# ECE 4012 Project Summary

| | |
|---|---|
| **Project Title** | **Ecosystem for Smart Glass Technologies (ESGT)** |
| **Team Members** (names and majors) | Najee Kitchens – CmpE<br>Kairi Kozuma – CmpE<br>Boa-Lin Lai – CmpE<br>Jonathan Osei-Owusu – CmpE<br>Nishant Shah – CmpE |
| **Advisor / Section** | Emmanouil Tentzeris – L3A |
| **Semester** | Spring 2016 – ECE 4012 |
| **Project Abstract** (250-300 words) | ESGT is a hardware and software ecosystem aimed towards the consumer market. The purpose is to provide a platform for integrating information into the daily routine seamlessly. It aims to replace traditional glass products – mirrors, windows, eyeglasses – providing relevant and useful information in an elegant package that does not interfere with the products' original purpose.<br><br>ESGT focuses on two main parts: the hardware technology to implement transparent displays and a software ecosystem to support hardware with application specific modules.<br><br>The prototype for ESGT is the smart mirror. This device maintains the qualities of a traditional mirror, while displaying useful information to the user. The mirror is implemented by attaching a traditional display behind a two-way mirror. Example data to be shown on the device include weather, time/date, calendar/agenda, news feed, and email. In addition, light, temperature, and weather sensors were incorporated to showcase the ease of adding new sensors. This can be extended to biosensors for health monitoring applications.<br><br>While the initial prototype draws power from an outlet, more portable versions can draw from a battery. To maintain security and privacy, the smart does not have a camera.<br><br>The user interface design takes advantage of the mirror's large screen size, but does not interfere with the mirror's original purpose; thus, information is shown at the edges of the display. In addition, the display can be turned off via gesture interaction. The design is minimalistic, resembling a normal mirror when off.<br><br>User interaction with the device is done through hover sensors, which detect gestures. Buttons diminish the aesthetic quality of the device, so it was not used for human machine interaction.<br><br>A smartphone application is used to pair and interact with the mirror via Wi-Fi for configuration. Each smart mirror can support multiple users, with the current user saved on Google Firebase. The configuration for each user is also backed up on Firebase, and can be instantly restored if needed. |

| **Project Title** | **Ecosystem for Smart Glass Technologies (ESGT)** |
| --- | --- |
| List **codes** and **standards** that significantly affect your project. Briefly describe how they influenced your design. | (i) IEEE 802.11 Wi-Fi - Since the mirror has wireless connectivity to communicate with the internet as well as the user's mobile device, the Wi-Fi chip on the microcontroller needs to be Federal Communications Commission(FCC) certified for use in the United States.<br>(ii) HDMI - To connect the microcontroller to the display behind the mirror.<br>(iii) SPI/I2C - To interface with the Hover gesture sensor.<br>(iv) Power – Standard 120V AC wall adapter to power the microcontroller and display.<br>(v) USB – Connect the mbed microcontroller as a slave device to read in analog inputs, using the virtual COM port over USB. |
| List at least two significant **realistic design constraints** that applied to your project. Briefly describe how they affected your design. | (i) The mirror needs to be aesthetically pleasing to look like a regular mirror to blend in with the surrounding area. A wooden frame with a stained façade was used for a minimalistic appearance.<br>(ii) Changes to user configuration on the mobile device had to be updated instantaneously on the smart mirror. To solve this, Firebase was used as the synchronization point for the mobile application, the backend server, and the frontend UI. |
| Briefly explain two **significant trade-offs** considered in your design, including options considered and the solution chosen. | (i) Touch screen vs. gesture sensor – The display was chosen with the touch screen to enable intuitive interaction with on-screen components; however, the capacitive screen could not detect touch behind the acrylic mirror. Instead, the Hover gesture sensor was used to receive input from the user. This gesture sensor had the added benefit of preventing smudging the display.<br>(ii) Raspberry Pi vs. mbed for analog input – The Raspberry Pi could support more analog inputs if a ADC was used; however, the mbed already natively supports six analog inputs. Ultimately, the mbed was used as a slave device to read analog inputs, and it transmitted data via the virtual COM port to the Raspberry Pi. |
| Briefly describe the **computing aspects** of your projects, specifically identifying **hardware-software** tradeoffs, interfaces, and/or interactions.<br><br>*Complete if applicable; required if team includes CmpE majors.* | The ESGT project is a complete design of an embedded system. Hardware aspects include materials, dimensions, circuits, power/energy, sensors, and aesthetics; the software component will deal with device setup, operating system, user interface, networks, and system security.<br><br>1. Powerful Microcontroller<br>The microcontroller must be powerful enough to run the frontend UI and backend database, with minimal performance lag.<br>2. Gesture Control<br>Different than traditional human interaction, the gesture sensor can let users to interact with the devices without touching it.<br>3. Low Resolution Display<br>Reduce the graphics computation load for the microcontroller, by limiting the number of pixels it needs to refresh.<br>4. Multi-User Support<br>Provide support for multiple users by using Firebase as a synchronization point between device and mobile application.<br>5. Networking Communication<br>Provide a wireless connection, so the device can be placed anywhere in the house. |