily through the mobile application. However, this functionality can be ignored entirely, and you can rely on the learning algorithm to figure out the optimal settings.

One major disadvantage to Nest is the need for a professional install, as it requires that the heat link be connected to the boiler, replacing its existing electronic thermostat.

### 2.2.6 Summary

There already exists quite a number of smart home controller hubs, which can be used to control devices and appliances around our home. From smart locks, which when used in conjunction with an application on our phone can open a door as we near, or vice versa as we move away, to smart sockets which record and report on the amount of electrical energy that is being consumed by a connected device.

We are in a truly fascinating time in this technological revolution, and some commercial industries have taken the initiative to get a head-start.

## 3. Platforms

---

The next logical step in our research, is to figure out what platform we should use for our home security solution. Initial investigation of existing home security systems would point towards the Raspberry Pi. However, as there are a few other hobbyist boards, such as Intel's Galileo, and the Arduino, both of which can be used for some form of home automation, we will take a close look at all three and provide a comparison.

### 3.1 Arduino

Arduino is an open-source platform used for building electronics projects. The Arduino consists of a programmable circuit board, and an IDE that runs on a computer. The IDE is used to write and upload a computer program to the board. Once the program is written, it is uploaded to the board via a USB cable. This functionality led to the Arduino's success over other programmable circuit boards, as these boards needed a separate piece of hardware, called a programmer, to load new code to the board. Additionally, the Arduino IDE uses a simplified version of C++, making it relatively easy to produce a program <sup>14</sup>.

There are five mainstream Arduino boards, including the Arduino Uno, the Lilypad Arduino, the Red Board, the Arduino Mega and the Arduino Leonardo. As the Arduino Uno is the most comparative to the Raspberry Pi, this is the board we will focus on.

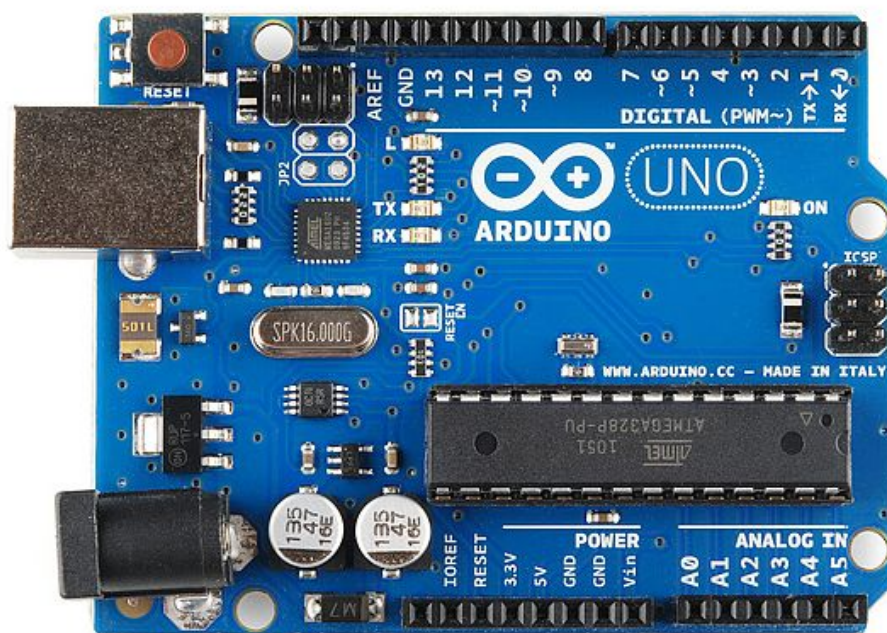


Fig. 12 Arduino Uno

**Specifications:**

- Microcontroller: ATmega328
- Operating Voltage: 5V
- Input Voltage (recommended): 7-12V
- Input Voltage (limits): 6-20V
- Digital I/O Pins: 14 (of which 6 provide PWM output)
- Analog Input Pins: 6
- DC Current per I/O Pin: 40 mA
- DC Current for 3.3V Pin: 50 mA
- Flash Memory: 32 KB of which 0.5 KB used by bootloader
- SRAM: 2 KB (ATmega328)
- EEPROM: 1 KB (ATmega328)
- Clock Speed: 16 MHz

**Price:** \$20

**Link:** <https://www.arduino.cc> <sup>15</sup>

The Arduino Uno, shown in Fig. 12, is first and foremost a microcontroller. It excels at controlling single devices, such as sensors, lights and motors. Sometimes used as a prototype for new electronic devices, if the prototype is successful, a printed circuit board would be the next step for the device <sup>16</sup>.

## 3.2 Raspberry Pi

The Raspberry Pi is a credit card sized computer originally designed for education, with the initial goal being that of creating a low-cost device, that would improve programming skills and hardware understanding of pre-university students. However, it quickly became noticed by tinkerers and electronic enthusiasts, who needed more than a basic micro controller for DIY projects <sup>17</sup>.

There are currently 6 Raspberry Pi models to choose from, ranging from respectively basic models, which need additional modules for bluetooth and WiFi connectivity, to the most recent Pi, the model 3. We will focus on the Raspberry Pi 3, as the difference in price between this model and its predecessors is minute.

The Raspberry Pi 3, displayed below in Fig. 13, boasts a 50-60% performance increase over the Pi 2 model B, which was released one year previous, and boasts a ten-fold increase in performance over the original Pi <sup>18</sup>.



**Fig. 13** Raspberry Pi 3

### Specifications:

- SoC: Broadcom BCM 2837
- CPU: 4× ARM Cortex-A53, 1.2GHz
- GPU: Broadcom VideoCore IV
- RAM: 1GB LPDDR2 (900 MHz)
- Networking: 10/100 Ethernet, 2.4GHz 802.11n wireless
- Bluetooth: Bluetooth 4.1 Classic, Bluetooth Low Energy
- Storage: microSD
- GPIO: 40-pin header, populated
- Ports: HDMI, 3.5mm analogue audio-video jack, 4× USB 2.0, Ethernet, Camera Serial Interface (CSI), Display Serial Interface (DSI)

**Price:** \$30-\$40

**Link:** <https://www.raspberrypi.org/blog/raspberry-pi-3-on-sale/><sup>19</sup>

The Broadcom BCM 2837 was built specifically for the Raspberry Pi 3 and is a powerful system-on-chip(SoC), which combines all the components of the miniature computer.

The Quad core CPU allows multi-threaded programming, and the GPIO pins allow several modules to communicate bi-directionally with the Pi<sup>20</sup>.



### 3.3 Galileo

Galileo, pictured below in Fig. 14, is a microcontroller board based on the Intel® Quark SoC X1000 Application Processor, a 32-bit Intel Pentium-class system on a chip. It's the first board based on Intel® architecture designed to be hardware and software pin-compatible with Arduino shields designed for the Uno R3. The Galileo runs the same IDE as the Arduino, and operates along the same lines, meaning writing code and uploading programs works the same way as the Arduino Uno (see 3.1).

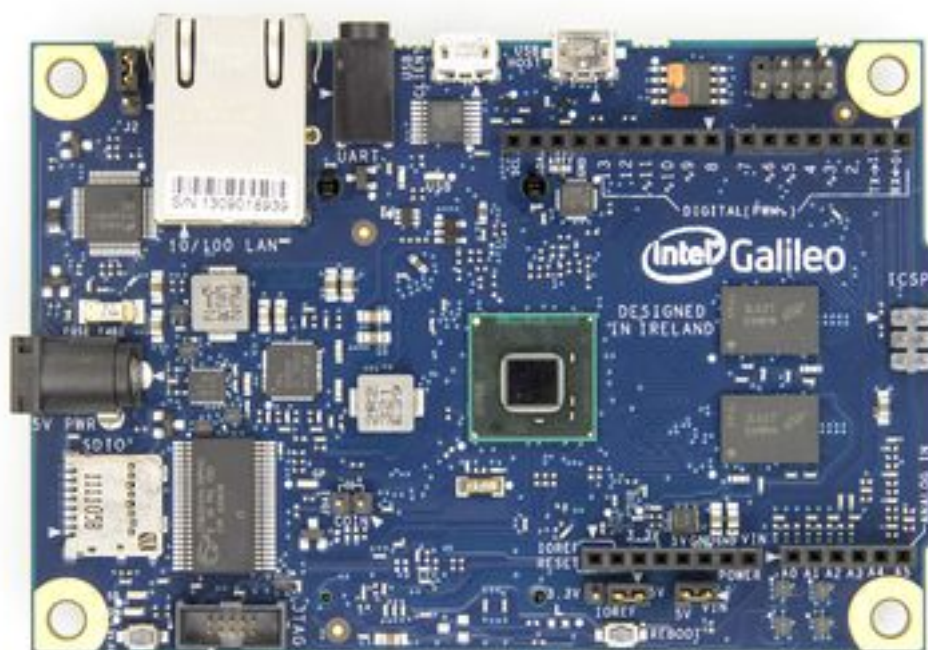


Fig. 14 Intel Galileo

#### Specifications:

- SoC: Quark X1000 @ 400MHz
- RAM: 256 MByte DDR3; 512 KByte embedded SRAM
- Flash Storage: 8 MByte NOR Flash; 8kb EEPROM
- LAN: 10/100 Ethernet (RJ-45 connector) with support for Power-over-Ethernet (PoE)
- PCI-E: Mini-PCI Express 1x Slot
- USB: USB 2.0 Host Port (Standard Type A); USB 2.0 Client Port (Micro-USB Type B)

**Price:** \$70

**Link:** <https://www.arduino.cc/en/ArduinoCertified/IntelGalileo> <sup>21</sup>

We included the Galileo in this list, as past students at IT Carlow have used the Galileo in their final year project. It is of particular interest to note that the Intel Galileo board is no longer in production, giving some insight into how quickly microcontrollers and minicomputers are growing in popularity. The Raspberry Pi and Arduino range seem to have taken control, leading to the demise of the Intel Galileo.

### 3.4 Which Platform?

With the Galileo out of the running, it boils down to the Raspberry Pi 3 and the Arduino Uno. The main difference between the two is that the Arduino is a microcontroller, used for running one process, repeatedly, whereas the Pi is a general purpose computer. Hence, you would only use the Arduino for repetitive tasks, such as sounding a buzzer when motion is detected, whereas one would use the Pi when needing more processing power, such as to control a complex robot.

It is difficult to provide a direct comparison as they are essentially two different things. One good rule of thumb when deciding whether to use the Pi or the Arduino is this; if the project can be described with less than two and's, use the Arduino, otherwise use the Pi <sup>22</sup>.

As our project will, at the very least, detect motion “and” notify a client, “and” provide video streaming, “and” open a door, “and” do all of this remotely, we definitely need the awesome power of the tiny computer, Raspberry Pi 3.



## 4. Pi Projects

---

We have already looked at DIY projects utilizing the Raspberry Pi for home security(see 1.1), the following will document other existing projects, which will highlight the versatility and power of the device.

### 4.1 Doorjam

Doorjam, an example of which is shown in Fig. 15, is a quirky project for the Pi, and one of the first you will find when you look for fun Pi projects. Created by the Redpepper agency, this project provides theme music for individuals. By making API calls to Spotify, and plugging in the Pi's auxiliary out into an old stereo, the Pi provides personalized entrance music for entire offices. The Doorjam application lives on your mobile phone, and is used to set your theme song. When you get within range of a bluetooth dongle, which is set up to sense proximate applications, your song starts playing.



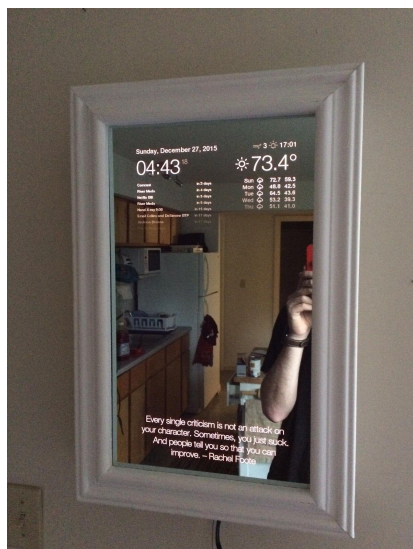
Fig. 15 Doorjam <sup>23</sup>

**Link:** <https://ideas.redpepper.land/doorjam-47f1a5bce2fd> <sup>24</sup>

The above link provides an overview and some code snippets to get started with this project.

## 4.2 Magic Mirror

Created by developer Michael Teeuw, the Magic Mirror, shown below in Fig. 16, is a high-tech mirror which shows you the weather, the time, and the day's headlines, while you are getting ready for work. By taking a one-way mirror, and attaching a display and a Raspberry Pi to the back, he was able to bring his concept to life.



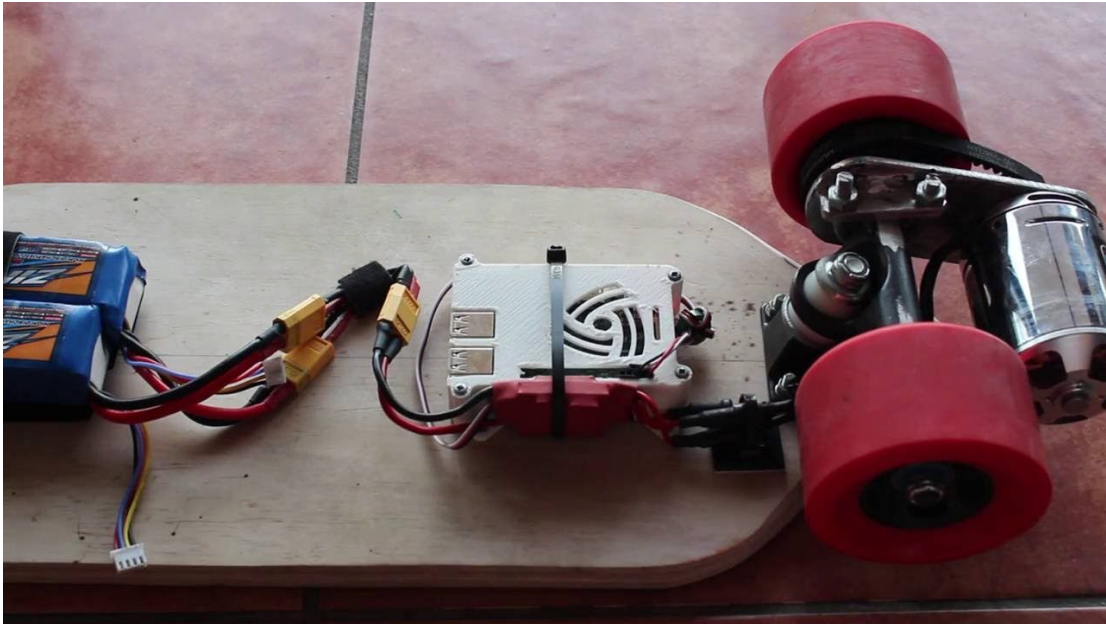
**Fig. 16** Magic Mirror

**Link:**

<http://michaelteww.nl/post/84026273526/and-there-it-is-the-end-result-of-the-magic> <sup>25</sup>

## 4.3 Electric Skateboard

The electric skateboard, shown below in Fig. 17, was imagined and created by “The Raspberry Pi Guy”(RPG). By taking a skateboard and attaching a motor, RPG was able to control the skateboard with a Nintendo Wii remote. The motor is controlled by a servo, which is sent signals from the Raspberry Pi, which receives signals from a the Nintendo Wii remote control. The skateboard is capable of reaching 30 kmph and has a range in excess of 10 km. The code for this project is written in Python, and is available on GitHub (along with a list of hardware needed) here: <https://github.com/the-raspberry-pi-guy/skateboard>.



**Fig. 17** Electric Skateboard

**Link:** <https://www.youtube.com/user/TheRaspberryPiGuy> <sup>26</sup>

#### 4.4 Summary

There are countless other projects involving the Raspberry Pi. From our office, to our home, and even to the great outdoors, the Internet Of Things will touch every aspects of our lives.

## 5. Development Environments

---

“The Raspberry Pi Foundation recommends Python as a language for learners. We also recommend Scratch for younger children. Any language which will compile for ARMv6 (Pi 1) or ARMv7 (Pi 2) can be used with the Raspberry Pi, though, you are not limited to using Python. C, C++, Java, Scratch, and Ruby all come installed by default on the Raspberry Pi”<sup>27</sup>.

As can be seen on the Raspberry Pi’s official website, there are a number of programming languages which can be used to program the device. It states that the Python language is the best for learners, however, we will also investigate Java and C/C++ before making any decisions as to which environment we will develop in. We will also document some integrated development environments(IDE) which can be used with each language.

### 5.1 C/C++

C is a very powerful language, second only to assembly, as this language speaks to the bare metal of a processor. C++ is an extension of C, introducing object orientation to the language, and mixing together with the procedural aspect of C. The difference between the two is out of scope for this document. C is amongst the most popular languages used by programmers nowadays, and is the language that Linux is written in.

Getting started with C programming on the Raspberry Pi is straightforward. The following steps highlight how simple it is to get up and running:

- 1) Download the Raspbian Wheezy SD card image from the Raspberry Pi website download page: <http://www.raspberrypi.org/downloads>.
- 2) Install it on a SD card using [win32diskimager](#) (if you are running Windows) or use the new noobs installer.
- 3) Boot your RPi, log in and start the GUI:  
<http://www.raspberrypi-projects.com/pi/pi-operating-systems/raspbian/raspbian-general>
- 4) Geany will already be installed.
- 5) If you want to control the IO pins of your Raspberry Pi install the bcm2835 library by Mike McCauley:  
<http://www.raspberrypi-projects.com/pi/programming-in-c/c-libraries/bcm2835-by-mike-mccauley>
- 5) Create a new Geany project :  
<http://www.raspberrypi-projects.com/pi/programming-in-c/compiler-and-ides/geany/creating-a-project><sup>28</sup>.

As can be seen from the steps above, the Geany IDE comes preinstalled on the Raspbian, which is the go to Operating System of the Raspberry Pi.

Geany is a text based editor with basic features of an IDE, as can be seen in Fig. 18. It is small and lightweight, making it ideal for the mini computer.

It comes with plenty of features including syntax highlighting, symbol name auto-completion, call tips, code navigation, simple project management and supported file types including C, Java, PHP, HTML, Python, Perl, Pascal and more. There is also a plugin interface for extending the IDE.

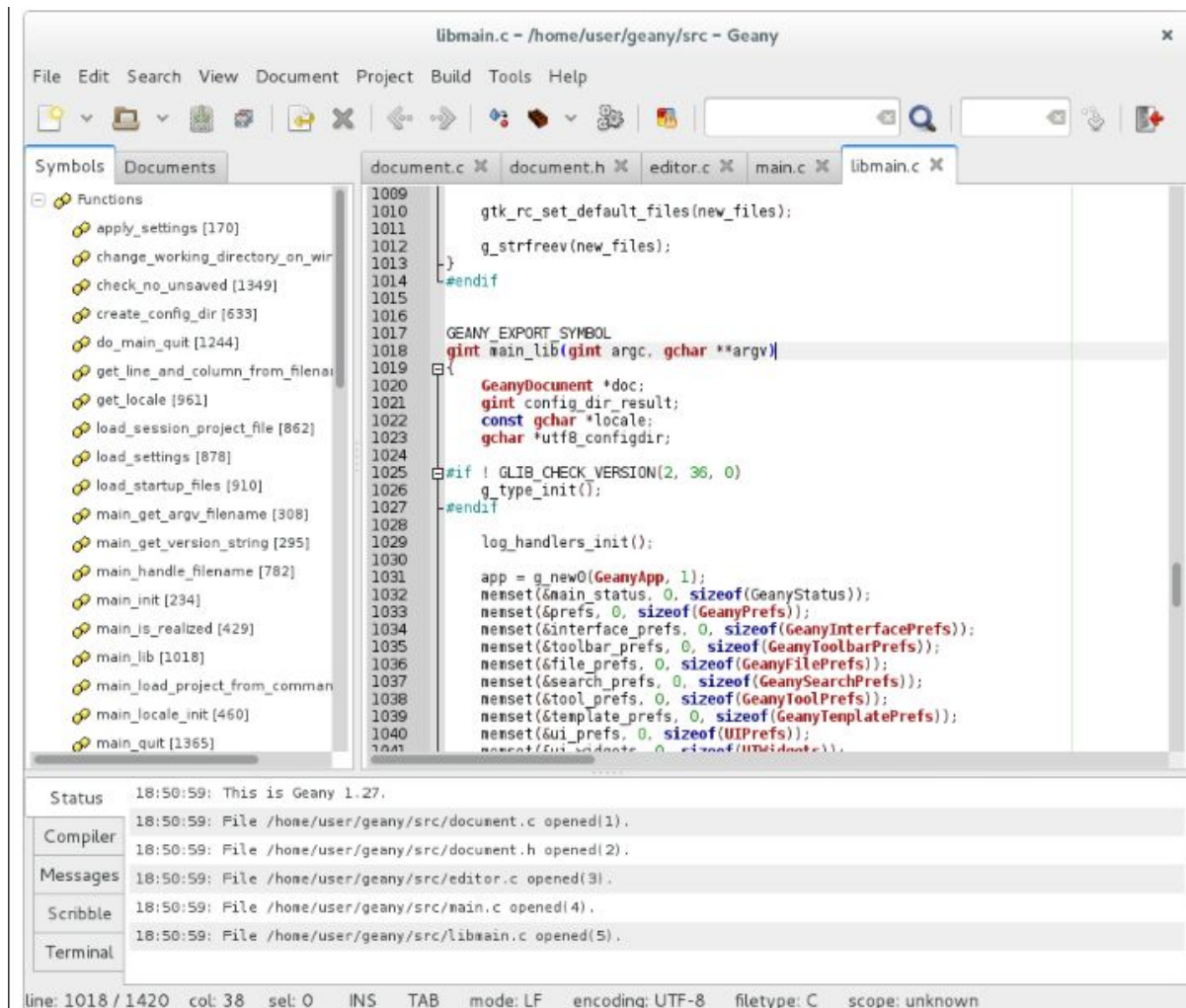


Fig. 18 Geany <sup>29</sup>





changed and experimented on. This interaction lends itself to the understanding of the object-oriented programming conception.

A useful tutorial for getting started with BlueJ on the Raspberry Pi can be found here: <https://www.bluej.org/raspberrypi/led.html>. This tutorial covers the go-to first project for any aspiring Pi hobbyist.

## 5.3 Python

“Python is a wonderful and powerful programming language that's easy to use (easy to read and write) and with Raspberry Pi lets you connect your project to the real world”<sup>31</sup>.

Python is a dynamic language, built for speed. With origins dating back as far as the late 1980's, the language borrows heavily from its predecessors like C and Haskell. Envisioned as an educational language by Guido van Rossum, to bring computer literacy to the general population, it has developed into a widely popular language used by the likes of Google and YouTube<sup>32</sup>.

There are a couple of IDE's that come preinstalled on the Raspbian operating system, IDLE and Thonny.

IDLE is Python's **I**ntegrated **D**evelopment and **L**earning **E**nvironment. Consisting of two main window types, the Shell window and the Editor window.



```
Python 3.5.2 (v3.5.2:4def2a2901a5, Jun 26 2016, 10:47:25)
[GCC 4.2.1 (Apple Inc. build 5666) (dot 3)] on darwin
Type "copyright", "credits" or "license()" for more information.
>>> WARNING: The version of Tcl/Tk (8.5.9) in use may be unstable.
Visit http://www.python.org/download/mac/tcltk/ for current information.
print("hello world")
hello world
>>> for each in range(10):
    print(each)

0
1
2
3
4
5
6
7
8
9
>>> |
```

Ln: 22 Col: 4

**Fig. 20** IDLE's Shell Window

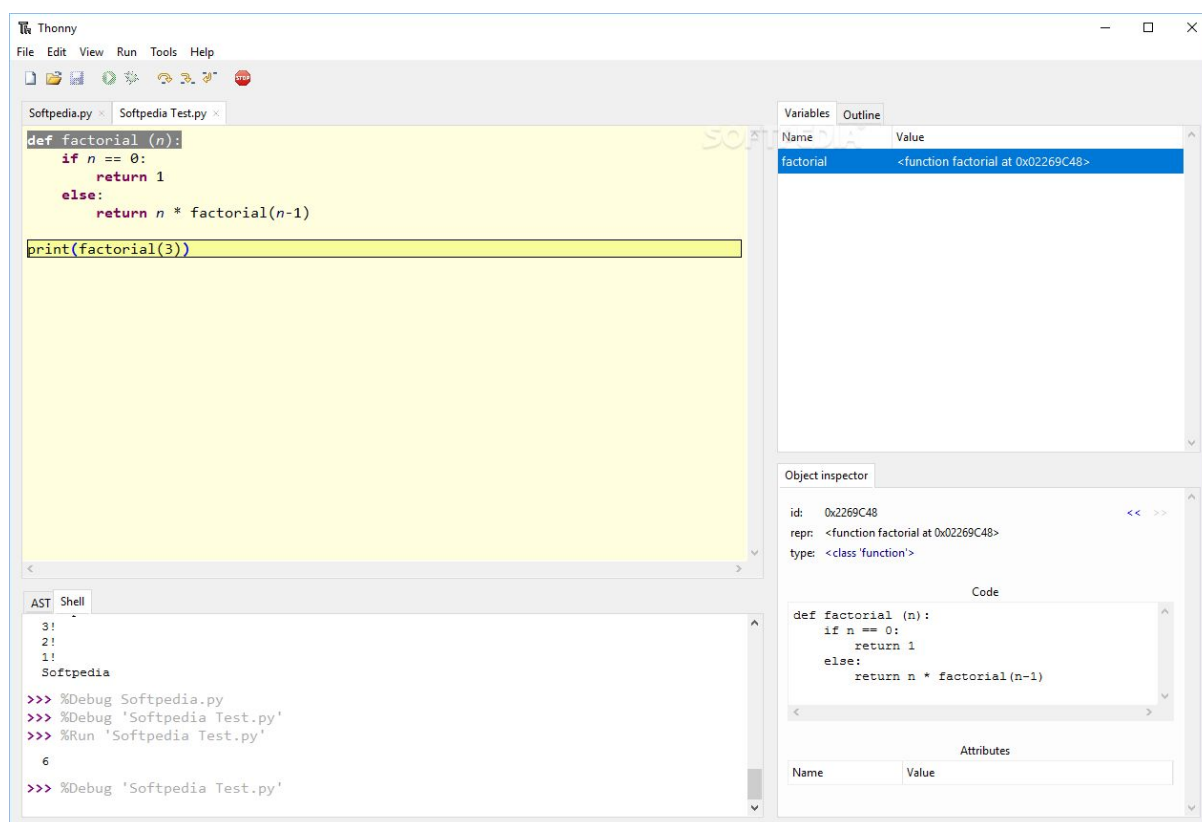
IDLE's Shell window (Fig. 20) allows us to experiment with Python commands, and let's us open an Editor window. With the Shell window the programmer can try out algorithms and



play around with different commands interactively, without having to write a complete program. In Fig. 18 we can see two simple commands, one which simply prints “hello”, and another which implements a simple for loop. This interactivity is the main purpose of IDLE’s Shell window.

As previously stated, we can open an Editor window from the Shell window, we would do this when we are ready to write a program. IDLE Editor window works the same as any other text editor, with the added ability to run the program from the menu bar, or by pressing `f5` <sup>33</sup>.

Thonny is a new IDE which comes with the latest version of Raspbian. The IDE comprises of a script editor and a shell. While this sounds the same as the IDLE IDE, the windows are merged onto the same window, and Thonny also provides some other useful features.



**Fig. 21** Thonny Interface <sup>34</sup>

As can be seen in Fig. 21, the programmer is able to inspect an object's state, and the state of any variables is also viewable. When the programmer changes these objects or variables in the shell, these states will be updated immediately, allowing for a truly interactive experience. Thonny also comes with a debugger, allowing one to step through the program line by line, and view any changes in state.

## 7. Production Research

---

As this project adopted an agile approach to production, the research was ongoing throughout development. This section will detail the technologies and languages that were researched in order to bridge any gaps or overcome barriers that were met.

### 7.1 Voice Communication Capability

Providing a means for voice communication became a distinct part of our product very early on in development. As we were in personally unexplored territory, a number of different options were explored, each being described in the following sections.

#### 7.1.1 Asterisk

Initial research into providing a functionality where an individual could communicate verbally with someone who has set off the motion sensor lead to the discovery of Asterisk.

“Asterisk is an open source framework for building communications applications. Asterisk turns an ordinary computer into a communications server”<sup>35</sup>.

There were some existing Raspberry Pi solutions which utilised Asterisk to provide some form of voice communication, such as this open source project: <https://opensource.com/article/17/4/asterisk-raspberry-pi-3> which outlines how to install and configure a phone system for a small business or home office.

It quickly became apparent that this approach was not the one we were looking for, as it was mostly focused on a phone system which would require IP handsets or connected internet browsers. This would infer that our Home SecuriPi product would need an external GUI for our guests interaction, and it was decided that this solution was out of scope for the project.

#### 7.1.2 Mumble

During investigation for our voice communication functionality, Mumble was occurring frequently in our peripherals, so we decided to take a closer look.

Mumble is a VoIP application which is primarily used by gamers who wish to have a team speaking platform. It is open source and runs on nearly all modern operating systems.

The architecture of mumble allows for groups of clients to communicate through a central server, called Murmur. Each member of the group installs the mumble application on their device, and communicate to the mumble server, which broadcasts any voice interaction to all

members in the group. This architecture was not easily implemented into a third party application, meaning research into this software ended here <sup>36</sup>.

## 7.2 Dynamic DNS

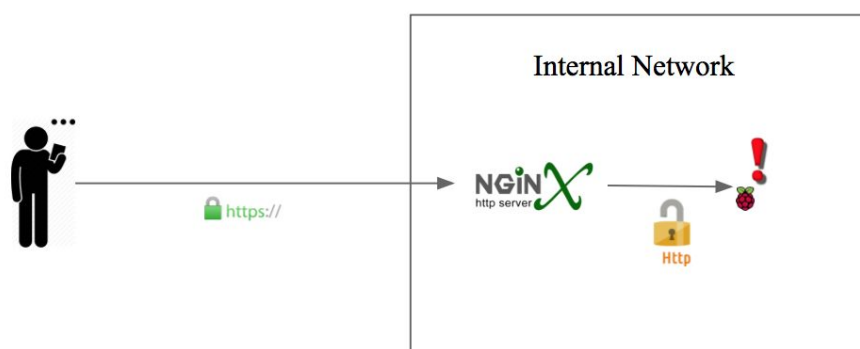
In order for users of Home SecuriPi to send commands remotely to our server living on the Raspberry Pi, which is installed in the home, we needed a way for the wide area network to know where our server was living. As internet service providers (ISP) change a client's IP address sporadically, we employed the use of a Dynamic Domain Name System (DDNS) provider. We could have purchased a static IP from our ISP, however this can be quite costly, and is easily circumvented using DDNS.

DDNS maps an internet domain name to an IP address. In the case of our proof of concept, the domain name chosen was karlredmond.gotdns.com. How this works is, whenever the ISP changes the IP address of the router, the router's software contacts the DDNS provider to notify them of the change, who then update the domain name to point to the newly appointed IP address. This means that whenever a newly appointed address is given to someone's home, the Home SecuriPi installation can still be communicated with.

## 7.3 NGINX

A growing criteria for web applications, is that they implement the use of HTTPS. HTTPS encrypts any request coming from a client browser to a server. Safari and Chrome have already implemented this requirement, and no longer allow HTTP requests to be sent to a server from their respective browsers, only allowing HTTP requests to be sent over an internal network (localhost) for development purposes.

In order to provide a HTTPS connection to our Raspberry Pi, research indicated that we needed to set up a reverse proxy, which would sit between our Home SecuriPi application and the outside world. The proxy would create the encrypted channel, and forward the requests across a HTTP channel to our application on the internal network.



**Fig. 22** HTTPS & Reverse Proxy

“NGINX is open source software for web serving, reverse proxying, caching, load balancing, media streaming, and more”<sup>37</sup>.

Figure 22 shows a high level view as to how a user would interact directly with the Home SecuriPi application. By configuring the NGINX server, as indicated below, the requirement could be met for connecting over HTTPS from a browser to our reverse proxy, which will forward the request over HTTP to our application.

```
server {
    listen 80;
    return 301 https://karlredmond.gotdns.com:5000
}

server {
    listen 443;
    server_name localhost
    ssl_certificate      /etc/nginx/cert.pem;
    ssl_certificate_key  /etc/nginx/key.pem;
    ssl on;

    location / {
        proxy_pass      http://192.168.1.7:5000;
    }
}
```

In the above configuration, any request coming in to `http://karlredmond.gotdns.com:5000`, which is the DDNS accompanied by a port number, will be redirected to `https://karlredmond.gotdns.com:5000`. The NGINX server will then set up the HTTPS connection, using a registered ssl certificate, and pass the request over HTTP to our application, which lives on 192.168.1.7, and is listening on port 5000.

Note: The configuration was successfully set up to pass the requests to our application over http, however, the ssl certificate implementation could not be confirmed, as in order to get a certification authority to allow registration of a certificate you need a registered domain name. This was considered out of scope for this project. To circumvent this issue, the HTTPS requests are sent to pythonanywhere, which forward the request to the Home SecuriPi application.

## 7.3 Apache

Apache was also investigated as a means to provide a HTTPS connection to our application. It works in much the same way, however the server configuration was different. The server configuration for Apache was as follows:

```
Listen 80
Listen 443
<VirtualHost *:80>
    ServerName localhost
    Redirect permanent / https://www.karlredmond.gotdns.com:5000
</VirtualHost>

<VirtualHost *:443>
    ServerName localhost
    SSLEngine On
    SSLCertificateFile /letsencrypt/cert.pem
    ProxyRequests Off
    ProxyPreserveHost On
    ProxyPass / http://192.168.1.7:5000/
    ProxyPassReverse / http://192.168.1.7:5000/
</VirtualHost>
```

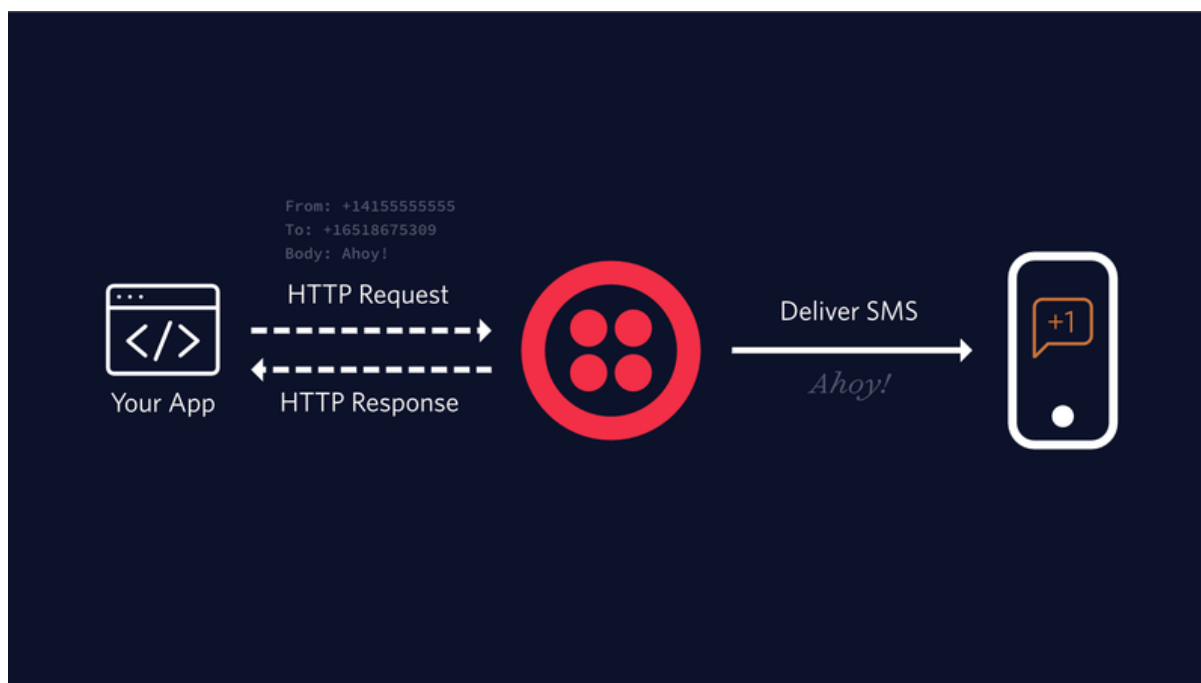
This configuration works in a similar fashion to that of NGINX. First it takes any HTTP request and sends a redirect to HTTPS. The HTTPS connection is then set up and the request is passed over HTTP to our internal server living on the Raspberry Pi at 192.168.1.7:5000.

## 7.6 Twilio

In order for our Home SecuriPi application to provide an alert to an end users mobile phone upon detection of movement, we employed the service provided by Twilio.

Twilio is a cloud communications platform as a service (PaaS). It provides a means for software developers to make/receive phone calls and send/receive sms messages. This service is harnessed through Twilio's web API <sup>38</sup>.

Home SecuriPi uses Twilio's API to send SMS messages to our end users. Setting up the service is simple. Upon registering with Twilio, you receive a virtual phone number and \$15.00 of free credit. On the free account, you can use the API to communicate with one physical phone.



**Fig. 23** Twilio SMS <sup>39a</sup>

```
from twilio.rest import Client
```

```
# put your own credentials here
```

```
account_sid = "ACc5f0347054e7b0d8ce2af0cf198b5a38"
```

```
auth_token = "your_auth_token"
```

```
client = Client(account_sid, auth_token)
```

```
client.messages.create(
```

```
    to="+353834449555",
```

```
    from_="+353861802301",
```

```
    body="Security Alert!! Motion Detected at Back Door!!
```

```
    karlredmond.gotdns.com:5000"
```

```
)
```

**Fig. ?** Send SMS Python <sup>39b</sup>

Figure 23 above shows how an application contacts a mobile phone. Basically, the application will send a HTTP request to a Twilio URI. The developer can easily create a client and message object using Twilio's Python helper library, as outlined in Figure ?. Twilio know which account to apply the message to from the `account_sid` and `auth_token` assigned to the account upon registration (*Twilio also provide a test account\_sid and auth\_token for testing purposes*). Upon receipt of the HTTP request the PaaS will send the message to the specified phone number. The application can form the body of the text message to the developer specifies, in the case above, a link is provided in the message body to

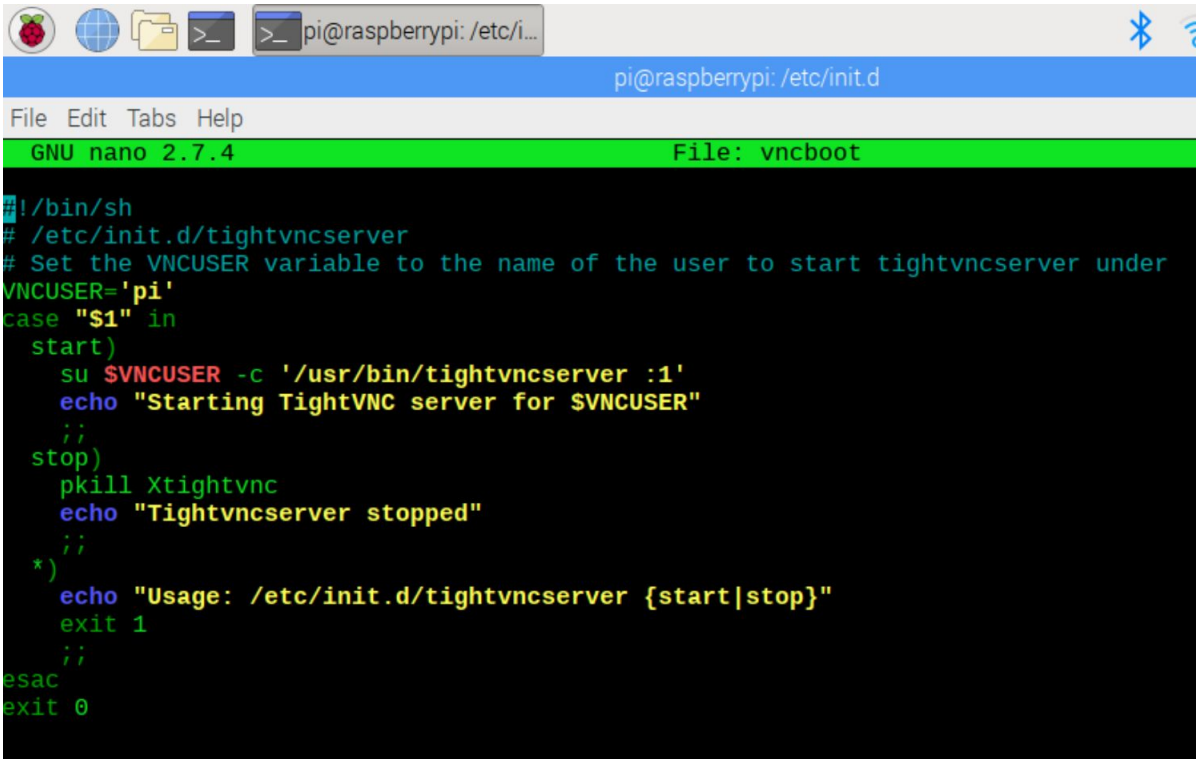
bring the user to the Home SecuriPi homepage, where the latest image captured by the camera will be displayed.

## 7.7 TightVNC

As this proof of concept involves home security, along with the fact that during development the project would not be easily transported, as it would mean disconnecting the wiring and the constant reconfiguration of network settings such as IP addresses and port numbers, it was decided to use the author of the project's house. This meant that the Raspberry Pi would not always be accessible when attending lectures and such. So, we needed a way to communicate to the Raspberry Pi remotely.

This was achieved through the installation and configuration of TightVNC. TightVNC is a client/server software package that allows remote access to graphical user interfaces(GUI). It is a free software available across most platforms under the GNU General Public License.

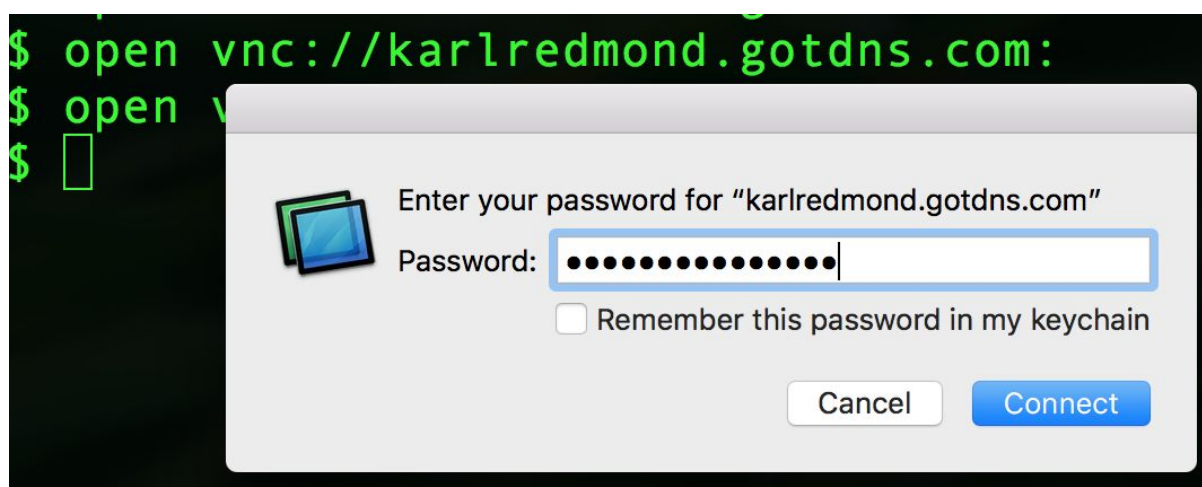
This was configured to run on startup as shown in Figure 24. This file is placed in the /etc/init.d directory, meaning it will be run with the other startup scripts at boot time.



**Fig. 24** TightVNC Startup Script

Once this server is up and running, the client application was installed onto the laptop. The following command is used for logging into the remote desktop:

```
open vnc://karlredmond.gotdns.com:???
```



**Fig. 25** TightVNC Password Screen

The URI is used to point at the home IP address (DDNS 7.2 of this doc), followed by a specific port number. The home router was configured to listen to this port and forward any traffic to the port number that the TightVNC server is listening on. Upon correct configuration you are presented with the screen displayed in Figure 25. When the correct password is entered, you are presented with the GUI of the Raspberry Pi’s Raspbian OS.

## 7.8 Flask

“Flask is a microframework for Python based on Werkzeug, Jinja 2 and good intentions. And before you ask: It's BSD licensed!”<sup>40a</sup>

Flask is a microframework, used for building web applications. This framework was considered the best option for development on this project, as it is extremely lightweight, and is specifically for Python, which is a core development language of the Raspberry Pi.

### Flask is Fun

*Latest Version: [0.12.2](#)*

```
from flask import Flask
app = Flask(__name__)

@app.route("/")
def hello():
    return "Hello World!"
```

### And Easy to Setup

```
$ pip install Flask
$ FLASK_APP=hello.py flask run
* Running on http://localhost:5000/
```

**Fig. 26** Flask Setup of “Hello World”<sup>40b</sup>



Getting started with Flask is as simple as the description given in Figure 26, which outlines every programming language's first application implementation of choice, "Hello World!".

The server debugger gives useful information should it crash for any reason during development, and there is huge online support for this framework, including extensive, easy to read and follow documentation <sup>40c</sup>.

While the framework is considered lightweight, as it doesn't require additional libraries or tools, there are a large number of packages associated with Flask which can be plugged into your application and used easily with Flask such as Flask-Security, Flask-SQLAlchemy and Flask-Cors <sup>41</sup>.

## 8. References

---

- [Title Page image] Raspberry Pi (2017). *RaspberryPi.org Home Page*[online], available: <https://www.raspberrypi.org> [accessed 15 Oct, 2017].
- [1] The MagPi (March 2017). *Sales Soar and Raspberry Pi beats Commodore 64*[online], available: <https://www.raspberrypi.org/magpi/raspberry-pi-sales/> [accessed 15 Oct, 2017].
- [2] PrivateEyePi (n.d). *Home Alarm System Project*[online], available: <http://www.projects.privateeyepi.com/home/home-alarm-system-project> [accessed 15 Oct, 2017].
- [3] Hackster.io(22<sup>nd</sup> March 2017). *DIY Smart Home Security System for under \$50*[online], available: <https://www.hackster.io/naran-inc/diy-smart-home-security-system-for-50767d> [accessed 16 Oct, 2017].
- [4] Danalock(2017). *Danalock.com Home Page*[online], available: <https://danalock.com/index.html> [accessed 16 Oct, 2017].
- [5] techhive.com(10<sup>th</sup> Dec 2015). *Samsung SmartThings Home Monitoring Kit Review*[online], available: <https://www.techhive.com/article/3013140/home-control/samsung-smarthings-home-monitoring-kit-review-a-solid-connected-home-foundation.html> [accessed 16 Oct, 2017].
- [6] Energenie(2017). *Energenie Home Page*[online], available: <https://energenie4u.co.uk> [accessed 21 Oct, 2017].
- [7] TECH REVIEWS(Sept, 2016). *Energenie: MiHome Smart Home Automation Review*[online], available: <http://tech-reviews.co.uk/reviews/energenie-mihome-smart-home-automation-review/> [accessed 21 Oct, 2017].
- [8] OxGadgets(2017). *Review: Energenie-Smart Home automation*[online], available: <http://www.oxgadgets.com/2015/09/review-energenie-smart-home-automation.html> [accessed 21 Oct, 2017].
- [9] Centrica (9th June 2017). *Centrica’s Smart Home Brand Hive Launches ‘Let’s Get Living’* [online], available: <https://www.centrica.com/news/centricas-smart-home-brand-hive-launches-lets-get-living> [accessed 5 Feb, 2018].

- [10 a,b] Bord Gais Energy (2018). Control your heating and hot water from anywhere with Hive[online], available: <https://www.bordgaisenergy.ie/homecare/hive-active-heating/> [accessed 5 Feb, 2018].
- [11] Centrica (2018). Hive[online], available: <https://www.centrica.com/about-us/our-brands/hive> [accessed 5 Feb, 2018].
- [12] Techradar (23rd Nov, 2017). Nest Learning Thermostat Review[online], available: <http://www.techradar.com/reviews/gadgets/appliances/nest-learning-thermostat-1256380/review> [accessed 12 Feb, 2018].
- [13] Argos (23rd Nov, 2017). Nest Learning 3rd Generation Thermostat[online], available: [http://www.argos.ie/static/Product/partNumber/5014785/c\\_1/1/%7Ccategory\\_root%7CGarden+and+DIY%7C14418702/c\\_2/3/%7C19536159%7CSmart+Home%7C50000397/c\\_3/4%7Ccat\\_50000397%7CSmart+heating%7C50000398.htm?\\_Sja=tsid%3A30390%7Cprd%3A6361382](http://www.argos.ie/static/Product/partNumber/5014785/c_1/1/%7Ccategory_root%7CGarden+and+DIY%7C14418702/c_2/3/%7C19536159%7CSmart+Home%7C50000397/c_3/4%7Ccat_50000397%7CSmart+heating%7C50000398.htm?_Sja=tsid%3A30390%7Cprd%3A6361382) [accessed 12 Feb, 2018].
- [14] sparkfun(n.d.). *What is an Arduino?*[online], available: <https://learn.sparkfun.com/tutorials/what-is-an-arduino> [accessed 25 Oct, 2017].
- [15] Arduino(n.d.). *What is a Arduino*[online], available: <https://www.arduino.cc> [accessed 25 Oct, 2017].
- [16] Beebom(September, 2016). *Arduino vs Raspberry Pi: A detailed Comparison*[online], available: <https://beebom.com/arduino-vs-raspberry-pi/> [accessed 25 Oct, 2017].
- [17] opensource.com(n.d.). *What is a Raspberry Pi*[online], available: <https://opensource.com/resources/raspberry-pi> [accessed 25 Oct, 2017].
- [18] lifehacker(March, 2016). *Raspberry Pi: Which Model Should You Buy*[online], available: <https://www.lifehacker.com.au/2016/03/raspberry-pi-which-model-should-you-buy/> [accessed 25 Oct, 2017].
- [19] Raspberry Pi(October, 2016). *Raspberry Pi 3 on sale now at \$35*[online], available: <https://www.raspberrypi.org/blog/raspberry-pi-3-on-sale/> [accessed 25 Oct, 2017].
- [20] The MagPi(October, 2016). *Pi 3 is out now! Specs, benchmarks & more*[online], available: <https://www.raspberrypi.org/magpi/raspberry-pi-3-specs-benchmarks/> [accessed 25 Oct, 2017].

- [21] Arduino(2017). *Intel Galileo*[online], available: <https://www.arduino.cc/en/ArduinoCertified/IntelGalileo> [accessed 25 Oct, 2017].
- [22] make(2017). *Raspberry Pi or Arduino Uno? One Simple Rule to choose the right board.*[online], available: <https://makezine.com/2015/12/04/admittedly-simplistic-guide-raspberry-pi-vs-arduino/> [accessed 25 Oct, 2017].
- [23]Doorjam (Sept 21st 2016). *Doorjam* [online], available: <https://ideas.redpepper.land/doorjam-47f1a5bce2fd> [accessed 30 Oct, 2017].
- [24] The Next Web(2016). *Hack a Raspberry Pi to play your them song when you enter a room* [online], available: <https://thenextweb.com/shareables/2016/10/10/hack-a-raspberry-pi-to-play-your-theme-song-when-you-enter-a-room/> [accessed 30 Oct, 2017].
- [25] Imgur(Dec 28th 2015). *Magic Mirror* [online], available: <https://imgur.com/gallery/Z8TxmRp> [accessed 30 Oct, 2017].
- [26] The Raspberry Pi Guy (2016). [online], available: <https://www.youtube.com/user/TheRaspberryPiGuy> [accessed 30 Oct, 2017].
- [27] RaspberryPi(2017). *Raspberry Pi FAQ's* [online], available:<https://www.raspberrypi.org/help/faqs/#softwareLanguages> [accessed 30 Oct, 2017].
- [28] IBEX(n.d.). *.Getting your pi ready for C programming* [online], available:<http://www.raspberry-projects.com/pi/programming-in-c/getting-your-raspberry-pi-ready-for-c-programming> [accessed 30 Oct, 2017].
- [29] Geany(2017). *Geany Screenshots* [online], available:<https://www.geany.org/Documentation/Screenshots> [accessed 30 Oct, 2017].
- [30] BlueJ(2017). *LED Tutorial* [online], available:<https://www.bluej.org/raspberrypi/led.html> [accessed 31 Oct, 2017].
- [31] Raspberry Pi(2017). *Python*[online], available:<https://www.raspberrypi.org/documentation/usage/python/> [accessed 31 Oct, 2017].
- [32] Software Engineer Insider(n.d.). *The Python Programming Language* [online], available:<https://www.softwareengineerinsider.com/programming-languages/python.html> [accessed 31 Oct, 2017].

- [33] The University of Arizona(July 2010). Using IDLE (Python’s IDE) [online], available:<https://www2.cs.arizona.edu/people/mccann/usingidle> [accessed 1 Nov, 2017].
- [34] softpaz (n.d.). Thonny [online], available: <https://www.softpaz.com/software/download-thonny-windows-161777.html> [accessed 1 Nov, 2017].
- [35] Asterisk (2017). Getting started with Asterisk[online], available: <https://www.asterisk.org/get-started> [accessed 12 Feb, 2018].
- [36] Lifewire (n.d.). Mumble - Group Voice Chat for Online Gaming[online], available: <https://www.lifewire.com/mumble-group-voice-chat-3426887> [accessed 19 Feb, 2018].
- [37] NGINX(2018). NGINX Home Page [online], available: <https://www.nginx.com/resources/glossary/nginx/> [accessed 19 Feb, 2018].
- [38]Twilio(2018). Programmable SMS [online], available: <https://www.twilio.com/sms> [accessed 20 Feb, 2018].
- [39 a,b] Twilio(2018). Send SMS and MMS Messages in Python[online], available: <https://www.twilio.com/docs/guides/how-to-send-sms-messages-in-python#sign-up-for-a-twilio-account> [accessed 20 Feb, 2018].
- [40a,b,c] Flask(2018). Flask, Web Development, one drop at time[online], available: <http://flask.pocoo.org> [accessed 20 Feb, 2018].
- [41] Flask(2018). Flask Extensions[online], available: <http://flask.pocoo.org/extensions/> [accessed 20 Feb, 2018].