FACIAL QUANTUM MATCHING MIRROR

By

[Akhil Morisetty]

[Alex Cheng]

[Ethan Zhang]

Final Report for ECE 445, Senior Design, [Spring 2026]

TA: [Wesley Pang]

[06] [May] 2026

Project No. 08

Abstract

This project involves the development of the "Facial Quantum Matching Mirror," an interactive prototype designed for the Illinois Quantum and Microelectronics Park. The goal was to create a visually engaging experience that uses facial recognition to "match" a user’s reflection with a scientist, entrepreneur, or an engineer. Our design uses a one-way mirror with a hidden display, a camera, and a control box with buttons to select the category. When a user interacts with the mirror, the system captures their image, identifies the best historical match, and displays that figure directly behind the glass, creating a "face-to-face" illusion. We successfully verified the hardware’s ability to manage power, control the lighting, and process the facial matching logic in real-time. This prototype serves as a proof-of-concept to help secure future investment for a full-scale museum installation.

Contents

1. Introduction

This report presents the Facial Quantum Matching Mirror, an interactive prototype that uses a one-way mirror and facial recognition to match a user with a visually similar scientist, engineer, or entrepreneur. The following sections describe the problem the prototype addresses, the proposed solution, the physical layout of the device, and the high-level requirements against which it was evaluated.

1.1 Problem

Illinois Quantum and Microelectronics Park is a major initiative aimed at advancing quantum computing and microelectronics research in Illinois. As part of this effort, Professor Kwait is seeking a viable prototype of a Facial Quantum Matching Mirror to demonstrate to potential investors and stakeholders. The goal of this project is to develop a functional and visually appealing prototype that can showcase the concept and support future development of a more advanced, museum-ready version for integration into the park.

1.2 Solution

We propose a Facial Quantum Matching Mirror, an interactive display device that combines a one-way mirror and facial recognition technology to match a user's facial features with notable figures in selected categories such as engineers, scientists, or entrepreneurs. The system creates an engaging interactive experience by allowing users to compare their facial features with recognized individuals from a chosen category.

The system consists of a one-way mirror, a display panel of equal size mounted behind the mirror, a camera, local storage, a microcontroller, and user input buttons, all integrated within a single frame. When the system is idle, the display remains dark, allowing the one-way mirror to function as a standard reflective surface so the user sees only their own reflection.

When a user selects a category through the input buttons, the microcontroller transmits the selected category to the main processing unit through Bluetooth communication. The camera then captures an image of the user, and the facial recognition backend processes the image to identify the most visually similar individual within the selected category. The system retrieves the corresponding media from local storage and displays the result on the screen behind the mirror, creating the illusion of interacting directly with the matched figure.

Compared to the original project proposal, the LED illumination subsystem was removed from the final design. The originally selected LED components required a minimum supply voltage of

9 V for proper operation, which exceeded the output capability of the custom PCB power system and would have introduced additional power regulation complexity. After further system integration, it was determined that the LED subsystem was not necessary for core functionality. The facial recognition program, implemented using OpenCV, was modified to support gesture-based activation through a thumbs-up detection process, allowing the system to initialize directly through camera input. This modification simplified the hardware design while preserving the intended user interaction workflow.

1.3 Visual Aid

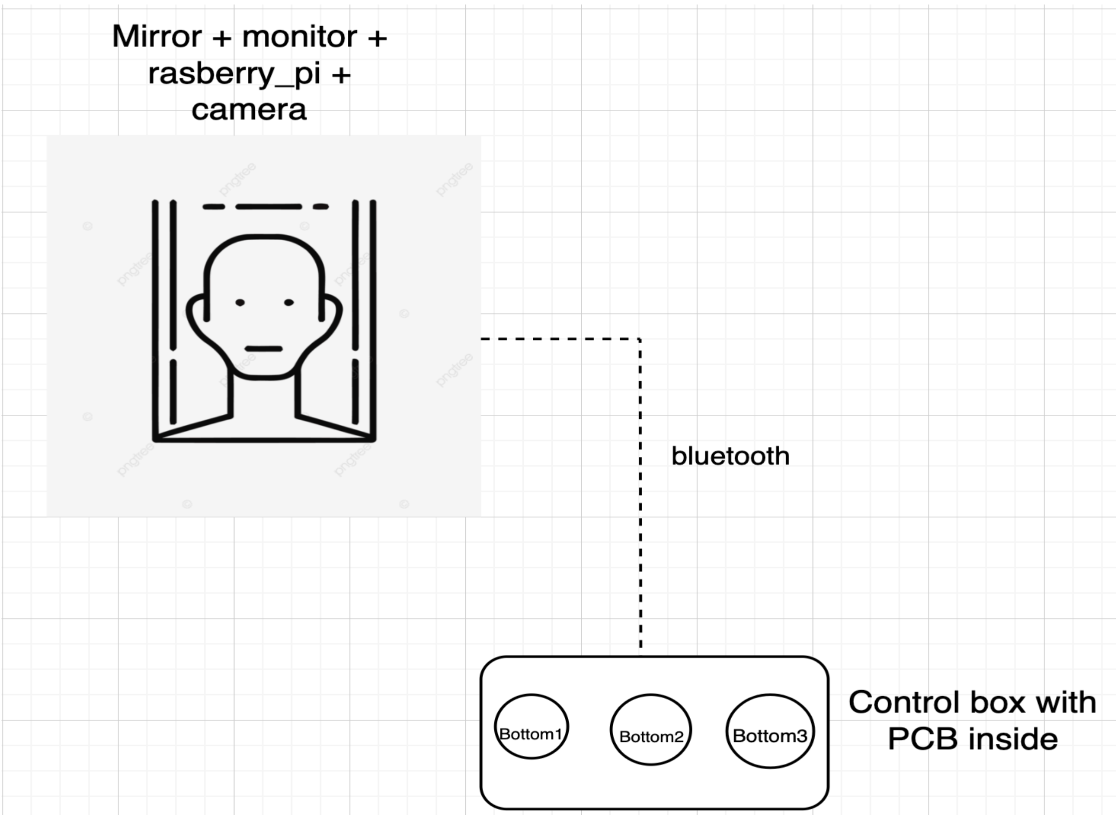


Figure 1: Visual Aid

1.4 High Level Requirements

1. The system should complete the full single-user interaction cycle (image capture -> facial matching -> display result) within 20 seconds of user activation
2. While the monitor is not displaying anything, the mirror should have 70% reflectivity and give the visual effect of a mirror, and when the monitor is on, the monitor screen should be visible even under 30% reflectivity.
3. The facial recognition model should return a visually accurate match that achieves at a minimum an 85% accuracy compared to a baseline recognition. (We believe baseline recognition should be a group of 3 or more human TAs)

2. Design

2.1 System Block Diagram

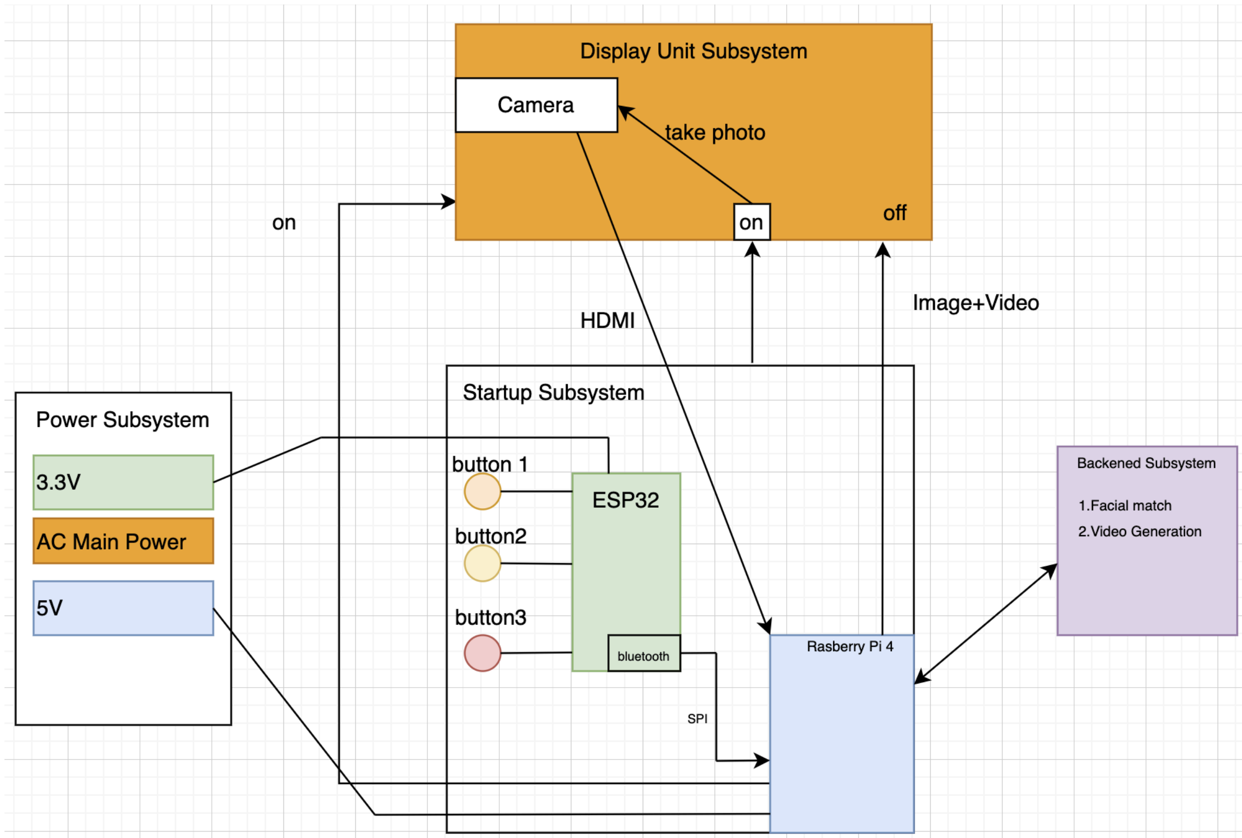


Figure 2: System Block Diagram

2.2 Raseberry Pi System Diagram

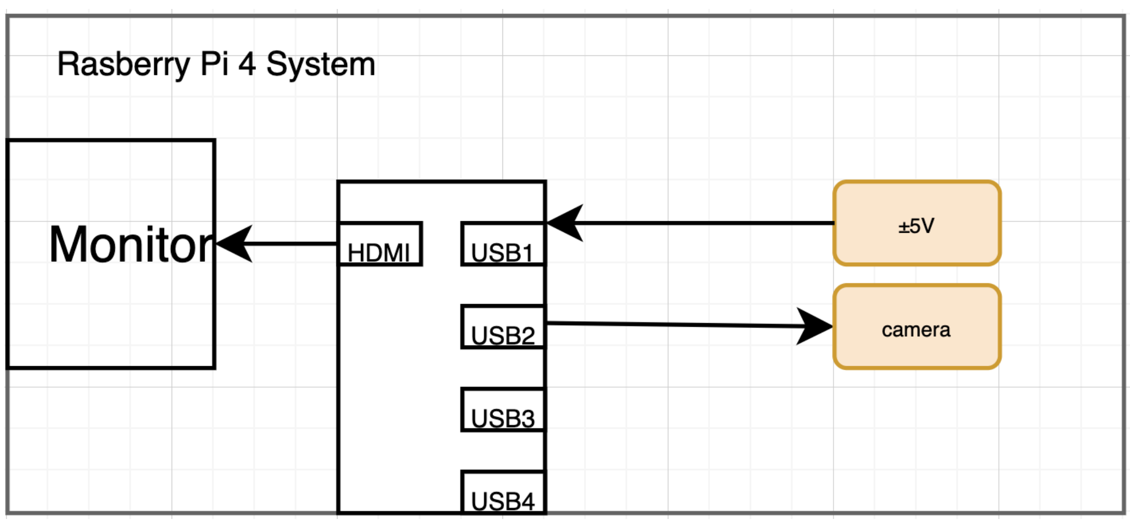


Figure 3: Raspberry Pi System Diagram

2.3 Design Procedure

2.3.1 Microcontroller Selection

The system uses an ESP32-S3 as the microcontroller for handling user input and wireless communication. The ESP32 was selected because it integrates Bluetooth functionality directly into the microcontroller, eliminating the need for an external Bluetooth module and reducing PCB design complexity. Compared with alternative microcontrollers that require additional wireless communication modules, the ESP32 provides a more compact and cost-effective solution while maintaining sufficient computational capability for input processing and communication tasks.

2.3.2 Processing Unit Selection

The main processing unit of the system is the Raspberry Pi 4. Facial recognition and image analysis require significantly more computational resources than can be provided by a microcontroller such as the ESP32. Therefore, a separate embedded processing platform was required. The Raspberry Pi 4 provides sufficient processing capability to execute the facial recognition model while maintaining reasonable system cost. An alternative option considered was the Raspberry Pi 5, which offers increased memory and improved processing performance, allowing for faster model execution and larger facial datasets. However, due to budget limitations, the Raspberry Pi 4 was selected as the more practical implementation.

2.3.3 Mirror Material Selection

A 70% reflective one-way mirror was selected as the primary display interface. Unlike a standard mirror, which provides nearly complete reflection of incident light, a partially reflective one-way mirror allows light from the display behind it to pass through when the display is active while maintaining reflective properties when the display is inactive. This property enables the mirror to function both as a conventional mirror and as a display surface for the facial matching result. The selected mirror material also provides flexibility in fabrication, allowing it to be cut and adjusted to fit the system frame dimensions.

2.3.4 Camera Placement Design

The camera was mounted externally on the system frame rather than behind the one-way mirror. Since the camera does not require concealment behind the reflective surface, external placement simplifies mechanical integration and improves image capture quality. This design choice allows the camera to receive direct ambient lighting, improving image clarity and increasing facial recognition accuracy compared to placement behind the partially reflective mirror.

2.4 Design Details

2.4.1 Microcontroller and Input Circuit

The startup subsystem is built around an ESP32-S3-WROOM-1 module mounted on a custom printed circuit board (PCB). Power enters the board through a power jack, the power jack would supply 5V and 0.2A to the regulation circuit described in Section 2.4.2. The 3.3 V output of the regulator powers the ESP32 module and is decoupled by a 22 μF bulk capacitor and a 0.1 μF high-frequency bypass capacitor placed adjacent to the module.

User input is provided through three momentary push buttons corresponding to the Scientists, Engineers, and Entrepreneurs categories. Each button is wired to a dedicated general-purpose input/output (GPIO) pin on the ESP32. A 10 kΩ pull-down resistor holds each GPIO line at ground when the button is unpressed, and a 1 μF capacitor in parallel with the resistor provides hardware debouncing. The resistor-capacitor (RC) time constant of this network is

$$\tau = R \cdot C = (10\ k\Omega)(1\ \mu F) = 10\ ms,$$

which exceeds the typical mechanical bounce duration of approximately 1–2 ms for a tactile push button while remaining short enough that the resulting input latency is imperceptible to the user.

Two additional switches on the board are wired to the enable and boot-strap pins of the ESP32, providing a hardware reset and a means of forcing the module into bootloader mode for firmware programming. These are not user-facing controls and are not exposed through the enclosure.

When a category button is pressed, the firmware encodes the selection as a single-byte identifier and transmits it to the Raspberry Pi over Bluetooth Low Energy (BLE), with the two devices paired automatically at boot time. Bluetooth was selected over a wired connection to allow the user control panel to be physically separated from the mirror frame.

The USB_C receptacle is used as an port to upload our Arduino code to the mcu, but we did not use it to supply the power. We also include the UART section in our pcb just in the case if the USB_C receptacle doesn't work; however our USB_C works well eventually so that part is redundant.

2.4.2 Power Regulation Circuit

The custom PCB receives 5 V from the power jack and steps it down to 3.3 V to supply the ESP32-S3. Figure 5 shows the regulation circuit, which is built around a Diodes Incorporated AP62250WU synchronous buck converter. A buck converter was chosen over a linear low-dropout (LDO) regulator because of its higher efficiency at this voltage step, which reduces thermal load on the small PCB and avoids the need for a heat sink.

The output voltage is set by the feedback resistor divider formed by R7 (31.6 kΩ) and R8 (10 kΩ), according to

$$V\_OUT = V\_FB \cdot (1 + R7 / R8),$$

where V_FB is the AP62250's internal reference voltage of 0.8 V. Substituting:

$$V\_OUT = 0.8\ V \cdot (1 + 31.6\ k\Omega / 10\ k\Omega) = 3.328\ V.$$

This places the output within 1% of the 3.3 V target and well inside the ESP32-S3's specified supply tolerance of 3.0 V to 3.6 V. A 2.2 μH inductor and a 0.1 μF bootstrap capacitor complete the switching network. The output filter consists of two 22 μF ceramic capacitors in parallel, providing approximately 44 μF of total output capacitance to keep ripple below 50 mV at the regulator's nominal switching frequency. The input is decoupled by a 10 μF ceramic capacitor placed close to the V_IN pin of the converter.

2.4.3 Mirror and Monitor Configuration

The display surface is an 18 × 24 inch tempered glass mirror with 70% reflectivity and 30% transmittance, mounted in the front face of the wooden frame. A Samsung S22D304 21.5 inch monitor with a rated brightness of at least 350 cd/m² and a native resolution of 1920 × 1080 pixels is mounted directly behind the glass, with its display surface positioned approximately 5 mm from the rear of the mirror.

The choice of mirror reflectivity and minimum monitor brightness was driven by the optical tolerance analysis introduced in Section 2.7. With an ambient illuminance $L = 500$ lux representative of indoor museum lighting, an effective scene reflectance $\rho = 0.3$, mirror reflectance $R = 0.7$, mirror transmittance $T = 0.3$, and a contrast factor $K = 3$, the ambient luminance reflected toward the user is

$$L\_ambient = (L \cdot \rho) / \pi = (500 \cdot 0.3) / \pi \approx 47.7 \text{ cd/m}^2,$$

and the minimum required display luminance behind the mirror is

$$L\_display \geq (L\_ambient \cdot K \cdot R) / T = (47.7 \cdot 3 \cdot 0.7) / 0.3 \approx 333.9 \text{ cd/m}^2.$$

The Samsung S22D304's rated brightness of at least 350 cd/m² therefore satisfies this requirement with a small margin. In practice, the matched figure is rendered against a dark background to further maximize the perceived contrast between the figure and the user's reflection.

2.4.4 Software Pipeline

The Raspberry Pi 4 runs a Python application built on OpenCV that handles gesture detection, face capture, embedding computation, database lookup, and result rendering. While idle, the application reads frames from the C270 and watches for a thumbs-up gesture; once detected, and given a category selection received over Bluetooth Low Energy from the ESP32, it captures a single 1280 × 720 frame, crops and normalizes the user's face, and passes it through a pretrained convolutional neural network to produce a fixed-length embedding. The embedding is compared by cosine distance against precomputed embeddings stored in a local database, with one entry per figure in the selected category..

2.5 Physical Project

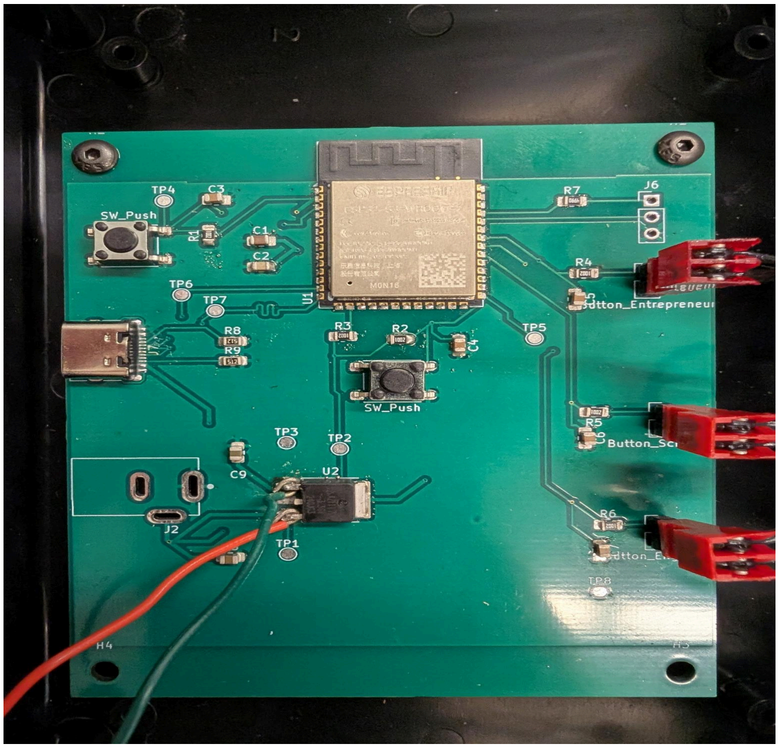


Figure 4: PCB

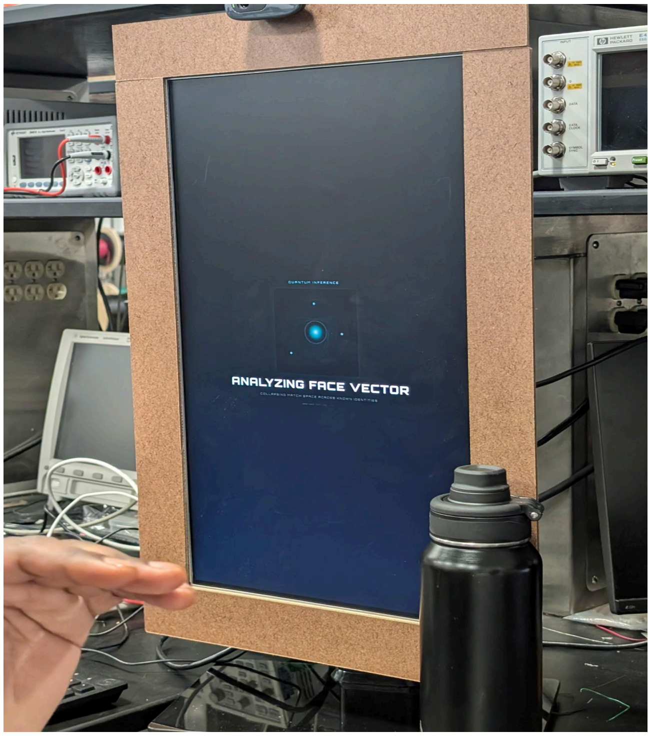


Figure 5: Monitor

3. Design Verification

3.1 Display Subsystem

This subsystem serves as the user interface layer of the smart mirror and is responsible for capturing the user's image and presenting the final matched result. It consists of a Logitech C270 HD webcam, a monitor with a minimum brightness of 350 cd/m², and a 70% reflective glass mirror, all mounted within a wooden frame. In idle mode, the monitor remains off so that the 70% reflective glass acts as a conventional mirror. When the user activates the system through the startup subsystem, the Raspberry Pi triggers the USB camera mounted at the top of the frame to capture an image of the user's face, which is then sent to the backend subsystem for facial recognition processing. While the image is being processed, the monitor displays a loading animation, and once the backend returns the closest match the corresponding figure is rendered on the monitor and becomes visible to the user through the 30% transmittance of the glass.

Requirements	Verification
The camera on the display unit must be able to capture an image of the participant's head in < 3s of activation	1. Set up trial testing upon camera activation to repeat every time a picture is taken. 2. Use a timer that starts when the camera activation begins, and when the picture is taken 3. Every trial should result in a picture being taken within 3 seconds.
The display unit must be reflective like a normal mirror while it is idle and on standby	1. Place the mirror in different lighting to test in variable environmental conditions. 2. When the display unit is in idle or standby, make sure the unit is reflective like a normal mirror

Table 1: Display Unit Requirements and Verification

3.2 Startup Button Subsystem

This subsystem handles the start of the matching process. The participant selects one of three quantum category buttons, which both chooses the category and triggers the camera and scan process, giving them direct control over when the interaction begins. The buttons are wired to an ESP32 microcontroller on a custom PCB housed in a separate control box, and on each press the microcontroller transmits the selected category to the Raspberry Pi over Bluetooth, where it is used to filter the facial recognition database.

The startup buttons directly contribute to the high-level requirements by providing the primary input from the participant and activating the mirror, taking the system through its states and producing an output. The microcontroller contributes to the high-level requirements by coordinating between the buttons and the Raspberry Pi to initiate the cycle.

Requirements	**Verification**
The microcontroller system must reliably receive user input signals from buttons and forward valid state data to the Raspberry Pi processing unit without loss in < 100 ms	1. Press each of the user input buttons and record timestamps each time the button is pressed 2. Record the time stamp when the Raspberry Pi received the data 3. Verify that the difference between the two time stamps is <100ms 4. Repeat process 20 times to ensure consistency
The microcontroller system must boot and enter operational state automatically when powered up without manual operations, and be able to operate continuously for 2 hours without communication failure or reset	1. Power cycle the microcontroller and verify the booting process as it enter an operational state without manuel help 2. Complete the power cycle process five times to verify consistency

	3. Have the system run continuously for 2 hours while sending input signal at regular intervals 4. Verify no failures or rests occurred during the 2 hours

Table 2: Startup Button Subsystem - Requirements & Verification

3.3 Power Unit Subsystem

The power unit powers all of the components within the project. Our power unit will power the monitor, LEDs, and microcontroller. The control pad/startup button will be powered by a three-volt battery. The Raspberry Pi power module will be used to power specifically the Raspberry PI which has the backend software programmed onto it. The power unit subsystem provides power to two major components: the monitor and the Raspberry Pi. ADD INFORMATION ABOUT 5 to 3.3V step down DC-DC linear regulator circuit that we added under the power unit subsystem

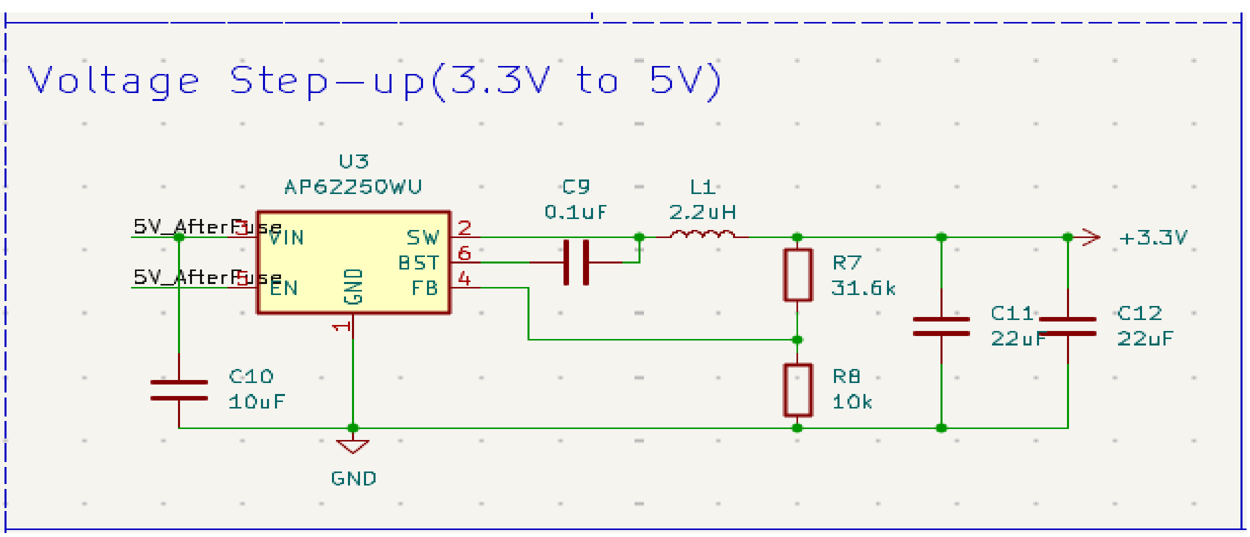


Figure 6: Power Subsystem Schematic

Requirements	Verification
The power unit must safely supply a stable 3.3 volts to our ESP-32 Microcontroller during peak usage	1. Connect a multimeter to the voltage input pins. 2. Run the ESP-32 when under the high load 3. Record the voltage across 2 hours 4. Verify that the voltage stays within 3.3V±.1V
The 5V to 3.3V step-down regulator must regulate the output voltage within ±.1V of 3.3V under all of the different load conditions	1. Connect a multimeter to the regulator output 2. Test under the different loads: no load, half load, and full load conditions. 3. Record the voltage output for each part of the regulator 4. Verify reading is between 3.2-3.4V

Table 3: Power Subsystem - Requirements and Verification

3.4 Backend: Database and Facial Recognition

This subsystem receives the captured user image from the Logitech camera and the selected category, computes a face embedding from that image using a pretrained and possibly fine-tuned recognition model. Those embeddings will then be compared to a local database of precomputed embeddings. The backend should return the closest matching match ID to the Raspberry Pi.

The backend: database and facial recognition subsystem performs all of the image processing and image matching to result in an output on the mirror and a matched image of the participant.

Requirements	Verification
The facial recognition system should return a match result within 10 seconds of image capture	1. Capture a test image using the Logitech camera 2. Start a time when the image is captured 3. Stop the timer when the Raspberry Pi receives the matched result 4. Repeat for 25 different images and verify that the results are returned in under 10 seconds
The system should achieve > 85% top-match accuracy per category using a defined validation image set	1. Get ready a set of validation images of individuals with accurate matches for each category 2. Run each of the images through a facial recognition system 3. Verify whether the top returned image is the correct one for the validation set.
The database should correctly store and retrieve records with a near 100% success rate	1. Add 50 test records 2. Simulate data to try to retrieve each of the records in the database 3. Record any retrievals that have failed or are incorrect 4. Verify that all the records are retrieved

Table 4: Backend Subsystem - Requirements and Verification

3.5 Tolerance Analysis

An important aspect of the design that poses a risk to the successful completion of the project is how the two-way mirror operates. If the mirror operates successfully, the mirror should act like a reflective mirror when the system is idle, and when the system changes states, the mirror should be transparent to display the output image. Depending on various lighting conditions and other variable factors, the mirror poses a risk of not acting correctly for the design.

Optical Tolerance Analysis:
L = Lux of the environment that the mirror is placed in
R = mirror reflectance from the human perspective
T = mirror transmittance from the human perspective
B = brightness of the monitor
p = effective reflectance of the mirror
K = contrast factor

In our case, we plan to find a 70% reflective piece of glass, use a monitor with a minimum of 350 cd/m^2, and place our system in replicable museum lighting, which can be equated to bright indoor lighting. We are also going to assume the mirror will be reflecting clothes, skin, and interior building items under average indoor circumstances. K is a contrast factor that means how many times brighter the display is than the reflection for the users to be comfortable seeing the content on the monitor. Thus our variables become:
L = 500 lux | R = 0.7 | T = 0.3 | B = 350 cd/m^2 | p = 0.3 | K = 3
Since our glass will be 70% reflective, when the monitor is in idle state, there wont be any issues with the glass acting as a mirror. The hard part is determined when the monitor is at 350 cd/m^2 brightness into a 30% transmittance. For the monitor to be visible from the human perspective we need:
L-ambient = (L * p) / pi = (500 * 0.3) / pi = 47.7 cd/m^2
L-display >= (L-ambient * K * R) / T
L-display >= (47.7 * 3 * 0.7) / 0.3 = 333.9 cd/m^2
This indicates to us that in order to have our monitor be visible with a contrast factor of 3 under 500 lux lighting conditions, we need a monitor that has a brightness specification of around 340 cd/m^2. Obviously, the luminance of the environment, contrast factor, reflectivity, and transmittance will not be exactly similar to our numbers, but a rough estimate guarantees us that we need to find a monitor with a minimum 300 cd/m^2 brightness and glass with a maximum 70% reflectivity.

4 Cost and Schedule

3.1 Cost Analysis

3.1.1 Labor

On average, the UIUC ECE Graduate makes approximately $86,000/year. That comes down to $41/hour. Based on the formula: $41/hr × 2.5 × 60 hrs = $6,150 per person

Team Member	Rate ($/hr)	Multiplier	Hours	Total
Akhil Morisetty	$41	2.5	60	$6,150
Alex Cheng	$41	2.5	60	$6,150
Ethan Zhang	$41	2.5	60	$6,150
Total Labor				**$18,450**

Table 5: Labor Costs

3.1.2 Parts

Description	Manufacturer	Quantity	Part #	Total
Raspberry Pi 4 Model B (2GB)	Raspberry Pi	1	SC15184	$65.95
ESP32 WROOM-32 Development Board (WiFi + BT)	HiLetgo	1	ESP-WROOM-32	$10.99
Logitech C270 HD Webcam (720p)	Logitech	1	960-000694	$29.99
WS2812B RGB LED Strip 5V, 16.4ft, 60LED/m	BTF-Lighting	1	WS2812B-5M	$15.99
18x24" Tempered Two-Way Mirror Glass (70% reflective)	SUPREMETECH	1	B09SBX9JS2	$89.99
21.5" IPS Monitor, 350+ cd/m²	Samsung	1	S22D304	$119.99

Thin Film Foot Pressure Sensor / FSR Pad	Garosa	1	FSR-PAD	$9.99
Raspberry Pi USB-C Power Supply (5V 3A)	Raspberry Pi	1	SC0218	$8.00
AMS1117-3.3V DC-DC Linear Voltage Regulator (10-pack)	SUNKEE	1	AMS1117-3.3	$5.99
Momentary Push Buttons 12mm (5-pack)	Generic	1	PBS-24B	$5.00
Lumber & hardware for wooden frame	Home Depot	1	N/A	$35.00
				$414.87

Table 6: Parts

5. Conclusion

5.1 Accomplishments

The final project successfully produced a smart mirror that fulfills all core design requirements, specifically maintaining 70% reflectivity for standard use while ensuring high-contrast digital legibility. We achieved our primary technical benchmarks by engineering a software pipeline that completes the full user interaction cycle—from image capture to data display—in under 20 seconds. However the average accuracy is a bit low due to the variants like the lightness, the hairstyle of users, whether they are wearing the glasses. By effectively integrating the custom power regulation circuits and the Raspberry Pi unit within the wooden frame, we have delivered a stable, responsive system that balances physical durability with personalized digital utility.

5.2 Uncertainties

While the system successfully demonstrates the core design concepts, two primary uncertainties remain regarding data and hardware. First, the facial recognition model utilized a limited dataset of 100 entries, which naturally impacted overall precision. However, the system consistently matched primary users and faculty—even when comparing against reference photos taken decades ago—proving that the underlying recognition logic is robust. Second, the physical interaction layer is affected by the age of the components; the integrated buttons, having been in use for years, exhibit inconsistent debouncing. While the ESP32 firmware handles the logic effectively, the physical wear requires a firmer press than modern hardware, suggesting that future iterations would benefit from newer tactile switches to ensure a more responsive user interface.

5.3 Ethical considerations

The primary ethical considerations for this project center on user privacy and algorithmic fairness. To uphold IEEE Code of Ethics Section I.1, the system prioritizes the "safety, health, and welfare of the public" by ensuring that all facial biometric data is processed in real-time and purged immediately after each interaction, preventing the creation of a persistent surveillance database. Additionally, we address the risk of "automated prejudice" by curating a diverse matching database that represents various ethnicities, genders, and backgrounds in STEM. This commitment to inclusivity ensures that the mirror serves as an empowering tool rather than reinforcing existing biases, aligning with ACM Principle 1.4 to be fair and take action not to discriminate.

5.4 Future work

References

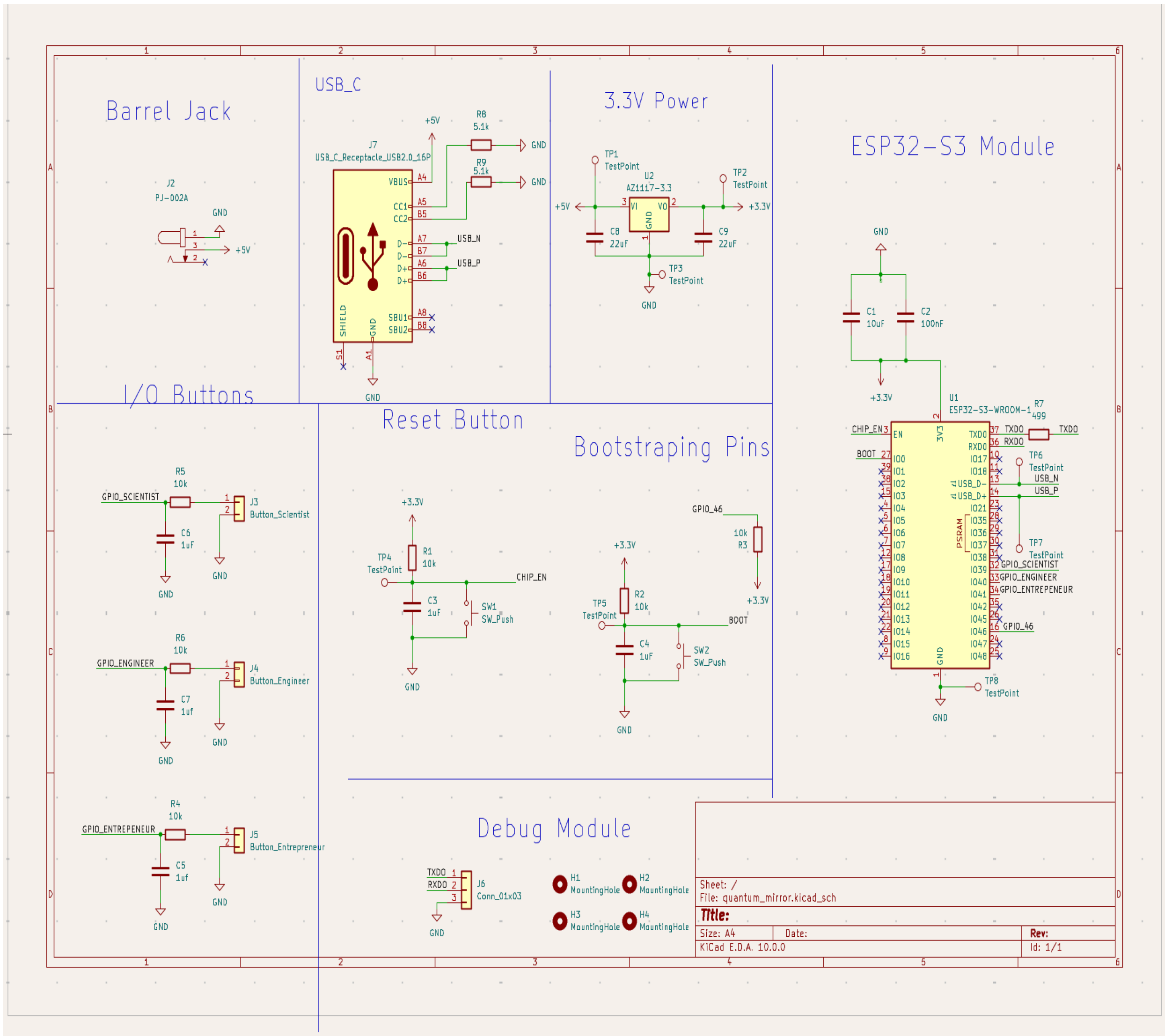
[1] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "ArcFace: Additive Angular Margin Loss for Deep Face Recognition," in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 4690–4699.

[2] Espressif Systems, "ESP32-S3 Series Datasheet," Version 2.2, 2024. [Online]. Available: https://www.espressif.com/sites/default/files/documentation/esp32-s3_datasheet_en.pdf

[3] H. V. Nguyen and L. Bai, "Cosine Similarity Metric Learning for Face Verification," in *Computer Vision – ACCV 2010*, Lecture Notes in Computer Science, vol. 6493, Springer, 2011, pp. 709–720.

[4] Espressif Systems, "ESP32-S3 Technical Reference Manual," 2024. [Online]. Available: https://www.espressif.com/sites/default/files/documentation/esp32-s3_technical_reference_manual_en.pdf

Appendix A Schematic



Appendix B: PCB