University of Central Florida

Department of Electrical Engineering and Computer Science
& College of Optics and Photonics

EEL 4914
Senior Design I

Smart Window

Initial Project and Group Identification Document
Divide and Conquer, Version 2.0

Authors:
Jake Pivnik - Photonic Science and Engineering
Pablo Calzada - Photonic Science and Engineering
Shaneal Findley - Computer Engineering
James Brunner - Electrical Engineering

Instructors:
Samuel Richie, PhD
David Hagan, PhD

Group #4

October 2, 2020

Table of Contents Page

1. Potential Customers

- Residents of larger cities that have limited space
- Residents of places that have harsh climates
- Small or large businesses
- Window and Glass Stores
- Homeowners

2. Project Narrative

2.1 Description

Windows have accompanied buildings for as long as humanity has constructed homes. Throughout their existence, the window has not changed much. However, as technology progresses ever further we find ourselves in a position to meaningfully update one of our oldest technologies. The Smart Window has a robust set of onboard sensors alongside WiFi connectivity to give accurate weather and forecast information pertaining to our immediate and regional areas. All the user needs to do is approach the window, which will automatically turn on, and display their local weather information including temperature, humidity, and UV index. The Smart Window will also have a transparency toggle to allow for the privacy of the user as well as everyone else inside the house. An early render of the Smart Window design can be seen below in Figure 1. Since windows are typically facing the Sun a good portion of the day, it makes sense to run our device on as much solar power as it can collect. This is done through the use of photovoltaic cells and batteries. All of these features will give its users a seamless experience to help them better decide how to spend their day with as little effort as possible.

Figure 1: Early Smart Window Project Rendering

2.2 Motivation

During the course of the year, many people have found themselves spending more time indoors. It isn't uncommon for people spending more time indoors to miss their outside areas. Typically, we only have one way to interact with the outside while still in our homes. Windows offer us a glimpse of the outside world, but don't give many details. We often use information like temperature, humidity, UV index, and a daily forecast to decide our outside activities. Unfortunately, this information often is conflicting, and comes from multiple remote sources. A weather forecast on the TV, and a zip-code forecast from our smartphones tell two different stories. This inspired our group to create this project. Additionally, the capability of the PDLC allows changing the windows opacity to reduce glare from outdoor lighting, and provide privacy. The motivation is to provide an adaptive device that informs people of the current atmospheric conditions indoors and outdoors while providing the optical appearance of a window with shutters or blinds.

2.3 Goals and Objectives

To design an easy to use smart window that displays real time weather information using on board sensors and WiFi connectivity while maintaining a low production cost. The system will have a privacy mode that makes the window opaque and the weather display should be integrated seamlessly to the glass pane to create an aesthetically pleasing user-friendly interface. The window should draw as little power from the home grid as possible, primarily relying on photovoltaic cells to charge up batteries built into the frame. If sunlight is scarce for a long enough time to drain the batteries then the window should switch to a separate supply coming from the home grid.

2.4 Project Function

The smart window will be designed to run in a household using solar energy coupled with batteries as the primary power source and a connection to the home power grid for cloudy or rainy days. The system will have real-time monitoring of basic weather conditions such as temperature, humidity and UV index as well as indoor temperature monitoring. Forecast information from internet sources will also be available to users. The system will have multiple privacy features such as the PDLC film that allows an opaque and transparent mode for the window. The proximity sensor will detect when the user is in front of the window and automatically switch from a low power mode to display the weather information in order to be energy efficient.

3. Requirements and Specifications

Marketing Requirement	Engineering Requirement	Justification
1,2	1600x900px Resolution Allows >50% light transmission in visible spectrum.	A large conventional LCD could be made translucent by backlight disassembly, yet offers a high quality image.
2,4,6	<60V | 2 ft x 3 ft	PDLC FIlm Offers privacy, and reduces glare
1,2	2 ft x 3 ft	Thickness should be enough to resist adverse weather
2,5	Custom built window pane, potentially wood	Should be aesthetically pleasing and be able to house sensors
2,4,5	Multiple lighting modes (12W Max power draw per color) 2700K / 6700K Light Temperatures	Using two lighting modes offers the user a color preference, and 1200 lumens should provide adequate lighting in any environment
2,5	LED Power control between 0-1 amp	Two knobs control current to offer variable lighting between 2700-6000K and between 0-2400Lumen
2	Matches length of window frame, completely covers LEDs	A scattering film and bulb housing gives the LED lighting a professional appearance.
3, 4, 5, 6	60V Max to PLDC film 3.3V to MCU 120VAC Input / Solar Panel Input	Photovoltaic cells charge batteries and connect to the home grid, given there is no sunlight. Power Supply to distribute electricity to components
1	ESP32/Raspberry Pi	Gets weather forecasting data to screen using wifi
1	Graphical display of temperature, date, and weather forecast in main screen	Display shows temperature, humidity and weather and serves to retrieve information from a website
1	MySQL database to contain user's first and last name, static face image file path in system, and main screen preferences	To offload memory from microcontroller, to retrieve database information from face recognition software when input face image is matched, and to saves settings for different users
1	3ft Range, in all lighting conditions	Proximity Sensor for the window to only display information when a user is nearby
1	+- 2% Relative Humidity / +- 0.5 C	Outdoor sensors to reliably determine the weather
1,3	+- 5% Relative Humidity / +- 2.0 C	Indoor Sensors determine the house conditions

Table 1: Requirements & Specifications

Market Requirements

1. The window must have components capable of delivering accurate information
2. The window must be aesthetically pleasing, without interfering with functionality.
3. The window should be affordable to the average consumer
4. The window should be accessible to a user of little technical ability
5. The window should use renewable energy/ have low power consumption
6. The window should be able to be transparent or opaque

3.1 Project Constraints

● Cost: The members of the group don't want to go over the $400 amount, in order to try and make the system affordable.

● Power Rating: The system must work primarily from solar energy and standard outlet as a backup

● Accessibility: The system must be able to handle users of basic technical knowledge.

● Time: The project must be completed within 2 semesters.

● Providing a quality product for a reasonable cost to the end user.

4. Project Block Diagrams

4.1 Project Constraints

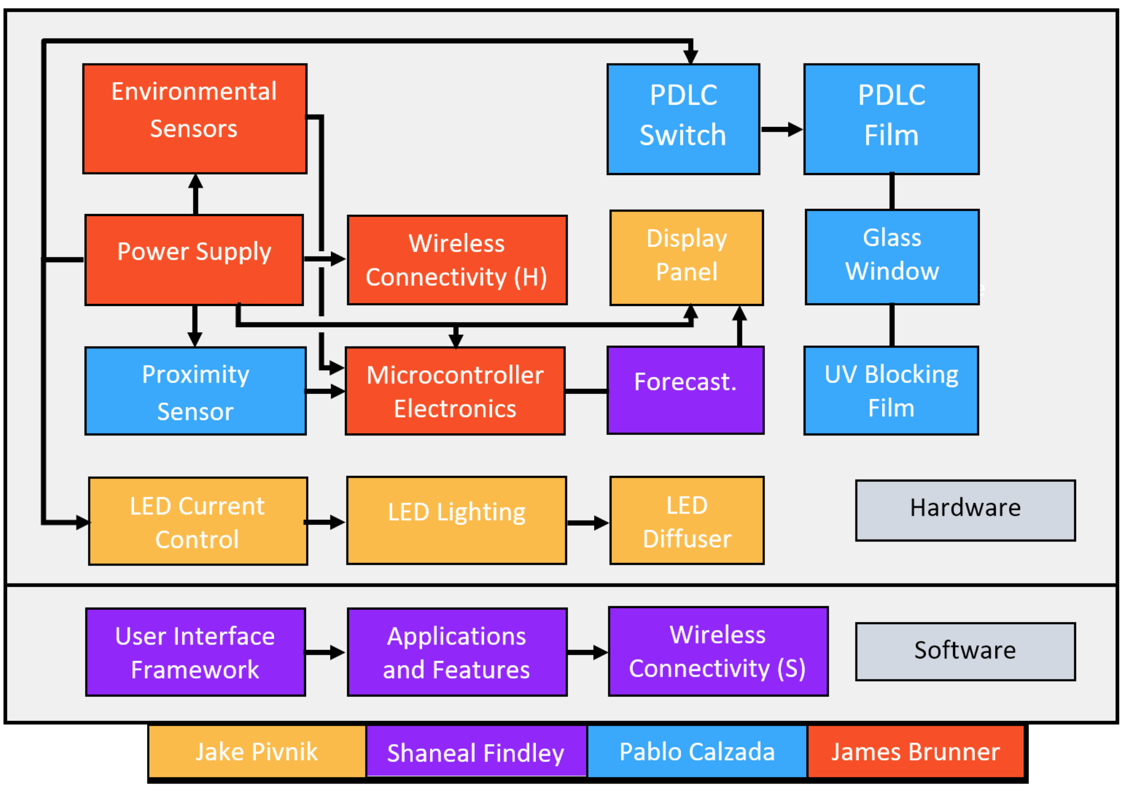

Figure 2: Overall Project Block Diagram

The Project Block diagram in Figure 2. shows a rough flow of component interaction. It has also been labelled with what group member is the lead on that component. The diagram is helpful for keeping track of essential parts of the design.

4.2 Software Flow Diagrams

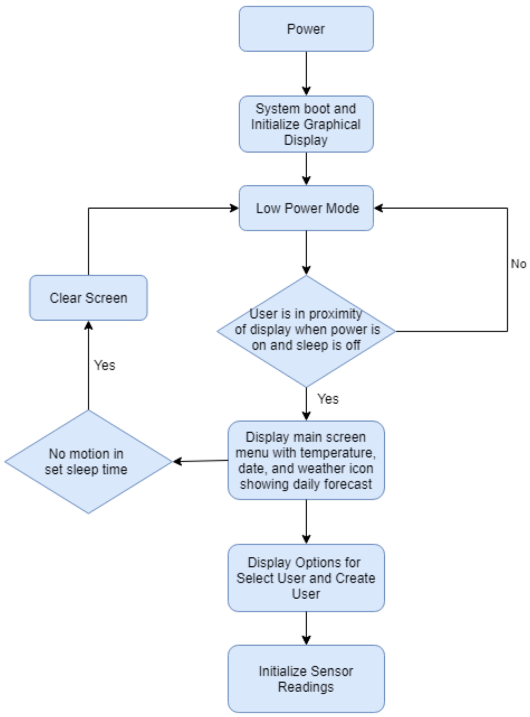

Figure 3: Software Block Diagram of Bootup/Startup Process

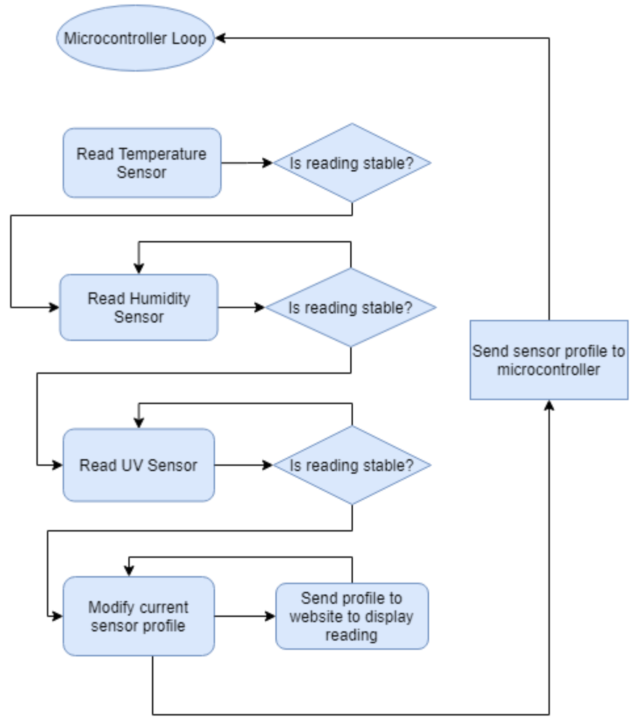

Figure 4: Software Block Diagram

Software flow diagrams are a useful tool for visualizing the processes that will allow our project to function. The startup process visualized in Figure 3 will aid in developing the protocols that wake the Smart Window from its sleep state. After initializing the device, it is important to continue to update the sensor readouts with accurate information as environmental conditions change. This will be handled by the microcontroller loop laid out in Figure 4.

4.3 Block Diagram Specific Information

4.3.1 Graphical Display:

The display will be used to interact with the users to display the option for the user to register themselves or to display the settings options for the user. The display will be operated through a graphics library API developed and hosted on GitHub. The display in low power 'sleep' mode will not display information. In the event that the proximity sensor reports motion it will trigger an interrupt to initialize the display software. The window will display its default screen with the temperature, date, and weather icon indicating the daily forecast retrieved from https://openweathermap.org/api. If no motion is reported in the default time or the time set by the user, then the window enters low power mode and has the display cleared. When the window initializes the display menu will display options to select and create users and set up the software to start voice and face recognition software. It will also communicate to the microcontroller to initialize screen and sensor readings.

4.3.2 Database:

A Raspberry PI's file system can be used to store the user's name and display settings. If Facial Recognition software is used we can use a remote database such as MySQL to store the sensor readings and store the image paths of each user static image in the Raspberry Pi filesystem.

4.3.3 Open Source Weather Software:

For the Raspberry to use the software it will require registering for an API key with OpenWeatherMap. Registration is free although the free API key has limits on the queries that you can make in a data. To poll the information in a local area it is required to fill in location, language, and preferred choice of units (metric, imperial or SI). OpenWeatherMap supports location by name, postal code, coordinates and ID.

DarkSky is an iOS weather app launched as a result of a Kickstarter campaign and is now purchased by Apple. The website provides wind, humidity, dew point, UV Index, visibility and pressure for the majority of metropolitan cities in the US. DarkSky APIs are not available to new sign ups after August 2020.

World Weather Online contains weather APIs to provide local weather, ski and mountain weather and marine and sailing weather data. Weather can be searched for using postal code and coordinates. It is delivered using standard HTTP requests and can be retrieved in the industry standard XML and JSON formats. In order to use the software, pricing starts at $5 per month. The standard 'starter' package includes 5 day local weather for major cities. The starter package allows for 2000 queries per day which make it possible to poll for the weather hourly.

4.3.4. Raspberry Pi:

A microcomputer is a small computer with a microprocessor as its central processing unit (CPU) with limited I/O, circuitry, and memory. The board that we will be discussing that fits into this category is the Raspberry Pi 3.

The Raspberry Pi 3 is a stand alone computer that can be used to connect to WiFi, connect to an OLED display. The Raspberry Pi is a Linux based microcomputer capable of running multiple distributions of embedded Linux Releases. The Raspberry Pi 3 has built in 4.2 Bluetooth and dual band Wi-Fi. This allows the sensor readings reported from the microcontroller selected to be reported to the internet without implementing the WiFi feature ourselves. This allows our device to be operated remotely, and with W-Fi, access a remote database that can store the users settings for the device.

The Raspberry Pi's 1.2 GHz clock speed and 1 GB RAM allows the Raspberry Pi to perform intensive processing actions such as facial recognition and voice in addition to processing sensor readings. It becomes inefficient for simple tasks due to its high power consumption.

As it is a microcomputer the Raspberry Pi has a filesystem that can host the required software to display on OLED and process sensor data. It can connect to an OLED display using 7 jumper wires and SPI (Serial Peripheral Interface) communication pins in the Raspberry Pi. If the microcontroller is present Implementing the connection between the Raspberry Pi and microcontroller will make the PCB design a challenge as the Raspberry Pi communicates through an SPI interface.

4.4 Microcontroller Communication Protocols:

4.4.1 Serial Peripheral Interface

One of the most popular interfaces to connect to sensors is SPI. SPI is a synchronous serial communication interface specification used for short distance communication. Applications include LCDs. SPI devices communicate in full-duplex mode using a master slave architecture with a single master and multiple slave communication. It requires four jumper wires. The graphical portion of the display can only be interfaced through SPI.

4.4.2 I2C: Inter-Intergrated Circuit

Another communication protocol to consider for this project is I2C. I2C is commonly used to attach lower-speed peripherals to processors and microcontrollers. I2C can be used to connect a microcontroller with different sensors or peripherals. It is a serial, half-duplex protocol that uses 2 bi-directional wires to communicate with other devices.

4.5 User Interface:

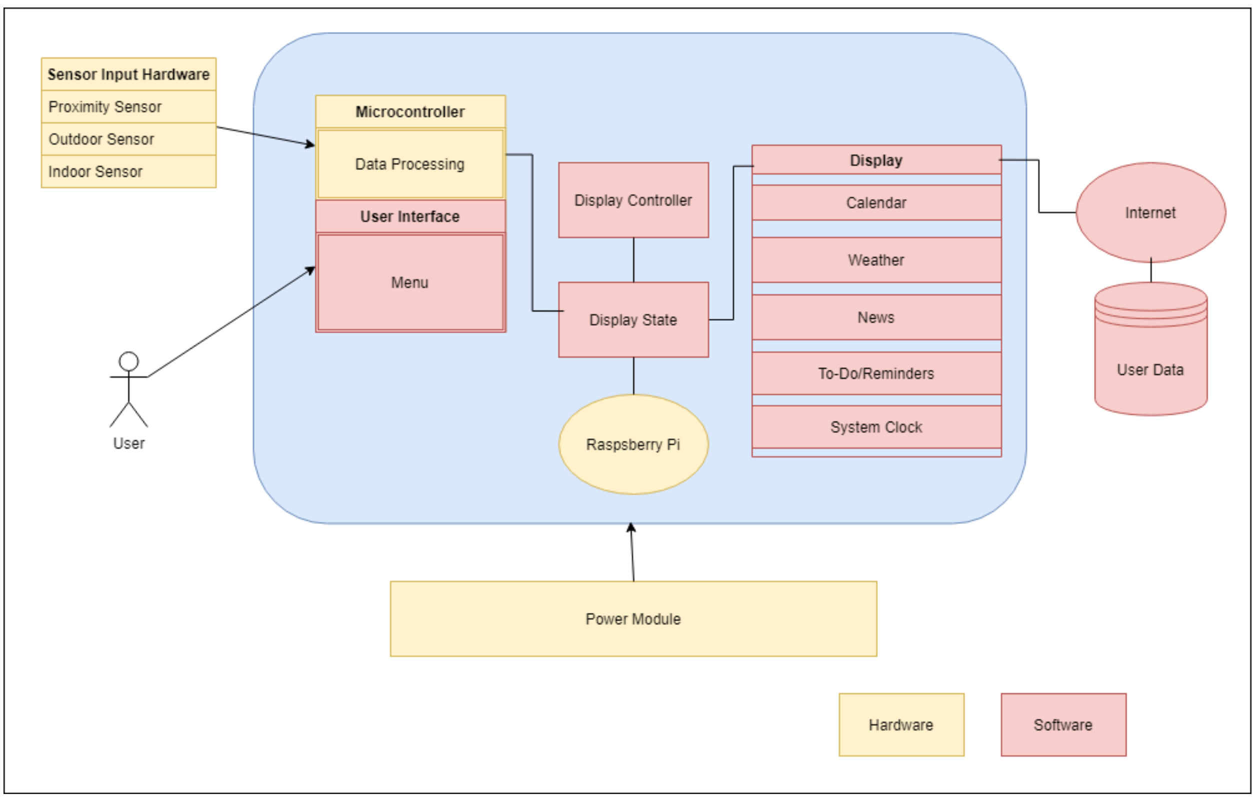

Figure 5: Data Flow in Smart Window Diagram

As the focus of the Smart Mirror is to provide a weather monitor that also serves as indoor furniture, it is required to develop an intuitive user interface. The Smart Window has the functionality of modern smartphone applications implemented in the form of widgets.The applications incorporated into the working design is the real time display of weather, a system clock to display the current system and time and can be obtained from the internet. A weather module to display the weather information based on manual latitude and longitude, the temperature based on the user's coordinates. The intent is to provide the user a Rich Site Summary (RSS) that allows the user to view updates directly and can choose what to view based on their preferences..

The User Interface must be large enough to display the temperature, data, and weather information in legible font and sparse enough for the user to be able to navigate the system to display additional temperature, weather information, and display information obtained from hardware sensors. Using an 1600 x 900 resolution that can display text can be used to suffice the basic implementation of the project.

Once the basic implementation has been achieved, we can include a more expanded interface to allow the user to store settings of their required display preferences and weather information that they choose to display. An online database will be used to store and obtain the user's calendar, and To-Do Reminders.

4.6 IEEE 802.11 (WiFi)

WiFi represents IEEE 802.11 and is part of the IEEE 802 set of local area network LAN protocol, and specifies the set of MAC and physical layer protocols for implementing wireless network communication. Our team is deciding to use WiFi as it is required to retrieve the RSS feed information from the internet to display in the user interface. The following table shows technical specifics of implementing WiFi for the smart window.

Specification	Value	Notes
Range	150 m indoors, 300 m outdoors	More than enough for project in use case scenarios
Data Rate	6.93 Gbps	Very high, more than enough for product
Latency	2ms - 3ms per RTT	Low, sufficient to achieve real-time accuracy
Power Consumption	400 mA (2.0 W) (3 B+ model)	Very high, will alway draw power unless disabled
Implementation Complexity	High	Very complex communication protocol to implement from scratch

While WiFi is used for wireless communication, there will be drawbacks to using WiFi as its power consumption in comparison to its benefits which include fast data transfer speeds, more range than other communication technologies such as Bluetooth (IEEE 802.15) in addition to more range and lower latency.

5. House of Quality Diagram

The house of quality diagram is used to investigate the relationships between the technical requirements of the project, and the expected market requirements of the consumer. Shown below in Figure 5, the HoQ diagram will aid in informing the groups decisions in prioritizing features and components of the design.

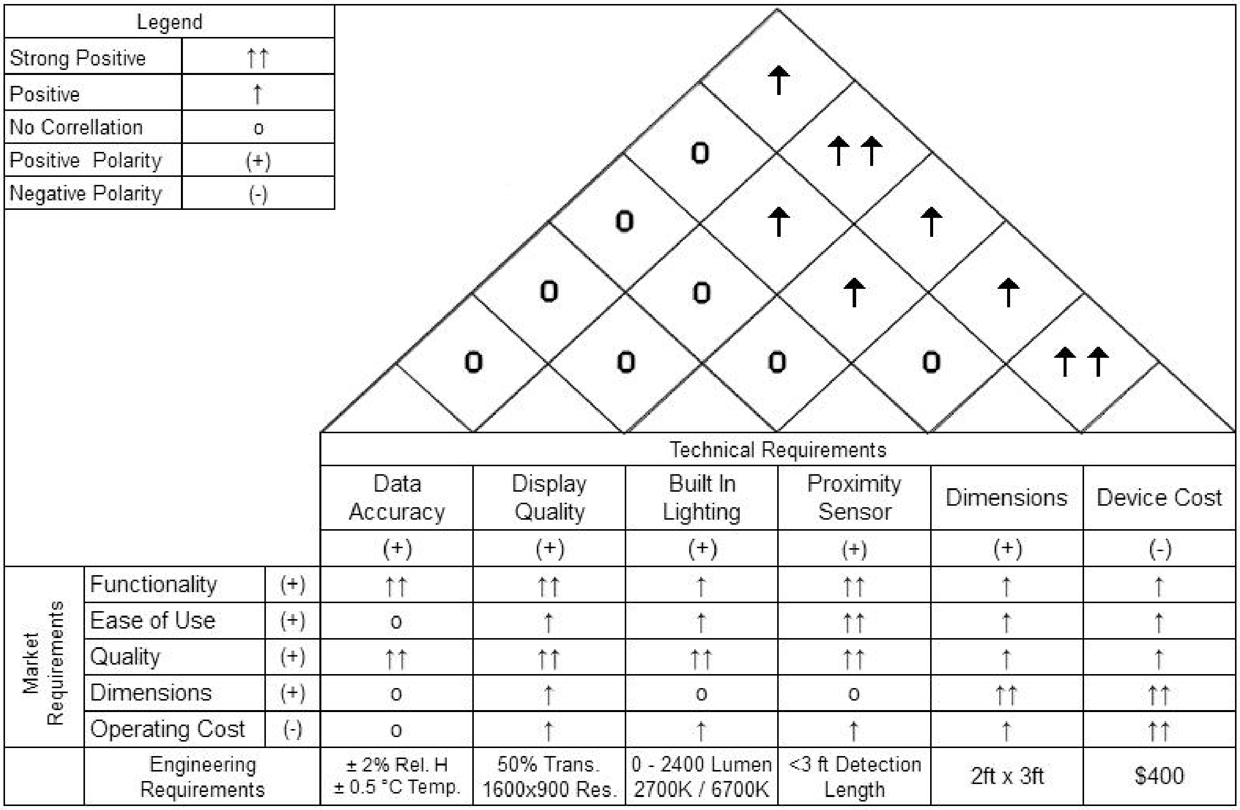

Figure 6: House of Quality Diagram

6. Project Budget And Financing

Our group estimates our project material cost to be in the range of $450 to $500 dollars. This budget is acceptable to the team. We also plan on using a modified, previously acquired, LCD as our main window display. Current estimations lead the group to believe a modified LCD may be translucent enough to function using sunlight as its primary backlight. If prototyping reveals this to not be the case, and we are unable to have a proper display sponsored, then the group will amend the budget to accommodate an appropriate display. Our current expected bill of materials has been tabulated below in table (2).

Expected Bill of Materials	**Expected Cost (USD)**
Window frame	<$30
Glass Pane	<$15
Interrupter switch	<$5
MCU	<$5
Solar Cells	<$80

UV Blocking Film	<$25
18650 Lithium ion batteries (8qty)	<$20
Battery Charger Protection Circuitry	<$5
Temperature/Humidity sensor (outdoor)	<$30
Temperature/Humidity sensor (indoor)	<$5
UV sensor	<$7
LED strips	<$25
Modified LCD	$0 (Already Owned)
LED diffuser	<$15
LED Control Electronics	<$10
Fuse	<$5
Power transformer	<$10
PDLC Film	$200
Total	$487
Group Member Contributions	$500 (Flexible)

Table 2: Project Budget & Financing

7. Initial Project Milestones

Major project milestones have been divided into two tables: Table 3 summarizes the important tasks for Senior Design I, and Table 4 lists the expected tasks for Senior Design II. These tables will be updated with more accurate dates and duration as the project continues.

7.1 Senior Design I

Fall 2020		
Description	**Duration**	**Dates**
Senior Design 1 Project Idea	5 days	August 24th - August 28th
Project Discussion	5 days	August 31st - September 4th
D&C Document 1.0	12 days	September 7th - September 18th
Initial Project Documentation	7 days	September 21st - September 25th
D&C Document 2.0	3 days	September 30th - October 2nd

60 Page Draft Report	5 days	November 9th - November 13th
100 Page Report	5 days	November 9rd - November 27th
Review Documentation	1 week	November 30th - December 7th
Final Document Due	1 week	December 8th, 2020

Table 3: SD1 Project Milestones

7.2 Senior Design II

Spring 2021		
Description	**Duration**	**Dates**
Build Prototype	TBD	TBD
Testing and Redesign	TBD	TBD
Finalize Prototype	TBD	TBD
Peer Presentation	TBD	TBD
Final Report	TBD	TBD
Final Presentation	TBD	TBD

Table 4: SD2 Project Milestones