Interactive Mirror Display

By

Hiraal Doshi

Stephen Hurwit

Pamela Sanan

Design Document for ECE 445, Senior Design, Spring 2019

TA: Nicholas Ratajczyk

21 February 2019

Project No. 29

Contents

1 Introduction . . . 1

1.1 Objective . . . 1

1.2 Background . . . 1

1.3 High-Level Requirements . . . 1

2 Design . . . 2

2.1 Control Unit . . . 4

2.1.1 Microcontroller . . . 4

2.1.2 Raspberry Pi . . . 6

2.2 User Interface . . . 8

2.2.1 Display Monitor . . . 8

2.2.2 Camera Status LED . . . 9

2.2.3 Microphone Status LED . . . 9

2.3 Audio I/O . . . 10

2.3.1 Microphone . . . 10

2.3.2 Speakers . . . 10

2.4 Visual Sensor . . . 11

2.4.1 Camera . . . 12

2.4.2 PIR Sensor Array . . . 13

2.4.3 Proximity Sensors . . . 15

2.5 Power . . . 16

2.5.1 AC Power . . . 16

2.5.2 AC/DC Power Converter . . . 16

2.5.3 Voltage Regulator . . . 16

2.5.4 Power Switch . . . 17

2.6 Requirement Summary . . . 18

2.7 Schematic Diagram . . . 20

2.8 Tolerance Analysis . . . 22

3 Cost and Schedule . . . 24

3.1 Cost Analysis . . . 24

3.1.1 Labor . . . 24

3.1.2 Parts . . . 24

3.2 Schedule 25
4 Ethics and Safety 27
Reference 28

1 Introduction

1.1 Objective

Technology has become a ubiquitous presence in our lives. We have many devices which take up space and clutter up our living environments, providing us with convenient access to interesting features and functionalities in exchange. However, it can be difficult to manage and find space for so many devices, especially those that present information visually.

We propose to address this issue by developing an interactive mirror display which provides context sensitive information and media access to the user through a discreet and unobtrusive device. We intend to integrate gesture recognition, through the use of visual sensors, and voice commands, through the use of Alexa integration, which will allow the user to interact with the device and access further information. The sensors will interface with a microcontroller, the user interface software will be run on a Raspberry Pi, and we will integrate a camera, microphone, and speakers.

1.2 Background

As technology is progressing, we are slowly entering a futuristic age. Smart devices are gaining popularity and are a part of peoples daily lifestyles and households. There is a clear interest from the public in these devices (51% of consumers in the United States are most excited about smart home technology), but many have concerns about cost, privacy and potential clutter with the accumulation of these devices [1]. We would like to design a smart mirror that can address some of these concerns. The appeal of a mirror is that it is already an everyday household object; it is discrete and concealed as a smart device. Furthermore, we want to be able to encompass many technologies to reduce clutter, so the mirror will be capable of presenting visual and audio content to the user directly from the same device. It will offer voice integration and gesture recognition in order to facilitate intuitive control of the device.

There are several other smart mirrors[2] with voice integration on the market today, however they are often little more than large Android devices hidden behind a mirror that do not effectively take advantage of their mirror form factor. As a reach goal, we hope to make our mirror more innovative by using computer vision and machine learning to integrate the capability to detect certain visual features of the users appearance. Based on these detections, the mirror can suggest commercial products that might be of interest to the user. As for privacy concerns, the camera and microphone on the mirror will be under the complete control of the user through the settings and the use of a simple off switch. Overall, we want to implement a smart mirror that is effective, user-friendly and secure.

1.3 High-Level Requirements

- The mirror should be able to recognize gestures, including left, right, up, and down swipes and hovering a hand over one of three proximity sensors, and perform the appropriate corresponding action.
- The mirror should be able to recognize speech and interpret specific voice commands. We will specifically be integrating open source Alexa SDK.
- The mirror should have a functional and practical display, which adjusts accordingly in order to ensure the information is clearly visible.

2 Design

The overall design of the device will be split into submodules which each provide a specific functionality. The user interface module will provide the hardware required to effectively communicate information to the user, including a display and status LEDs. The audio I/O module will enable the user to interact with the device via voice commands through a microphone and will provide speakers through which the device can respond and provide audio media. The visual sensor module will make use of a camera, PIR sensor array, and proximity sensors to enable gesture detection and perform visual analysis on the user, as well as typical photo and video functionality. The control unit will consist of a microprocessor which will process input from the sensor modules and relay information to the Raspberry Pi which will run the virtual user interface software. The power module will ensure that the devices in the rest of the modules receive the power that they need to operate. The composition of and interaction between these modules is presented visually in Figure 1.

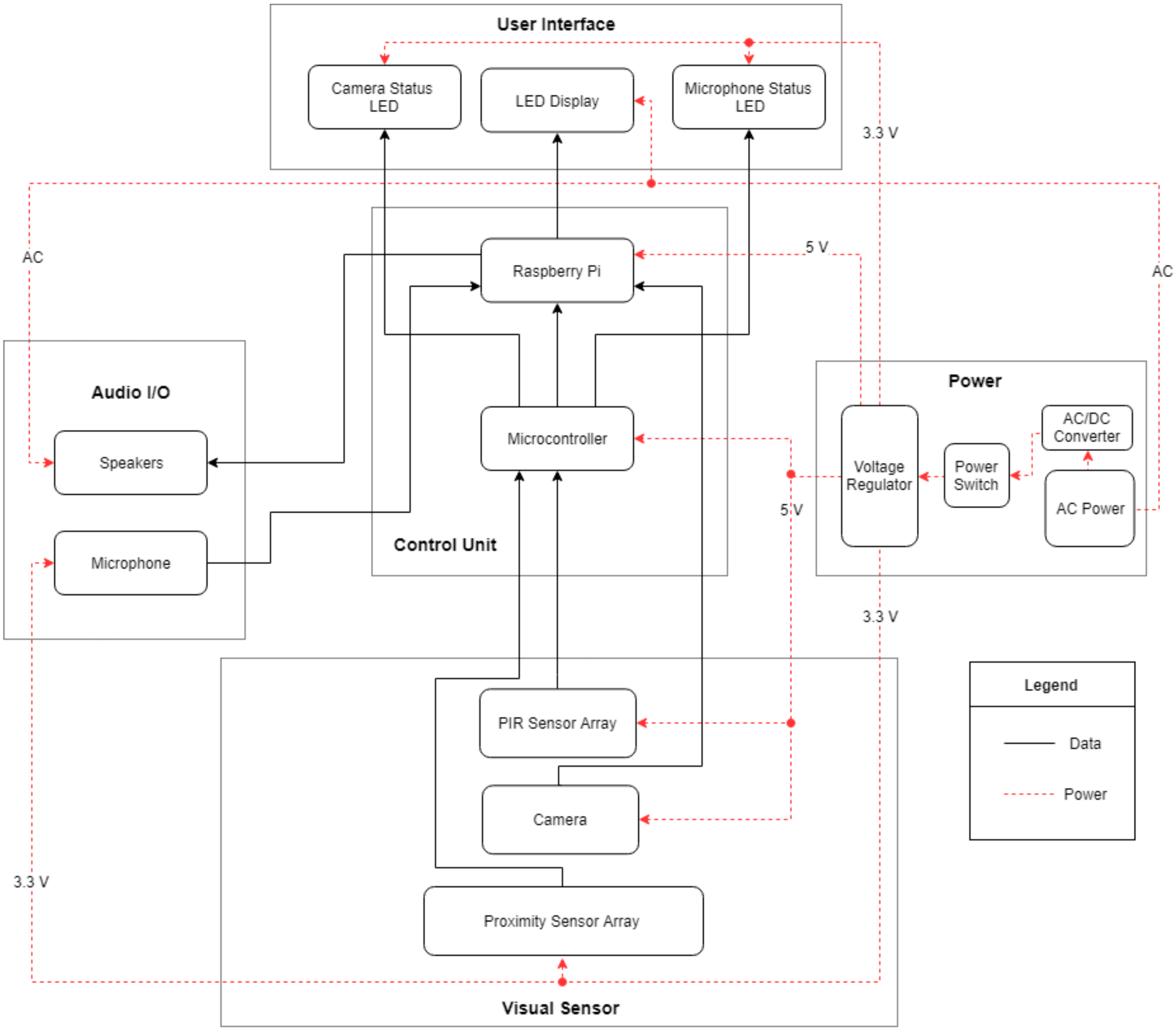


Figure 1: Interactive Mirror Display Block Diagram

The physical design of the device will consist of a single self-contained unit. The front will consist of a two-way mirror surrounded by a bezel at the bottom of which the two status LEDs will be inlaid. The back of the device will consist of a monitor which will fill the upper portion of the display and a sensor module below which will include the camera, PIR sensors, proximity sensors, and microphone. By placing the sensor module at the bottom of the device, we ensure that they will have an unobstructed view through the two-way mirror while still being concealed effectively. The bottom of the frame will contain holes to ensure that sound travels to the microphone and from the speakers effectively as well as the power switch. The physical layout and dimensions of the device are presented visually in Figure 2.

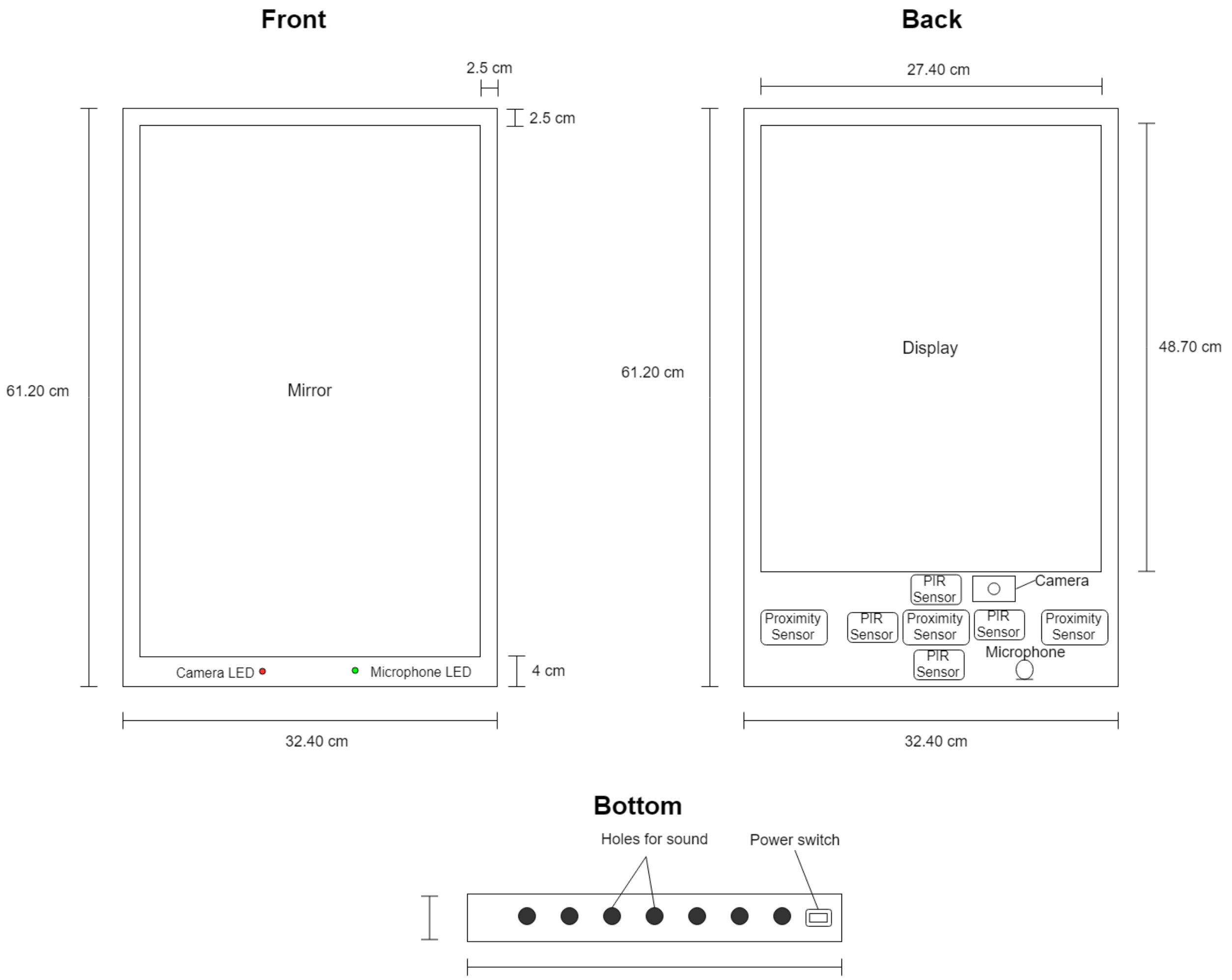


Figure 2: Interactive Mirror Display Physical Diagram

2.1 Control Unit

The control unit will contain a microcontroller and a Raspberry Pi. The microcontroller will be responsible for coordinating the data flow between all of the sensors and outputs. It will process the data it receives and communicate with the Raspberry Pi, which will be responsible for providing the virtual user interface.

2.1.1 Microcontroller

The microcontroller (ATmega328p [3]) will receive sensor data from the PIR and proximity sensors and interpret it as specific gestures including swipes and palm holds. It will then send the gesture data to the Raspberry Pi over the serial port. It will also be responsible for activating the status LEDs when the camera or microphone is in use.

Table 1: Microcontroller Requirements and Verifications

Requirement	Verification
Operating voltage of 5 V ± 5%	1. Attach input leads to voltmeter 2. Observe and ensure that the value does not leave the range of 5 V ± 5%
Data rate of 115200 baud ± 5%	1. Write simple program which sends data packets with alternating bit values 2. Attach data line to oscilloscope 3. Set oscilloscope to trigger on pulse 4. Measure duration of shortest pulse and ensure that the reciprocal falls within 115200 baud ± 5%

Figure 3 shows the pinout diagram for the ATmega328p. We will make use of the serial communication pins, I^2C pins, and analog pins of this device to communicate with the sensors and the Raspberry Pi.

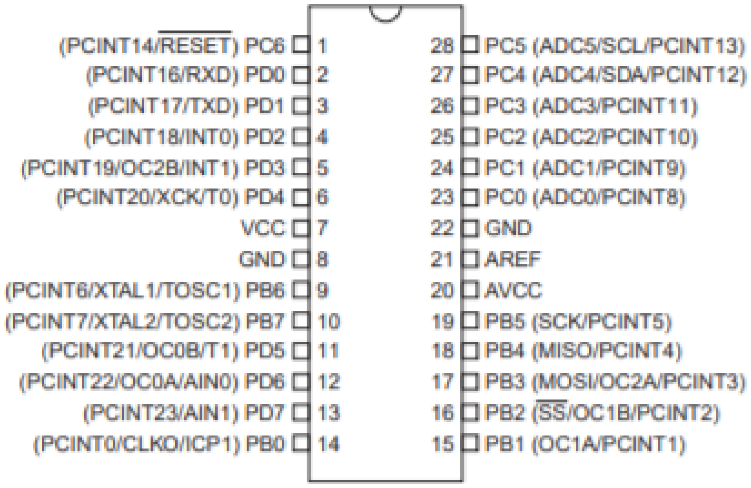


Figure 3: ATmega328P Pinout [4]

2.1.2 Raspberry Pi

The Raspberry Pi 3 [5] will be responsible for running the virtual user interface which provides visual and audio media to the user through the Raspberry Pi's wireless network chip. It will run an application which provides the full-featured user interface and will receive inputs from the microcontroller which communicate processed sensor data and controls.

Table 2: Raspberry Pi Requirements and Verifications

Requirement	Verification
Operating voltage of 5 V ± 5%	1. Attach input leads to voltmeter 2. Observe and ensure that the value does not leave the range of 5 V ± 5%
Max current of 1A ± 1%	1. Attach input leads to multimeter 2. Ensure that the current through the load is at 1A ± 1% using a multimeter (positive lead attached to 10A and negative lead attached to COM) in series with a 12V battery
Data rate of 115200 baud ± 5%	1. Write simple program which sends data packets with alternating bit values 2. Attach data line to oscilloscope 3. Set oscilloscope to trigger on pulse 4. Measure duration of shortest pulse and ensure that the reciprocal falls within 115200 baud ± 5%

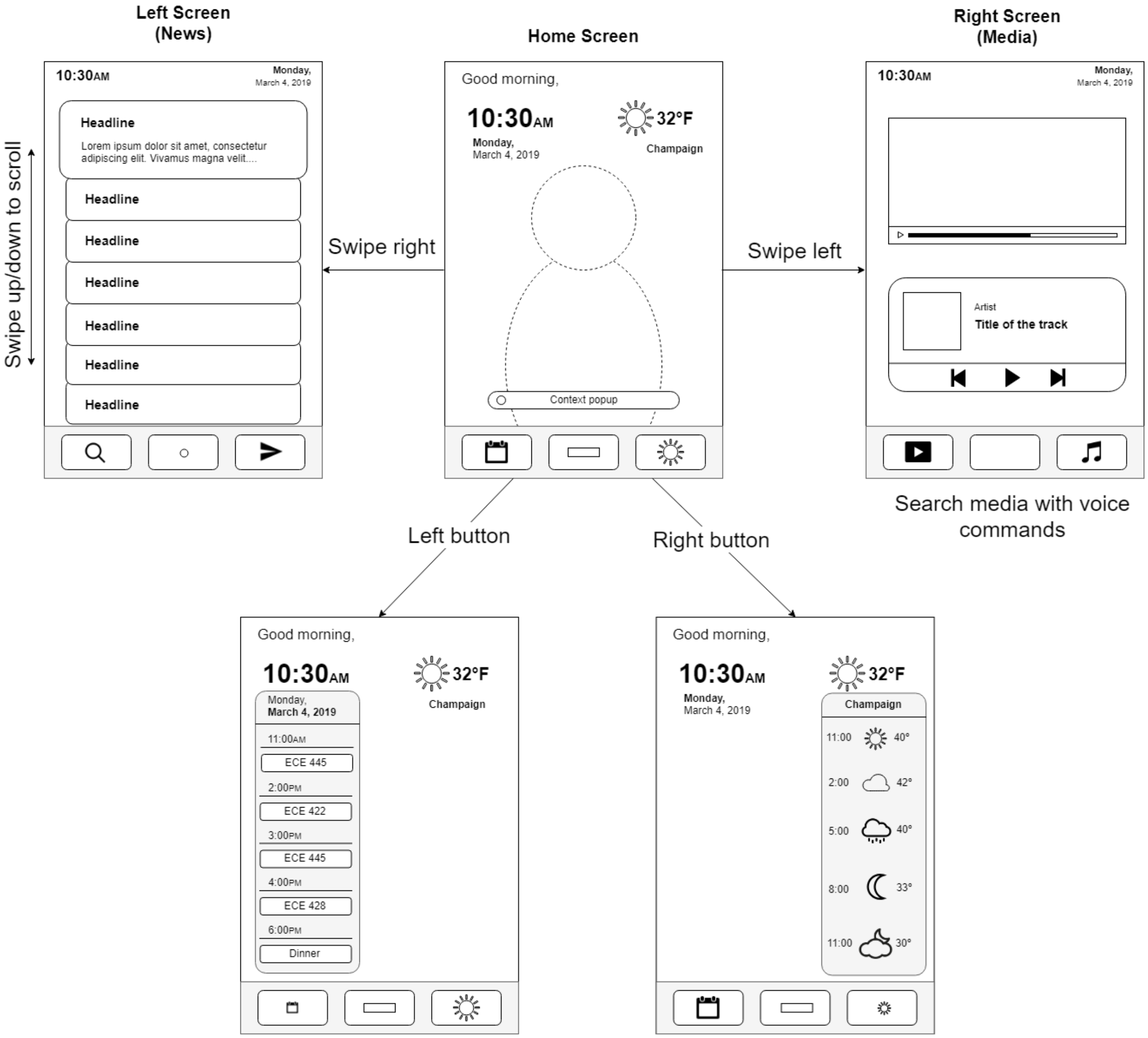


Figure 4: User Interface Diagram

Figure 4 contains a prototype of the user interface application which will be run on the Raspberry Pi and displayed on the monitor. It demonstrates some of the content that the device will provide and illustrates how the gestures will be used to navigate between screens and contexts. Each of the buttons contained within the overlay at the bottom of the screen will be activated by the corresponding proximity sensor, and the swipe gestures will be recognized by the PIR sensor array.

2.2 User Interface

The user interface will supply the hardware necessary to communicate information to the user effectively. The virtual user interface will be presented on the display monitor and two status LEDs will notify the user when the camera or microphone are active.

2.2.1 Display Monitor

The display monitor will be the main interface through which information and content will be provided to the user. We will use a 1080p monitor to provide a clear picture at a common resolution with a reasonable cost. Data will be supplied to the display by the Raspberry Pi over HDMI, and power will be supplied through the wall outlet.

Table 3: Display Monitor Requirements and Verifications

Requirement	Verification
50 mA ± 1% to connect display to Raspberry Pi via HDMI port	1. Attach input leads to ammeter 2. Ensure that the current through the load is at 50 mA ± 1% using a multimeter (positive lead attached to 10 A and negative lead attached to COM) in series with a 12 V battery
Power consumption of 22 W ± 5%	1. Test the power by plugging the monitor into a Kill a Watt meter 2. Make sure the power reading is 22 W ± 5%
Supports up to 1080p resolution	1. Check the on screen control in the settings of the display to check if 1080p is supported

2.2.2 Camera Status LED

The camera status LED will indicate when the camera is currently enabled so that the user is always aware that it is watching. Additionally, a mechanical switch will be incorporated to control whether or not the camera is being utilized.

Table 4: Camera Status LED Requirements and Verifications

Requirement	Verification
The status LED should be visible from 1 m away and have a drive current of 10 mA ± 5% - 20 mA ± 5%	1. Set up test circuit shown in Figure 5 to verify that current is between 10 mA ± 5% - 20 mA ± 5% 2. Ensure LED is visible from 1 m away

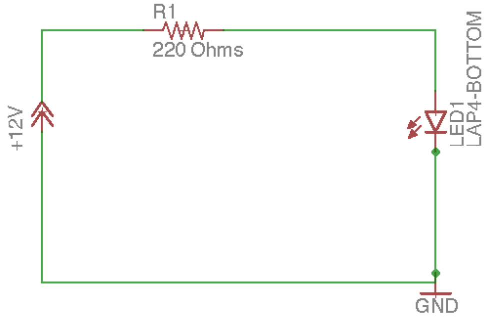


Figure 5: LED Test Circuit - Turns LED on when connected to battery and off otherwise

2.2.3 Microphone Status LED

The microphone status LED will indicate when the microphone is currently enabled so that the user is always aware that it is listening.

Table 5: Microphone Status LED Requirements and Verifications

Requirement	Verification
The status LED should be visible from 1 m away and have a drive current of 10 mA ± 5% - 20 mA ± 5%	1. Set up test circuit shown in Figure 5 to verify that current is between 10 mA ± 5% - 20 mA ± 5% 2. Ensure LED is visible from 1 m away

2.3 Audio I/O

The audio module will take input through the microphone and output through the speakers in order to provide voice control and audio media playback capabilities.

2.3.1 Microphone

The microphone (Mini USB Microphone [6]) will be used to collect data as the user speaks in order to process voice commands. It will send the data to the Raspberry Pi and be powered through the power supply.

Table 6: Microphone Requirements and Verifications

Requirement	Verification
Audio bandwidth of 300 Hz - 5 kHz	1. Plug microphone in and play audio signals ranging from 300 Hz - 5 kHz 2. Check to see if microphone picks up those signals
Max current draw of 1 A ± 1%	1. Attach input leads to multimeter 2. Ensure that the current through the load is at 1 A ± 1% using a multimeter (positive lead attached to 10 A and negative lead attached to COM) in series with a 12 V battery

2.3.2 Speakers

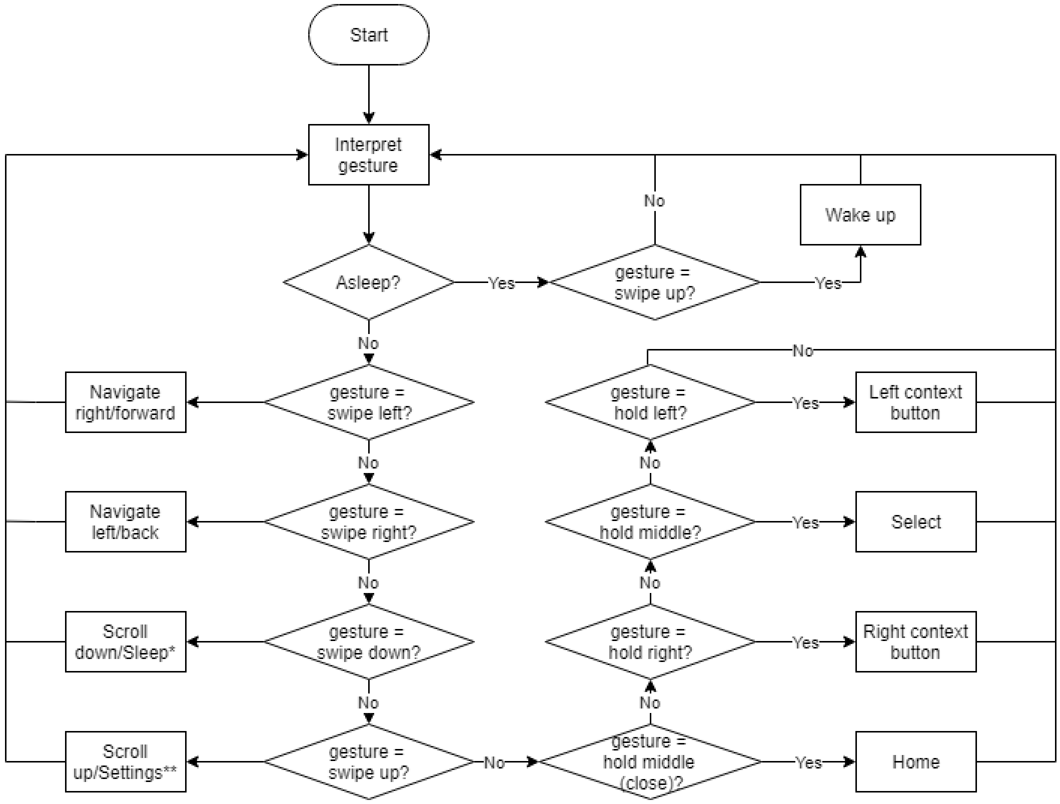
The speakers will output audio to the user including responses to voice commands and audio media. Data will be provided by the Raspberry Pi over HDM.

Table 7: Speakers Requirements and Verifications

Requirement	Verification
Integrated speakers in monitor (powered by monitor): sound should span 30 Hz ± 10% to 17 kHz ± 5%	1. Output an audio file to the monitor that has frequencies ranging from 33 Hz to 16,150 Hz 2. Check to see if the speakers from the monitor successfully output all the audio and one can hear it

2.4 Visual Sensor

The visual sensor system will provide various visual information to the processor in order to facilitate gesture recognition. These gestures will include swiping left, swiping right, swiping up, swiping down, and holding your hand in front of a sensor for a period of time.



*Swiping down while on the home screen will raise a "sleep" dialog

**Swiping up while on the home screen will navigate to the settings screen

Figure 6: Gesture Control Algorithm

The flow chart in Figure 6 outlines the gesture recognition system that will be implemented for the Interactive Mirror Display. The mirror reads a gesture, then performs a task accordingly. If the mirror is asleep, then it will turn on if the gesture is recognized as an upward motion. If the mirror was not asleep in the initial state, then it will interpret a sequence of gestures to either navigate or perform certain actions, such as select an item or a button. The flow chart illustrates the state space of the mirror, and which of the eight gestures can be used to transition from a given state to another.

When the mirror is on and the user wishes to transition it to sleep mode, the user must input a given voice command such as ”Hey mirror, go to sleep” using open source Alexa SDK to do so. Before going into sleep mode, the mirror will ask the user for confirmation to do so to avoid accidentally going to sleep.

2.4.1 Camera

The camera (Raspberry Pi Camera Module V2 [7]) will provide a real time stream of visual information to the microprocessor which will be used to send pictures and videos. The feed will be sent to the Raspberry Pi for processing.

Table 8: Camera Requirements and Verifications

Requirement	Verification
Max current draw of 250 mA ± 1%	1. Attach input leads to multimeter 2. Ensure that the current through the load is at 250 mA ± 1% using a multimeter (positive lead attached to 10 A and negative lead attached to COM) in series with a 12 V battery
Pixel resolution between 2592 x 1944 pixels and 3280 x 2464 pixels	1. Take a picture using the camera 2. Check properties of the image file to verify pixel resolution matches the requirement
Can support 640p, 720p and up to 1080p video mode	1. Take a video using the camera 2. Check properties of the video file to verify resolution matches the requirement

2.4.2 PIR Sensor Array

The PIR sensors (HC-SR501 [8]) detect motion within their field of view and output a digital signal accordingly. We will be using an array of these sensors in order to detect motion, specifically hand gestures such as swipes. In order to interpret these gestures, we will use the delay between the activation of the sensors in the array to determine the direction of the motion.

Table 9: PIR Sensor Array Requirements and Verifications

Requirement	Verification
Needs 5 V ± 1% power (powered by ATmega328)	1. Attach input pin to voltmeter 2. Observe and ensure that the value does not leave the range of 5 V ± 1%
Can detect motion within the range of 10 cm ± 5% to 30 cm ± 5% (it is acceptable for the range to extend past this limit as long as it functions as expected within the specified range)	1. Set up the circuit shown in Figure 7 on a breadboard and wait 30-60s for the PIR to stabilize 2. During this time, the LED may blink - wait until the LED is off 3. Move hand from 0 cm to 30 cm incrementing your position by 10 cm every 10 seconds. When the sensor detects motion, the output pin goes high to 3.3V and lights up the LED 4. Ensure that LED turns on while within the specified range
Field-of-view of 9°± 0.5°	1. Set up test harness consisting of two dowels connected at one end and separated by 9.5 such that they are properly aligned with the tangent from the base of the sensor 2. Set up the circuit shown in Figure 4 on a breadboard and wait 30-60s for the PIR to stabilize

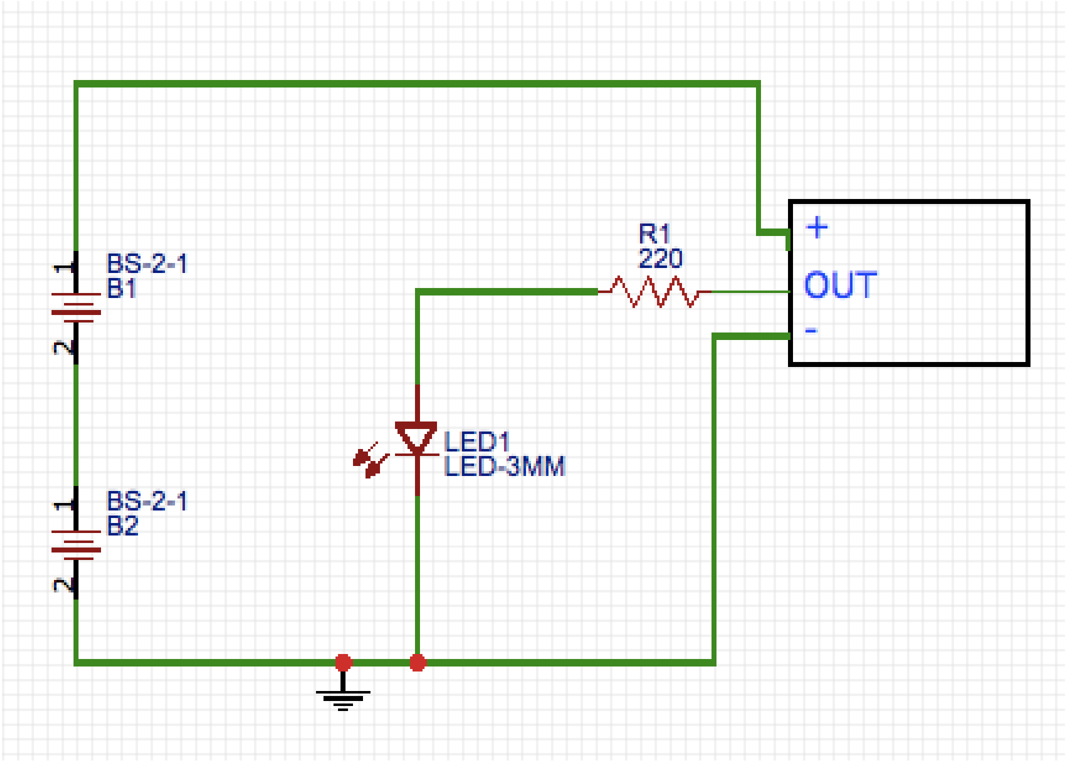


Figure 7: PIR and Proximity Sensor Test Circuit - LED will turn on when the sensor detects a hand in range

2.4.3 Proximity Sensors

The infrared proximity sensors (VCNL4200 [9]) will be used to enable selection of the two auxiliary action buttons when the user places their hand within 30cm of the sensor for over 1 second. The distance data will be sent to the microcontroller over the I^2C bus.

Table 10: Proximity Sensor Requirements and Verifications

Requirement	Verification
Has a range from 10 cm ± 5% to 30 cm ± 5%	1. Configure proximity sensor for desired range 2. Set up the circuit shown in Figure 7 where OUT is the interrupt pin 3. Move hand from 0 cm to 30 cm incrementing your position by 10 cm every 10 seconds. When the sensor detects presence, the output pin goes high to 3.3 V and lights up the LED 4. Ensure that LED turns on while in the range and off while outside of it
Operation voltage from 2.5 V ± 5% to 3.6 V ± 5%	1. Attach input leads to voltmeter 2. Observe and ensure that the value does not leave the range of 2.5 V - 3.6 V

Figure 8 shows the pinout diagram for the VCNL4200. We will make I^2C communication pins to send distance data to the microcontroller when a user places their hand near the proximity sensor.

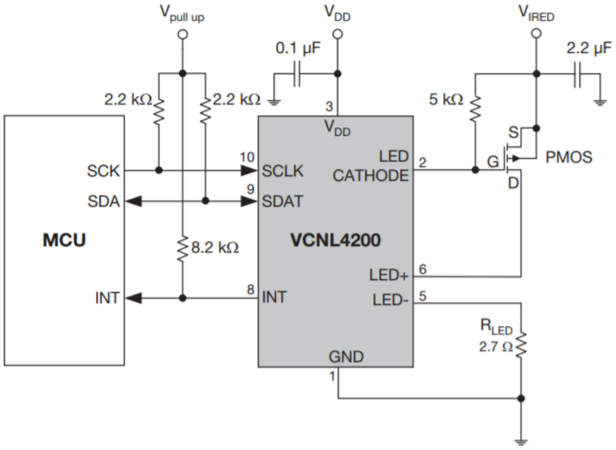


Figure 8: VCNL4200 Pinout [10]

2.5 Power

The power module will be responsible for adapting the power provided by the wall outlet and supplying power to all of the other components.

2.5.1 AC Power

AC power will be supplied via a wall outlet.

2.5.2 AC/DC Power Converter

The power converter will convert the AC power supplied by the wall to DC power usable by the devices in our design.

Table 11: AC/DC Power Converter Requirements and Verifications

Requirement	Verification
Convert the 120V AC to 12V ± 5% DC.	1. Attach input leads to voltmeter 2. Observe and ensure that the value does not leave the range of 120 V ± 5%

2.5.3 Voltage Regulator

The voltage regulator will regulate the voltage supplied by the power converter at the specific voltages required for the various devices throughout the design.

Table 12: Voltage Regulator Requirements and Verifications

Requirement	Verification
Regulate the voltage to 5 V ± 5% for the microcontroller, Raspberry Pi, gesture sensor	1. Attach output to voltmeter 2. Observe and ensure that the value does not leave the range of 5 V ± 5%
Regulate the voltage to 3.3 V ± 5% for the status LEDs	1. Attach output to voltmeter 2. Observe and ensure that the value does not leave the range of 3.3 V ± 5%

2.5.4 Power Switch

The power switch will turn the device completely on or off. The mirror may be woken with gestures and/or voice commands as well, but in order to eliminate privacy concerns and allow the users complete control over what features are enabled, the switch is provided to turn the whole device off when not in use.

Table 13: Voltage Regulator Requirements and Verifications

Requirement	Verification
The power switch should power off the entire system	1. Once switch is flipped, ensure that all devices are no longer receiving power

2.6 Requirement Summary

Table 14: Points Summary

Module	Component	Requirement	Points
Control Unit	Microcontroller	Operating voltage of 5 V ± 5%	1
		Data rate of 115200 baud ± 5%	1
	Raspberry Pi	Operating voltage of 5 V ± 5%	1
		Max current of 1A ± 1%	1
		Data rate of 115200 baud ± 5%	1
		UI software functional	10
		Total	**15**
User Interface	Display Monitor	50 mA ± 1% to connect display to Raspberry Pi via HDMI port	1
		Power consumption of 22 W ±5%	1
		Supports up to 1080p resolution	1
	Camera Status LED	The status LED should be visible from 1 m away and have a drive current of 10 mA ± 5% - 20 mA ± 5%	1
	Microphone Status LED	The status LED should be visible from 1 m away and have a drive current of 10 mA ± 5% - 20 mA ± 5%	1
		Total	**5**
Audio I/O	Microphone	Audio bandwidth of 300 Hz - 5 kHz	1
		Max current draw of 1 A ± 1%	1
		Speech recognition functional	3
	Speakers	Integrated speakers in monitor(powered by monitor): sound should span 30 Hz ± 10% to 17kHz ± 5%	1
		Total	**6**

Continued on next page

Table 14 – continued from previous page

Module	Component	Requirement	Points
Visual Sensor	Camera	Max current draw of 250 mA ± 1%	1
		Pixel resolution between 2592 x 1944 pixels and 3280 x 2464 pixels	1
		Can support 640p, 720p and up to1080p video mode	1
	PIR Sensor Array	Needs 5 V ± 1% power (poweredby ATmega328)	1
		Can detect motion within the range of 10 cm ± 5% to 30 cm ± 5% (it is acceptable for the range to extend past this limit as long as it functions as expected within the specified range)	1
		Field-of-view of 9°± 0.5°	3
		Button select gesture detection functional	5
	Proximity Sensors	Has a range from 10 cm ± 5% to 30 cm ± 5%	1
		Operation voltage from 2.5 V ± 5% to 3.6 V ± 5%	1
		Swipe gesture detection functional	5
		Total	**20**
Power	AC/DC Power Converter	Convert the 120V AC to 12V ± 5%DC	1
	Voltage Regulator	Regulate the voltage to 5 V ± 5% for the microcontroller, RaspberryPi, gesture sensor	1
		Regulate the voltage to 3.3 V ± 5% for the status LEDs	1
	Power Switch	The power switch should power offthe entire system	1
		Total	**4**

2.7 Schematic Diagram

Figure 9 shows the schematic diagram containing the microcontroller and its peripheral devices. Figure 10 shows the schematic diagram of the proximity sensors. Figure 11 shows the schematic diagram of the PIR sensor array. All of these modules are connected together, the sensors sending data to the microcontroller over the pins indicated in the diagrams.

We will have four PCBs in total. One will contain the microcontroller and its peripherals, as well as the PIR sensor array which will be centered horizontally at the bottom of the device. The other three will simply contain a single proximity sensor and the pins necessary to connect it to the main PCB. These will be spaced evenly along the bottom of the device to form the buttons.

Missing from the diagram is the Raspberry Pi which will be connected to the microcontroller on the main PCB over the serial pins.

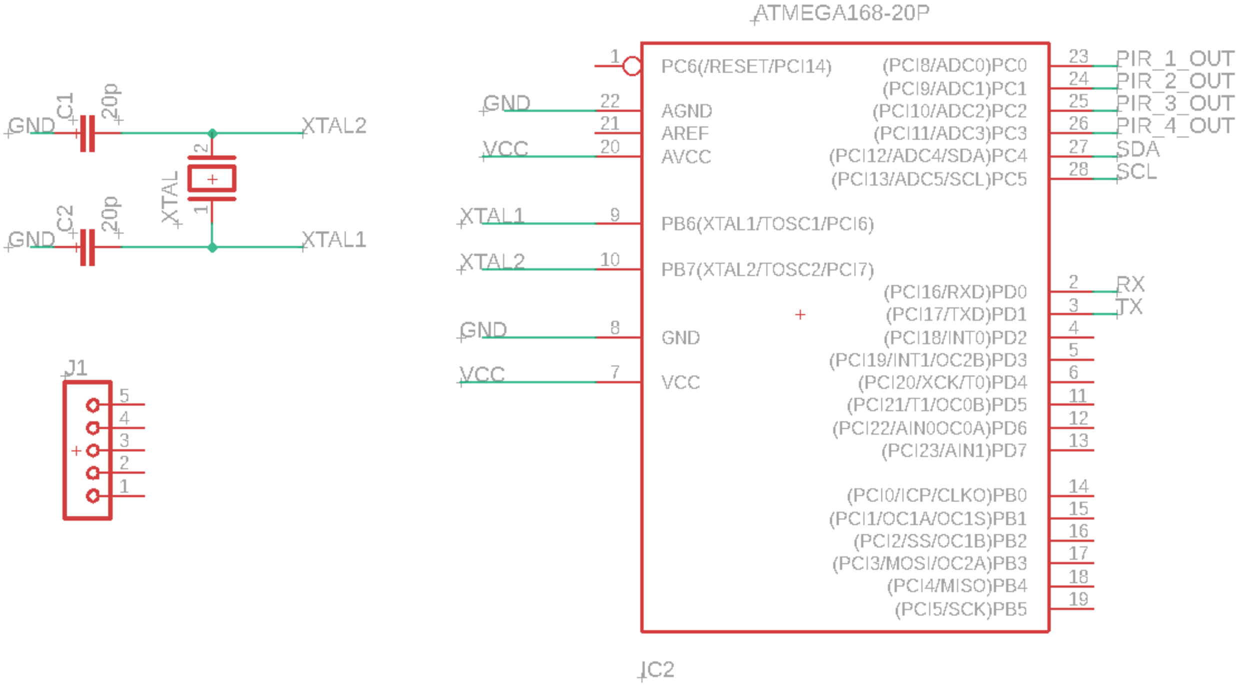


Figure 9: Microcontroller Schematic Diagram

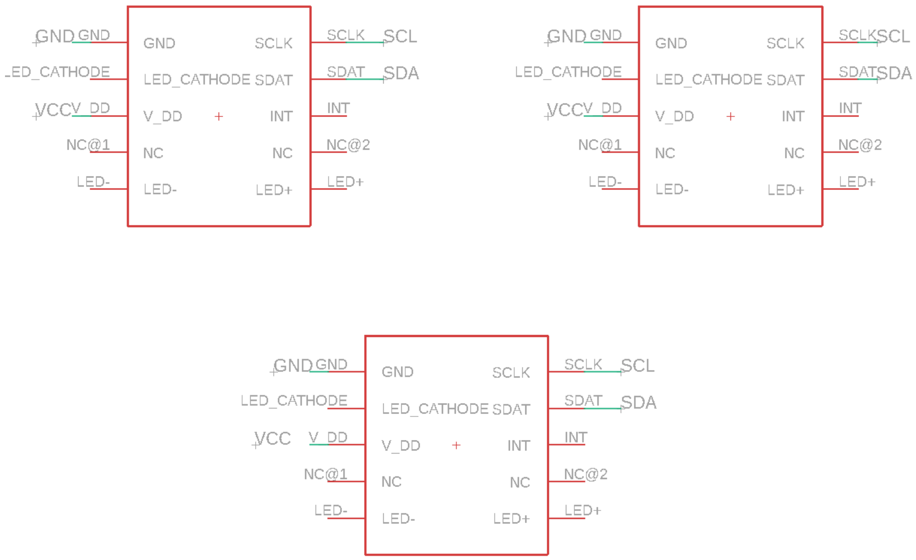


Figure 10: Proximity Sensor Schematic Diagram

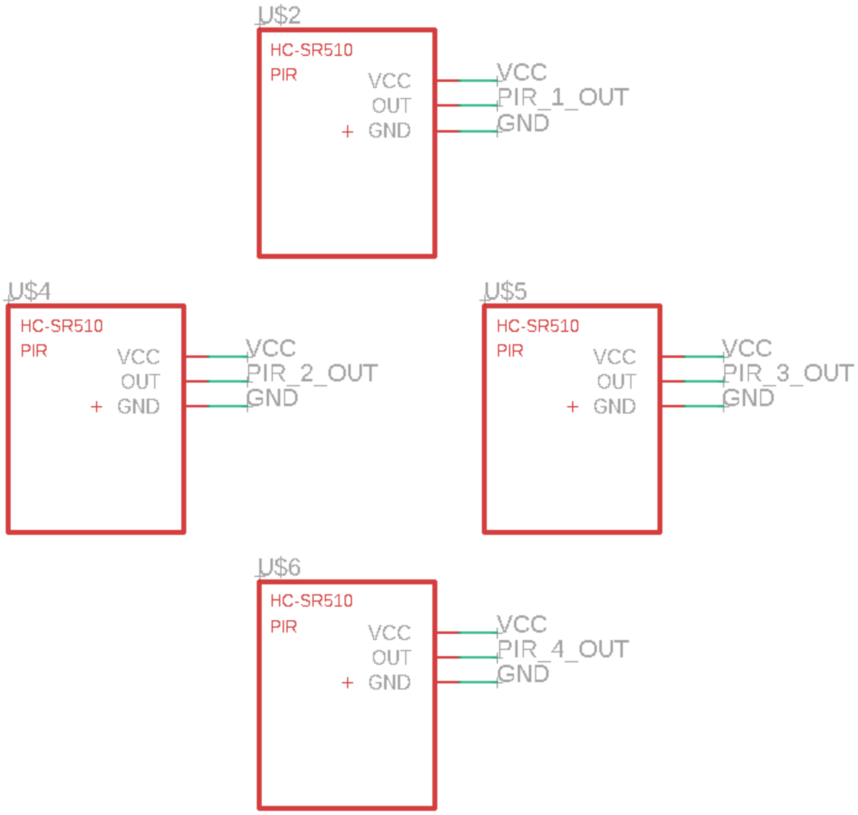


Figure 11: PIR Sensor Array Schematic Diagram

2.8 Tolerance Analysis

One key tolerance which will have a significant impact on the functionality of the device is the field of view of the PIR sensors. In order to ensure that we can extract meaningful data from the array of PIR sensors, we must first ensure that the interference between them will be minimal at the distances specified. If the field of view is not limited as specified, it is possible that multiple sensors will be triggered simultaneously and thus we will not be able to get any information about the directionality of the motion as intended.

The key factors which will determine whether or not the coverage of the sensors overlap will be the distance between them, $d_{spacing}$, which will be fixed at 2 cm due to space limitations in the physical design, the distance at which the user will operate the device, which is specified to fall within a range of 10 cm and 30 cm ±5%, as well as the field-of-view of the sensor, which must therefore fall within the specified tolerance.

The width of the PIR sensors w_{sensor} is 3.2 cm, so the distance between the center axes of any pair of sensors $d_{sensor} = 5.2$ cm will be described by Equation 1.

$$d_{sensor} = 2(\frac{w_{sensor}}{2}) + d_{spacing} \tag{1}$$

The angle of the field of view of the devices, θ_{FOV}, must therefore be defined by the relationship shown in Equation 2, where d_{max_op} is the furthest tangential distance from the sensor at which the user is expected to be able to perform the gesture, in order to avoid interference between a pair of sensors.

$$\theta_{FOV} = 2 \cdot tan^{-1}(\frac{d_{sensor}/2}{d_{max\_op}}) \tag{2}$$

The consequence of this is that in order to maintain the necessary separation between a pair of PIR sensors at the operating ranges described, the PIR sensors must have a field of view of 9.9°, as shown in Figure 12.

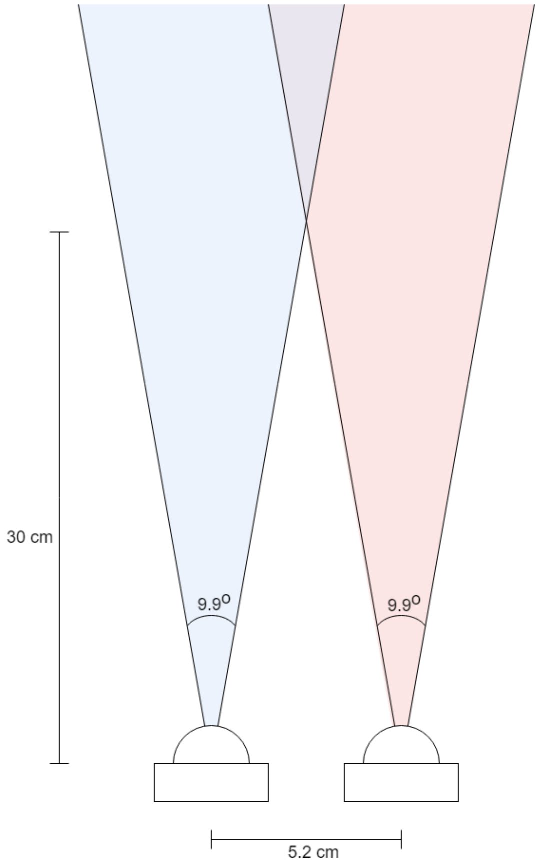


Figure 12: PIR Sensor FOV Diagram

By setting our requirement for field-of-view to be 9°± 0.5°, we ensure that even at the maximum distance from the device, the sensors will pick up motion independently from each other. This will ensure that we can glean insight into the direction of the motion of the users hand when they perform a swipe gesture in front of the sensors. By placing one pair of sensors along the horizontal axis of the bottom of the mirror and another along the vertical axis, we can simply compare the times at which the sensors in a pair were activated to determine the direction of the motion. In order to ensure that the FOV requirement is met, we will modify the Fresnel lens on each sensor by occluding the edges of the lens such that the field of view is reduced.

3 Cost and Schedule

3.1 Cost Analysis

3.1.1 Labor

The cost of development will consist of 15 hours of work per week for 3 people at $40/hour, as shown in Equation 3.

$$3 \cdot \$40/\text{hour} \cdot 15 \text{ hours/week} \cdot 9 \text{ weeks} \cdot 2.5 = \$40,500 \tag{3}$$

3.1.2 Parts

Table 15 illustrates the parts which will be required to complete the design and the quantities and costs associated with each. We have provided an estimate of both the retail cost for the prototype that we will build in class this semester, as well as the bulk costs that would be applicable for larger scale production.

Table 15: Parts Costs

Part	Retail Cost	Bulk Purchase Cost
Microcontroller (ATmega328p)	$1.96	$1.62
Raspberry Pi 3	$35	$35
Display Monitor	$50	$50
Camera (Raspberry Pi Camera Module V2)	$29.95	$22.50
Microphone (Mini USB Microphone)	$5.95	$4.76
Proximity Sensors (VCNL4200) x3	$8.85 ($2.95)	$5.70 ($1.90)
PIR Sensor (HC-SR501) x5	$8.99	$8.99
Assorted components	$9.30	$9.00
Total	$150	$137.57

3.2 Schedule

Table 16: Schedule

Week	Hiraal	Stephen	Pamela
2/25/19 (Design Review)	Begin schematic/PCB design	Design UI software	Acquire components needed for design
3/4/19 (Teamwork Eval I/Soldering Assignment)	Continue PCB design	Begin vision sensor software design	Develop UI software
3/11/19 (First PCB round)	Test and verify components	Vision sensor software development	Continue UI development and testing
3/18/19 (Spring Break)	Work on Individual Progress Report - complete any pending work		
3/25/19 (Individual Progress/2nd PCB Round)	Revise PCB for second round	Continue vision sensor development	Continue UI development and begin integration for speech recognition
4/1/19	Begin physical construction: two-way mirror, sensors, status LEDs, microphone and display	Test and iterate vision sensor software development	Continue speech recognition integration (Begin designing computer vision for reach goals if time allows)

Continued on next page

Table 16 – continued from previous page

Week	Hiraal	Stephen	Pamela
4/8/19	Complete and test physical device	Finalize vision sensor software development (Continue computer vision algorithm if time allows)	Test/Debug speech recognition integration
4/15/19 (Mock Demo)	Conduct finished product testing	Fix any issues found while testing	Prepare mock demonstration (Finish computer vision algorithm if time allows)
4/22/19 (Demo/Mock Presentation)	Prepare final presentation	Fix any remaining bugs	Work on final report

4 Ethics and Safety

The ethical and safety concerns that our project presents relate to data management and privacy. As the ACM Code of Ethics (1.6) [11] explains, we need to ensure that potential consumers know whether or not their personal information will be collected or monitored. To specifically address this issue, we will explicitly state how and when the mirror will be gathering and using information. To work toward the goal of full disclosure, we will incorporate LEDs used to alert the consumer that the mirror and/or camera is on. Additionally, we need to ensure that all data and metadata gathered will be kept confidential in order to comply with the ACM Code of Ethics (1.7) [11]. This means that we will not share data that can identify a consumer with third-parties (only specified personnel will have limited access), unless there is a clear violation of the law. In such a case, we will inform the consumer that their data may be shared with proper authorities.

Another concern we will be addressing is the security of our systems. The Interactive Mirror Display will have the potential ability to gather and monitor information about the consumer, we will be implementing certain security measures. In order to provide the consumer with a way to manage the usage of the mirror, we will be implementing a button to turn the mirror on or off. This provides a measure of physical security, and is simple enough for the general consumer to understand, which complies with the ACM Code of Ethics (2.9) [11]. Furthermore, we will be designing a system that implements common networking security protocols to ensure that all data being sent to a server is secured to a satisfactory level. In the case of a data breach, we will notify all parties affected in the ”most expedient time possible and without unreasonable delay” [11]. By implementing and following these standards, we would be adhering to the Illinois Personal Information Protection Act (815 ILCS 530, et seq.) and the Federal Trade Commission Act (15 U.S. Code 41, et seq.) [11].

While our device does not raise many safety concerns, it is possible that damage could be caused by improper mounting of the device, or if the screen were to shatter. In order to reduce the likelihood of these events and mitigate the damage that could be caused if they did occur, we will provide two mounting hooks, one on either side of the device, to ensure stability and support. We will also use acrylic coated in two-way mirror film rather than actual glass in order to avoid shattering.

Another potential concern is that the consumer would come into contact with some of the circuitry. To avoid direct contact with the electrical components, the components will be enclosed by the actual acrylic described above and the back of the display.

References

[1] "Impact of Smart Home Technology - IoT Innovation," Web page, IoT Innovation, accessed February 2019. [Online]. Available: https://internet-of-things-innovation.com/insights/the-blog/impact-of-smart-home-technology/#.XFtTyxlKiqA

[2] "The 6 Best Raspberry Pi Smart Mirror Projects You Can Make," Web page, MakeUseOf.com, 2018, accessed February 2019. [Online]. Available: https://www.makeuseof.com/tag/6-best-raspberry-pi-smart-mirror-projects-weve-seen-far/

[3] "ATmega328, P Product Description," Web page, DigiKey, accessed February 2019. [Online]. Available: https://www.digikey.com/product-detail/en/microchip-technology/ATMEGA328P-PU/ATMEGA328P-PU-ND/1914589

[4] "ATmega328," Datasheet, Atmel, 2009, accessed February 2019. [Online]. Available: https://www.sparkfun.com/datasheets/Components/SMD/ATMega328.pdf?fbclid=IwAR2WZGy-vAtoW_0tXzsM_vME84chuFD0gCNQRNyZ-wkCdcsYvCdljCg8dhg

[5] "Raspberry Pi 3 Product Description," Web page, Adafruit Industries, accessed February 2019. [Online]. Available: https://www.adafruit.com/product/3055?src=raspberrypi

[6] "Mini USB Microphone Product Description," Web page, Adafruit Industries, accessed February 2019. [Online]. Available: https://www.adafruit.com/product/3367

[7] "Raspberry Pi Camera Module V2 Product Description," Web page, SparkFun Electronics, accessed February 2019. [Online]. Available: https://www.sparkfun.com/products/14028

[8] "HC-SR501 PIR Sensor Product Description," Web page, Amazon.com, Inc, accessed February 2019. [Online]. Available: https://www.amazon.com/DIYmall-HC-SR501-Motion-Infrared-Arduino/dp/B012ZZ4LPM/ref=sr_1_3?s=electronics&ie=UTF8&qid=1550559971&sr=1-3&keywords=pir+sensor

[9] "VCNL4200 Proximity SensorProduct Description," Web page, Mouser Electronics, accessed February 2019. [Online]. Available: https://www.mouser.com/ProductDetail/Vishay-Semiconductors/VCNL4200?qs=gt1LBUVyoHnG9%252beWUQ9nRg%3D%3D

[10] "VCNL4200," Datasheet, Vishnay Semiconductors, 2017, accessed February 2019. [Online]. Available: https://www.vishay.com/docs/84430/vcnl4200.pdf

[11] "ACM Code of Ethics," Web page, ACM, 2018, accessed February 2019. [Online]. Available: http://www.linxtechnologies.com/products/rf-modules/lt-series-transceiver-modules/