Ecosystem for Smart Glass Technologies (ESGT)

ECE4011 Senior Design Project

Section ET2, The Magicians
Project Advisor, Dr. Manos Tentzeris

Najee Kitchens
Kairi Kozuma
Boa-Lin Lai
Jonathan Osei-Owusu
Nishant Shah

Submitted

November 21, 2016

Table of Contents

Executive Summary

The Internet of Things (IoT) device market has exploded in recent years, and is expected to reach 6.7 billion device shipments with $1.7 trillion in value added to the global economy in 2019 [1]. To take advantage of this trend, companies have begun exploring innovative ways to add smart capabilities to everyday products. Ecosystem for Smart Glass Technologies (ESGT) is a complete hardware and software solution that brings IoT devices to the transparent realm. ESGT aims to enhance traditional glass products, by adding information display and user interaction capabilities. Example data to be displayed include weather, traffic conditions, directions, personal agenda, and news. ESGT has two main parts: hardware technology to implement transparent displays for any glass application and a software ecosystem to provide backend support. Application specific modules will be implemented for each glass product, leveraging the common software framework. Users will be able to interact with ESGT products through stereoscopic gesture sensors or capacitive floating touch technology. The expected outcome of the design is a smart mirror prototype. This device will use transparent glass technology with a reflective material behind it to provide mirror capabilities; furthermore, the device will use a microcontroller that connects to the local Wi-Fi connection and leverages the ESGT software framework. All computation will be done over the cloud to minimize the power consumption on the device. A smartphone application will be developed for setup and configuration of the smart mirror via Bluetooth. The prototype will cost approximately $64.10. More modules could be added using the common hardware and software foundation, to develop devices such as smart windows, eyeglasses, and windshields.

Ecosystem for Smart Glass Technologies

1. Introduction

Ecosystem for Smart Glass Technologies (ESGT) is a complete hardware and software solution that lays the foundation for rapid development of glass IoT devices. The ESGT development team will design a smart mirror that displays useful user information as a prototype application of the ecosystem. The team is requesting $64.10 to develop this prototype.

1.1 Objective

The objective of ESGT is to design and develop a robust hardware and software ecosystem to support all glass IoT devices. The prototype to demonstrate ESGT is a smart mirror. The user will be able to interact with the smart mirror using gestures, captured either through capacitive floating touch or a stereoscopic sensor. The gestures will be used to turn on and off the display, interact with on screen data, or cause a refresh if necessary. The data will be pulled through Wi-Fi from the common ESGT software framework, which is hosted remotely on the cloud. Information such as calendar, date, time, and weather will be displayed near the edges of the mirror, so that the user's own reflection remains unobstructed. A smartphone application will be used for first time setup and configuration. The application will use Bluetooth to communicate with the mirror.

1.2 Motivation

Although IoT has found its way into thermostats, vacuums, and refrigerators [2], it has not had a huge impact on transparent products, due to the difficulty implementing the technology cost effectively. For instance, Planar’s LookThru OLED is a commercial-grade 55-inch transparent display with a base price of $14,995 [3]. To make transparent glass technology accessible for the average consumer, the hardware must be developed with minimal cost and adequate software support. The ESGT is looking at alternatives such as inkjet printing to print transparent circuitry directly onto glass to achieve a low cost solution. This will allow rapid prototyping and deployment of any glass IoT device at a reasonable price.

The smart mirror was chosen as the prototype for ESGT, as there are no commercial smart mirrors available to purchase. Furthermore, nearly all implementations of smart mirrors from the Do-It-Yourself (DIY) community today rely on a traditional LCD display behind a two-way mirror [4]. The ESGT team will develop a smart mirror using transparent glass, making it easier for other glass devices to be implemented.

1.3 Background

1.3.1 Transparent Display

The electronic display industry has done extensive research on transparent displays. The most common implementation is transparent OLED circuitry, which is constructed from electrically conductive material with low light absorption, called transparent conducting oxides (TCOs). Indium tin oxide (ITO) is the most widely used TCO because of its electrical conductivity, optical transparency, and ease with which the material can be deposited as a thin film [5].

Each pixel in a transparent OLED display is made up of four sub-pixels: red, blue, green and clear. The clear sub-pixel is responsible for transparency; furthermore, the ratio of clear to colored sub-pixels is directly proportional to resolution. A ratio above one creates less space for clear sub-pixels, resulting in more occluded displays but higher-resolution images [6][7].

ESGT will use the same ITO material, but rely on inkjet printing to directly print the transparent circuits onto glass. This reduces the cost of developing the transparent display and allows for rapid prototyping.

1.3.2 Floating Touch

Floating touch adds a third dimension to the touch screen, allowing users to interact by hovering over the display. This is necessary for ESGT to provide interaction capabilities without directly touching the glass surface, preventing fingerprints and smudges.

Floating touch can be implemented by innovations to traditional projected capacitive touch. A touch event is detected by measuring changes in capacitance at an electrode. When a conductive object approaches an electrode, it interferes with the electromagnetic field around it and alters its capacitance [8]. Since the electric field extends outside of the screen, the electrode can detect objects without physical contact, provided the capacitance change is above a threshold value [9]. To obtain X and Y coordinates on a touchscreen, electrodes are arranged in a two-dimensional matrix [8].

Floating touch technology requires the same three core components as traditional capacitive touch screens: touchscreen panel, touch controller unit, and an Analog Front End (AFE). Changes in capacitance can be as little as tens of pF, so the sensing circuit must detect changes of that magnitude [10]. The AFE converts the capacitance to a voltage using a differential sensing circuit, which amplifies the signal to voltage levels ranging from 20mV to 50mV[11]. The voltage is then converted to digital data with and Analog to Digital Converter (ADC). This data is then sent to the processor, typically via I^2C or SPI interface standards [11].

1.3.3 3D Image Processing and Gesture Recognition

An alternative to capacitive floating touch is 3D image processing to capture gestures. 3D image processing can be setup with a cameras or 3D depth sensors. The data captured is passed to a 3D image processing software embedded in the computer [12]. For instance, the Microsoft Kinect is a commercially available gesture recognition device that has three cameras, one for the RGB (red, green, blue) colors and two for 3D depth sensors. The RGB camera helps to distinguish the user from the background, and the 3D depth sensors track the user's body [13]. This type of technology can be used to track fingers for the smart mirror, allowing users to control the device with simple gestures.

The main operations of gesture detection are image capture, digitization, segmentation, model fitting, motion prediction, and qualitative or quantitative conclusion. During image capture, a 3D object is projected to a 2D intensity image through a pinhole camera. Then, digitization converts the analog signal to a digital one. The digital image is segmented into parts, simplifying its representation for processing [14]. Different model fitting and motion prediction algorithms - different motion analysis (DMA) method, the Block-Matching Algorithm (BMA), and boundary pixel decimation (BPD) - are used to extract meaningful information from the segmented image [15]. Finally, the gesture is determined to be valid or invalid by qualitative or quantitative measures. The main software program on the smart mirror will use the gesture to respond to user input.

1.3.3 Microcontroller

Microcontroller choice is important for embedded device, has it determines the performance and affects the power consumption. The smart mirror will use a Raspberry Pi 3 Model B as its prototype microcontroller. This is a single board computer with a 1.2Ghz 64-bit quad-core ARMv8 CPU as its main processor. The board offers built in capabilities for Wi-Fi and Bluetooth, making it an attractive option. The Pi 3 requires 2.5A at 5V, making the total power consumption 12.5W [16]. It is also cheap, available for $35 [17]. Another option is the Raspberry Pi Model A+ board, designed for low consumption by operating between 0.5W to 1W [17]. However, there is no built in Wi-Fi capability, and the processor only runs at 700Mhz on a ARMv6 CPU. This may not be enough to do all the gesture or touch recognition processing and drive a large display.

1.3.4 Wireless Connectivity

The smart mirror will use Wi-Fi to connect to the Internet as an IoT device. Wi-Fi refers to any type of IEEE 802.11 Wireless Local Area Network (WLAN), extending the reach of wired Local Area Networks (LANs) [18]. Since Wi-Fi ranges are typically 30 to 50 meters, the smart mirror can be placed far from the router and still connect to the Internet. In addition, the common 802.11n standard offers a theoretical “maximum data rate of about 540Mbps”, which is more than adequate for sending simple API calls and data.

A smartphone application will be used to set up the mirror. This includes Wi-Fi authentication setup and other software configurations. The application will use Bluetooth to connect to the mirror. Bluetooth networks are called piconets, which are established dynamically and automatically as Bluetooth devices enter and leave radio proximity [19]. The smartphone will act as the master device and pair with the slave smart mirror device. Although Bluetooth can only achieve speeds of around 2.5Mbps and has a short range of 60m, it is sufficient for setup configuration [19].

2. Project Description and Goals

The goal of ESGT is to design a complete hardware and software ecosystem for electronic glass devices. The smart mirror will be a prototype that will act as a proof-of-concept device, consisting of a transparent glass display with reflective material attached to the back to provide mirror functionality. A microcontroller will drive the display and connect to the Internet using Wi-Fi. The software framework will include APIs common to all glass devices with a database implementation. The application specific modules of ESGT will have the graphical user interfaces (GUI) and interaction methods necessary for smart mirror. An accompanying smartphone application will be used to set up and configure software on the smart mirror, communicating via Bluetooth. Project goals include the following:

ESGT General Features:

- Low cost transparent displays achieved through inkjet printing
- Support for user interaction, either via gesture sensors or capacitive touch
- Framework of software APIs common to all glass devices, including database support
- Extensibility to add support for any glass product

Smart Mirror Features:

- Display weather, agenda, news, and other relevant information to the user
- Smartphone application to set up and configure the smart mirror
- Appearance of a regular mirror when display is off

3. Technical Specifications

Table 1. Smart Mirror Specifications

Feature	**Specification**
Mirror Glass Dimensions	400mm x 225mm x 1mm
Mirror Package Dimension	410mm x 235mm x 50mm
Chassis Anchor Holes	2 holes
Power Supply Voltage	90-120V
Display Refresh Rate	60 Hz
Interaction Response Time	< 100ms

4. Design Approach and Details

4.1 Design Approach

4.1.1 System Diagram

The smart mirror will consist of microcontroller, glass, display module, and user input device (gesture sensor, and capacitive input interface). Figure 1 shows the block diagram of the system.

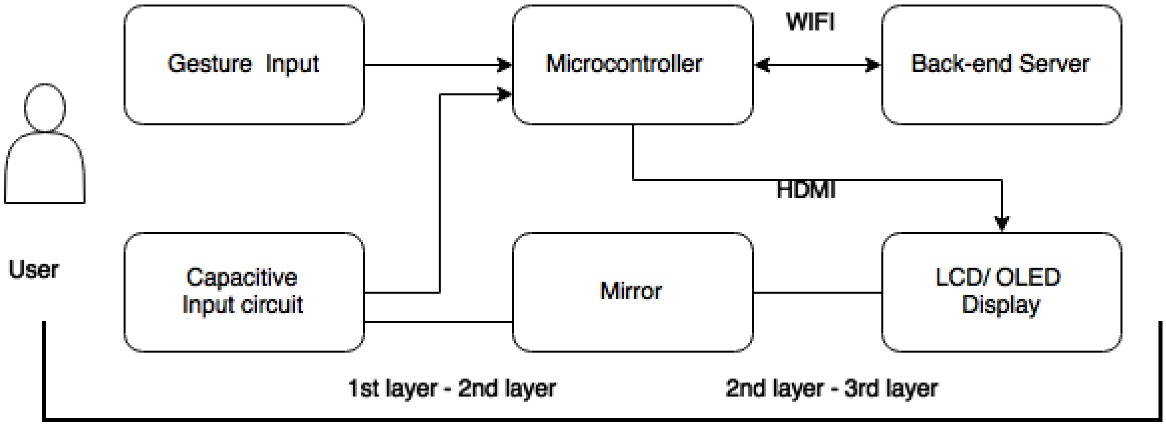


Figure 1. Block diagram of smart mirror system.

4.1.2 Hardware

Microcontroller

The Raspberry Pi™ 3 Model B was selected as the main microcontroller for this development. It has a Wi-Fi (802.11n) module for the network communication and an HDMI port for display. The collected user input will be sent to a backend server over Wi-Fi to minimize processing power consumption on the device. Figure 2 shows the layout of the microcontroller.

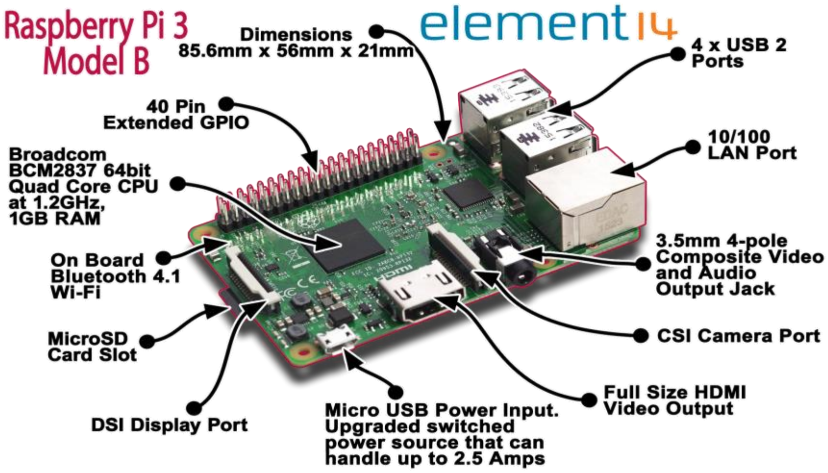


Figure 2. Raspberry Pi™ 3 Model B features [20].

Gesture Input Module - LeapMotion

The Leap Motion will be used as input sensor. Its data will be sent to a backend server to reduce computation time. It can interpret four different gestures: a circle (single finger tracing a circle), swipe (long, linear movement of a finger), key tap (tapping movement by a finger as if tapping a keyboard key), and screen tap (a tapping movement by the finger as if tapping a vertical computer screen). Figure 3 shows the illustrations of the gestures.

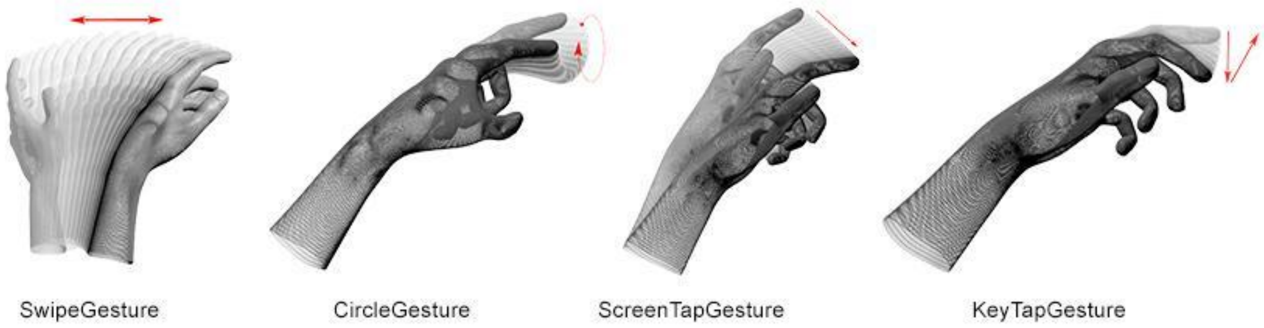


Figure 3. Leap Motion gestures [21].

Mirror

A rectangular glass with dimensions 400mm x 225mm x 1 mm will be used for the smart mirror. It will be placed between monitor and capacitive input circuit.

LCD

Computer monitor with an HDMI port will be used in the initial prototype for the smart mirror. This will be used until the transparent display technology can be implemented in ESGT. The range for the price is from $50 to $100.

4.1.2 Software

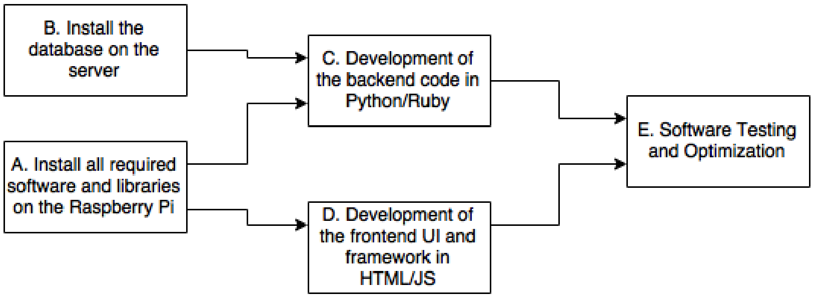


Figure 4. Critical path for software development and testing.

The major software aspects for the smart mirror include the GUI, API integration, and database components. Figure 4 shows a development chart for the software aspect of ESGT.

Graphical User Interface (GUI)

The GUI for the Smart Mirror will be implemented as a website with HTML/CSS for the layout and JavaScript to make it interactive. The front-end code will interact with the back-end code using a REST APIs. The backend will be developed using a scripting language such as Python or Ruby, since these languages have mature libraries for developing networking APIs and interfacing with databases. Figure 5 shows a simple mockup of an example GUI implemented for the smart mirror.

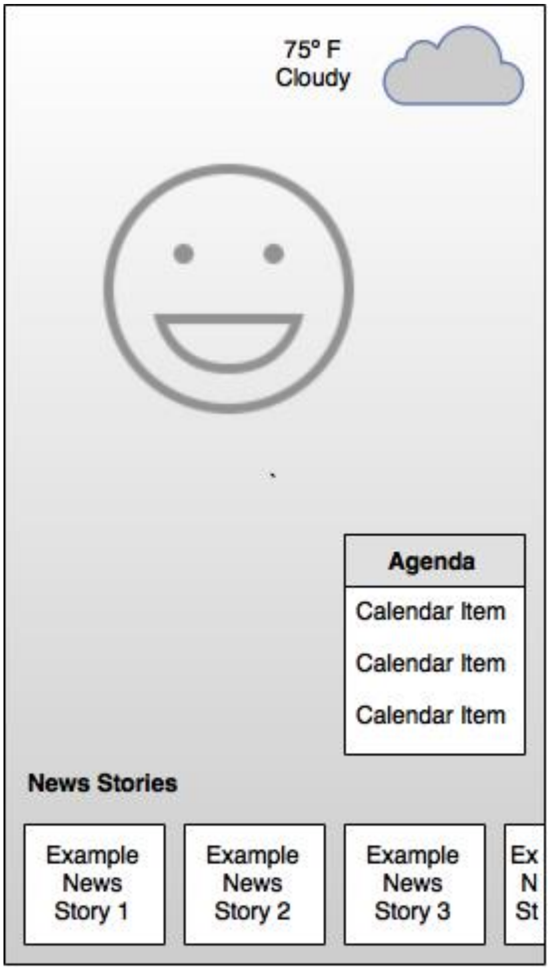


Figure 5. A mockup of the mirror GUI showing example components for weather, news and calendar agenda.

API Integration Framework

In order to make this project applicable to other smart glass devices, a framework to add UI components will be included. Using this framework, developers can add custom elements to the smart glass after the initial release. The initial API integrations that will be added to demonstrate the framework will include weather, calendar information and news. This information will be fetched over the network periodically.

Database

In order to improve initial load times and subsequent performance when displaying information on the GUI, the ESGT framework will use a database for data persistence. In case there is an issue with wireless networking or API integrations and new data cannot be fetched for some time, having cached data in the local database will ensure that the user experience remains consistent.

4.2 Codes and Standards

Since the mirror will have wireless connectivity to communicate with the Internet as well as the mobile device of the user, the Wi-Fi and Bluetooth modules need to be Federal Communications Commission (FCC) certified for use in the United States. The Wi-Fi also needs to follow the IEEE 802.11 protocol [22].

The microcontroller and the transparent display circuitry will be connected using an HDMI cable, so the ports of both need to be the correct shape and size with 19 pins [23].

The microcontroller has a USB port to connect a USB cable to the gesture sensor. The Raspberry Pi™ 3 Model B fits the requirements of the design, since it has both an HDMI and USB port [17].

The United States outlet voltage standard is 120V; thus, this limits the amount of power that can be supplied to both the microcontroller and the display [24].

Using two plastic anchors would have a failure weight of 160 lbs, which limits the weight of the product materials [25].

4.3 Constraints, Alternatives, and Tradeoffs

GUI Design Language

Another method that could be used to create the GUI is to use native C/C++ libraries such as Qt, which are faster and more memory efficient compared to web technologies and do not require a browser application to run. The advantages of using HTML, CSS, and JavaScript is that they are platform independent and can be easily ported to run on different kinds of hardware. Another advantage is that since Python and Ruby are Object Oriented Programming languages, it is easier to build and extensible framework as well as the modules for that framework, which lowers the barrier for developers who want to make their own custom GUI components.

5. Schedule, Tasks, and Milestones

Appendix A contains a table of major tasks and their respective owners. Appendix B contains a Gantt chart, showing the projected timeline for all tasks and milestones. Appendix C contains the PERT chart, with the critical path shown with a bold line. The difficulty and risk associated with each task is denoted for each path in the PERT chart.

Most of the difficulty will be in implementing the hardware technology for transparent displays, as this is a nascent field with a variety of options and difficulty in manufacturing. The hardware prototype and software ecosystem are scheduled to be completed before Spring Break, in order to provide ample time for software development of the smart mirror prototype.

6. Project Demonstration

The prototype demonstration can be in any classroom in Van Leer or Klaus, since the smart mirror will be portable. The demonstration will consist of the following:

1. The user will connect the device to an outlet and turn on the power.
2. Following the commands displayed on the smart mirror, the user will use the smartphone application to connect to the device via Bluetooth.
3. Through the smartphone application, the user will connect the mirror to a wireless access point, so that it can access the Internet.
4. The user will proceed through the configuration steps in the smartphone application to personalize the data to be displayed onto the mirror.
5. Finally, the user will perform gestures to interact with the smart mirror, turning on/off the display and refreshing data.

7. Marketing and Cost Analysis

7.1 Marketing Analysis

The intended market consists of consumers who wish to gain more functionality from their wall-mounted mirrors. Because transparent display technology is a concept still being researched and developed, no commercially available smart mirrors exist on the market. Corning, Inc. Intel, Inc. and MemoryMi are collaborating to create the MemoryMirror™ to allow shoppers to “try on” new clothing without entering the dressing room. While the MemoryMirror™ claims to be a digital mirror, it is actually a translucent glass that displays images captured by a mounted camera. To avoid latency and security concerns associated with placing a camera in a home, the smart mirror will, instead, feature a two-way mirror.

7.2 Cost Analysis

7.2.1 Cost of Components

The total development cost for a smart mirror prototype using an LCD and a two-way mirror ("Prototype 1") is approximately $405.94. Table 2 itemizes the equipment necessary to construct Prototype 1. Successful construction of this prototype will serve as a proof-of-concept for the final product, while offering the software team ample time to develop the glass technologies ecosystem.

Table 2. Prototype 1 Equipment Costs

Component	**Cost**
Raspberry Pi™ 3 Model B	$35.00 [27]
Raspberry Pi™ Power Supply	$8.99 [28]
Two-Way Mirror	$72.00 [29]
Touchscreen LCD Monitor	$289.95 [30]
HDMI Cable	$0.00 (Received for free)
Total	**$405.94**

Once Prototype 1 is functioning correctly, Prototype 2 will be built using glass and transparent circuitry. As shown in Table 2, this new prototype will cost an additional $20.11. The inkjet printer and conductive silver ink are being provided to the team free of charge. The HDMI cable will also be procured by the senior design team at no cost. The Raspberry Pi and its power supply will be recycled from Prototype 1, to not incur additional expenses. If these two components were not reused, then this prototype would cost $64.10.

Table 3. Prototype 2 Equipment Costs

Component	**Cost**
Rectangular Annealed Glass	$20.11 [26]
Raspberry Pi™ 3 Model B	$0.00 (Reuse from Prototype 1)
Raspberry Pi™ Power Supply	$0.00 (Reuse from Prototype 1)
Conductive Silver Ink	$0.00 (Received for free)
HDMI Cable	$0.00 (Received for free)
Inkjet Printer	$0.00 (Received for free)
Total	**$20.11**

7.2.2 Development Costs

Development hours invested per engineer is itemized in Table 4. Five engineers will work to develop the smart mirror. Besides group meetings, assembly will require the most labor hours because the touchscreen capable transparent display is a recent concept and has little assembly documentation.

Table 4. Development Hours Invested Per Engineer

Task	**Hours**
Group Meetings	110
Report Preparation	25
Presentation	3
Demo Preparation	10
Research	5
Fabrication	15
Assembly	30
Testing	20
Total	**218**

Assuming fringe benefits are 34% of labor and overhead is 120% of labor and equipment, Table 5 shows the total development costs associated with one smart mirror prototype. At an assumed salary of $57,000, 218 labor hours for all engineers yields $29,870.00 in labor expenses.

Table 5. Total Development Costs

Development Component	**Cost**
Equipment for Both Prototypes	$426
Labor	$29,870
Fringe Benefits, % of Labor	$10,155
Subtotal	$40,451
Overhead, % of Equipment, Labor, & Fringe Benefits	$48,541
Total	**$88,992**

7.2.3 Selling Price and Profit

The production of 4,000 smart mirror units employed with transparent display circuitry will be sustained over five years (800 units per year). When buying in bulk, the Raspberry Pi can be purchased for $34.50 per unit [27]. Buying in bulk yields a total equipment cost of $64 per unit. Advertising the product presents a 7% sales expense of the final selling price of $28. A group of workers will be employed at an hourly rate of $25 to assemble and test final products. At a market price of $400, the expected total revenue is $1,600,000. This reflects a $160 profit per unit sold, and a $800,000 profit over the five-year production period. Table 6 shows the selling price and expected profit per unit.

Table 6. Selling Price and Profit Per Unit (4,000 units)

Expense or Income	**Dollar Amount**
Equipment cost	$63
Assembly Labor	$10
Testing Labor	$10
Total Labor	$20
Fringe Benefits, % of Labor	$6
Overhead, % of Equipment, Labor & Fringe Benefits	$107
Subtotal, Input Costs	$196
Sales Expense	$28
Amortized Development Costs	$16
Subtotal, All Costs	$240
Profit	$160
Selling Price	***$400***

8. Current Status

Presently, the team has determined the equipment that will be ordered for prototype creation. Each member took a tour of Dr. Tentzeris' lab and has gained permission to utilize his ink jet printers for transparent display circuitry. Printing a circuit to display images still needs to be researched. Furthermore, combining floating touch with the aforementioned circuitry must be tested after thorough investigation. After researching and ordering the necessary equipment, the team will begin to construct the system.

9. References

[1] J. Greenough, “The ‘Internet of Things’ will be the world’s most massive device market and save companies billions of dollars”, *Business Insider*, Feb. 18, 2015. [Online]. Available: http://www.businessinsider.com/the-internet-of-things-market-growth-and-trends-2015-2. [Accessed Nov. 20, 2016].

[2] R. Henderson, “Best smarthome and IoT gadgets of CES 2016”, *Pocket Lint*, Jan. 11, 2016. [Online]. Available: http://www.pocket-lint.com/news/136306-best-smarthome-and-iot-gadgets-of-ces-2016-samsung-lg-and-more. [Accessed Nov. 14, 2016].

[3] D. MacCormack, “Planar Launches LookThru OLED Transparent Display”, *Commercial Integrator*, Nov. 05, 2015. [Online]. Available: http://www.commercialintegrator.com/article/planar_launches_lookthru_oled_transparent_display. [Accessed Nov. 17, 2016].

[4] M. Braun, “My Bathroom Mirror Is Smarter Than Yours”, *Medium*, Jan. 29 2016. [Online]. Available: https://medium.com/@maxbraun/my-bathroom-mirror-is-smarter-than-yours-94b21c6671ba#.rfe8jsoel

[5] A. Stadler, "Transparent conducting Oxides—An up-to-date overview," *Materials*, vol. 5, no. 12, pp. 661–683, Apr. 2012.

[6] K. Kiyokawa, Y. Kurata, and H. Ohno, "An optical see-through display for mutual occlusion with a real-time stereovision system," Proc. IEEE, vol. 25, no. 5, pp. 765–779, Oct. 2001.

[7] Planar Systems Inc., “Transparent OLED Displays”, *Planar Systems Inc.*, n.d. [Online]. Available: http://www.planar.com/innovations/transparent-oled/. [Accessed Nov. 20, 2016].

[8] G. Barrett and R. Omote, “Projected-Capacitive Touch Technology,” Information Display, vol. 32, no. 3, p.16, March, 2010. [Online serial]. Available: http://www.walkermobile.com/March_2010_ID_Projected_Capacitive.pdf [Accessed Nov. 20, 2016].

[9] Cypress Semiconductor Corp., “TrueTouch® Multi-Touch All-Points Touchscreen Controller,” CYTMA568 datasheet, May 2016.

[10] G. Walker. Part1: Fundamentals of Projected-Capacitive Touch Technology. Presented at SID Display Week ’14. [Online]. Available: http://www.walkermobile.com/Touch_Technologies_Tutorial_Latest_Version.pdf. [Accessed Nov. 20, 2016].

[11] S. H. Bae and Y. S. Lee, “Implementation of a High-Performance Touch Controller and Differential Sensing Circuit,” International Journal of Computer and Information Technology, vol. 3, no. 6, p. 1177, November, 2014. [Online serial]. Available: http://www.ijcit.com/archives/volume3/issue6/Paper030602.pdf [Accessed Nov. 20, 2016].

[12] Technavio, “Gesture Recognition, Coming to a Smartphone Near You”, *Infiniti Research Limited*, Jun. 30, 2015. [Online]. Available: http://www.technavio.com/blog/gesture-recognition-coming-to-a-smartphone-near-you. [Accessed Nov. 20, 2016].

[13] Microsoft, “Kinect Sensor for Xbox 360 Components”, *Microsoft*, n.d. [Online]. Available: https://support.xbox.com/en-US/xbox-360/accessories/kinect-sensor-components [Accessed Nov. 20, 2016].

[14] R. Boyle, V. Hlavac, and M. Sonka, *Image Processing, Analysis, and Machine Vision*, 4th ed., Stamford, CT: Cengage Learning, 2014. [E-book]. Available: https://goo.gl/yv3m0r [Accessed Nov. 20, 2016].

[15] Y. Lin, H. Tang, and T. Wu, “Real-time Object Image Tracking Based on Block-Matching Algorithm”, *University of Wisconsin-Madison*, 2006. [Online]. Available: https://homepages.cae.wisc.edu/~ece734/project/s06/lintangwuReport.pdf. [Accessed Nov. 20, 2016].

[16] S. Hymel, “Single Board Computer Benchmarks - learn.sparkfun.com.”, *learn.sparkfun.com*. [Online]. Available: https://learn.sparkfun.com/tutorials/single-board-computer-benchmarks. [Accessed Nov. 20, 2016].

[17] Raspberry Pi Foundation, “Raspberry Pi FAQs - Frequently Asked Questions,” *Raspberry Pi Foundation*, n.d. [Online]. Available: https://www.raspberrypi.org/help/faqs/. [Accessed Nov. 20, 2016].

[18] V. Varma, “Wireless Fidelity—WiFi”, *IEEE Emerging Technology Portal*, 2012. [Online serial]. Available: https://www.ieee.org/about/technologies/emerging/wifi.pdf. [Accessed Nov. 20, 2016].

[19] Bluetooth SIG, Inc., “What is Bluetooth?”, *Bluetooth SIG, Inc.*, 2016. [Online]. Available: https://www.bluetooth.com/what-is-bluetooth-technology/bluetooth-technology-basics [Accessed Nov. 20, 2016].

[20] P. F. plc, "Raspberry pi 3 model B technical specifications,". [Online]. Available: https://www.element14.com/community/docs/DOC-80899/l/raspberry-pi-3-model-b-technical-specifications. [Accessed Nov. 21, 2016].

[21] Redacacia, "Leap-motion-gesture," GitHub, 2016. [Online]. Available: https://github.com/Redacacia/leap-motion-processing-1. [Accessed Nov. 21, 2016].

[22] GPO, "ECFR — code of federal regulations," in *U.S. Government Publishing Office*, 2016. [Online]. Available: https://goo.gl/VEVGCy. [Accessed Nov. 20, 2016].

[23] HDMI, "HDMI: Installers: Inside an HDMI cable," in *HDMI*, 2003. [Online]. Available: http://www.hdmi.org/installers/insidehdmicable.aspx. [Accessed Nov. 20, 2016].

[24] World Standards, "Complete list: Plug, socket & voltage by country," in *Worldstandards*, World Standards, 2016. [Online]. Available: http://www.worldstandards.eu/electricity/plug-voltage-by-country/. [Accessed Nov. 21, 2016].

[25] B. Erickson, "Testing Wall Anchors and Picture Hangers," in *Today's Homeowner*, Today's Homeowner, 2016. [Online]. Available: http://www.todayshomeowner.com/testing-wall-anchors-and-picture-hangers/. [Accessed Nov. 20, 2016].

[26] One Day Glass. (2016). *Custom Glass Cutting* [Online]. Available: https://www.onedayglass.com- /custom-orders/custom-glass-processing/custom-glass-cutting/

[27] [27] MCM Electronics. (2016). *Raspberry Pi™ 3 Model B 1GB Project Board* [Online]. Available: https://goo.gl/ThFwjJ. [Accessed Nov. 20, 2016].

[28] [28] MCM Electronics. (2016). *Official Raspberry Pi™ Foundation 5V, 2.5A Power Supply* [Online]. Available: https://goo.gl/5bMRGz

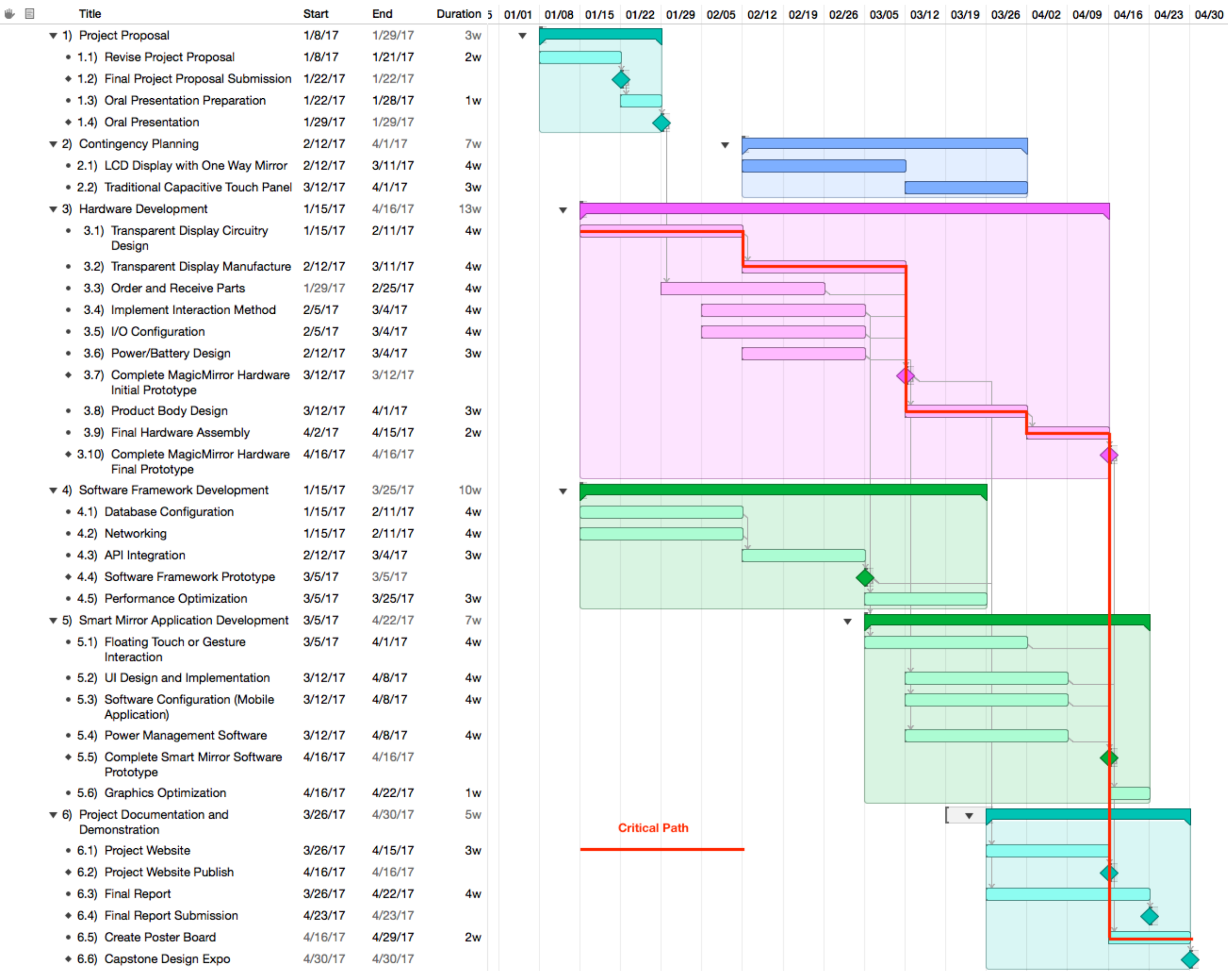
[29] TAP Plastics. (2016). *2-Way Mirrored Acrylic Sheets* [Online]. Available: http://www.tapplastics.com/product/plastics/cut_to_size_plastic/two_way_mirrored_acrylic/558. [Accessed Nov. 20, 2016].

[30] Newegg Inc. (2016). *Dell P2314T Black 23" IPS 8ms (GTG) Touchscreen LCD/LED Monitor, 270 cd/m2 8000000:1, USB 3.0 Port with 10 Touch Points Function with Smooth Operation* [Online]. Available: https://goo.gl/5RCi8K. [Accessed Nov. 20, 2016].

Appendix A - Task Assignment and Risk Level

Task Name	**Task Lead**	**Difficulty**	**Risk Level**
Proposal, Documentation, and Presentation	All	Low	Low
Project Proposal	All	Low	Low
Oral Presentation	All	Low	Low
Project Website	All	Low	Low
Final Report	All	Low	Low
Demonstration	All	Medium	Medium
Capstone Design Expo Preparation	All	Medium	Medium
Hardware Development	KK, BL, JO	High	High
Transparent Display Circuitry Design	JO, KK	High	High
Transparent Display Manufacture	JO, KK	High	High
Order and Receive Parts	KK, BL, JO	Low	Medium
Implement Interaction Method	KK	Medium	High
I/O Configuration	BL	Low	Low
Power/Battery Design	BL	Medium	High
Product Body Design	All	Low	Low
Final Hardware Assembly	KK, BL, JO	Low	Medium
Software Framework Development	NK, NS	Low	Low
Database Implementation	NK, NS	Low	Low
Networking Configuration	NK, NS	Low	Low
API Integration	NK, NS	Low	Low
Performance Optimization	NK, NS	Medium	Medium
Smart Mirror Application Development	All	Low	Low
UI Design and Implementation	NK	Medium	Low
Floating Touch or Gesture Interaction	KK, JO	Medium	Medium
Mobile Application and Setup Software	All	Low	Low
Power Management Software	BL, NS	Medium	Medium
Graphics Optimization	NK	Medium	Low
Contingency Planning	KK, BL, JO	Medium	Medium
LCD Display with Two-Way Mirror	KK, BL, JO	Low	Medium
Traditional Capacitive Touch Panel	KK, BL, JO	Medium	Medium

Appendix B - Gantt Chart



Title	Start	End	Duration
▼ 1) Project Proposal	1/8/17	1/29/17	3w
• 1.1) Revise Project Proposal	1/8/17	1/21/17	2w
◆ 1.2) Final Project Proposal Submission	1/22/17	1/22/17	
• 1.3) Oral Presentation Preparation	1/22/17	1/28/17	1w
◆ 1.4) Oral Presentation	1/29/17	1/29/17	
▼ 2) Contingency Planning	2/12/17	4/1/17	7w
• 2.1) LCD Display with One Way Mirror	2/12/17	3/11/17	4w
• 2.2) Traditional Capacitive Touch Panel	3/12/17	4/1/17	3w
▼ 3) Hardware Development	1/15/17	4/16/17	13w
• 3.1) Transparent Display Circuitry Design	1/15/17	2/11/17	4w
• 3.2) Transparent Display Manufacture	2/12/17	3/11/17	4w
• 3.3) Order and Receive Parts	1/29/17	2/25/17	4w
• 3.4) Implement Interaction Method	2/5/17	3/4/17	4w
• 3.5) I/O Configuration	2/5/17	3/4/17	4w
• 3.6) Power/Battery Design	2/12/17	3/4/17	3w
◆ 3.7) Complete MagicMirror Hardware Initial Prototype	3/12/17	3/12/17	
• 3.8) Product Body Design	3/12/17	4/1/17	3w
• 3.9) Final Hardware Assembly	4/2/17	4/15/17	2w
◆ 3.10) Complete MagicMirror Hardware Final Prototype	4/16/17	4/16/17	
▼ 4) Software Framework Development	1/15/17	3/25/17	10w
• 4.1) Database Configuration	1/15/17	2/11/17	4w
• 4.2) Networking	1/15/17	2/11/17	4w
• 4.3) API Integration	2/12/17	3/4/17	3w
◆ 4.4) Software Framework Prototype	3/5/17	3/5/17	
• 4.5) Performance Optimization	3/5/17	3/25/17	3w
▼ 5) Smart Mirror Application Development	3/5/17	4/22/17	7w
• 5.1) Floating Touch or Gesture Interaction	3/5/17	4/1/17	4w
• 5.2) UI Design and Implementation	3/12/17	4/8/17	4w
• 5.3) Software Configuration (Mobile Application)	3/12/17	4/8/17	4w
• 5.4) Power Management Software	3/12/17	4/8/17	4w
◆ 5.5) Complete Smart Mirror Software Prototype	4/16/17	4/16/17	
• 5.6) Graphics Optimization	4/16/17	4/22/17	1w
▼ 6) Project Documentation and Demonstration	3/26/17	4/30/17	5w
• 6.1) Project Website	3/26/17	4/15/17	3w
◆ 6.2) Project Website Publish	4/16/17	4/16/17	
• 6.3) Final Report	3/26/17	4/22/17	4w
◆ 6.4) Final Report Submission	4/23/17	4/23/17	
• 6.5) Create Poster Board	4/16/17	4/29/17	2w
◆ 6.6) Capstone Design Expo	4/30/17	4/30/17	

Appendix C - PERT Chart