Cloud-Enabled Raspberry Pi Security System

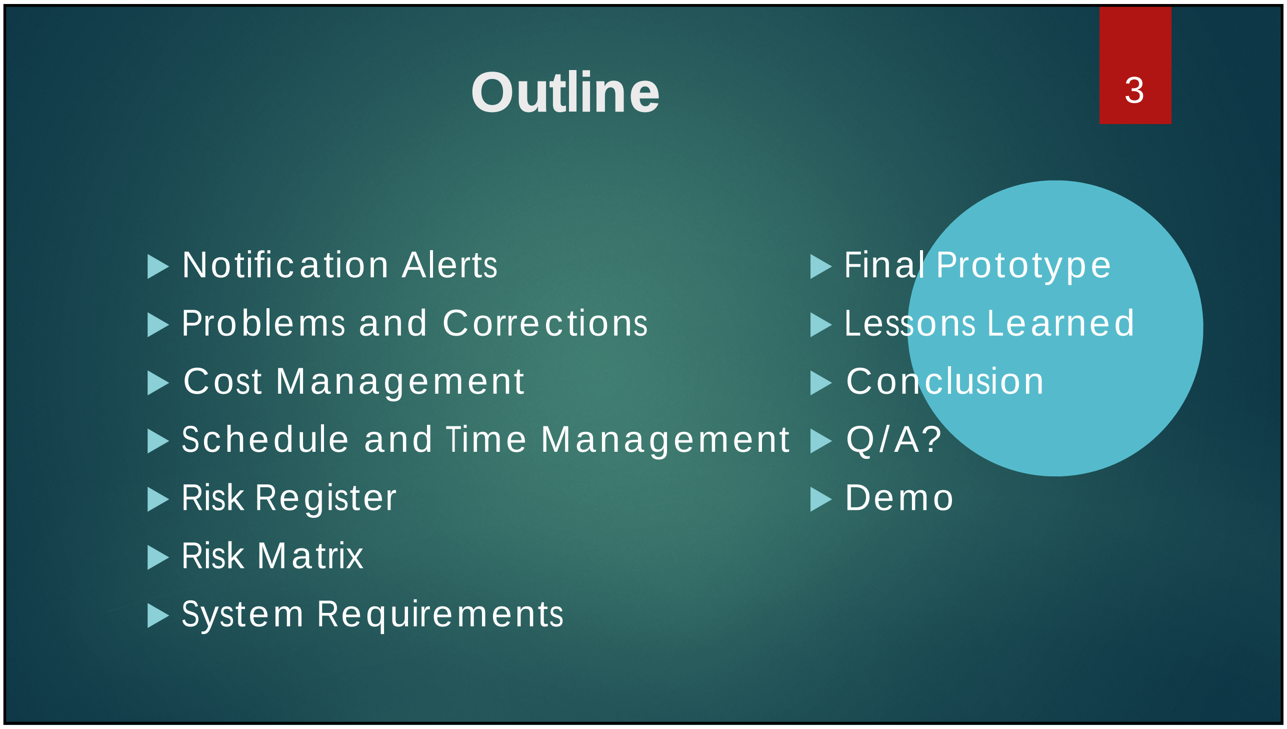
ARCHIE BRADY
ADVISORS: PROFESSOR LIN & PROFESSOR STEFFEN

CPET 491
12/09/2016

Outline

2

- Home Security
- Executive Summary
- Overview
- Circuit Design
- Software Design
- Notification Alerts
- Project Architecture
- Components
- Raspberry Pi Pinout
- Circuit Design
- Circuit Testing (Breadboard)
- Circuit Testing (Perf-board)
- PrivateEyePi Website
- Software Flow Chart
- Software Design



Outline

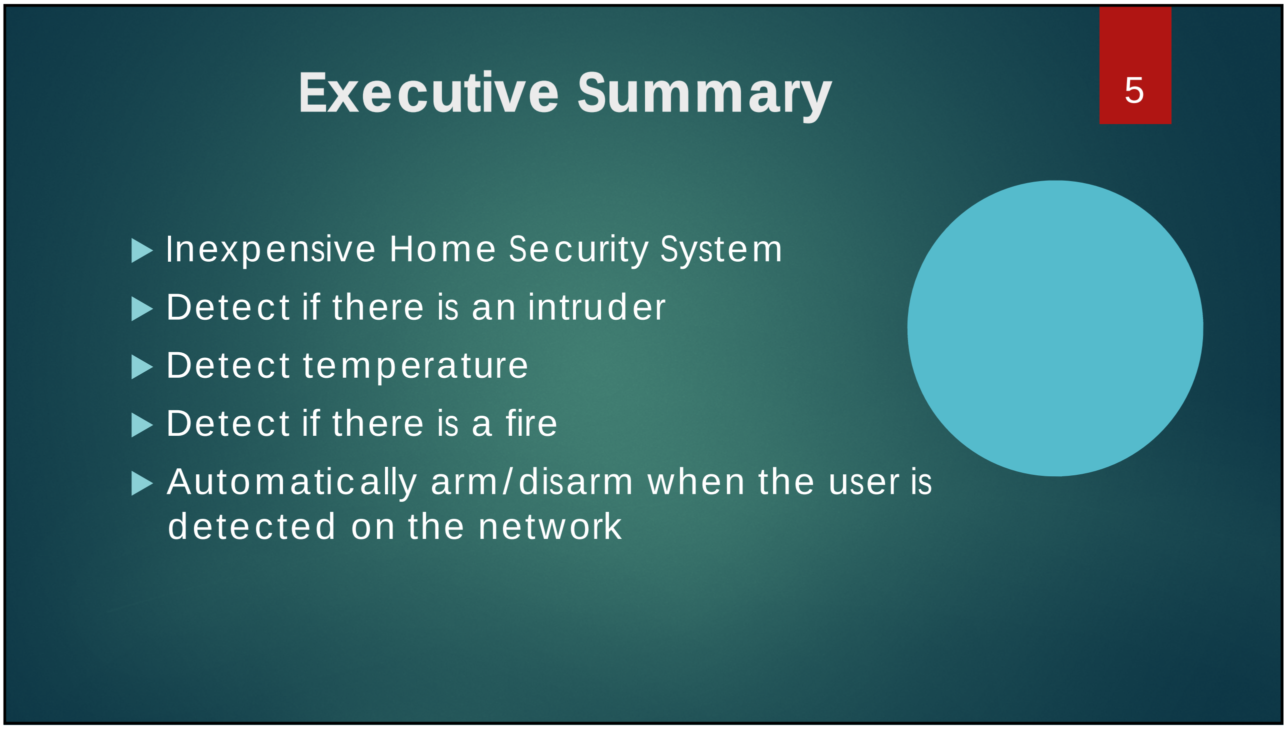
3

- Notification Alerts
- Problems and Corrections
- Cost Management
- Schedule and Time Management
- Risk Register
- Risk Matrix
- System Requirements
- Final Prototype
- Lessons Learned
- Conclusion
- Q/A?
- Demo

Home Security

4

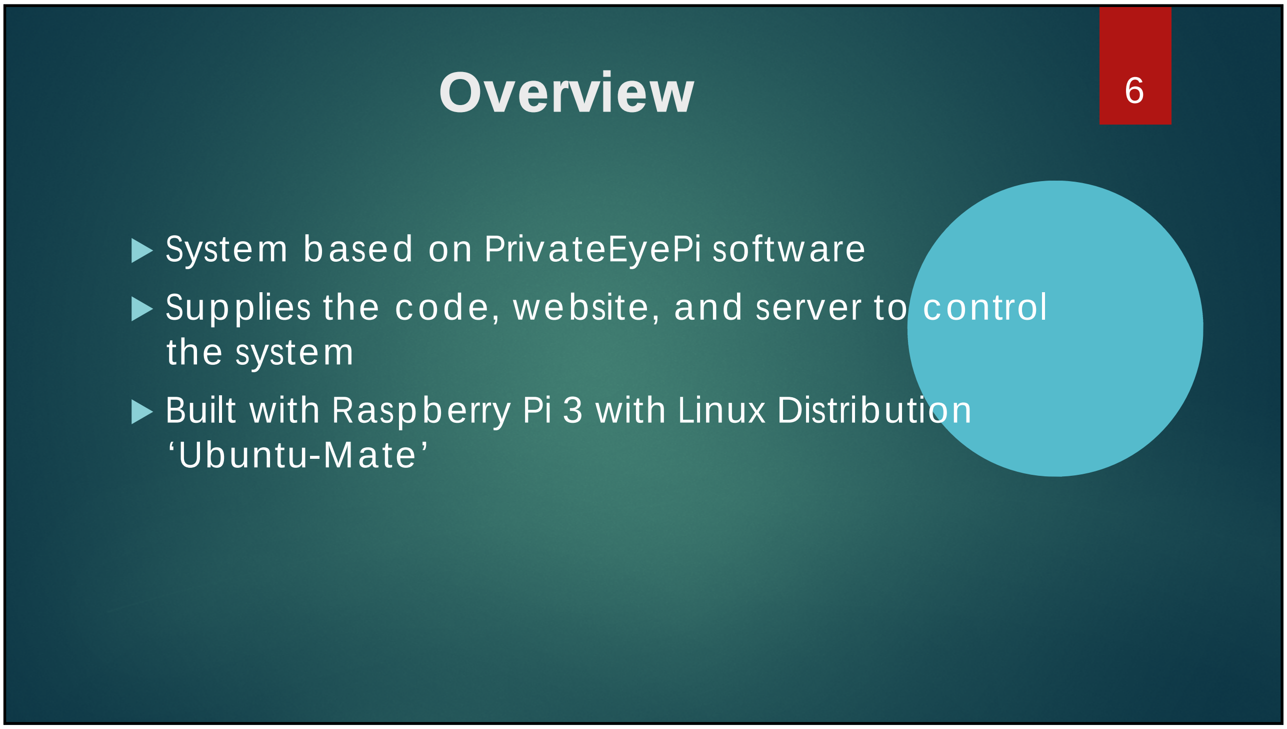
- According to www.safewise.com, in 2014, only 13.6% of home invasions resulted in arrests
- Over 86% go unsolved
- Can be prevented, solved with Home Security System



Executive Summary

5

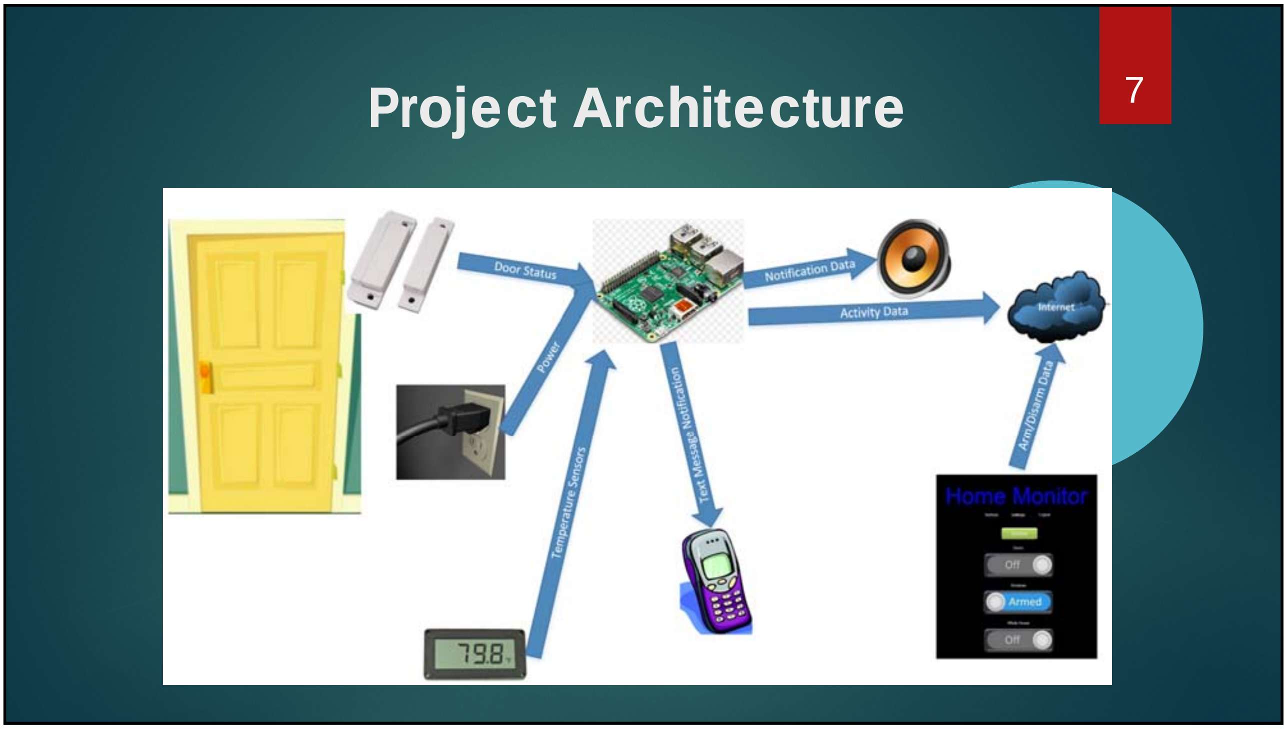
- Inexpensive Home Security System
- Detect if there is an intruder
- Detect temperature
- Detect if there is a fire
- Automatically arm/disarm when the user is detected on the network



Overview

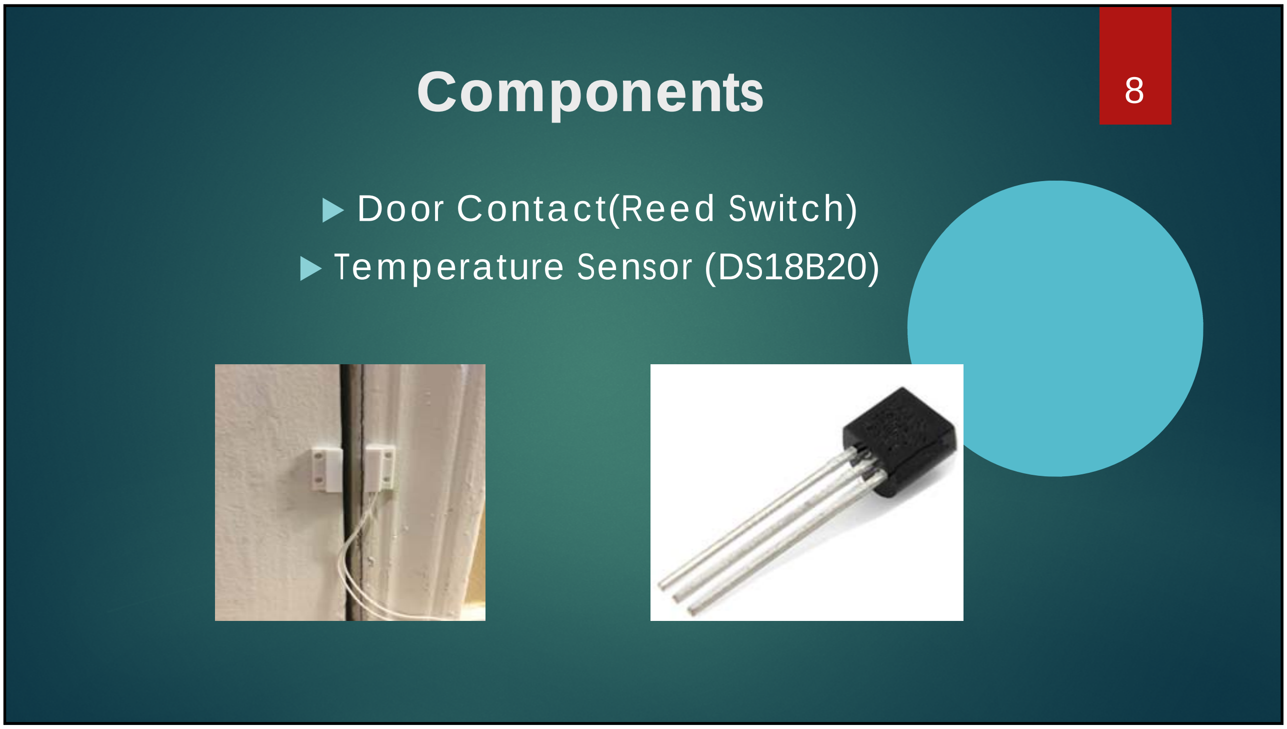
6

- System based on PrivateEyePi software
- Supplies the code, website, and server to control the system
- Built with Raspberry Pi 3 with Linux Distribution ‘Ubuntu-Mate’



Project Architecture

7



Components

8

- Door Contact(Reed Switch)
- Temperature Sensor (DS18B20)

Components

9

- Siren (Active Buzzer 2500 Hz)
- Flame Sensor

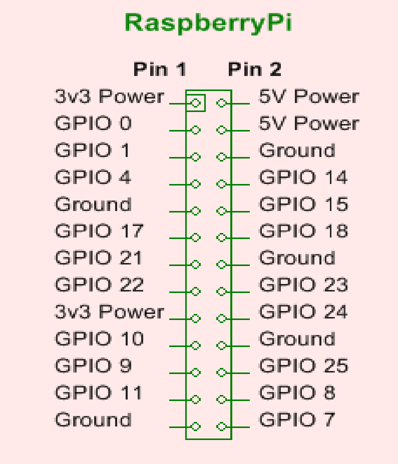
Components

10

- Raspberry Pi 3

Raspberry Pi Pinout

11

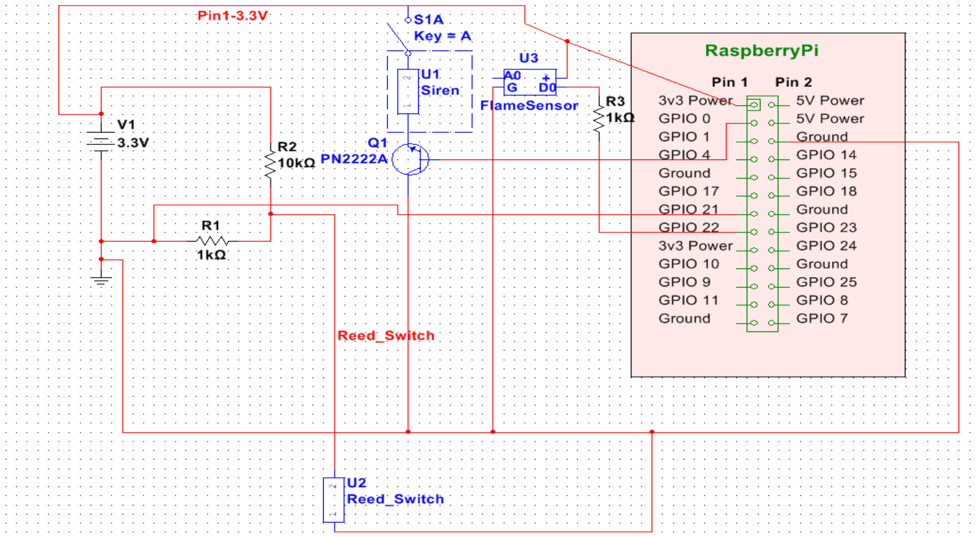


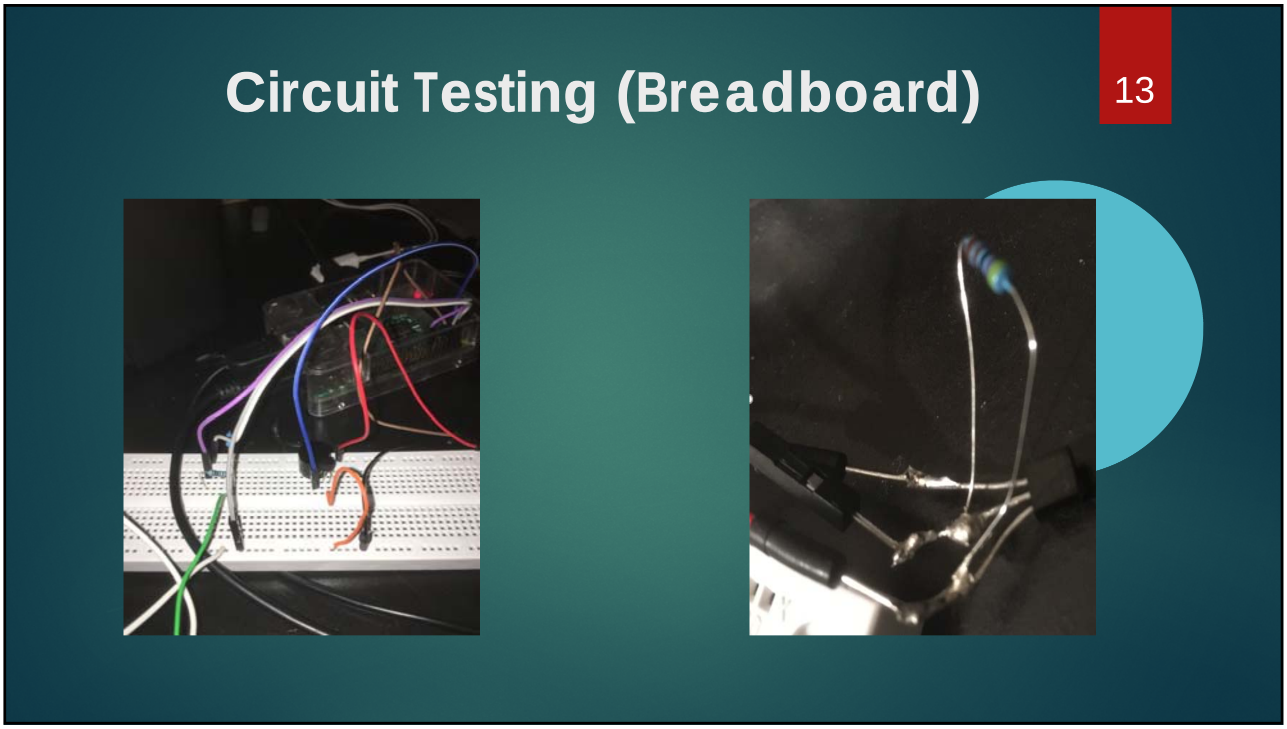
RaspberryPi

Pin 1	Pin 2
3v3 Power	5V Power
GPIO 0	5V Power
GPIO 1	Ground
GPIO 4	GPIO 14
Ground	GPIO 15
GPIO 17	GPIO 18
GPIO 21	Ground
GPIO 22	GPIO 23
3v3 Power	GPIO 24
GPIO 10	Ground
GPIO 9	GPIO 25
GPIO 11	GPIO 8
Ground	GPIO 7

Circuit Design

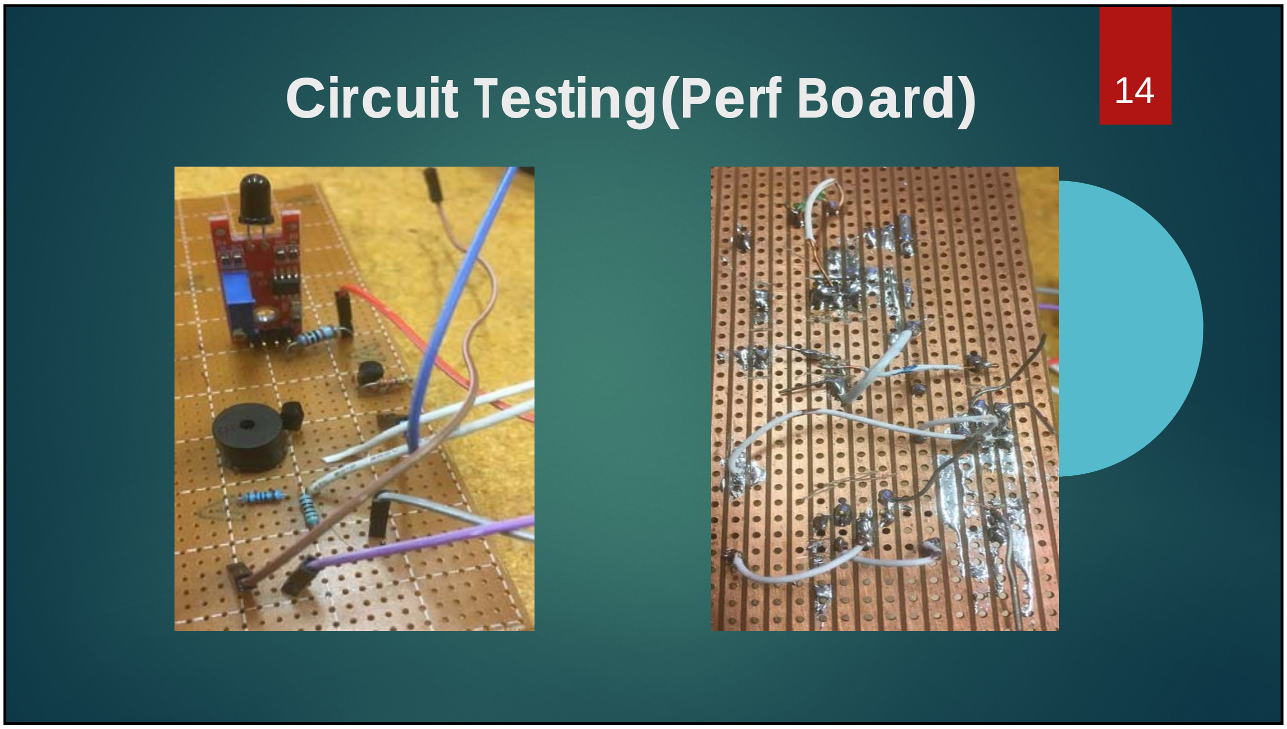
12





Circuit Testing (Breadboard)

13

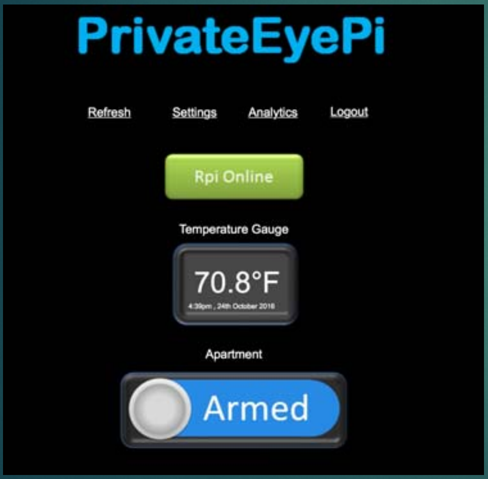


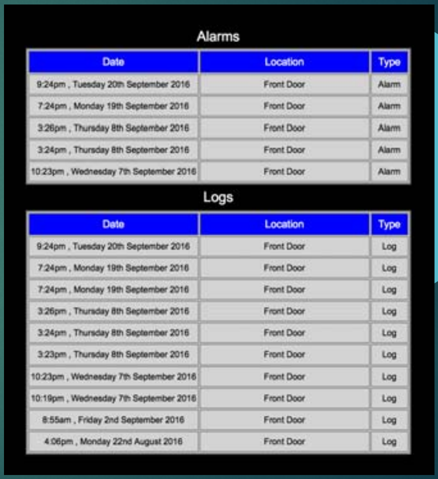
Circuit Testing(Perf Board)

14

PrivateEyePi Website

15





Alarms

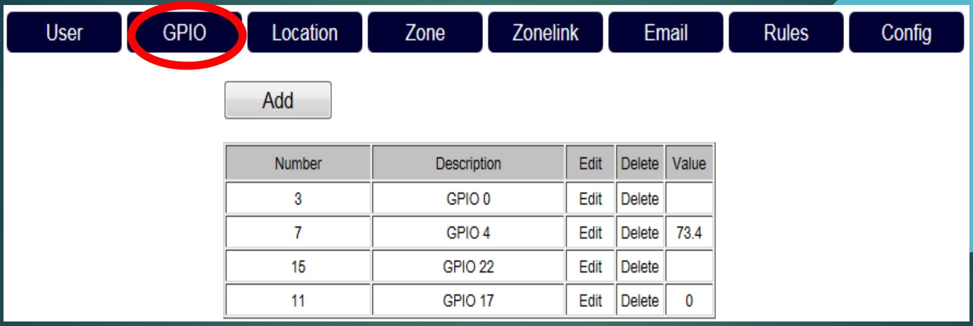
Date	Location	Type
9:24pm , Tuesday 20th September 2016	Front Door	Alarm
7:24pm , Monday 19th September 2016	Front Door	Alarm
3:26pm , Thursday 8th September 2016	Front Door	Alarm
3:24pm , Thursday 8th September 2016	Front Door	Alarm
10:23pm , Wednesday 7th September 2016	Front Door	Alarm

Logs

Date	Location	Type
9:24pm , Tuesday 20th September 2016	Front Door	Log
7:24pm , Monday 19th September 2016	Front Door	Log
7:24pm , Monday 19th September 2016	Front Door	Log
3:26pm , Thursday 8th September 2016	Front Door	Log
3:24pm , Thursday 8th September 2016	Front Door	Log
3:23pm , Thursday 8th September 2016	Front Door	Log
10:23pm , Wednesday 7th September 2016	Front Door	Log
10:19pm , Wednesday 7th September 2016	Front Door	Log
8:55am , Friday 2nd September 2016	Front Door	Log
4:06pm , Monday 22nd August 2016	Front Door	Log

PrivateEyePi Website

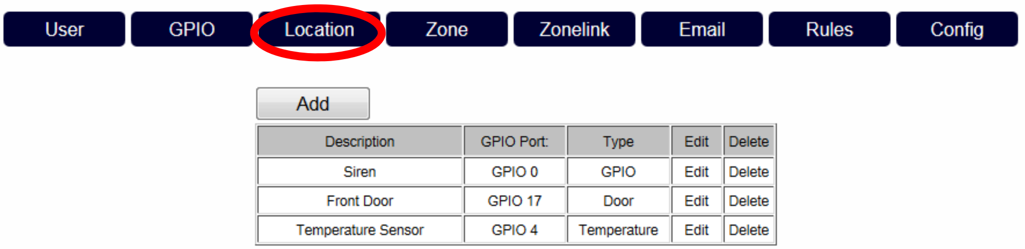
16



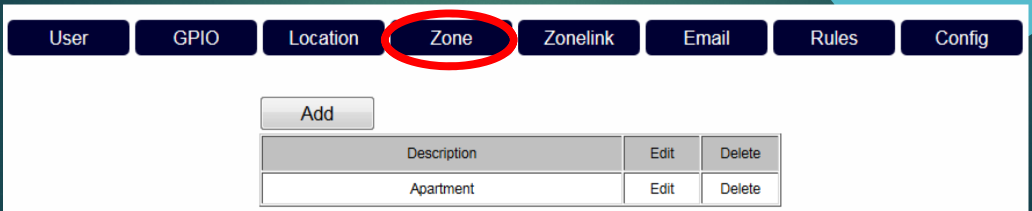
Number	Description	Edit	Delete	Value
3	GPIO 0	Edit	Delete	
7	GPIO 4	Edit	Delete	73.4
15	GPIO 22	Edit	Delete	
11	GPIO 17	Edit	Delete	0

PrivateEyePi Website

17



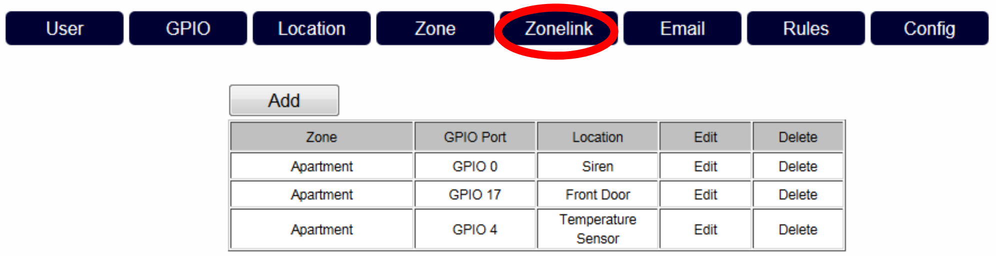
Description	GPIO Port:	Type	Edit	Delete
Siren	GPIO 0	GPIO	Edit	Delete
Front Door	GPIO 17	Door	Edit	Delete
Temperature Sensor	GPIO 4	Temperature	Edit	Delete



Description	Edit	Delete
Apartment	Edit	Delete

PrivateEyePi Website

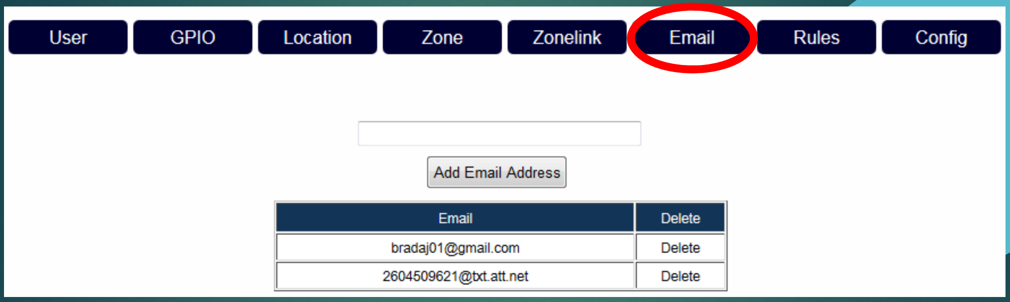
18



Zone	GPIO Port	Location	Edit	Delete
Apartment	GPIO 0	Siren	Edit	Delete
Apartment	GPIO 17	Front Door	Edit	Delete
Apartment	GPIO 4	Temperature Sensor	Edit	Delete

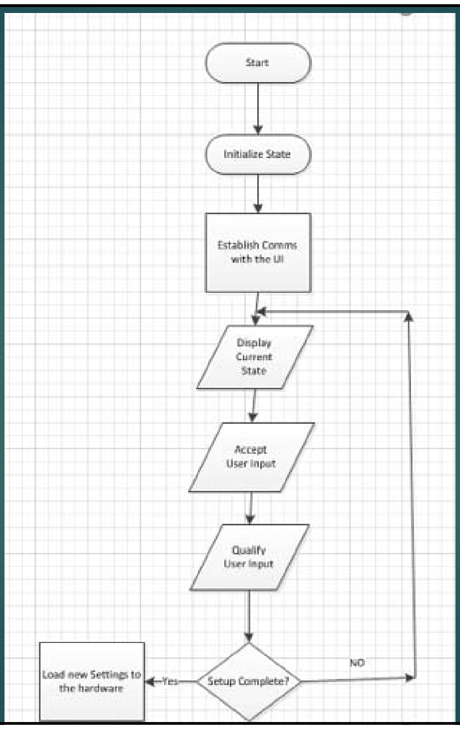
PrivateEyePi Website

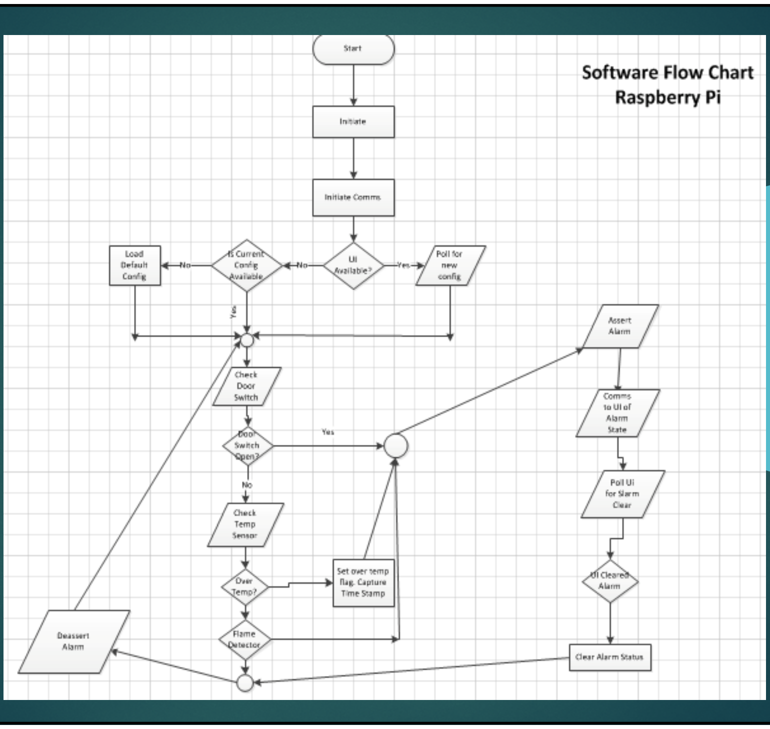
19



Email	Delete
bradaj01@gmail.com	Delete
2604509621@txt.att.net	Delete

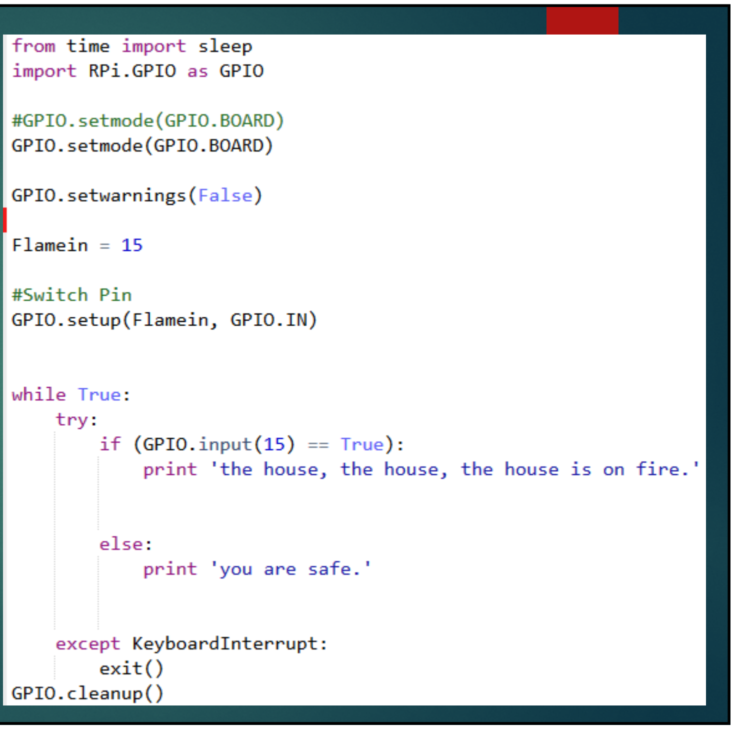
20





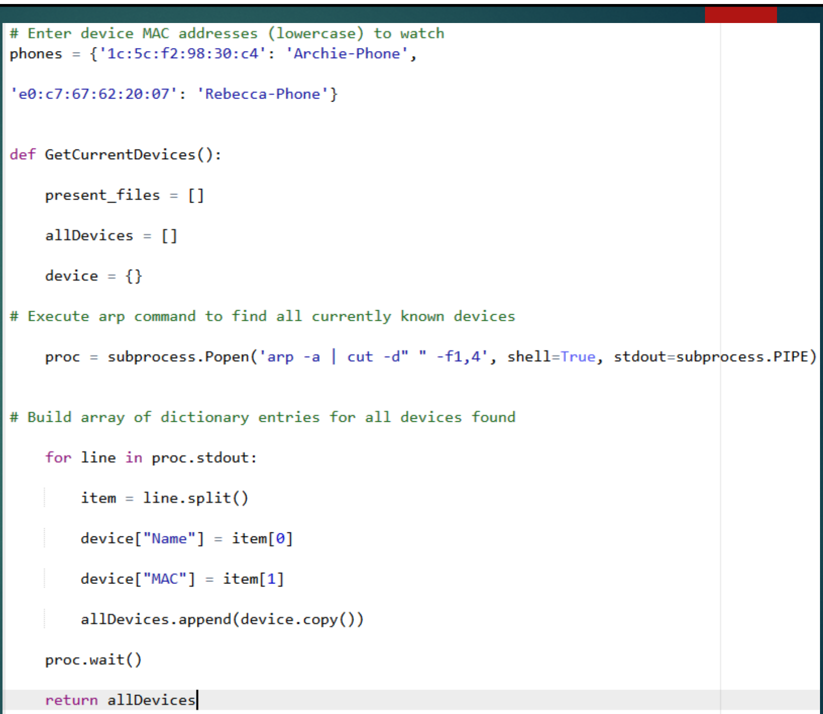
21

Flame Sensor



```
from time import sleep
import RPi.GPIO as GPIO

#GPIO.setmode(GPIO.BOARD)
GPIO.setmode(GPIO.BOARD)

GPIO.setwarnings(False)

Flamein = 15

#Switch Pin
GPIO.setup(Flamein, GPIO.IN)


while True:
    try:
        if (GPIO.input(15) == True):
            print 'the house, the house, the house is on fire.'


        else:
            print 'you are safe.'


    except KeyboardInterrupt:
        exit()
GPIO.cleanup()
```

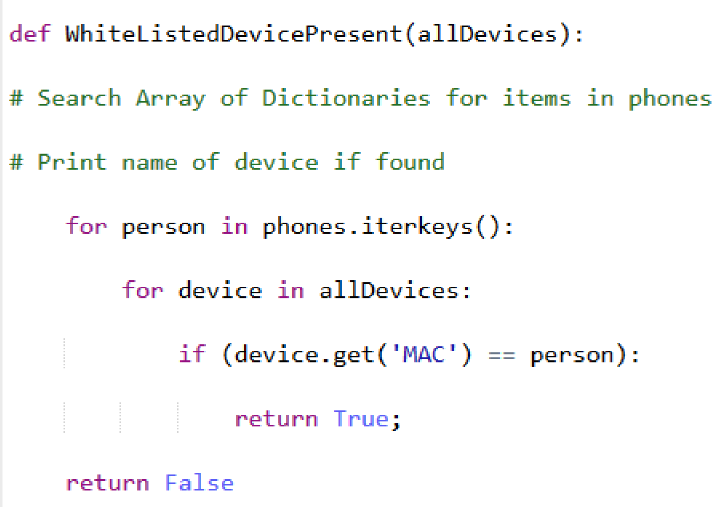
Network Scanner 2



```
# Enter device MAC addresses (lowercase) to watch
phones = {'1c:5c:f2:98:30:c4': 'Archie-Phone',

'e0:c7:67:62:20:07': 'Rebecca-Phone'}


def GetCurrentDevices():

    present_files = []

    allDevices = []

    device = {}

# Execute arp command to find all currently known devices

    proc = subprocess.Popen('arp -a | cut -d" " -f1,4', shell=True, stdout=subprocess.PIPE)

# Build array of dictionary entries for all devices found

    for line in proc.stdout:

        item = line.split()

        device["Name"] = item[0]

        device["MAC"] = item[1]

        allDevices.append(device.copy())

    proc.wait()

    return allDevices
```

NetworkScanner2

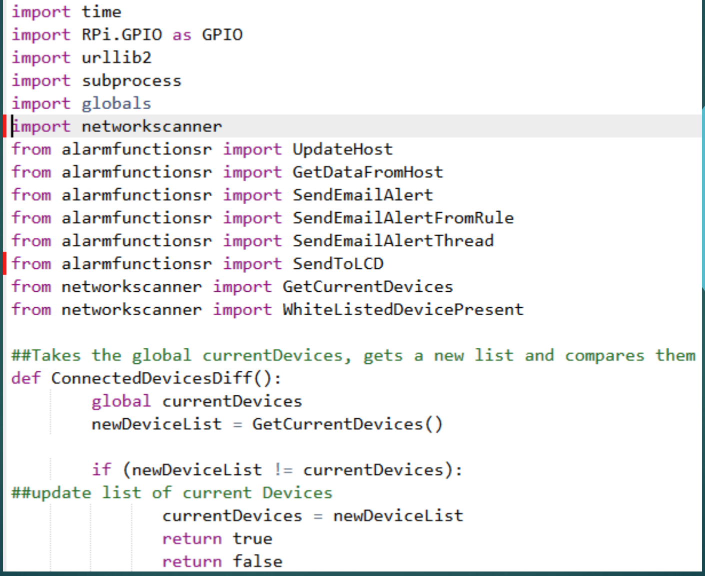
24



```
def WhiteListedDevicePresent(allDevices):

# Search Array of Dictionaries for items in phones

# Print name of device if found

    for person in phones.iterkeys():

        for device in allDevices:

            if (device.get('MAC') == person):

                return True;

    return False
```

Alarmup2.py

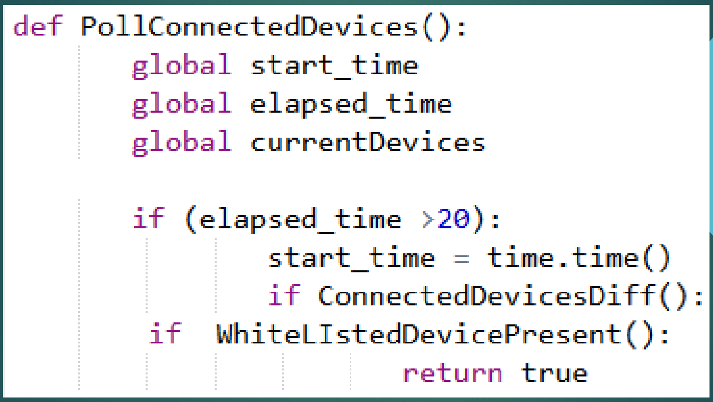
25



```
import time
import RPi.GPIO as GPIO
import urllib2
import subprocess
import globals
import networkscanner
from alarmfunctionsr import UpdateHost
from alarmfunctionsr import GetDataFromHost
from alarmfunctionsr import SendEmailAlert
from alarmfunctionsr import SendEmailAlertFromRule
from alarmfunctionsr import SendEmailAlertThread
from alarmfunctionsr import SendToLCD
from networkscanner import GetCurrentDevices
from networkscanner import WhiteListedDevicePresent

##Takes the global currentDevices, gets a new list and compares them
def ConnectedDevicesDiff():
        global currentDevices
        newDeviceList = GetCurrentDevices()

        if (newDeviceList != currentDevices):
##update list of current Devices
               currentDevices = newDeviceList
               return true
               return false
```

Alarmup2.py

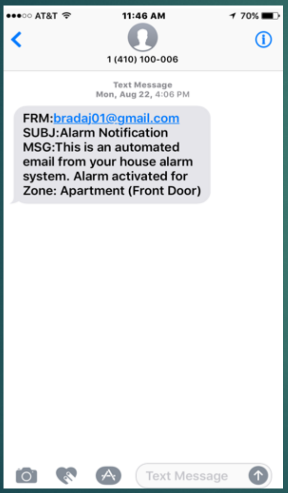
26



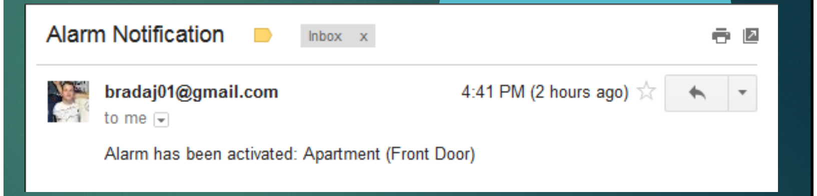
```
def PollConnectedDevices():
       global start_time
       global elapsed_time
       global currentDevices

       if (elapsed_time >20):
               start_time = time.time()
               if ConnectedDevicesDiff():
        if  WhiteLIstedDevicePresent():
                       return true
```

Notification Alerts

27



FRM:bradaj01@gmail.com
SUBJ:Alarm Notification
MSG:This is an automated email from your house alarm system. Alarm activated for Zone: Apartment (Front Door)

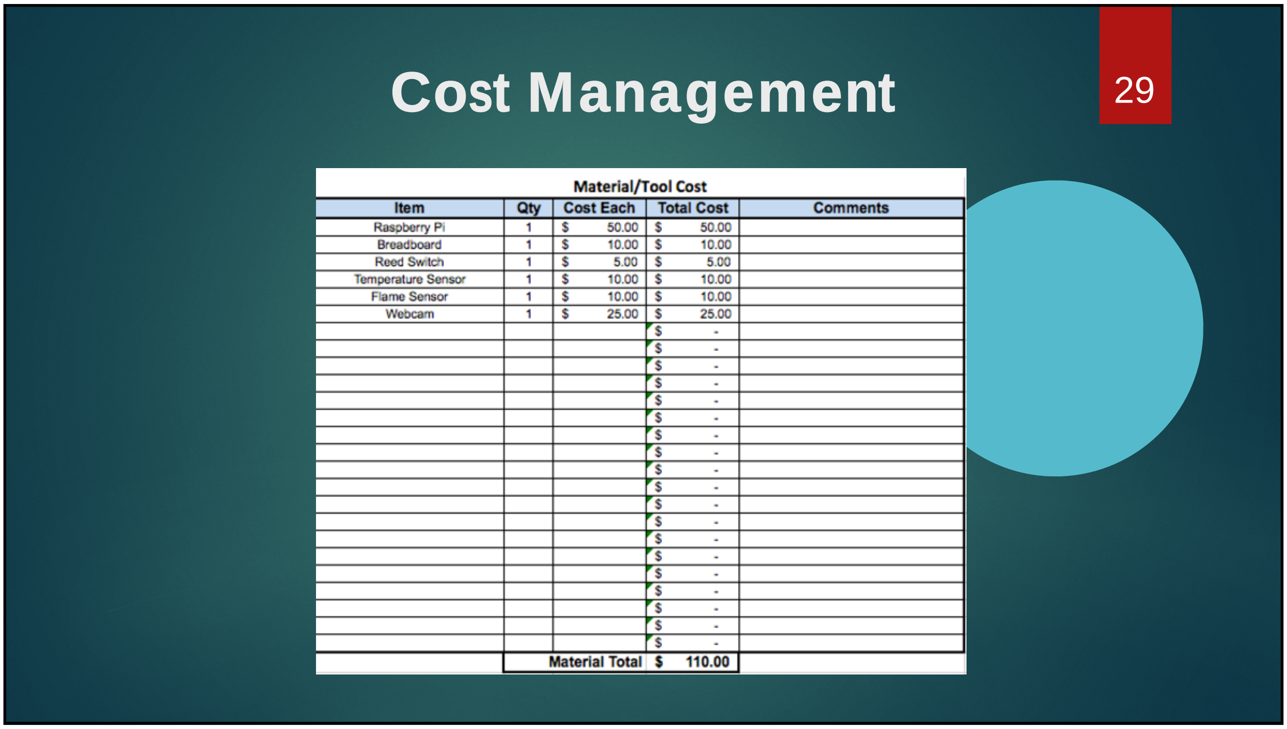


Alarm has been activated: Apartment (Front Door)

Problems and Corrections

28

- Changed from scanning IP address to MAC address
- Switched operating systems from Raspbian to Ubuntu-Mate due to Internet connectivity issues
- Issues with the automatically arming/disarming

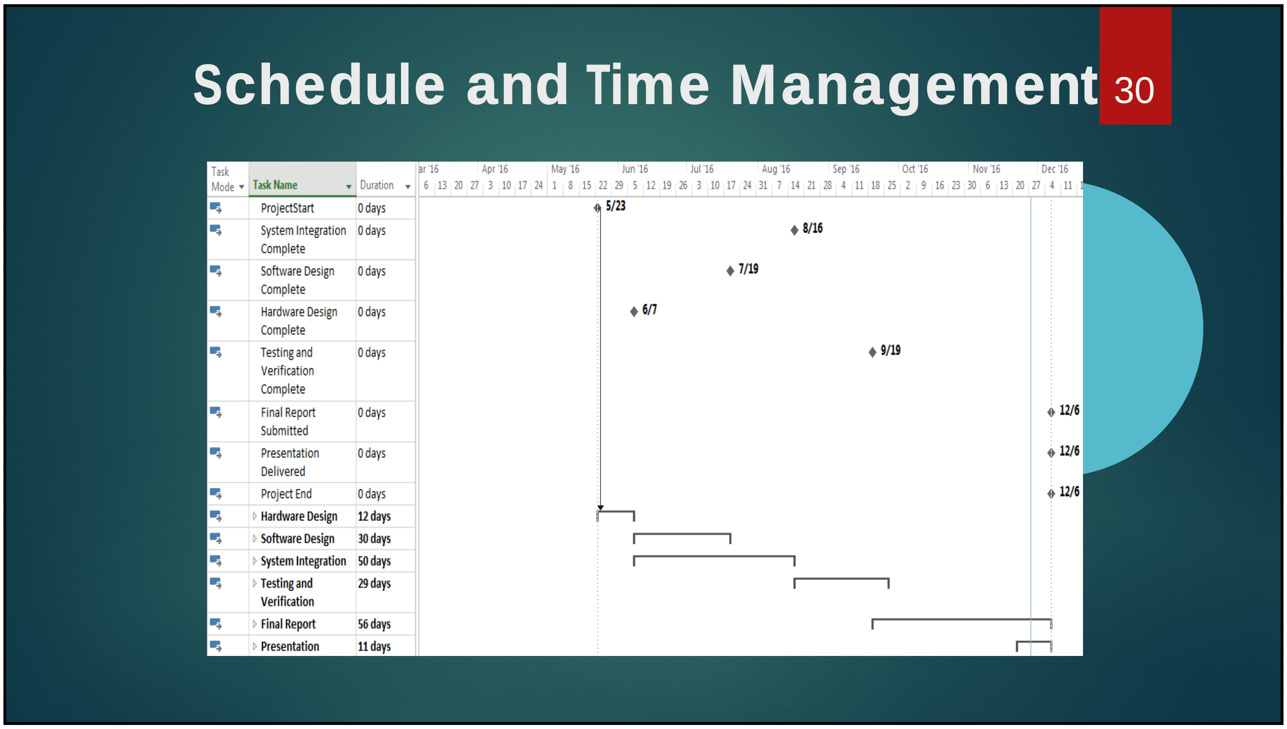


Cost Management

29

Material/Tool Cost

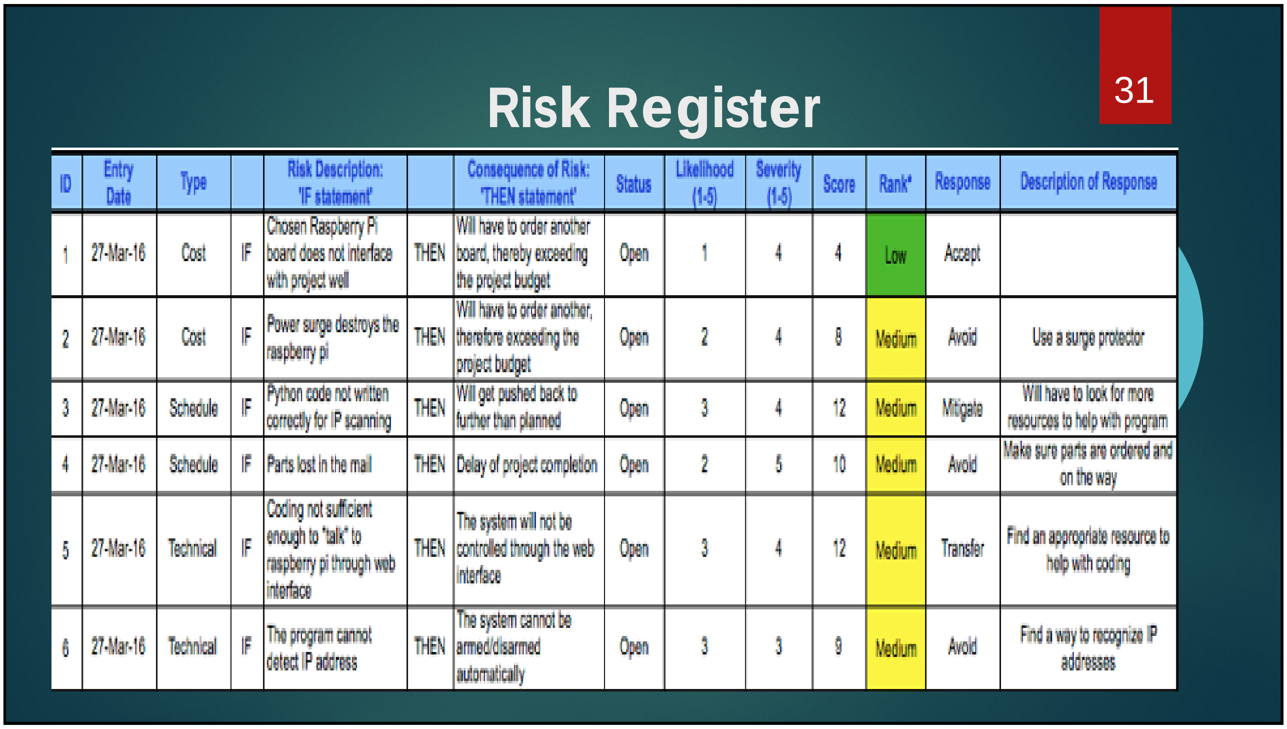
Item	Qty	Cost Each	Total Cost	Comments
Raspberry Pi	1	$ 50.00	$ 50.00	
Breadboard	1	$ 10.00	$ 10.00	
Reed Switch	1	$ 5.00	$ 5.00	
Temperature Sensor	1	$ 10.00	$ 10.00	
Flame Sensor	1	$ 10.00	$ 10.00	
Webcam	1	$ 25.00	$ 25.00	
			$ -	
			$ -	
			$ -	
			$ -	
			$ -	
			$ -	
			$ -	
			$ -	
			$ -	
			$ -	
			$ -	
			$ -	
			$ -	
			$ -	
			$ -	
			$ -	
			$ -	
			$ -	
			$ -	
		Material Total	$ 110.00	



Schedule and Time Management

30

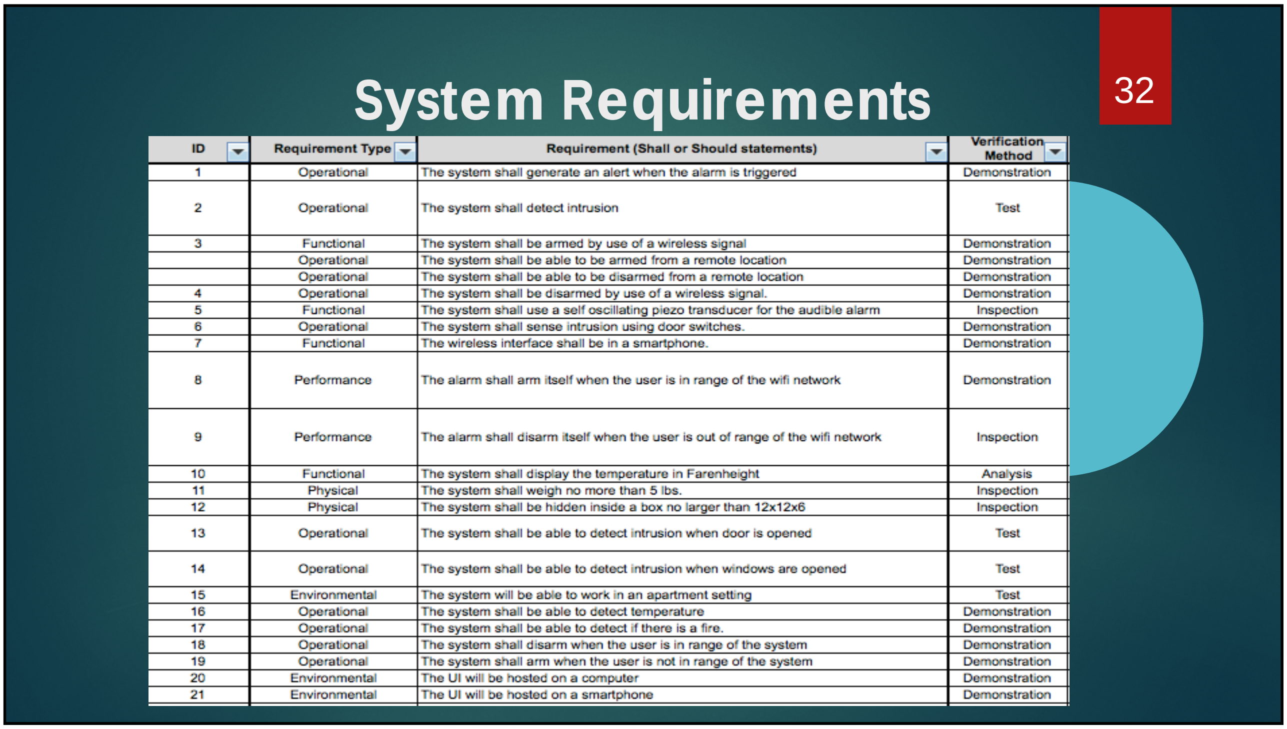
Task Mode	Task Name	Duration
	ProjectStart	0 days
	System Integration Complete	0 days
	Software Design Complete	0 days
	Hardware Design Complete	0 days
	Testing and Verification Complete	0 days
	Final Report Submitted	0 days
	Presentation Delivered	0 days
	Project End	0 days
	Hardware Design	12 days
	Software Design	30 days
	System Integration	50 days
	Testing and Verification	29 days
	Final Report	56 days
	Presentation	11 days



31

Risk Register

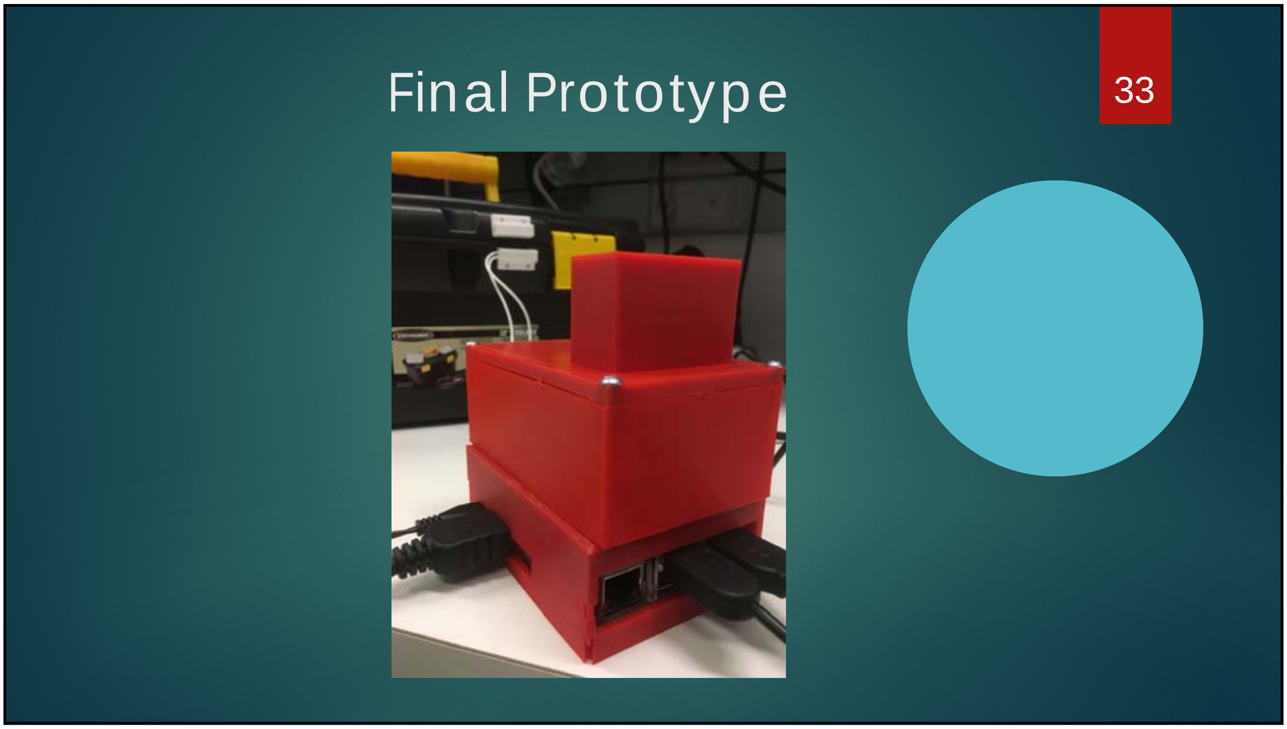
ID	Entry Date	Type		Risk Description: 'IF statement'		Consequence of Risk: 'THEN statement'	Status	Likelihood (1-5)	Severity (1-5)	Score	Rank*	Response	Description of Response
1	27-Mar-16	Cost	IF	Chosen Raspberry Pi board does not interface with project well	THEN	Will have to order another board, thereby exceeding the project budget	Open	1	4	4	Low	Accept	
2	27-Mar-16	Cost	IF	Power surge destroys the raspberry pi	THEN	Will have to order another, therefore exceeding the project budget	Open	2	4	8	Medium	Avoid	Use a surge protector
3	27-Mar-16	Schedule	IF	Python code not written correctly for IP scanning	THEN	Will get pushed back to further than planned	Open	3	4	12	Medium	Mitigate	Will have to look for more resources to help with program
4	27-Mar-16	Schedule	IF	Parts lost in the mail	THEN	Delay of project completion	Open	2	5	10	Medium	Avoid	Make sure parts are ordered and on the way
5	27-Mar-16	Technical	IF	Coding not sufficient enough to "talk" to raspberry pi through web interface	THEN	The system will not be controlled through the web interface	Open	3	4	12	Medium	Transfer	Find an appropriate resource to help with coding
6	27-Mar-16	Technical	IF	The program cannot detect IP address	THEN	The system cannot be armed/disarmed automatically	Open	3	3	9	Medium	Avoid	Find a way to recognize IP addresses



32

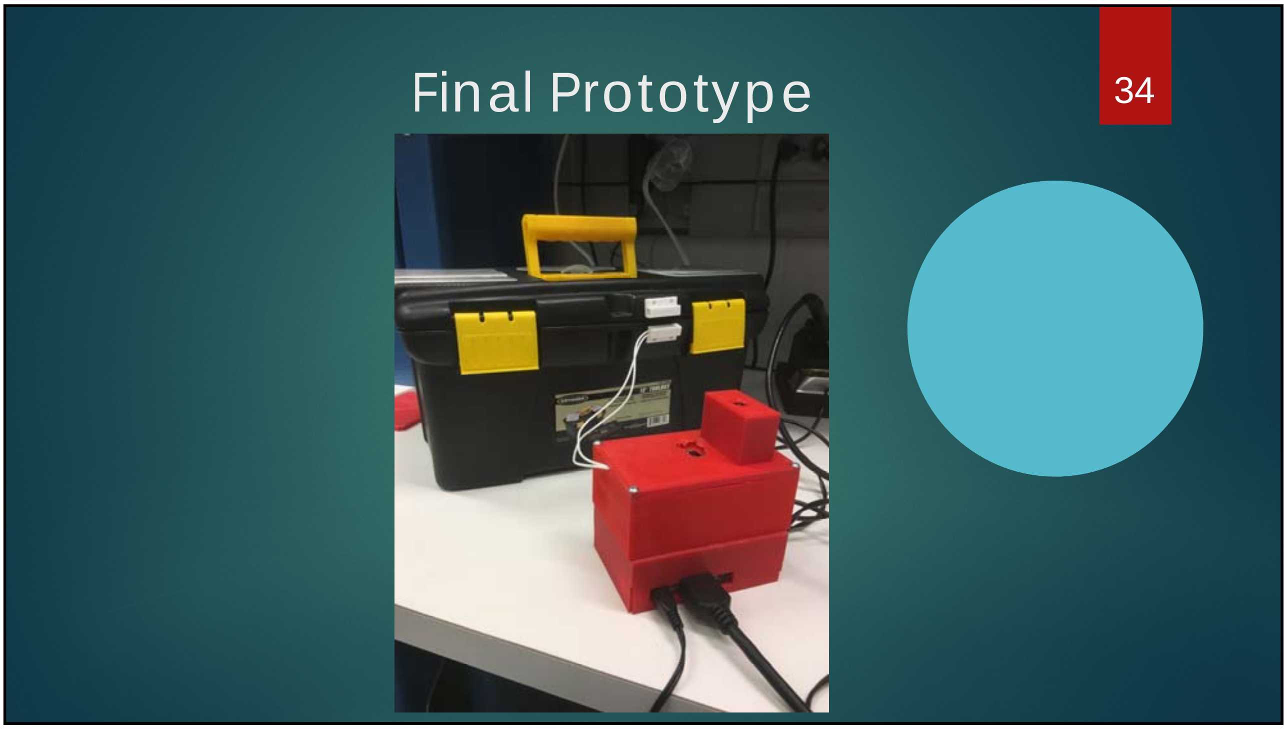
System Requirements

ID	Requirement Type	Requirement (Shall or Should statements)	Verification Method
1	Operational	The system shall generate an alert when the alarm is triggered	Demonstration
2	Operational	The system shall detect intrusion	Test
3	Functional	The system shall be armed by use of a wireless signal	Demonstration
	Operational	The system shall be able to be armed from a remote location	Demonstration
	Operational	The system shall be able to be disarmed from a remote location	Demonstration
4	Operational	The system shall be disarmed by use of a wireless signal.	Demonstration
5	Functional	The system shall use a self oscillating piezo transducer for the audible alarm	Inspection
6	Operational	The system shall sense intrusion using door switches.	Demonstration
7	Functional	The wireless interface shall be in a smartphone.	Demonstration
8	Performance	The alarm shall arm itself when the user is in range of the wifi network	Demonstration
9	Performance	The alarm shall disarm itself when the user is out of range of the wifi network	Inspection
10	Functional	The system shall display the temperature in Farenheight	Analysis
11	Physical	The system shall weigh no more than 5 lbs.	Inspection
12	Physical	The system shall be hidden inside a box no larger than 12x12x6	Inspection
13	Operational	The system shall be able to detect intrusion when door is opened	Test
14	Operational	The system shall be able to detect intrusion when windows are opened	Test
15	Environmental	The system will be able to work in an apartment setting	Test
16	Operational	The system shall be able to detect temperature	Demonstration
17	Operational	The system shall be able to detect if there is a fire.	Demonstration
18	Operational	The system shall disarm when the user is in range of the system	Demonstration
19	Operational	The system shall arm when the user is not in range of the system	Demonstration
20	Environmental	The UI will be hosted on a computer	Demonstration
21	Environmental	The UI will be hosted on a smartphone	Demonstration



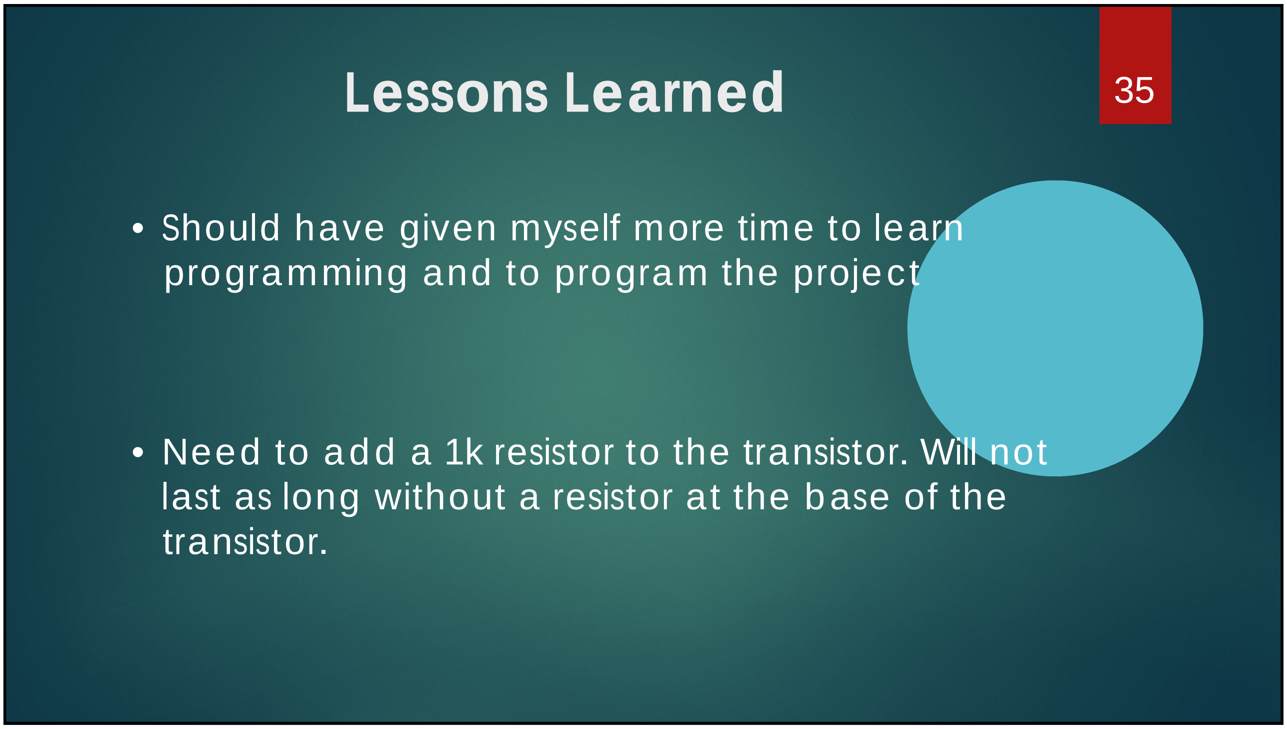
Final Prototype

33



Final Prototype

34

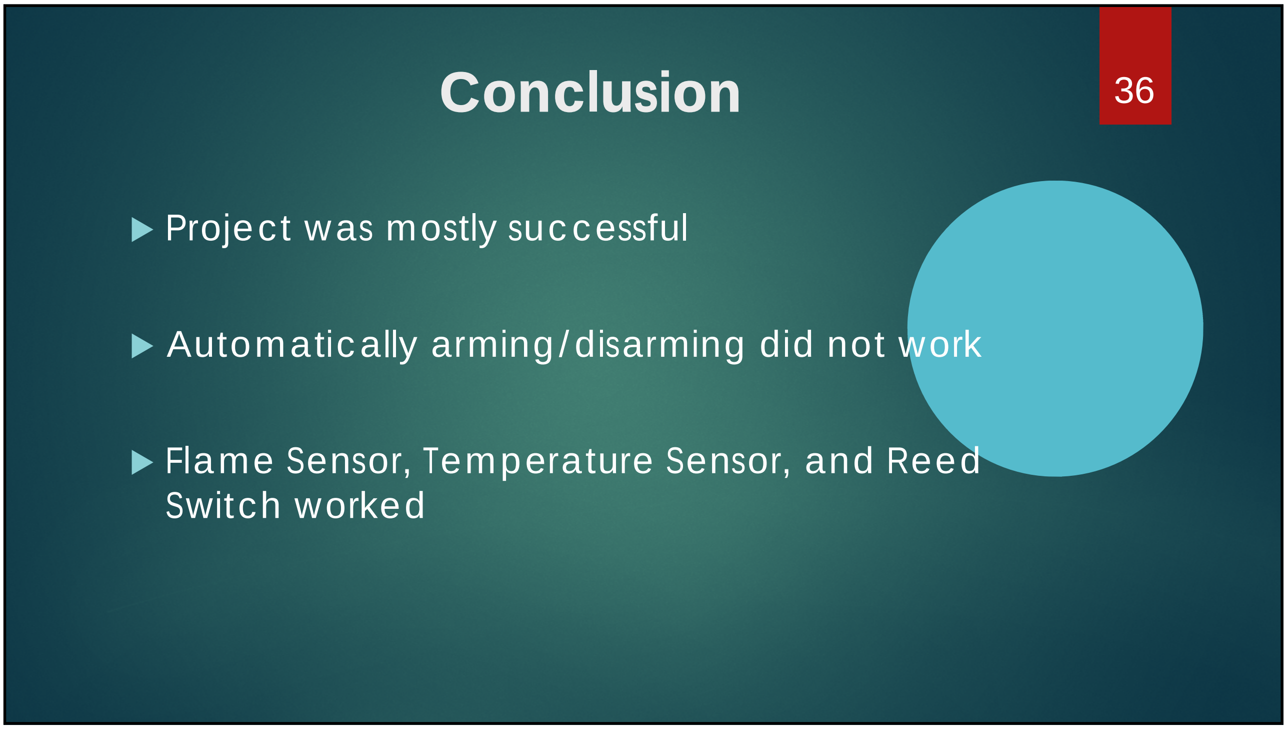


Lessons Learned

35

- Should have given myself more time to learn programming and to program the project

- Need to add a 1k resistor to the transistor. Will not last as long without a resistor at the base of the transistor.



Conclusion

36

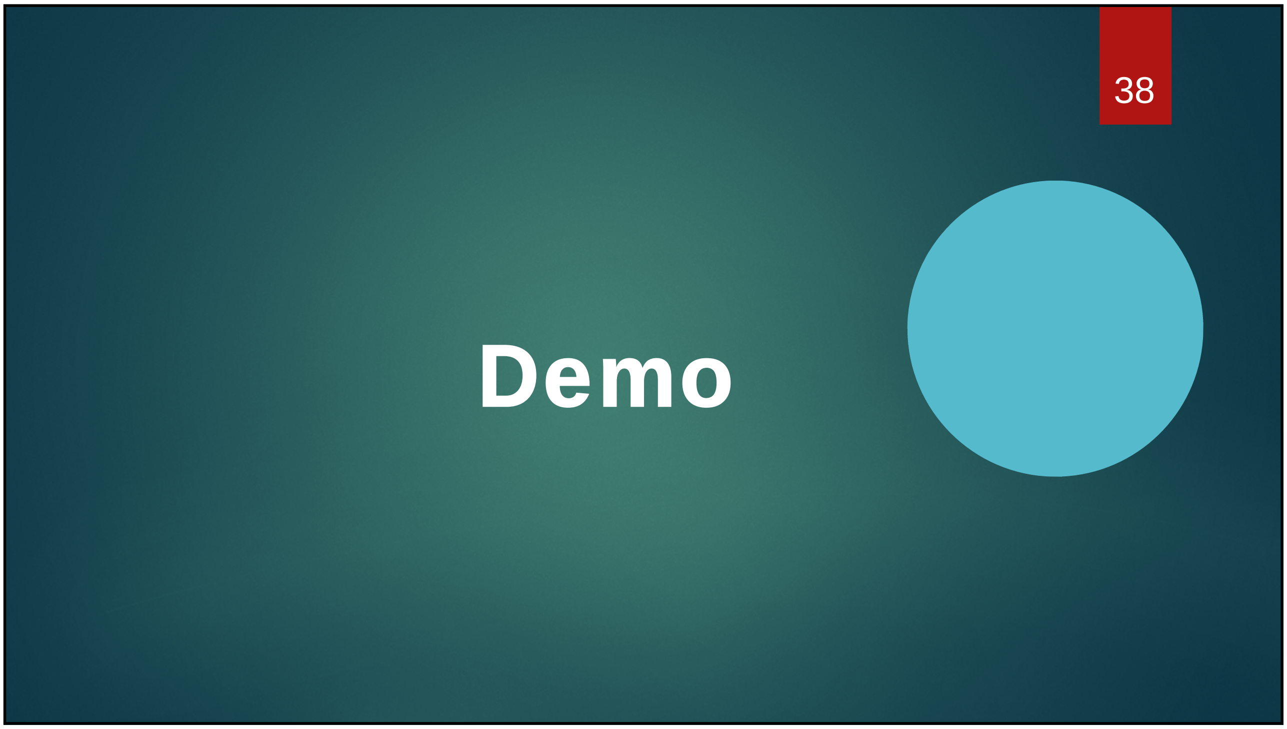
- Project was mostly successful

- Automatically arming/disarming did not work

- Flame Sensor, Temperature Sensor, and Reed Switch worked

37

Q/A?



38

Demo