

Contents

[1 Introduction 1](#_Toc512841687)

[2 Initial Idea 1](#_Toc512841688)

[3 Idea Development 1](#_Toc512841689)

[4 Plan & Research 2](#_Toc512841690)

[5 The Build 2](#_Toc512841691)

[5.1 Hardware 2](#_Toc512841692)

[5.2 Software 3](#_Toc512841693)

[5.3 Complex Prediction Model 4](#_Toc512841694)

[6 Development Issues 6](#_Toc512841695)

[7 Future Plans 6](#_Toc512841696)

[8 Final Thoughts 9](#_Toc512841697)

[9 Conclusion 10](#_Toc512841698)

[10 Appendix 12](#_Toc512841699)

[Appendix A 12](#_Toc512841700)

[Appendix B 14](#_Toc512841701)

[Appendix C 14](#_Toc512841702)

[Appendix D 16](#_Toc512841703)

[Appendix E 25](#_Toc512841704)

[Appendix F 25](#_Toc512841705)

[Appendix G 26](#_Toc512841706)

[Appendix H 26](#_Toc512841707)

[Appendix I 27](#_Toc512841708)

[Appendix J 27](#_Toc512841709)

[Appendix K 28](#_Toc512841710)

[Appendix S 28](#_Toc512841711)

[11 Glossary 29](#_Toc512841712)

[12 Bibliography 30](#_Toc512841713)

# 1 Introduction

Automation will be a big part of our futures and this project will give an insight into what this might look like. As this becomes more and more prominent in our lives the prices will fall and the complexity will rise; this is why making this artefact now has been so interesting.

My home automation system is based around the Wi-Fi model, using a WAMP web server to act as the front end and a raspberry pi, running a python script, to control the state of various channels in the relay. I could have opted to host the web app on the Pi itself but using a WAMP server allowed me to access the light state outside the home network.

# 2 Initial Idea

My initial idea stemmed from constantly forgetting to turn my bedroom light off when leaving the house. I then investigated solutions to the problem, how can I change the state of my light without being in the house. Films such as Iron Man and Batman incorporated smart homes to make the scene more futuristic and who doesn’t want a little bit of superhero in their house. I knew this project would be a challenge and these I love so the idea was decided upon and now could be developed.

# 3 Idea Development

As a computing student, breaking down how things work comes naturally, so when looking into how these gadgets work I was surprised to see how they function. The Bluetooth light bulbs work using a short frequency radio wave from either a remote control or a smartphone. Similarly, the Wi-Fi bulbs work by having their own locally hosted network web app that receives data packets from various networked devices to affect its state.

the idea developed more than expected, starting with the idea of using a remote control similar to that of a TV, then progressing to Bluetooth before settling on the web-controlled method.

The raspberry pi runs a Linux distribution that allows the user to host a web page from the device itself. The initial idea was to host the page and control the state all from the one device but when I thought about future expansion, considering this is a proof of concept project, I decided to have a centralized hub where many devices could be controlled from would be far more effective than opting for the completely self-contained setup. this meant I would have to think of a way to host the website and transmit data to the Pi.

This will be discussed further in the software section but essentially, I chose to use a computer as a server and the Pi to request the state from.

# 4 Plan & Research

As there are many stages to this project planning was essential. I created a basic Gantt Chart to manage my time for not only the build but also the write-up and presentation and found this extremely useful when trying to meet deadlines. This can be found in appendix S.

As the project was very technical I needed to learn how to use the technology I was implementing before starting. This was a lot of my research and planning.

I started by looking online for raspberry Pis as my existing one was too old to run Raspbian competently. I settled on the Pi 3 Model B as it is extremely fast for the size and was on offer. I then went to eBay and purchased all the components I needed for the learning phase of the build. I brought a breadboard, LEDs, jumper cables and a few resistors. These arrived, and I used the components to construct simple circuits controlled by a python program. The circuits I already knew how to construct from my physics lessons so using the Pi as a switch was the only new stage. I used a GPIO Pinout that allowed me to work out which pins to use to make the relay work.

# 5 The Build

## 5.1 Hardware

I started by purchasing various electronic components that would be used in the project:

* Raspberry Pi 3 w/ WIFI
* 8 channel relay board
* Assorted LEDs
* Breadboard
* A fused power strip
* Resistors
* A lamp
* An SD card
* MDF
* Cable ties and tacks
* GPIO jumper cables

I started by flashing the latest Raspbian image to the micro sd card. I then plugged this into the Pi and booted. The first thing I did was use the “sudo apt-get update” (and upgrade) commands to ensure everything was up to date. Raspbian comes with python pre-installed, this is the programming language that the Pi uses to interact with the GPIO pins. During school I learnt to write python but has to set some time aside to familiarise myself with the RPi.GPIO library. To learn how this worked I used LEDs and plugged them into the breadboard, creating simple circuits that could be switched on and off from the program. The gpio library has inbuilt functions that relate to the physical via a pinout, see Appendix F.

Once the ‘hello world’ stage was complete I then had to get to know the channel relay. To use this I had to connect the jumper cables from the board to the relay. This allowed the Pi to control the state of 8 separate switches, similar to the way a Solenoid valve works. To test that I had this working correctly I connected a channel switch in series with the breadboard circuit, now powered by a 9V battery, and turned the circuit on and off.

Once the relay had been configured correctly I purchased a light from Argos and used a screw driver to remove the casing of the switch. The switch was soldered to the wires so using my soldering iron I removed this, leaving two bare wires. These were then put into the channel relay and screwed tightly in place. Once this connection was secured I used a python program to switch the light on and off in sequence just to ensure it was working correctly.

## 5.2 Software

After completing the hardware section, for now, I then moved over to the front-end interface. For this I would use previously known PHP skills to create a simple button to toggle the state of hardware channels. I started by dusting off an old laptop and installing WAMP, a program that allows users to host websites on their local network. This program has many features including the ability to allow connections from WLAN IP addresses. This means that after changing a few settings in the config file, using DHCP reservation to create a static address for the server, allowing a firewall exception and port forwarding its port 80, I could connect to and view websites on the server from anywhere in the world.

The site was very quick to create as only a few buttons were necessary. I decided on a system that would follow the process found on the left of this page:

Lights

Button

On Click

LightStatus Page

Home Page

If Currently True, Change State To False. Visa Versa

Screenshots from the site after the first button had been created are shown below, first showing the device in the off state and the other showing the on state with its respective colours. See appendix G.

I settled on a futuristic theme, with bright text colours, large text and a ghosted background for the button. I added a grayscale golden ratio wallpaper for contrast and was happy with how the site looked. This setup would allow me to add more buttons for more devices in future as they are added.

The Raspberry Pi will then request the source code of the status pages every second, changing the state of the relays based on their state. The algorithm for this is shown in Appendix E.

Now that I had a way to interact between the two devices and a suitable circuit design I then mounted all the required components on the MDF. This was so I could put the product where I wanted without many components hanging by wires. This was then put onto a shelf where the lamp could be placed on the desk in my office.

I purchased a raspberry pi with wifi specifically so the product could be transported and so it didn't require an ethernet tether. This meant I could put the whole product onto the shelf with only a power cable and the lamp plug running from it. Rather than having to plug the Pi into a monitor I could use a utility called Putty to run programs and upload new versions of the program as it evolved. Putty works using Secure Shell, this is a method of data transfer that is more commonly used when remotely accessing machines for command execution, so it seemed perfect! I have included a screenshot of putty being used in Appendix K.

As a laptop running a stock version of windows 10 pro has been used as a server it will often have forced updates that cause the laptop to restart without warning. The Pi, initially, would periodically request the status of the devices and if the server could not reply within a set time the program would crash as it did not know how to deal with this error. To overcome this I used a simple try catch statement that would mean the Pi would keep the same status until it could reconnect.

Once the light was working fully I needed to add the next feature. When watching Mr Robot and Iron Man the levels of automation in the futuristic houses inspired me to include a timed state change. This would work by allowing the user to set a time for the light to automatically turn on and off each day. An image of this can be found in Appendix H. This would then alter the state of the device completely automatically. The convenience of this becomes clear when you imagine synchronizing your alarm with your lights or the lights with the time you leave for work in the morning or arrive home. The process of checking if the pre-set switch needed to be activated was implemented in the source code of the complex prediction model to reduce the number of programs that needed to be running.

## 5.3 Complex Prediction Model

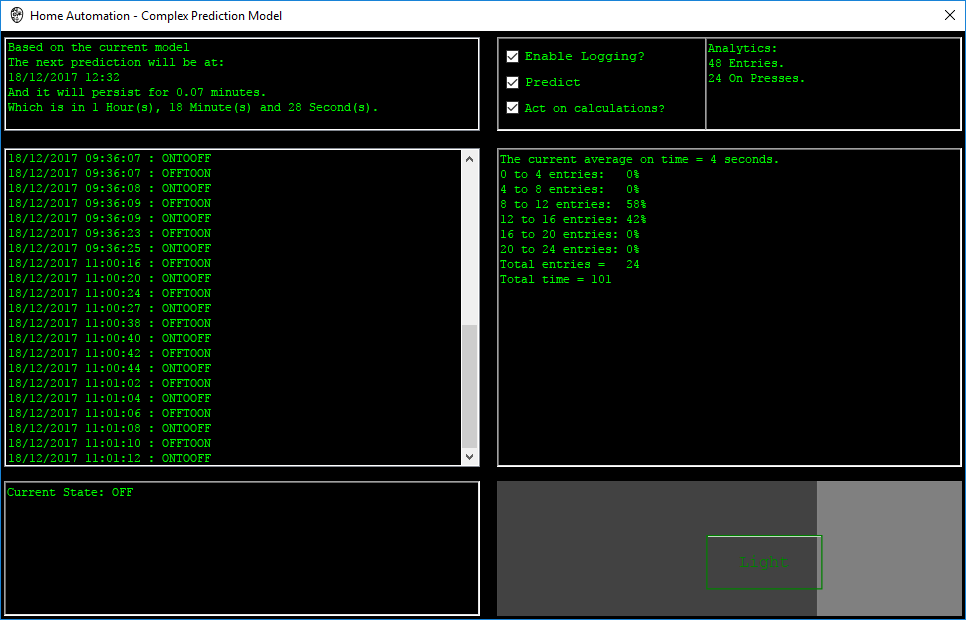
As a final challenge I decided to act on my interest in machine learning and create a simple learning bot that would record the state of the devices at intervals throughout the day and store this in its own file. These are the different prediction options I considered:

* Complex Prediction Model
  + Use of simple algorithms to produce a prediction based on trends
* Deep Learning
  + Use of *neural networks* to form predictions, based on large data sets
* Machine Learning
  + The way a program may take large data sets, learn from trends it encounters **and improve the way it predicts**
* Artificial Intelligence
  + Broad term to describe how a computer may carry out a task in a *smart* way. Machine Learning and Deep Learning are subcategories of this

*\*neural networks: a model based on a biological neural network that increases the weight of its ‘synapses’ along a path that reaches a better result*

The Machine Learning concept I had in mind was far too complex as I have no experience with neural networks, but to produce a similar result I opted for a complex prediction model which works by analysing data from previous events to produce a report outlining when the next state change is due to occur. The way it works can be explained simply using the following steps:

1. Run a check every second to see if the light has had its state alerted. if it has, then store this along with its time in a file. This timer is also responsible for checking the current time and seeing if it is equal to the prediction, if this is true then it will proceed in changing the state.
2. Run another timer that will calculate the prediction time
   1. Get all file entries and store in a 2d array
   2. Group all of the “OFF TO ON” state changes
   3. Group these remaining entries into one of the following categories
      1. 00:00 AM to 04:00 AM
      2. 04:00 AM to 08:00 AM
      3. 08:00 AM to 12:00 PM
      4. 12:00 PM to 16:00 PM
      5. 16:00 PM to 20:00 PM
      6. 20:00 PM to 24:00 PM
   4. If there is 30% of the entries in a given group, then produce a prediction for this time. This is to avoid a prediction being made for a single, one off, entry
   5. The program then converts all entries in the groups to ‘ticks’, which is time format, puts these values into an array and then takes an average after removing the outliers. This tick value is then converted back into a date time and then displayed.



Behind the scenes, this program is also responsible for checking to see if the set times on the web app are the same as the current time and changing the state as required. The source code for this program can be found in Appendix D

# 6 Development Issues

The development was relatively straight forward as I took the time to learn how things worked before attempting to create a solution but still had a few issues.

As discussed, the server not responding came as no surprise due to windows forcing updates and the server restarting. To overcome this, I added WAMP to the computer’s start up procedures so that if it did restart it would run the server again and added a try-catch on the Pi Script If there was no response.

# 7 Future Plans

The program could easily be improved to create a better overall product, incorporating new features. These are listed below:

* The current light pre-set configuration allows for an ON/OFF time to be set that will be active every single day. In future I would like to add the option to turn the pre-set times off altogether, an option to set different times for different days in the week and the ability to set one off times on specific dates.
* Allow the user to see the toggle logs from the web page.
* Use the darkSKY API to allow the prediction algorithm to tell if it is currently light outside to ensure the light isn’t turned on while it is day time.
* Use a weather API to affect the predictions, if the weather is rainy then it is more likely to be darker outside.
* Improve the prediction algorithm by using smaller time intervals and better data processing algorithms.
* The relay can have other devices added to it easily, using power strips and motors for curtains to make it more and more futuristic.

As a future development, I could include a motor to operate the blinds. This is the algorithm that I would use, in the form of pseudo code:

‘If the motor is set to up it will appear as Green on the site and True in the database

if server.MotorStatus = True

Set motorState = UP

End if

Do

Try

Request status from server

Set lightStatus = server.LightStatus

Set MotorStatus = server.MotorStatus

Set PowerStripStatus = server.PowerStripStatus

If lightStatus = True then

GPIO.1.enabled = True

Else

GPIO.1.enabled = False

End if

If PowerStripStatus = True then

GPIO.2.enabled = True

Else

GPIO.2.enabled = False

End if

‘Gpio 3 will have the + and - on the correct terminals on the

Motor. GPIO will have - on - and + on +. The motor is DC so

These will both move the motor in opposite directions.

If MotorStatus = True then

If MotorState = UP then

‘Do nothing

Else

GPIO.3.enabled = True

Sleep(3)

GPIO.3.enabled = False

MotorState = UP

End if

Else

If MotorState = DOWN then

‘Do nothing

Else

GPIO.4.enabled = True

Sleep(3)

GPIO.4.enabled = False

MotorState = DOWN

End if

End if

Catch

console.log(“Server Down Or WiFi Connection Lost.”)

End try

loop

It became more and more apparent that the solution I would create could be transferred to an array of other problems. These include more home automation devices such as motorised curtains and plug sockets but also more advanced medical applications such as automating on mass for large buildings, such as hospital lights or hotel corridor lighting systems. Furthermore, new advances in individual hospital rooms for longer term patients could benefit from systems controlled remotely via web app. An article recently discussed the way hospital beds can be fitted with sensors to determine whether patients are currently in them, this is useful when monitoring dementia and Alzheimer's patients to ensure their safety as they will be alerted so a human can view the room via CCTV and ensure the patient is okay.

Despite public concern over handing control over our homes to computers, the applications of it reach further than just a simple light. The software I have developed could be easily modified to encapsulate a detector for smoke, gas or pressure leaks in industry that could save the company thousands in an emergency as the computers could detect a change much faster than humans. While on the topic of man vs machine, the reaction time to a disaster in a factory will be far quicker coming from an automated system and could even prevent a bigger catastrophe. This could be a fire, a trapped arm or a gas leak. Automation is also brilliant at allowing users to control their energy use and save them some money, automated heating and lights mean there won’t be any need for them to be on while nobody is in as you can switch them off from anywhere if you forget.

These points come together to form an extremely strong argument for automation and can be seen in market grown figures, but it does have its draw backs. These come in the form of security issues and involve the entire IoT Community. IoT stands for Internet of Things and can be quickly explained by describing a concept house. If this house has a fridge that tracks what’s inside it and automatically sends its contents to an app on your phone, or has an oven that can be controlled wirelessly, or a kettle or a toaster or central heating or lights or TVs or door locks. Anything that can be wirelessly controlled that has some sort of effect on the world we live in is considered an IoT device. These devices are new technology and so are being made quickly and, by association, in a rush to meet the demands of the market. This means that manufacturers copy software for one product and alter it only slightly to make it work for another. This means a large percentage of the IoT community is running on rushed software that has been copied and pasted and made in a rush. The problem with this is that if a vulnerability is found on one of the devices then the same vulnerability is likely to be active on a range of other devices that don’t even require the part of the software the vulnerability relates to! Furthermore, default account passwords will no doubt become a pressing issue. As these devices are distributed to the world with the username as “Admin” and the password as “password” the vulnerability doesn’t even need to relate to software, the vulnerability it people’s laziness. It takes 2 seconds to log in with default information and have a live feed to a baby monitor watching a child. Another, even more serious, scenario is something called a Bot Net whereby the vulnerable IoT devices become susceptible to a bug that allows their NIC, Network Interface Card, to be controlled. This means they are essentially slaves to a ‘master’ computer owned by the hacker. The hacker could then send a request to its slaves, instructing them to send millions of data packets at a specified target, overwhelming the target to the point where it cannot respond to any genuine requests from users. This is called a DDOS attack and is extremely dangerous on a scale like described.

These security vulnerabilities don’t exist in my system as the software is custom written and the light control panel sits behind MD5 encryption.

# 8 Final Thoughts

In a world running on computers, doing an EPQ on such a trending topic has given me an insight in to what the future will be like and I have had a lot of fun doing it. whether it was reading about security of IoT devices or just creating the website each element has been very fun. If I was to do this again I feel I would spend more time on the website front end as there are so many features I wish I included but there may be added in future.

During the project, I have learned how to effectivly manage my time thorugh numernous miscalculations and missed deadlines, although I quickly corrected deadlines and progress as the final deadline came closer to ensure I didn’t over run overall. The time management skills that I had never had to use before definitely improved durastically as I would become far better at estimating how long things would take. I have also learned many technical skills such as how to bridge the gap between software and hardware and found this so interesting that it has given me more of a focus for future career paths. I have also learned how to properly lay out a report and include a good glossary, while also managing to explain the technical elements of the device to a non specialist audience which took planning and rehearsing.

I have most enjoyed learning about the Pi’s GPIO Pins and getting the software to interact with the real world. I have found building the web pages most valuable technical part of the process as it is a skill that will be most transferable to future jobs. I also enjoyed researching about IoT security as I was unaware of the risks that the public face.

My main strength would definitely be the problem solving aspect of the project, wether it was trying to stop the light from flickering if the server didn’t respond or if it was getting the prediction algorithm to stop choosing the closest time and making sure it was always in the future there were many challenges that were fun to overcome. My biggest weakness would have to be my knowledge on how to produce a professional report as I had never had to do one before, this meant I had to rewrite a lot of content. My organisation has been okay, I feel it could have been better when it came to keeping on top of what paper work needed to be handed in and when.

As mentioned previously, the skills that have progressed the most are the technical skills involving web design and report writing but also time management as I had never had to manage my time with a large project before.

If I was to attempt this project again I would opt for a different product, rather than building a lighting system I would build a product that had more real world uses. An example of this would be something like a pressure sensor that could be used to detect a leak and automattically alert an engineer or rectify the issue. The reason why I would avoid the lighting system is that it seemed at times that it was more of a toy than a practical device and a more useful product may have been more interesting to present and research.

If anyone else was to undertake the same project I would advice them to focus more on researching the IoT security as this is a very complex topic that I still don’t understand as much as id like to. I would also say the amount of time spent writing the research diary should have been scheduled better as I found myseld back filling every two weeks or so until I decided to keep on top of it.

# 9 Conclusion

* Does it work?

Yes, the product works as desired when the server is running. The server currently runs from an old laptop that turns off due to overheating sometimes which causes the device to stop responding.

* Is it reliable?

As stated above, the device is extremely reliable while the server is running. If reliability was essential, then the server would run from a proper server computer but as it is a prototype the occasional shut down isn’t worth worrying about too much.

* Was it expensive?

The total for the build was around £30, not including the server. This is more than a WIFI lightbulb would cost regularly but as it is a prototype and can be customised and improved in future I don’t feel this is too expensive.

* What would I do differently?

I would opt to automate a detecting device like a gas leak detector rather than a bulb as I feel it would have allowed me to explore the industry side of automation more.

* What have I learnt?

I have learnt how to write prediction algorithms, solder wires together, research efficiently and present to a large audience but most of all I have learnt how to manage my time and schedule activities that need to be done. I also learnt to prioritise certain things to ensure the deadlines were met. I had met some deadlines although over ran on the essay submission as I had to add a glossary after finding out the moderator wouldn’t be familiar with the topic. I have also learnt what makes a good presentation, by watching my classmates before my own and deciding what I felt were good features to include as well as what I felt were things to avoid.

# 10 Appendix

## Appendix A

WEB PAGE INDEX SOURCE CODE

|  |
| --- |
| <!doctype html>  <html lang="en">  <head>  <link rel="shortcut icon" type="image/png" href="https://image.flaticon.com/icons/png/512/248/248093.png" />  <meta charset="utf-8">  <title>Home[Made] Automation</title>  <meta name="Hub For Controlling Office Light" content="Login - Time Set - Toggle Switch">  <meta name="Luke Price" content="Home Automation System">  </head>  <style>  input[type=text] {  width: 14em;  font-family:'courier new';  padding: 12px 20px;  margin: 8px 0;  box-sizing: border-box;  border: 3px solid #ccc;  -webkit-transition: 0.5s;  transition: 0.5s;  outline: none;  }  input[type=text]:focus {  border: 3px solid #555;  color:lightgreen;  background-color:black;  }  input[type=password] {  width: 14em;  padding: 12px 20px;  margin: 8px 0;  font-family:'courier new';  box-sizing: border-box;  border: 3px solid #ccc;  -webkit-transition: 0.5s;  transition: 0.5s;  outline: none;  }  input[type=password]:focus {  border: 3px solid #555;  color:lightgreen;  background-color:black;  }  </style>  <?php  $username = "mediumfanta";  $password = "pompey111";  $nonsense = "supercalifragilisticexpialidocious";  if (isset($\_COOKIE['PrivatePageLogin'])) {  if ($\_COOKIE['PrivatePageLogin'] == md5($password.$nonsense)) {  ?>  <?php include('header.php'); ?>  <?php  exit;  } else {  echo "Bad Cookie.";  exit;  }  }  if (isset($\_GET['p']) && $\_GET['p'] == "login") {  if ($\_POST['user'] != $username) {  echo "Sorry, that username does not match.";  exit;  } else if ($\_POST['keypass'] != $password) {  echo "Sorry, that password does not match.";  exit;  } else if ($\_POST['user'] == $username && $\_POST['keypass'] == $password) {  setcookie('PrivatePageLogin', md5($\_POST['keypass'].$nonsense));  header("Location: $\_SERVER[PHP\_SELF]");  } else {  echo "Sorry, you could not be logged in at this time.";  }  }  ?>  <script src="http://code.jquery.com/jquery-1.10.2.js"></script>  <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">  <style>.wrapper {  text-align: center;  position: absolute;  top: 20%;  left:40%;  }  .wrapper1 {  text-align: center;  position: absolute;  top: 75%;  left:43%;  }  #login-form{  height:0px;  width:0px;  }  .button {  background:none;  border-size:5px;  border-width:5px;  border-style: solid;  margin:0;  padding:.8em;  font-size:44px;  font-family:'courier new';  }  .button2 {  background:none;  border-size:5px;  border-width:0px;  border-style: solid;  margin:0;  padding:.8em;  font-size:44px;  font-family:'courier new';  }    body {  background-image: url("https://i.pinimg.com/originals/e5/5a/9e/e55a9e4765d66bad42264249b5944277.jpg");  background-size: 100% auto;  font-family:'courier new';  color:lightgreen;  }  #submit {  background-image:url('https://image.flaticon.com/icons/svg/26/26053.svg');background-size: contain;background-repeat: no-repeat;background-position: center;  }  #submit:hover { background-image:url('https://www.shareicon.net/download/2016/06/15/781471\_lock\_512x512.png');background-size: contain;background-repeat: no-repeat;background-position: center;  cursor: pointer;}  </style>  <body>  <a href="https://docs.google.com/document/d/1jSAJopbnhooEX2\_9oN3L3WqEwQJyQDIJ5QTK7hppytw/edit?usp=sharing" ><img id="logobutton" src='homemadeautomation.png' style="position: absolute; bottom: 10px; width:12em; height:auto;cursor:position;"/> </a>  <div id="theDiv" style="display: none;"></div>  <div class="wrapper">  <form style="text-align:center;" action="<?php echo $\_SERVER['PHP\_SELF']; ?>?p=login" id="login-form" method="post">  <label>Name<input type="text" name="user" id="user" /> </label><br />  <label>Password<input type="password" name="keypass" id="keypass" /> </label><br/>  <br><input type="submit" id="submit" class="button2" value=" " />  </form>  </div>  </body> |

## Appendix B

WEB PAGE LIGHT STATE UPDATE SCRIPT

<?php

// disable cache by HTTP

header("Expires: Thu, 01 Jan 1970 00:00:00 GMT");

header('Cache-Control: max-age=0, no-cache, no-store, must-revalidate');

// disable cache by IE cache extensions

header('Cache-Control: post-check=0, pre-check=0', false);

// disable cache by Proxies

header("Pragma: no-cache");

$file = 'lights.html';

if ($\_SERVER['REQUEST\_METHOD'] === 'POST')

{

// POST request

$previous = file\_get\_contents($file);

if ($previous === 'true')

{

file\_put\_contents($file, 'false');

}

else

{

file\_put\_contents($file, 'true');

}

}

else

{

// not POST request, we assume it's GET

echo file\_get\_contents($file);

}

exit();

?>

## Appendix C

WEB PAGE TIMED LIGHT UPDATE SCRIPT

<?php

if(isset($\_POST['field1']) && isset($\_POST['field2'])) {

unlink("mydata.txt");

$data = $\_POST['field1'] . "\n" . $\_POST['field2'] . "\n";

$ret = file\_put\_contents('mydata.txt', $data, FILE\_APPEND | LOCK\_EX);

if($ret === false) {

die('There was an error writing this file');

}

else {

echo "$ret bytes written to file";

}

}

else {

die('no post data to process');

}

header( "refresh:0.5;url=index.php" );

## Appendix D

COMPLEX PREDICTION MODEL ALGORITHM – VB.NET

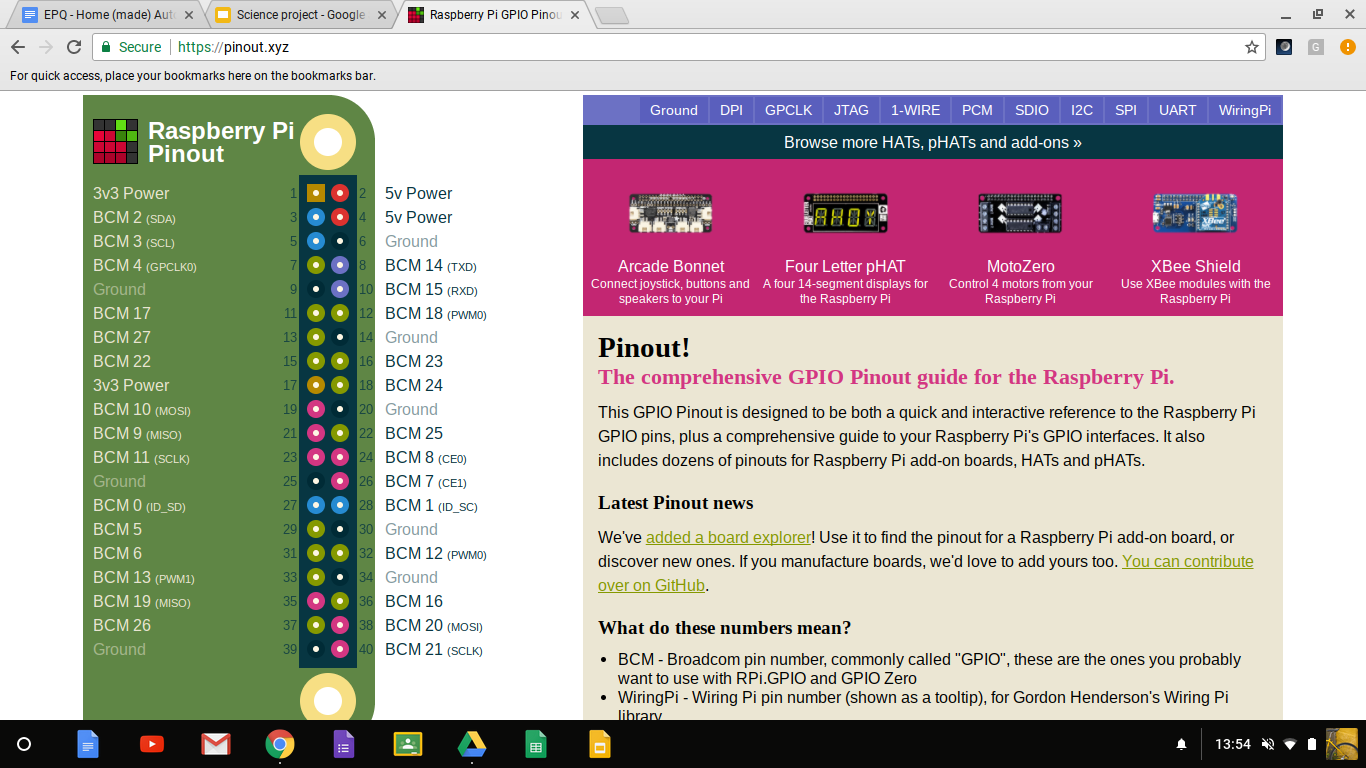
|  |
| --- |
| Imports System.ComponentModel  Public Class PredictionUI  Structure log  Dim log\_date As DateTime  <VBFixedString(7)> Dim state\_change As String  End Structure  Public log\_entry As New log  Public prediction As New DateTime  Public current\_state As String = "OFF"  Dim total\_time As Integer  Dim on\_time As Integer  Private Sub Form1\_Load(sender As Object, e As EventArgs) Handles MyBase.Load  PredictionWindowOutput.HorizontalScrollbar = True  FileOpen(1, "LogFile.dat", OpenMode.Random, , , Len(log\_entry))  populate()  Dim request As System.Net.HttpWebRequest = System.Net.HttpWebRequest.Create("http://82.19.253.90/homeautomation/lights")  Dim response As System.Net.HttpWebResponse = request.GetResponse()  Dim sr As System.IO.StreamReader = New System.IO.StreamReader(response.GetResponseStream())  Dim sourcecode As String = sr.ReadToEnd()  If sourcecode = "true" Then  CurrentStateWindow.Text = ("Current State: ON")  current\_state = "ON"  Else  CurrentStateWindow.Text = ("Current State: OFF")  current\_state = "OFF"  End If  End Sub  Sub populate()  LogList.Items.Clear()  For I = 1 To LOF(1) / Len(log\_entry)  FileGet(1, log\_entry, i)  LogList.Items.Add(log\_entry.log\_date & " : " & log\_entry.state\_change)  Next  End Sub  Private Sub tmrLog\_Tick(sender As Object, e As EventArgs) Handles tmrLog.Tick  analytics()  set\_time()  If EnableLoggingCheckbox.Checked Then  Dim request As System.Net.HttpWebRequest = System.Net.HttpWebRequest.Create("http://82.19.253.90/homeautomation/lights")  Dim response As System.Net.HttpWebResponse = request.GetResponse()  Dim sr As System.IO.StreamReader = New System.IO.StreamReader(response.GetResponseStream())  Dim sourcecode As String = sr.ReadToEnd()  Dim new\_state\_boolean As String = sourcecode  Dim next\_record As Integer = (LOF(1) / Len(log\_entry)) + 1  If new\_state\_boolean.ToLower = "true" Then  CurrentStateWindow.Text = ("Current State: ON")  If current\_state = "ON" Then  Else  current\_state = "ON"  log\_entry.log\_date = Now  log\_entry.state\_change = "OFFTOON"  FilePut(1, log\_entry, next\_record)  End If  Else  CurrentStateWindow.Text = ("Current State: OFF")  If current\_state = "ON" Then  current\_state = "OFF"  log\_entry.log\_date = Now  log\_entry.state\_change = "ONTOOFF"  FilePut(1, log\_entry, next\_record)  Else  End If  End If  populate()  LogList.TopIndex = LogList.Items.Count - 1  End If  If ActOnCalculationCheckbox.Checked And Now = prediction Then  IFrameWebUI.Document.All("buttt").InvokeMember("click")  End If  If LogList.Items.Count = 0 Then  PredictCheckbox.Enabled = False  Else  PredictCheckbox.Enabled = True  End If  End Sub  Private Sub Form1\_Closing(sender As Object, e As CancelEventArgs) Handles Me.Closing  FileClose(1)  End Sub  Private Sub tmrCalculate\_Tick(sender As Object, e As EventArgs) Handles tmrCalculate.Tick  If PredictCheckbox.Checked Then  PredictionWindowOutput.Items.Clear()  PredictionWindowCalculation.Text = String.Empty  IFrameWebUI.Refresh()  PredictionWindowOutput.Items.Add("Based on the current model")  PredictionWindowOutput.Items.Add("The next prediction will be at: ")  Dim predicted\_date As DateTime = get\_prediction()  Dim time\_until\_prediction As TimeSpan = predicted\_date - Now  PredictionWindowOutput.Items.Add(predicted\_date)  PredictionWindowOutput.Items.Add("And it will persist for " & Math.Round(on\_time / 60, 2) & " minutes.")  PredictionWindowOutput.Items.Add("Which is in " & time\_until\_prediction.Hours & " Hour(s), " & time\_until\_prediction.Minutes & " Minute(s) and " & time\_until\_prediction.Seconds & " Second(s).")  End If  End Sub  Function get\_prediction()  Dim amount\_of\_records As Integer = LOF(1) / Len(log\_entry)  Dim array\_times(amount\_of\_records, 1) As String  total\_time = 0  Dim avg\_ontimes As New List(Of Integer)  Dim previous\_time As New DateTime  Dim cluster\_0to4 As New List(Of DateTime)  Dim cluster\_4to8 As New List(Of DateTime)  Dim cluster\_8to12 As New List(Of DateTime)  Dim cluster\_12to16 As New List(Of DateTime)  Dim cluster\_16to20 As New List(Of DateTime)  Dim cluster\_20to24 As New List(Of DateTime)  For I = 1 To amount\_of\_records  FileGet(1, log\_entry, i)  If log\_entry.state\_change = "OFFTOON" Then  If DateTime.Parse(log\_entry.log\_date.ToLongTimeString) < DateTime.Parse("#04:00:00#") Then  Console.WriteLine(log\_entry.log\_date.ToString("hh:mm:ss") & "Is b4 4am")  cluster\_0to4.Add(log\_entry.log\_date.ToString("hh:mm:ss"))  ElseIf DateTime.Parse(log\_entry.log\_date.ToLongTimeString) < DateTime.Parse("#08:00:00#") Then  Console.WriteLine(log\_entry.log\_date.ToString("hh:mm:ss") & "Is b4 8am")  cluster\_4to8.Add(log\_entry.log\_date.ToString("hh:mm:ss"))  ElseIf DateTime.Parse(log\_entry.log\_date.ToLongTimeString) < DateTime.Parse("#12:00:00#") Then  Console.WriteLine(log\_entry.log\_date.ToString("hh:mm:ss") & "Is b4 12am")  cluster\_8to12.Add(log\_entry.log\_date.ToString("hh:mm:ss"))  ElseIf DateTime.Parse(log\_entry.log\_date.ToLongTimeString) < DateTime.Parse("#16:00:00#") Then  Console.WriteLine(log\_entry.log\_date.ToString("hh:mm:ss") & "Is b4 4pm")  If DateTime.Parse(log\_entry.log\_date.ToLongTimeString).Hour = 12 Then  cluster\_12to16.Add(log\_entry.log\_date.ToString("00:mm:ss"))  Else  cluster\_12to16.Add(log\_entry.log\_date.ToString("hh:mm:ss"))  End If  ElseIf DateTime.Parse(log\_entry.log\_date.ToLongTimeString) < DateTime.Parse("#20:00:00#") Then  Console.WriteLine(log\_entry.log\_date.ToString("hh:mm:ss") & "Is b4 8pm")  cluster\_16to20.Add(log\_entry.log\_date.ToString("hh:mm:ss"))  ElseIf DateTime.Parse(log\_entry.log\_date.ToLongTimeString) < DateTime.Parse("#23:59:59#") Then  Console.WriteLine(log\_entry.log\_date.ToString("hh:mm:ss") & "Is b4 12pm")  cluster\_20to24.Add(log\_entry.log\_date.ToString("hh:mm:ss"))  End If  previous\_time = log\_entry.log\_date  Else  Try  Dim secondsDiff As Integer = DateDiff(DateInterval.Second, previous\_time, log\_entry.log\_date)  avg\_ontimes.Add(secondsDiff)  Catch ex As Exception  'first time set  End Try  End If  Next  For h = 0 To avg\_ontimes.Count - 1  total\_time += avg\_ontimes(h)  Next  Dim cluster\_line As String = String.Empty  Dim cluster\_0to4\_ticks(cluster\_0to4.Count - 1) As Long  Dim cluster\_4to8\_ticks(cluster\_4to8.Count - 1) As Long  Dim cluster\_8to12\_ticks(cluster\_8to12.Count - 1) As Long  Dim cluster\_12to16\_ticks(cluster\_12to16.Count - 1) As Long  Dim cluster\_16to20\_ticks(cluster\_16to20.Count - 1) As Long  Dim cluster\_20to24\_ticks(cluster\_20to24.Count - 1) As Long  For I = 0 To cluster\_0to4.Count - 1  cluster\_line &= cluster\_0to4(i) & " "  cluster\_0to4\_ticks(i) = Date.Parse(cluster\_0to4(i)).Ticks / 1000  Next  Dim a As Date  Try  a = New Date(Math.Floor(cluster\_0to4\_ticks.Average) \* 1000)  Catch ex As Exception  End Try  cluster\_line = String.Empty  For I = 0 To cluster\_4to8.Count - 1  cluster\_line &= cluster\_4to8(i) & " "  cluster\_4to8\_ticks(i) = Date.Parse(cluster\_4to8(i)).Ticks / 1000  Next  Dim b As Date  Try  b = New Date(Math.Floor(cluster\_4to8\_ticks.Average) \* 1000)  Catch  b = New Date(0)  End Try  cluster\_line = String.Empty  For I = 0 To cluster\_8to12.Count - 1  cluster\_line &= cluster\_8to12(i) & " "  cluster\_8to12\_ticks(i) = Date.Parse(cluster\_8to12(i)).Ticks / 1000  Next  Dim c As Date  Try  c = New Date(Math.Floor(cluster\_8to12\_ticks.Average) \* 1000)  Catch  c = New Date(0)  End Try  cluster\_line = String.Empty  For I = 0 To cluster\_12to16.Count - 1  cluster\_line &= cluster\_12to16(i) & " "  cluster\_12to16\_ticks(i) = Date.Parse(cluster\_12to16(i)).Ticks / 1000  Next  Dim d As Date  Try  d = New Date(Math.Floor(cluster\_12to16\_ticks.Average) \* 1000)  If d.Hour = "00" Then  Dim ts As New TimeSpan(12, d.Minute, d.Second)  d = d.Date + ts  End If  Catch  d = New Date(0)  End Try  cluster\_line = String.Empty  For I = 0 To cluster\_16to20.Count - 1  cluster\_line &= cluster\_16to20(i) & " "  cluster\_16to20\_ticks(i) = Date.Parse(cluster\_16to20(i)).Ticks / 1000  Next  Dim f As Date  Try  f = New Date(Math.Floor(cluster\_16to20\_ticks.Average) \* 1000)  Catch  f = New Date(0)  End Try  cluster\_line = String.Empty  For I = 0 To cluster\_20to24.Count - 1  cluster\_line &= cluster\_20to24(i) & " "  cluster\_20to24\_ticks(i) = Date.Parse(cluster\_20to24(i)).Ticks / 1000  Next  Dim g As Date  Try  g = New Date(Math.Floor(cluster\_20to24\_ticks.Average) \* 1000)  PredictionWindowCalculation.Text &= vbNewLine & (cluster\_line & "Ticks: " & cluster\_20to24\_ticks.Average \* 1000 & " Guessed Time: " & g)  Catch  g = New Date(0)  End Try  cluster\_line = String.Empty  PredictionWindowCalculation.Text &= ("The current average on time = " & Math.Round(total\_time / avg\_ontimes.Count) & " second(s).")  PredictionWindowCalculation.Text &= vbNewLine & ("0 to 4 entries: " & Math.Round((cluster\_0to4.Count / avg\_ontimes.Count) \* 100) & "%")  PredictionWindowCalculation.Text &= vbNewLine & ("4 to 8 entries: " & Math.Round((cluster\_4to8.Count / avg\_ontimes.Count) \* 100) & "%")  PredictionWindowCalculation.Text &= vbNewLine & ("8 to 12 entries: " & Math.Round((cluster\_8to12.Count / avg\_ontimes.Count) \* 100) & "%")  PredictionWindowCalculation.Text &= vbNewLine & ("12 to 16 entries: " & Math.Round((cluster\_12to16.Count / avg\_ontimes.Count) \* 100) & "%")  PredictionWindowCalculation.Text &= vbNewLine & ("16 to 20 entries: " & Math.Round((cluster\_16to20.Count / avg\_ontimes.Count) \* 100) & "%")  PredictionWindowCalculation.Text &= vbNewLine & ("20 to 24 entries: " & Math.Round((cluster\_20to24.Count / avg\_ontimes.Count) \* 100) & "%")  PredictionWindowCalculation.Text &= vbNewLine & ("Total entries = " & avg\_ontimes.Count)  PredictionWindowCalculation.Text &= vbNewLine & ("Total time = " & total\_time)  on\_time = total\_time / avg\_ontimes.Count  If Now.Hour < 4 Then  If Math.Round((cluster\_0to4.Count / avg\_ontimes.Count) \* 100) > 30 And Now < a Then  prediction = a  ElseIf Math.Round((cluster\_4to8.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = b  ElseIf Math.Round((cluster\_8to12.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = c  ElseIf Math.Round((cluster\_12to16.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = d  ElseIf Math.Round((cluster\_16to20.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = f  ElseIf Math.Round((cluster\_20to24.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = g  Else  no\_prediction\_error()  End If  ElseIf Now.Hour < 8 Then  If Math.Round((cluster\_4to8.Count / avg\_ontimes.Count) \* 100) > 30 And Now < b Then  prediction = b  ElseIf Math.Round((cluster\_8to12.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = c  ElseIf Math.Round((cluster\_12to16.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = d  ElseIf Math.Round((cluster\_16to20.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = f  ElseIf Math.Round((cluster\_20to24.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = g  ElseIf Math.Round((cluster\_0to4.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = a  Else  no\_prediction\_error()  End If  ElseIf Now.Hour < 12 Then  If Math.Round((cluster\_8to12.Count / avg\_ontimes.Count) \* 100) > 30 And Now < c Then  prediction = c  ElseIf Math.Round((cluster\_12to16.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = d  ElseIf Math.Round((cluster\_16to20.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = f  ElseIf Math.Round((cluster\_20to24.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = g  ElseIf Math.Round((cluster\_0to4.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = a  ElseIf Math.Round((cluster\_4to8.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = b  Else  no\_prediction\_error()  End If  ElseIf Now.Hour < 16 Then  If Now.Hour = 12 Then  If Math.Round((cluster\_12to16.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = d  ElseIf Math.Round((cluster\_16to20.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = f  ElseIf Math.Round((cluster\_20to24.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = g  ElseIf Math.Round((cluster\_0to4.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = a  ElseIf Math.Round((cluster\_4to8.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = b  ElseIf Math.Round((cluster\_8to12.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = c  Else  no\_prediction\_error()  End If  Else  If Math.Round((cluster\_12to16.Count / avg\_ontimes.Count) \* 100) > 30 And Now < d Then  prediction = d  ElseIf Math.Round((cluster\_16to20.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = f  ElseIf Math.Round((cluster\_20to24.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = g  ElseIf Math.Round((cluster\_0to4.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = a  ElseIf Math.Round((cluster\_4to8.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = b  ElseIf Math.Round((cluster\_8to12.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = c  Else  no\_prediction\_error()  End If  End If  ElseIf Now.Hour < 16 Then  If Math.Round((cluster\_16to20.Count / avg\_ontimes.Count) \* 100) > 30 And Now < f Then  prediction = f  ElseIf Math.Round((cluster\_20to24.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = g  ElseIf Math.Round((cluster\_0to4.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = a  ElseIf Math.Round((cluster\_4to8.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = b  ElseIf Math.Round((cluster\_8to12.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = c  ElseIf Math.Round((cluster\_12to16.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = d  Else  no\_prediction\_error()  End If  ElseIf Now.Hour < 20 Then  If Math.Round((cluster\_20to24.Count / avg\_ontimes.Count) \* 100) > 30 And Now < f Then  prediction = g  ElseIf Math.Round((cluster\_0to4.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = a  ElseIf Math.Round((cluster\_4to8.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = b  ElseIf Math.Round((cluster\_8to12.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = c  ElseIf Math.Round((cluster\_12to16.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = d  ElseIf Math.Round((cluster\_16to20.Count / avg\_ontimes.Count) \* 100) > 30 Then  prediction = f  Else  no\_prediction\_error()  End If  Else  no\_prediction\_error()  End If  Return prediction  End Function  Private Sub CheckBox3\_CheckedChanged(sender As Object, e As EventArgs) Handles PredictCheckbox.CheckedChanged  If PredictCheckbox.Checked = False Then  PredictionWindowOutput.Items.Clear()  PredictionWindowOutput.Items.Add("Prediction Window")  PredictionWindowCalculation.Text = "Verbose Prediction Calculation"  End If  End Sub  Private Sub tmrWebUIResize\_Tick(sender As Object, e As EventArgs) Handles tmrWebUIResize.Tick  tmrWebUIResize.Enabled = False  IFrameWebUI.Focus()  SendKeys.SendWait("^-")  SendKeys.SendWait("^-")  SendKeys.SendWait("^-")  End Sub  Sub analytics()  AnalyticsWindow.Text = "Analytics:"  AnalyticsWindow.AppendText(vbNewLine & LOF(1) / Len(log\_entry) & " Entries.")  AnalyticsWindow.AppendText(vbNewLine & Math.Ceiling(LOF(1) / Len(log\_entry) / 2) & " On Presses.")  End Sub  Sub no\_prediction\_error()  PredictionWindowCalculation.Text &= vbNewLine & (" No Valid Prediction")  End Sub  Sub set\_time()  Dim request1 As System.Net.HttpWebRequest = System.Net.HttpWebRequest.Create("http://82.19.253.90/homeautomation/lights")  Dim response1 As System.Net.HttpWebResponse = request1.GetResponse()  Dim sr1 As System.IO.StreamReader = New System.IO.StreamReader(response1.GetResponseStream())  Dim sourcecode1 As String = sr1.ReadToEnd()  Dim new\_state\_boolean1 As String = sourcecode1  Dim request As System.Net.HttpWebRequest = System.Net.HttpWebRequest.Create("http://82.19.253.90/homeautomation/mydata.txt")  Dim response As System.Net.HttpWebResponse = request.GetResponse()  Dim sr As System.IO.StreamReader = New System.IO.StreamReader(response.GetResponseStream())  Dim sourcecode As String = sr.ReadToEnd()  timeSetVal.Text = "Set Times: " & vbNewLine & sourcecode  Dim parts As String() = sourcecode.Split(vbLf)  Dim turn\_on As DateTime = parts(0)  Dim turn\_off As DateTime = parts(1)  timeSetVal.Text &= "Current Time: " & Now.ToShortTimeString  Try  If Now.ToShortTimeString = turn\_on Then  If new\_state\_boolean1 = "false" Then  IFrameWebUI.Document.All("buttt").InvokeMember("click")  timesetLog.Text &= "As per preset value, the light was set from off to on at " & Now & vbNewLine  End If  End If  If Now.ToShortTimeString = turn\_off Then  If new\_state\_boolean1 = "true" Then  IFrameWebUI.Document.All("buttt").InvokeMember("click")  timesetLog.Text &= "As per preset value, the light was set from on to off at " & Now & vbNewLine  End If  End If  Catch ex As Exception  MsgBox("Please Login")  End Try  End Sub  End Class |

## Appendix E

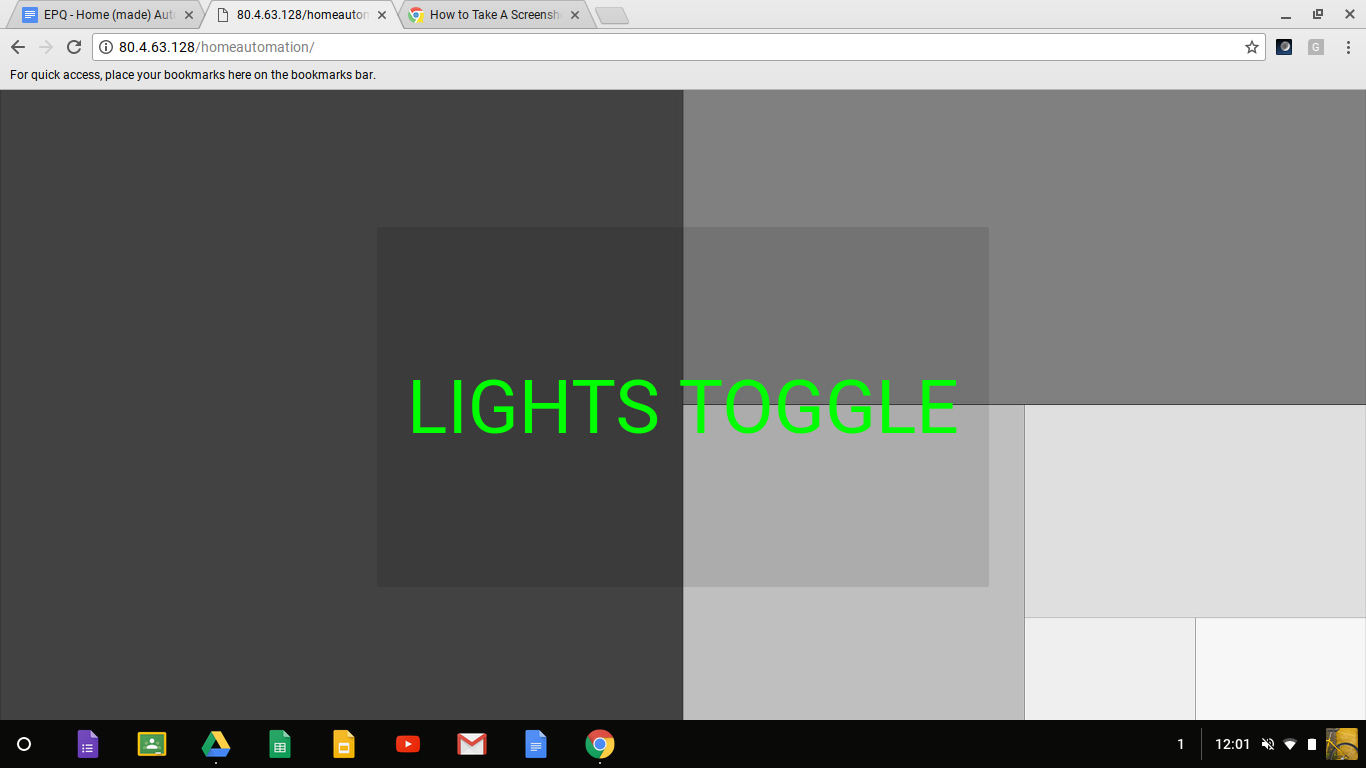
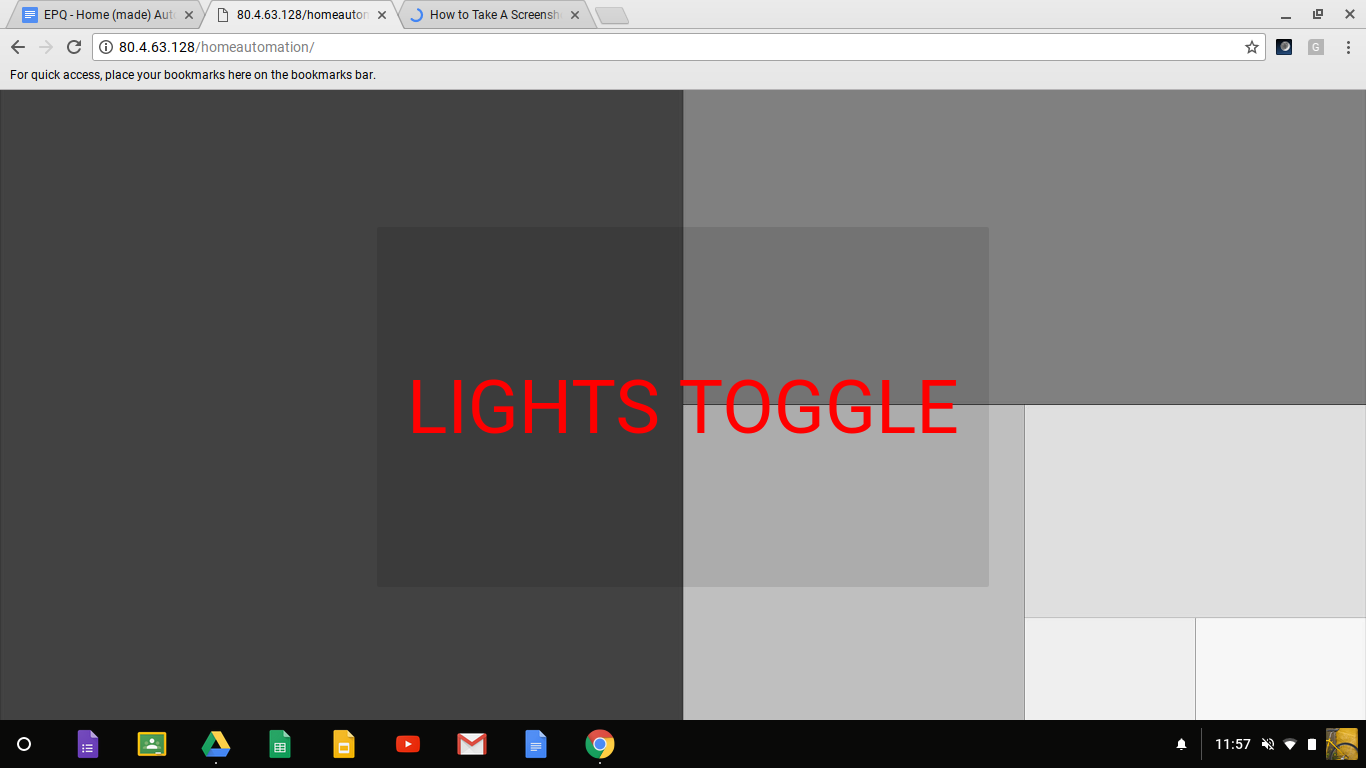
RASPBERRY PI STATE CONTROL SCRIPT

|  |
| --- |
| **import** **urllib2**  **import** **RPi.GPIO** **as** **GPIO**  **import** **time**  GPIO.cleanup()  GPIO.setmode(GPIO.BCM)  gpioList = [26, 19, 13, 06, 12, 16, 20, 21]  **for** I **in** gpioList:  GPIO.setup(i, GPIO.OUT)  GPIO.output(i, GPIO.HIGH)  *# Sleep time variables*  sleepTimeShort = 0.2  sleepTimeLong = 2  **while** True:  **try**:  response\_7 = urllib2.urlopen("http://82.19.253.90/homeautomation/LIGHTS")  page\_source\_7 = response\_7.read()  response\_7.close()  **except**:  page\_source\_7 = "false"  **if** page\_source\_7 == "false":  GPIO.output(12, GPIO.LOW)  GPIO.output(13, GPIO.HIGH)  **print** ("LIGHTS: OFF")  **else**:  GPIO.output(12, GPIO.HIGH)  GPIO.output(13, GPIO.LOW)  **print** ("LIGHTS: ON")  time.sleep(1) |

## Appendix F

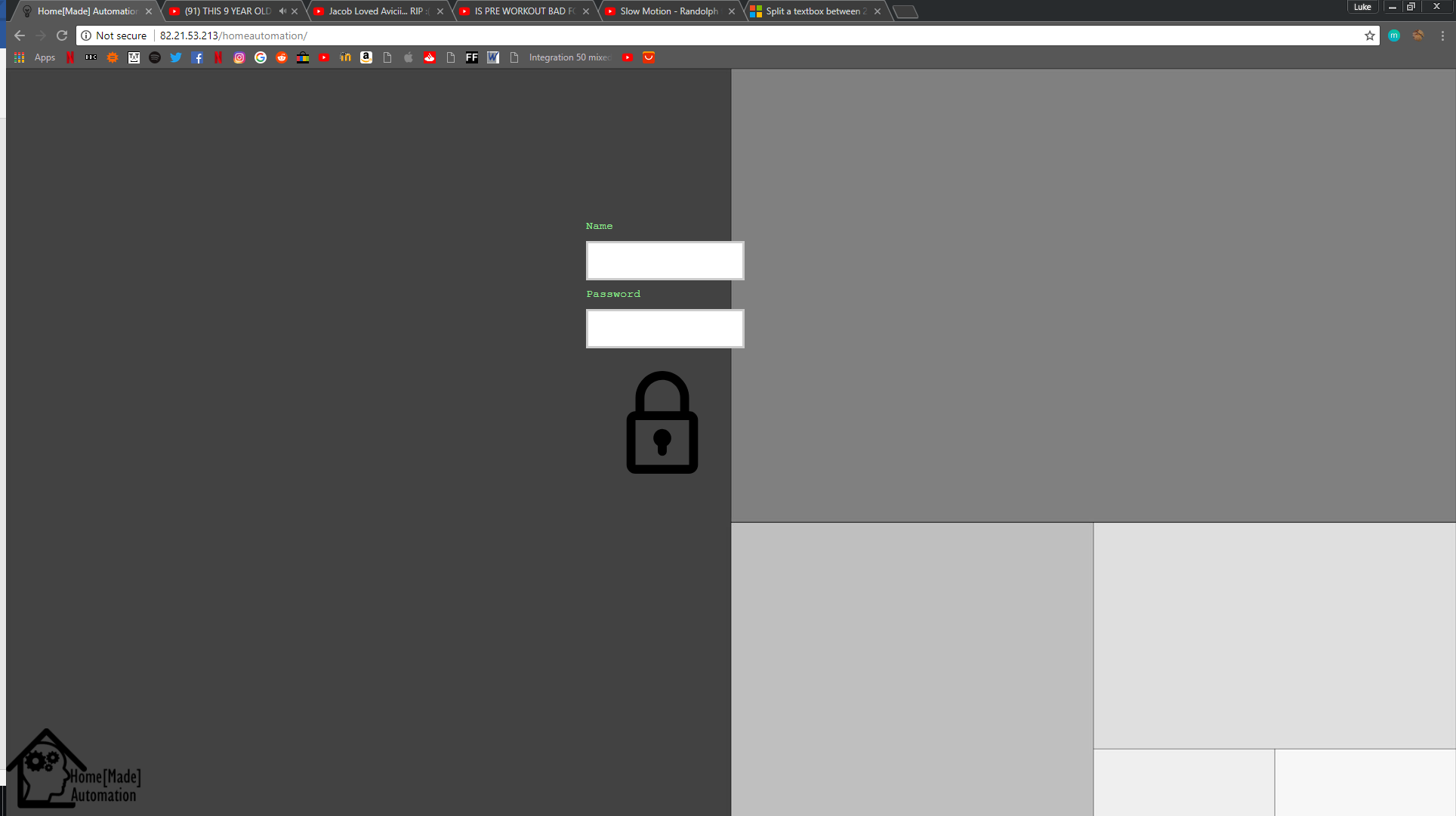


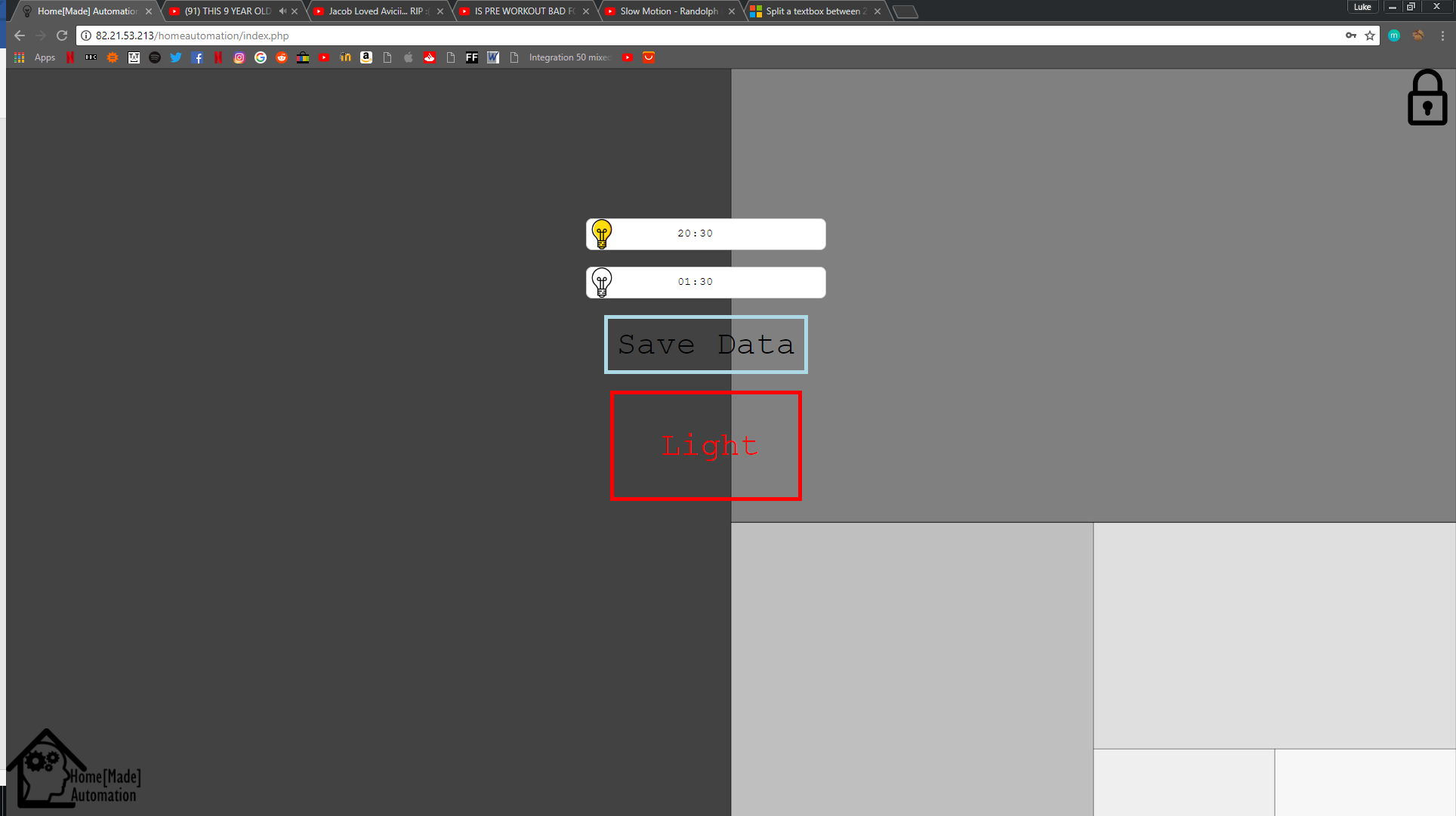
## Appendix G



## Appendix H

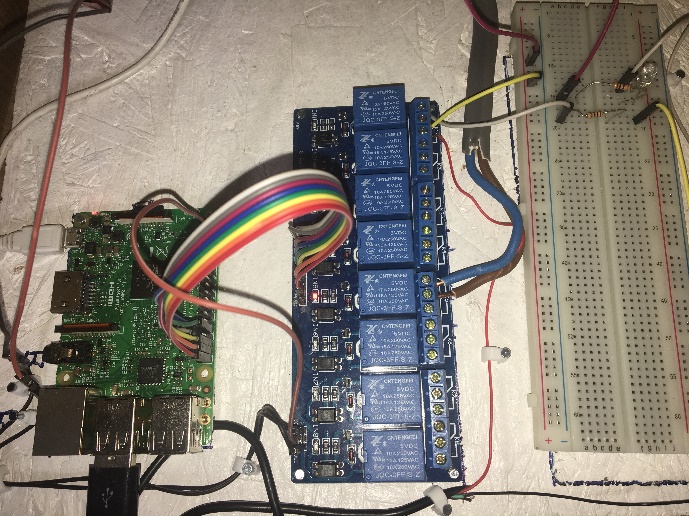
SC OF INDEX PAGE

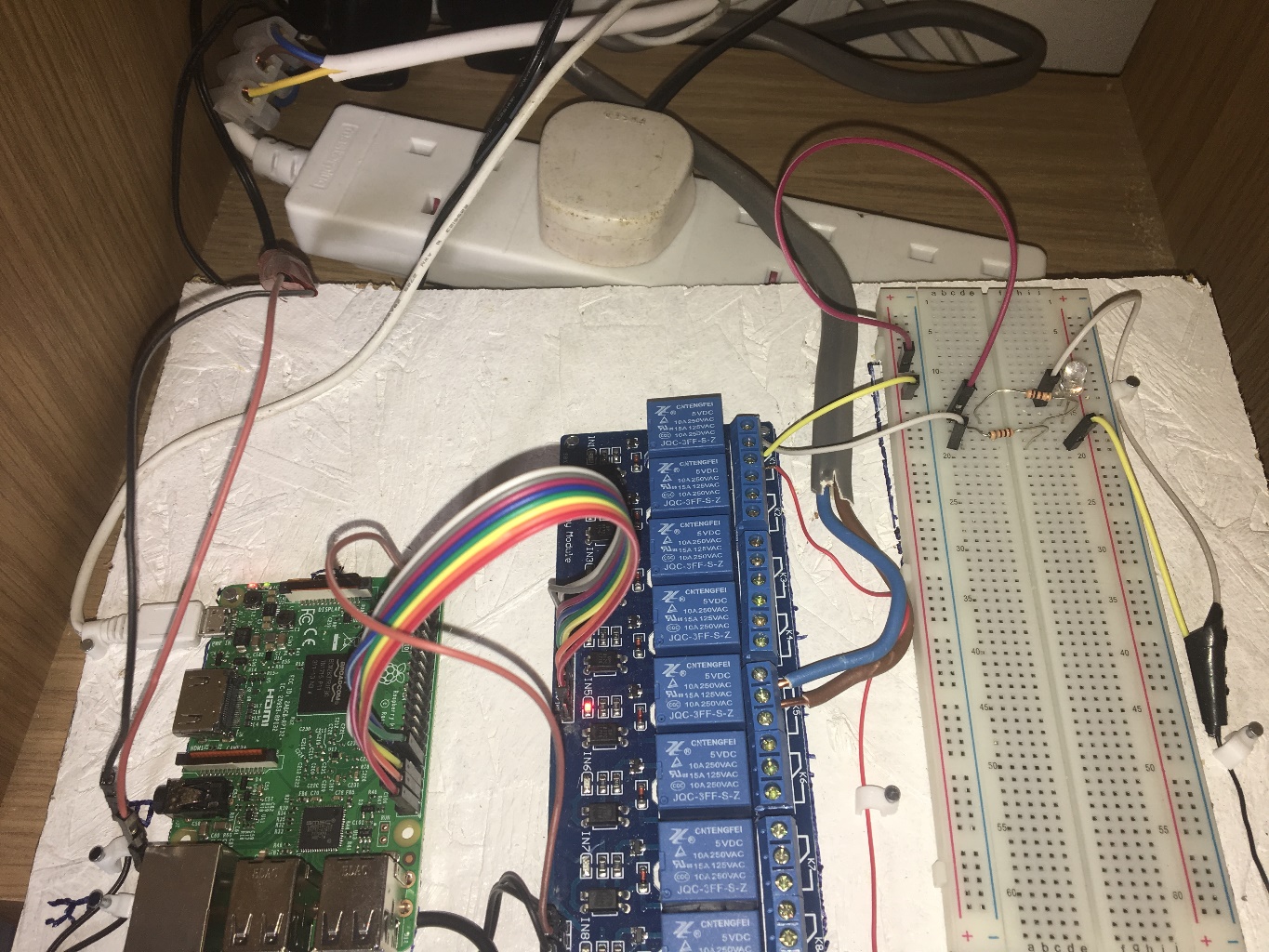




## Appendix I

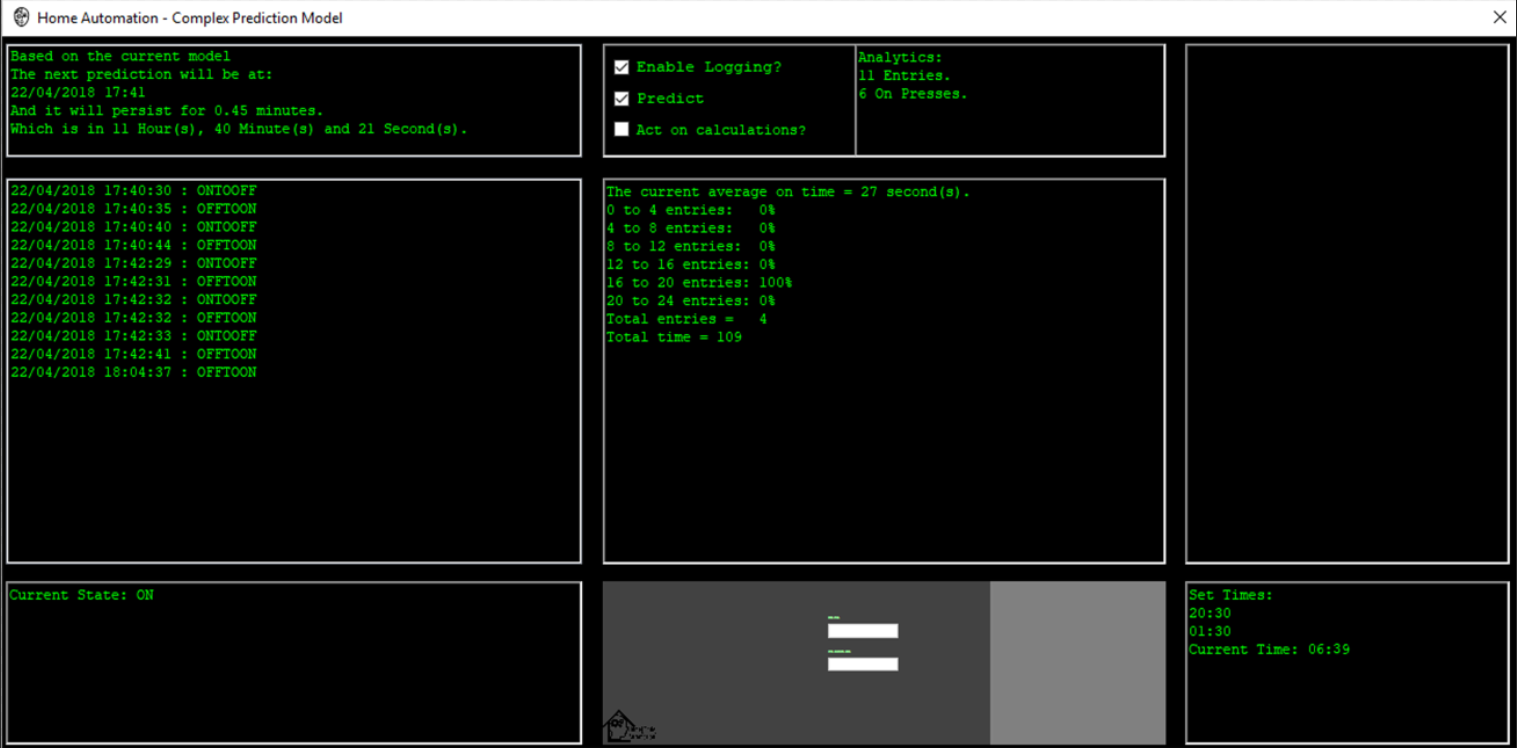
Photographic evidence of the product mounted on MDF





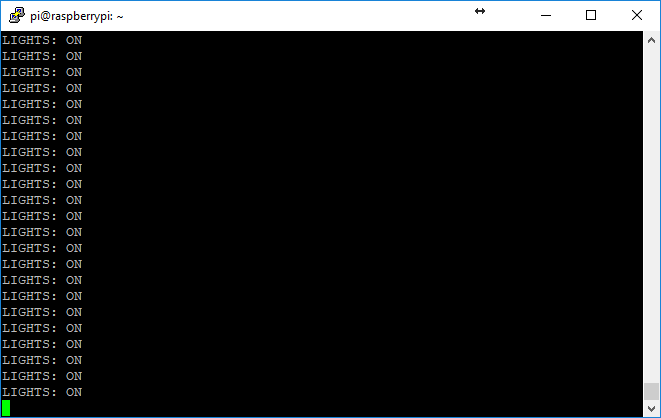
## Appendix J

Photographic evidence of final Prediction Model



## Appendix K

Putty running the Python Script



## Appendix S

Gantt Chart – Submitted as separate document

# 11 Glossary

|  |  |
| --- | --- |
| Wamp | WAMP. Stands for "Windows, Apache, MySQL, and PHP." WAMP is often installed as a software bundle (Apache, MySQL, and PHP). It is often used for web development and internal testing, but may also be used to serve live websites. |
| Pi | The Raspberry Pi is a small single-board computer developed in the United Kingdom |
| Network | a set of computers connected together for the purpose of sharing resources. Other shared resources can include a printer or a file server. |
| Python | Python is an interpreted high-level programming language for general-purpose programming. |
| Script | A script is a program that carries out a specific task and is written in a programming language. |
| Front end | What the client sees of the product, in this case it would be the web site as they would never see the Pi script or prediction algorithm |
| Relay channels | Allows a voltage to be applied to it which will close or open a circuit connected to it. |
| Web app | In computing, a web application or web app is a client–server computer program which the client runs in a web browser. |
| Data packets | A data packet is a unit of data made into a single package that travels along a given network path. |
| Networked devices | Any device that is part of the network |
| Linux | Linux is a family of free and open-source software operating systems built around the Linux kernel. |
| Server | a computer or computer program which manages access to a centralized resource or service in a network. |
| Raspbian | Raspbian is the Pi’s official supported operating system. |
| 'hello world' stage | The first stage of any computer based project, stemming from the first ever program created that simply displayed ‘hello world’ |
| PHP | PHP is a general-purpose scripting language that is especially suited to server-side web development |
| WLAN IP | External IP address, like the address of your house but for your computer |
| DHCP Reservation | DHCP reservation is a permanent local IP address assignment |
| Firewall exception | Allow certain data into your computer, by default windows blocks the ports required to host websites so we must unblock it. |
| Port forwarding | Gateway from local to public network. Allowing the data from your home devices to be accessible all over the world via the internet. |
| Secure Shell | SSH is a software package that enables secure system administration and file transfers over insecure networks. |
| try-catch | A try statement is used to catch exceptions that might be thrown as your program executes. You should use a try statement whenever you use a statement that might throw an exception That way, your program won't crash if the exception occurs. |
| API | A service that a programmer can use so when the program is running it can request live data from an external source. |
| IoT | The Internet of Things (IoT) is the network of physical devices, vehicles, home appliances and other items embedded with electronics, software, sensors, actuators, and connectivity which enables these objects to connect and exchange data. |
| MD5 Encryption | A cryptographic hashing function. |

# 12 Bibliography

Any sources used for research purposes have been submitted in a separate document – Research diary