ECE 445

Design Document

Fall 2025

RFID Poker Table

Team #18

Satyam Singh (satyams2)
Darren Liao (darrenl4)
Khuselbayar Bolor-Erdene (kb40)

Professor: Rakesh Kumar

TA: Eric Tang

Oct 12, 2025

Contents

1 Introduction

1.1 Problem

Poker is a popular card game that blends elements of chance, psychology, and strategy, where players wager on the strength of their hands in an effort to win chips or money from opponents. At its core, poker is both a competitive and social activity, often played in casual home settings among friends or in professional tournaments. The game's appeal lies in its mix of skill, probability, and human interaction, qualities that make it engaging yet occasionally prone to confusion or disputes.

In friendly or informal games, casual players frequently encounter issues such as inconsistent rule enforcement, unclear betting procedures, or disagreements about the order of play. Misunderstandings often arise from players' differing interpretations of hand rankings, blinds, or side pots, especially when no designated dealer or rule adjudicator is present. Studies on recreational gaming have noted that these informal contexts can lead to friction or accidental rule violations due to variations in experience levels and communication styles [1],[2].

Live poker has a unique social and tactile appeal, but it also comes with challenges that online poker has largely solved. In casual games, players and dealers must manually track cards, blinds, and betting order, which often leads to mistakes such as misdeals, misreads, or losing track of the action. Unlike online platforms, live games lack conveniences like automatic hand evaluation, win probability calculations, and game state visualization. As a result, live poker can feel slower, less precise, and less accessible, especially for new players who may struggle to keep up with the rules and flow of the game.

1.2 Solution

To address the challenges faced by casual poker players, we propose a smart poker table that integrates RFID technology with a companion app to track the game state in real time. The table will automatically detect each player's hole cards and the community cards, reducing errors and eliminating the need for players to constantly monitor the game manually. A companion user interface for each player will display key information, such as their own cards, the community cards, whose turn it is, blinds and dealer positions, and fold status. The table will also track chips in play using computer vision, which can additionally detect common player actions, such as tapping to indicate a check or folding when cards are thrown into the pot. By providing these visual cues and real-time game updates, the system brings the benefits of online poker into the live experience.

In addition to the core MVP, several stretch goals could further enhance the live poker experience. LED indicators embedded in the table could visually highlight the dealer position, blinds, active turns, and folded players, giving players intuitive cues without constantly checking the app. A GTO-based move suggester could provide strategic recommendations, helping casual players make better decisions and learn optimal play patterns. Recording action history and hand logs would allow players to review previous hands for analysis or learning purposes. Finally, a spectator mode could present all hole cards and game probabilities in a view-only interface, enhancing the social and entertainment aspects of live poker.

1.3 Visual Aid

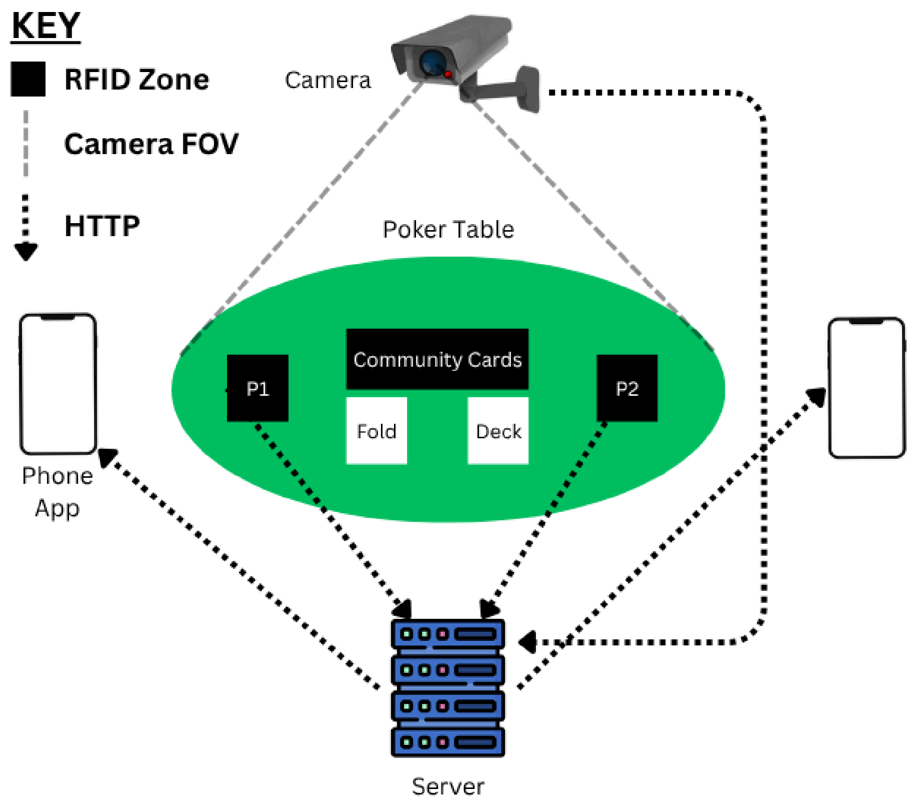

Figure 1. Overview of block diagram of RFID poker board system

The following is a brief overview of the system shown in Figure 1. RFID antennas embedded in the poker table will connect to a central microcontroller that scans cards in real time and communicates updates to a central server over the HTTP protocol. For the sake of the demo, this will be a single laptop running the core game server. The fold and deck piles are for demonstration purposes and can be placed anywhere the players see fit, as long as they are outside the RFID zones. The server also receives a feed from a camera mounted above the table, which monitors the game. The server will have all the information it needs to keep track of the game state and will compile this information in the best format for the players' apps. The server runs a computer vision model to interpret the camera feed and also runs a Monte Carlo simulation to predict winning odds for each player, based on the information they are allowed to know. Once all this information has been processed and formatted, it will be sent to the players' phone apps. This loop will continue until the end of the game, or until the board is unplugged or reset.

Figure 2. Example of a UI that can be displayed on the players' phones

Our system is designed specifically for Texas Hold'em, following standard rules. The MVP focuses on core functionality: RFID scanning for players' and community cards, a cross-platform user interface (UI) for each player to display the current game state, automated tracking of dealer position, blinds, and turn order, and computer vision to monitor player actions and chips. Figure 2 above shows what the UI could look like for a player on their phone. It is important to note that the final implementation will be slightly different from the image above. A player will not be able to see other players' cards unless the view is in broadcast mode. However, key information, such as community cards, their own cards, bet sizes, remaining balance, and turn order, will be displayed. Additional information, like the player's winning probability, will also be shown to help the player make better decisions.

1.4 High-Level Requirements

To consider our project successful, it must meet the following requirements:

- The system must accurately detect and track each player’s hole cards and the community cards with at least 95% reliability during a standard Texas Hold’em hand.
- The companion app must update the game state within a second of any change on the table.
- The computer vision can accurately detect player chips with an accuracy of over 90% (this could be further improved by refining the process or requiring players not to stack their chips).
- Player movements that indicate actions, such as folding or checking, are detected over 90% of the time and correctly update the game state on the companion app.

2 Design

2.1 Physical Design

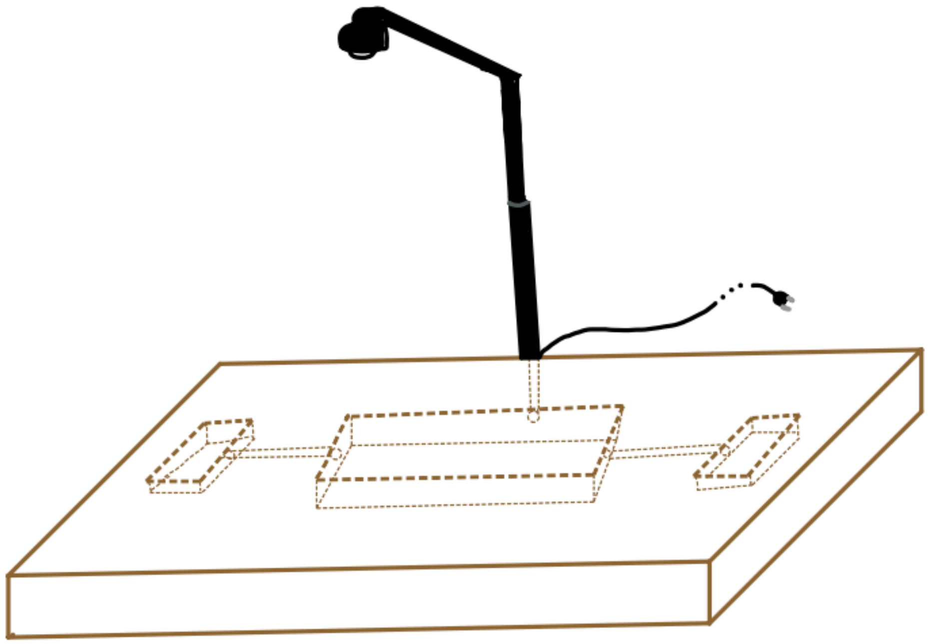

Figure 3. Overview of the physical design of the table

The physical design of the board is relatively simple, as shown above in Figure 3. It consists mostly of a single rectangular block of wood with indents to house the electronics for both the RFID readers and the community cards, along with extra space in the middle for other components, such as the central PCB with the ESP32. Small holes are drilled through the indents to allow intermediate wiring between the RFID antennas and the central PCB for power and communication. If time allows, we also plan on laying an additional layer of felt on top purely for aesthetic purposes. Research has shown that felt won't interfere with RFID signals and should still work as intended [3]

The additional physical component is the stand, which positions the camera at an angle above the board so that it can ideally view the entire table within its field of view. The hope is that this stand will be adjustable, although this will be determined by the machine shop when they follow up on the request. The final component is the power cable, which is attached at the same end as the camera stand so that power can be efficiently distributed to all components of the board.

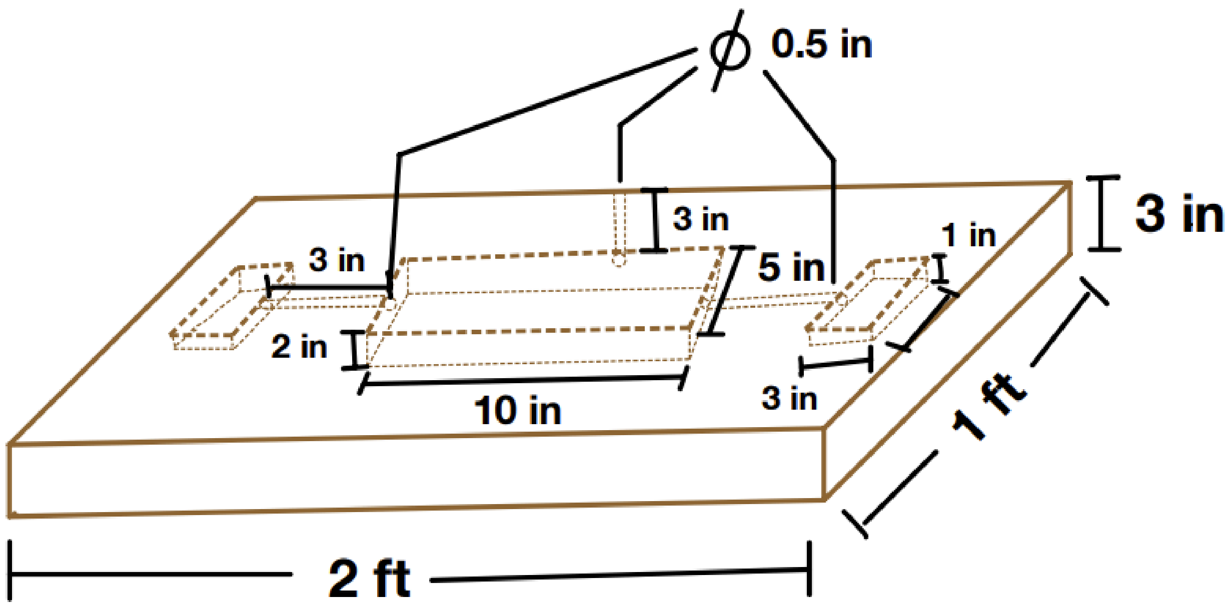

Figure 4. Dimension breakdown of the poker table

A real poker table would have to be much larger than the dimensions shown in Figure 4 above, but for the sake of proof of concept, cost savings, and demonstrating the capability of our prototype, we used two layers on a relatively smaller table. We designed it so that more players can be added easily by adding more indents and holes, and connecting them to our multiplexer on the primary PCB.

The dashed lines in Figure 4 outline the size and shape of the indents—holes in the table used to store the electronics—so that the final result resembles a traditional poker table and the electronics do not interfere with its natural feel. The holes between the indents are large enough for wires to be routed for both power and communication, ensuring the RFID antennas are properly connected.

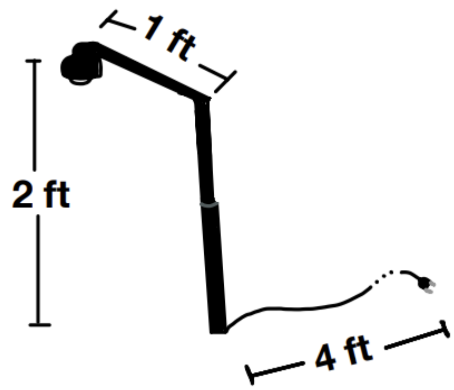

Figure 5. Dimensions breakdown of the camera stand and power cable

Figure 6. Example of height adjustable stand implementation

In order for the camera's field of view to cover the entire table, it must be placed at a bird's-eye view. A top-down perspective also helps reduce errors that could result from placing the camera at an angle, ensuring it can more easily distinguish chips and player movements and accurately recognize them. In Figure 5, the camera is placed at a height of 2 ft and can be positioned over the middle of the table with an additional arm that extends 1 ft toward the center. However, the goal is to make this stand adjustable. This could be implemented using a series of pins and holes extending from the base of the camera stand, similar to Figure 6 above. Ultimately, the final design will depend on the machine shop's ability to implement these adjustments, though a fixed-height stand would also be completely viable for our system.

2.2 Block Diagram

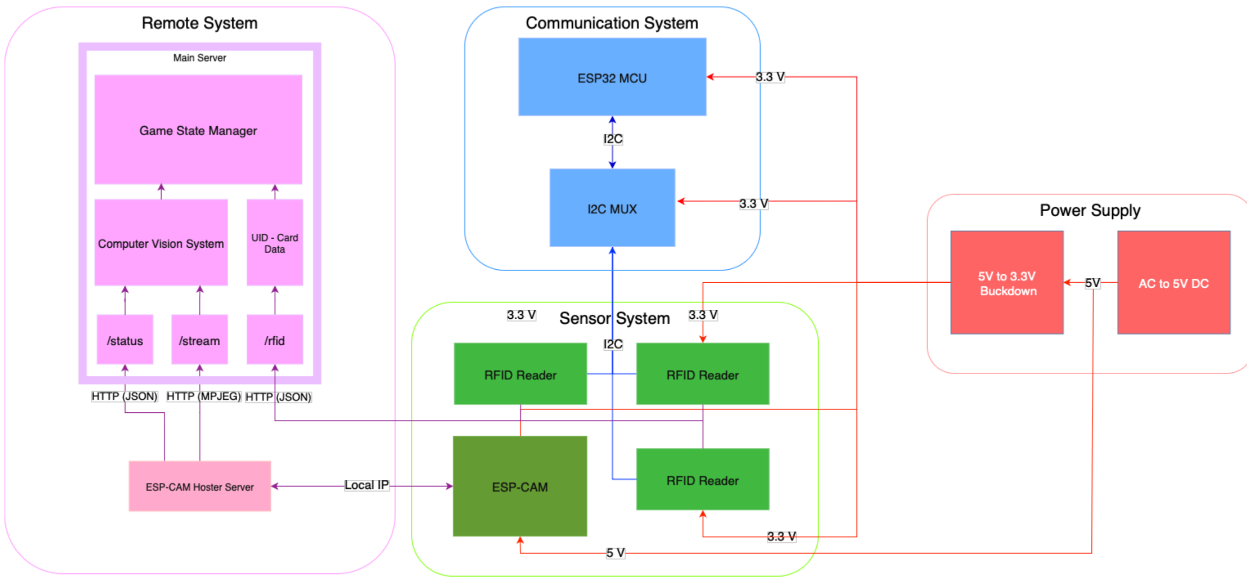

Figure 7. Block diagram of the RFID poker table

2.3 Subsystem Overview & Block Diagram Requirements

Our design is divided into four primary subsystems: the power system, the communication system, the sensor system, and the remote system. Each subsystem performs a specialized role but is closely interconnected to achieve seamless detection of playing cards and real-time display of game information. The power system ensures stable operation across all electronics, the communication system manages control and data flow, the sensor system detects and identifies cards using RFID, and the remote system provides the user-facing game interface. Together, these subsystems form a complete design that converts physical card placement into a digital game state accessible to players and dealers.

2.3.1 Power System

The power system supplies stable voltages to all other subsystems. AC to DC 5V wall plug acts as the main energy source. The 5 V output powers the ESP-CAM and is also stepped down through 3.3 V via a dedicated regulator to power the ESP32, multiplexer, and RFID reader (PN532) modules. By distributing the correct voltage rails, the power system ensures reliable operation and protects sensitive electronics across the table.

Contribution: Without regulated and stable voltage, RFID readers, camera, I2C MUX or the ESP32 would fail, preventing game state detection.

Interfaces:

- Provides 5.0 V ± 0.05 V at ≥250 mA to ESP32-CAM
- Provides 3.3 V ± 0.05 V at ≥250 mA for ESP32, RFID readers (PN532), I2C MUX

Table 1: Power System Subsystem – Requirements & Verification

Requirements	**Verification**
Must continuously supply ≥300 mA at 3.3 V ± 0.05 V to ESP32, RFID Readers, and MUX	1. Power the board with a wall adapter or bench supply at nominal VIN. 2. Connect electronic load to 3.3 V rail at the load sense pads. 3. Set load to 0 mA; record VOUT (should be 3.30 V ±0.05 V ⇒ **3.25–3.35 V**). 4. Sweep ILOAD: 50 mA → 100 mA → 200 mA → **300 mA** → 350 mA (margin). 5. At each point, record VOUT and ripple. **Pass if:** VOUT stays **3.25–3.35 V** up to **≥300 mA**; no oscillation;

Must continuously supply ≥300 mA at 5 V ± 0.05 V to ESP-CAM	1. Power the board with a wall adapter or bench supply at nominal VIN. 2. Connect electronic load to 5 V rail at the load sense pads. 3. Set load to 0 mA; record VOUT (should be 5.0 V ±0.05 V ⇒ **4.95–5.05 V**). 4. Sweep ILOAD: 50 mA → 100 mA → 200 mA → **300 mA** → 350 mA (margin). 5. At each point, record VOUT and ripple. **Pass if:** VOUT stays **4.95–5.05** up to **≥300 mA**; no oscillation;
Must operate with no time limit	1. Assemble the complete system: 3.3 V rail powering ESP32 + MUX + readers; 5 V rail powering ESP-CAM; all firmware running worst-case traffic simultaneously. 2. Start continuous operation: readers polling, ESP32 Wi-Fi active (bi-directional traffic), camera streaming. 3. Run **24 hours** minimum (longer is better). Log every 30–60 min: ● VIN, 3.3 V, 5 V ● Total input current ● Board hotspot temps ● Any watchdog resets, Wi-Fi disconnects, PN532 read errors. **Pass if:** No resets, no comms dropouts, rails stay within their specified bands the entire time.

2.3.2 Communication System

The communication system serves as the control and coordination hub. It is built around the ESP32 MCU, which manages Wi-Fi connectivity and system logic, and the TCA9548A I^2C multiplexer, which enables multiple RFID readers to share the same I^2C bus without address conflicts. The ESP32 communicates with the remote system over HTTP, sending detected card data and game states.

Within the local system, it handles I^2C commands to select and read from specific PN532 modules. This subsystem bridges the sensor data with the remote interface while managing all logic flow.

The communication system is built around the ESP32 and I^2C multiplexer. It processes sensor inputs, runs game logic, and sends updates wirelessly to the remote system.

Contribution: Provides the logic backbone for detecting and distributing game state.

Interfaces:

- Reads card IDs from the sensor system via I^2C (3.3 V logic, 100–400 kHz clock).
- Sends JSON packets to the remote system via Wi-Fi (2.4 GHz, latency ≤200 ms).
- Controls RFID readers via the multiplexer.

Table 2: Communication System Subsystem – Requirements & Verification

Requirements	**Verification**
Must process RFID reads and broadcast updates within ≤1 second	1. Put a scope probe on a GPIO that toggles high on PN532 detection and low when the network send completes. 2. Sniff network (PC) and record packet arrival times. 3. Measure GPIO pulse width (processing time) and GPIO rising edge → packet arrival (end-to-end). **Pass if:** All measured end-to-end intervals ≤ 1 s over 25+ trials.
Must support simultaneous polling of up to 3 RFID readers	1. Connect 3 PN532 to distinct MUX channels, place three different tags 2. Run firmware that round-robins channels with minimal inter-channel gap and no per-channel blocking (use per-channel timeouts ≤ 10 ms). 3. Log detections (UID + channel) for 2 minutes.

	Pass if: All three channels report stable, continuous presence with correct channel attribution; no channel starves for >200 ms.
Must maintain a Wi-Fi link at ≥90% uptime during operation	1. Enable camera streaming and 3-reader polling simultaneously. 2. Run the same 2-hour health checks. **Pass if:** Uptime still **≥ 90%** with the load; no watchdog resets.

2.3.3 Sensor System

The sensor system consists of PN532 RFID readers paired with antennas, positioned under the table at each player's position and the community card area and an ESP32-CAM camera module. The readers detect HF (13.56 MHz) RFID tags embedded in the playing cards, identifying which cards are in play. Data from the readers is transmitted over I^2C into the TCA9548A MUX, which routes it to the ESP32 for processing. While the camera module hosts its own server to provide continuous video streaming footage to our main remote server. The ESP32-CAM server maintains two ports 80 and 81, one for status/capture and the other fully reserved for streaming video, on its local IP address. The sensor system provides the raw game state input, directly translating the physical playing card data into digital data for the rest of the system.

Contribution: Converts physical card placement into digital input for the system.

Interfaces:

- Communicates card IDs to the communication system via I^2C. Requires 3.3 V ± 0.1 V power supply. Antennas detect RFID tags embedded in playing cards at distances of 2–4 cm.
- Streams the video stream to the main server over HTTP

Table 3: Sensor System Subsystem – Requirements & Verification

Requirements	**Verification**
RFID Reader must detect standard HF RFID tags (13.56 MHz) with ≥95% reliability	1. Connect 1 PN532 module to ESP32. 2. Place one 13.56 MHz tag directly above the antenna at the designed mounting distance 3. Disable Wi-Fi and other readers to isolate the test. 4. Present the tag for 2 s, remove it for 2 s. 5. Repeat for **50 cycles**. 6. Log every detection/removal event on ESP32 (UID detected, timestamp). **Pass if:** ● At least **48 detections** out of 100 cycles succeed. ● No false positives during “no tag” intervals.
RFID Reader must detect a card placement/removal within ≤1 seconds	1. Use firmware timestamps (*t_detect_start* when PN532 read loop sees new UID; *t_publish* when message sent to network). 2. On PC, log message arrival time *t_rx*. 3. Perform 50 card placements and 50 removals spaced by ~2 s. 4. Compute *t_rx* − *t_placement* and *t_rx* − *t_removal* using synchronized clocks (ESP32 NTP + PC). **Pass if:** ● **100%** of events detected and broadcast ≤ 1.0 s after physical movement. ● Median latency ≤ 300 ms.

ESP-CAM must stream realtime video with at least 24fps	1. Connect ESP-CAM to Wi-Fi 2. Start MJPEG stream 3. Mark time at each JPEG boundaries 4. Serial info log the time difference Repeat 3 times under: ● Idle (no RFID polling) ● Full load (RFID polling + Wi-Fi traffic) ● 10 min continuous operation. **Pass if:** Average FPS ≥ 24; frame drop < 10% per 30 s window.
ESP-CAM must stay connected to network	1. Start video stream and heartbeat ping (1 Hz) from PC to ESP-CAM for **2 hours**. 2. Log ping success/failure and response time. 3. Count disconnects or stream dropouts > 2 s. **Pass if:** ● Connection uptime ≥ 90% (consistent with earlier Wi-Fi spec). ● No single outage > 10 s without auto-reconnect. ● Reconnect latency ≤ 10 s after forced drop.

2.3.4 Remote System

The remote system is responsible for user interaction and game visualization. It consists of a mobile user interface that displays player hands, community cards, and card odds, player chip stack and pot chip stack all in real time. It is a fully software backend that:

- ingests RFID events from the Communication/Sensor systems,
- ingests video stream from Sensor system,
- looks up UID data from look card lookup table,
- runs computer vision program to calculate chip stacks,

- maintains authoritative game state,
- exposes REST endpoints that the mobile UI uses to read state and control the camera.

Function. Hosts the REST API and stream proxy. Implements

- */rfid*,
- */camera/stream*
- */camera/capture*
- */camera/status*.

Implementation. Python/Flask app with camera proxy helpers; HTTP endpoints and their handlers, and the video streaming routes from ESP32-CAM and RFID data and address from ESP32. With this data it maps RFID UID to specific cards from the lookup table, runs computer vision analysis on video footage to calculate chip size of each player and updates the game state on the server. The index page embeds the ESP32-CAM stream and provides quick actions .

External interface.

- /rfid (HTTP POST, JSON):
    - Returns JSON:

      { "reader": <int>,

      "value": "<hex address> # <hex UID>" } →

      { "status":"ok", "received_value":"<hex>" } (200 on success).

- /camera/stream (HTTP GET):
    - Returns binary MJPEG HTTPS Response

"received_value":"<JPEG frame data><boundary><JPEG frame data><boundary><JPEG frame data> " (200 on success).

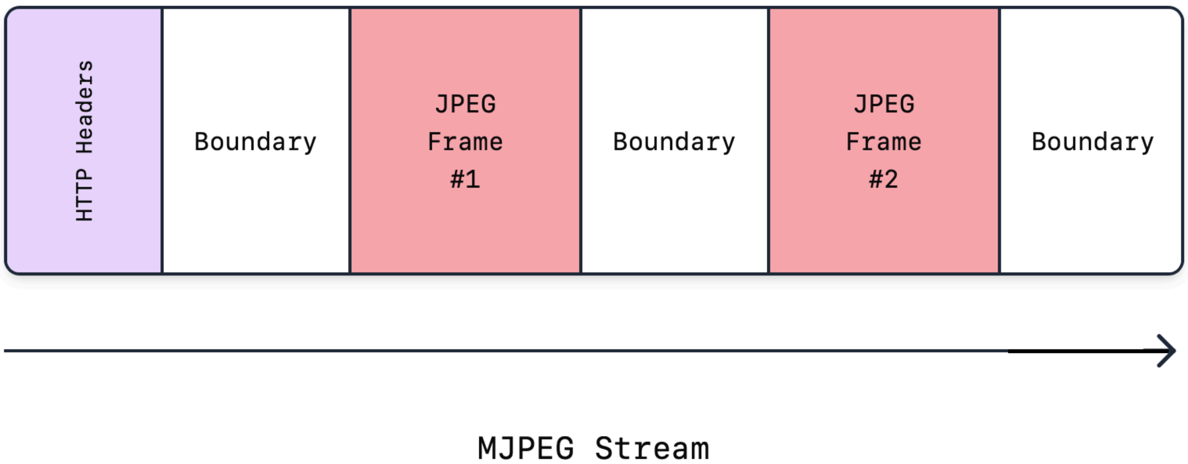

Figure 8. Data Layout of the MJPEG Stream

- /camera/capture (HTTP GET):
    - Returns a single JPEG frame (image/jpeg).

      { "status":"ok", "received_value":"<JPEG frame data>" } (200 on success).
- /camera/status (HTTP GET):
    - returns camera sensor/status JSON (proxy to http://<cam>/status) .

Table 4: Remote System Subsystem – Requirements & Verification

Requirements	**Verification**
Must display updated game state within ≤1 seconds of change	1. Insert or remove a tagged card, and mark the physical action time with a pushbutton tied to ESP32 GPIO (firmware logs timestamp).

	2. Insert or remove a chip from the pot, and mark the physical action time with a pushbutton tied to another ESP32 GPIO (firmware logs timestamp). 3. The ESP32 publishes an update message with an embedded timestamp. 4. The client app logs the timestamp when the UI element changes. 5. Collect 100 + samples in various conditions (idle network, camera streaming, full table). **Pass if** ● 100 % of updates ≤ 1.0 s. ● No missed or duplicated state updates including cards and stack sizes
Must run on both iOS and Android devices	1. Test on at least: ○ iPhone 12/13/14 or newer (iOS 16+) ○ Pixel 5/6 or equivalent Android 12+ 2. Perform a **full functional run**: ○ Login/authentication ○ Game lobby join ○ Card display updates ○ Betting actions (if applicable) ○ Video or camera view (if implemented) 3. Compare UI layout, scaling, and performance (FPS, input latency). **Pass if** ● All features behave identically and correctly on both platforms. ● No critical UI errors, crashes, or missing assets. ● Startup time ≤ 3 s on either platform.
Must enforce access control so players only see their own hole cards	1. Deal two cards to each player; verify backend associates correct UID with each player. 2. On every client, check visible cards:

	○ Player's own two hole cards visible. ○ All others' hole cards are hidden or back-faced. ○ Community cards visible to all. 3. Perform "role switch" tests: ○ Log in as different user → UI updates to show only that user's cards. **Pass if** ● Each user sees only their own hole cards and shared cards.

2.4 Tolerance Analysis

2.4.1 Stacked RFID Detection

One critical challenge for the RFID Poker Table is reliable detection of multiple cards placed on a single antenna, as each player must hold two hole cards. RFID readers like the PN532 can sometimes fail to distinguish between closely stacked tags due to electromagnetic coupling or orientation misalignment.

The HF RFID protocol (ISO 14443A) supports anti-collision mechanisms, but in practice, simultaneous reads can be inconsistent. For example, if one PN532 antenna can reliably detect a single card at 95% accuracy, stacking two cards reduces detection reliability to ~85–90% without careful tuning. To ensure feasibility, we plan to:

- Use card-grade thin RFID inlays with tuned antenna orientations to minimize coupling.
- Optimize PN532 polling frequency (~30 ms per cycle) to allow multiple read attempts within the 2-second requirement.

- Run tests on stacked-tag detection and adjust antenna placement (e.g., small lateral offset in the fold zone) to improve accuracy.

Mathematically, the probability of failing to detect both cards in a 2-second polling window can be modeled as:

$$P_{fail} = (1 - R)^{N}$$

Where R is the per-cycle detection reliability (e.g., 0.90 for stacked tags) and N is the number of polling attempts in 2 seconds (≈66 at 30 ms per attempt). With R=0.90,

$$P_{fail} = (0.10)^{66} \approx 10^{-66}$$

which is effectively zero, demonstrating feasibility if rapid polling is implemented.

2.4.2 Power Delivery Analysis

One concern that we had was that since the camera was far away from the power source if the voltage and current drop over the distance would be important to consider . To determine whether a 3-foot wire is suitable for powering an ESP32-CAM module at 5 V and 500 mA, we first analyze the voltage drop along the wire. The wire's resistance directly impacts the voltage delivered to the device. Assuming the use of 22 AWG copper wire, which has a resistance of approximately 0.0161 Ω per foot, the total length of the current loop (forward and return paths) is 6 feet. The total resistance of the wire is therefore:

$$R_{total} = 6 \text{ ft} \times 0.0161\ \Omega/\text{ft} = 0.0966\ \Omega$$

Using Ohm's law, the voltage drop V_{drop} across the wire is given by:

$$V_{drop} = I \times R_{total}$$

Substituting the current I=0.5 A, we obtain:

$$V_{drop} = 0.5\ \text{A} \times 0.0966\ \Omega \approx 0.0483\ \text{V} \approx 48\ \text{mV}$$

The voltage at the ESP32-CAM module is therefore:

$$V_{ESP32} = 5\ \text{V} - 0.0483\ \text{V} \approx 4.95\ \text{V}$$

This voltage is well within the ESP32-CAM's operational range of 4.5–5.5 V, indicating that a 3-foot 22 AWG wire will not cause significant power loss or performance issues.

For longer runs or higher currents, the voltage drop can become more substantial. In such cases, several mitigation strategies can be employed. Using thicker wires (e.g., 20 AWG or 18 AWG) reduces resistance and consequently the voltage drop. Another approach is to transmit power at a higher voltage (e.g., 9–12 V) over the wire and then use a local buck converter to step down to 5 V near the ESP32-CAM module. Additionally, twisting the power and ground wires together can help reduce inductive effects in environments with switching currents.

In conclusion, for the current 3-foot, 0.5 A setup, the voltage drop is minimal, and no special mitigation is necessary. However, careful consideration of wire gauge or local voltage regulation is recommended for longer distances or higher power requirements.

3 Cost and Schedule

3.1 Cost Analysis

The cost of this project can be divided into labor, testing/prototyping materials, and final construction. The total cost of labor can be estimated using the average annual earnings of a Computer Engineering graduate from UIUC, which is approximately $118,752 per year, or $57.09 per hour [4]. Assuming each team member works an average of 6 hours per week for 14 weeks, the total labor cost for the engineers amounts to $14,386. However, in order to build the actual poker table an estimated 4 hours of work from the machine shop was quoted. With the average of a machine shop worker in Champaign, IL being around $26.00/hr the total for this work would be roughly $104, bringing the total labor cost to $14,409.68 [5].

During the process of testing and developing the RFID board, several development and breakout boards were used to expedite prototyping and demonstrate proof of concept. The costs of these boards and their respective prices are itemized in the following section.

Finally, all individual components used in the final version of the RFID board are listed below. Combining the total cost of development boards ($70.58), the final parts ($16.53), and labor ($14,386), the estimated total project cost is approximately $14,577.79.

Labor					
Name	*$/hr*	*Hours/week*	*Weeks worked*		*Total Cost*
Satyam Singh	$57.09	6	14		$4,795.56
Darren Liao	$57.09	6	14		$4,795.56
Khuselbayar Bolor-Erdene	$57.09	6	14		$4,795.56
Machine Shop	$26.00	4	1		$104.00
				Total	**$14,490.68**
Development					
Description	*Manufacturer*	*Part Number*	*Quantity*	*Unit Cost*	*Total Cost*
PN532 NFC / RFID reader	HiLetgo		3	$11.49	$34.47
ESP32-S3-DevKitC	Espressif		1	$15.00	$15.00
ESP32-CAM	Hosyond		1	$7.99	$7.99
TCA9548A I^{2}C multiplexer	Adafruit		1	$13.12	$13.12
				Total	**$70.58**
Parts					
Description	**Manufacturer**	**Part Number**	**Quantity**	**Unit Cost**	**Total Cost**
0.1 µF ±10% 50V Ceramic Capacitor	Yageo	CC0805KRX7R9BB104	2	$0.08	$0.16
22 µF ±20% 25V Ceramic Capacitor	Yageo	CL21A226MAQNNNE	2	$0.08	$0.16
1 µF ±10% 50V Ceramic Capacitor	Yageo	CL21B105KBFNNNE	1	$0.08	$0.08
0.1 µF ±10% 50V Ceramic Capacitor	Yageo	CC0805KRX7R9BB104	1	$0.08	$0.08
4mm 12V 3mm Round Button	XUNPU	TS-1088-AR02016	2	$0.42	$0.84
10K OHM 1% 1/4W 0805	Vishay Dale	CRCW080510K0FKEA	2	$0.10	$0.20
100K OHM 1% 1/4W 0805	Vishay Dale	CRCW0805100KFKEA	11	$0.10	$1.10
USB-C connector, 24P	GCT	USB4105-GF-A	1	$0.78	$0.78
5.11K OHM 1% 1/4W 0805	Vishay Dale	CRCW08055K11FKEA	2	$0.10	$0.20
ESP32-S3-WROOM-1U	Espressif	ESP32-S3-WROOM-1U-	1	$6.56	$6.56
Straight Header Male 8-pin	Sparkfun	(SparkFun 8-pin header)	4	$0.75	$3.00
IC REG LINEAR 3.3V 1A	UMW	AMS1117-3.3	1	$0.68	$0.68
10 µF Molded Tantalum 16V	KYOCERA AVX	TAJA106K016RNJ	1	$0.35	$0.35
22 µF 25V Tantalum	KYOCERA AVX	TCJB226M025R0150E	1	$1.01	$1.01
Conn Power RCP Through-Hole	Nebj	NEBJ21R	1	$0.71	$0.71
8-channel I^{2}C switch / MUX	Texas Instruments	TCA9548A	1	$0.62	$0.62
				Total	**$16.53**
				TOTAL	**$14,577.79**

Table 5: Cost analysis for ClassroomClarity production

3.2 Schedule

Week of	Task	Person
9/8	Research problem and poker rules to address	Darren Liao
	Research RFID and limitations	Satyam Singh
	Research power requirements	K.B. Erdene
9/15	Brainstorming plan to build	Everyone
	Research parts	K.B. Erdene
	Meet with TA to revise plan	Everyone
9/22	Order parts to make first demo	Satyam Singh
	Start PCB design with ESP32 chip	Darren Liao
	Proposal + feedback review	Everyone
9/29	Complete first iteration of main ESP32 board PCB design	Darren Liao
	Build breadboard for RFID detection, MUX, and server communication	Satyam Singh
	Getting initial PCB checked by TA + fixing issues	Everyone
10/6	Getting camera feed to server with breadboard setup	K.B. Erdene
	Adding MUX support and power to PCB	Darren Liao
	Working on design document	Everyone
	Breadboard demo #1	Everyone
10/13	Passing audit and ordering first PCB	Everyone
	Implementing computer vision with the camera to detect chips and players	K.B. Erdene

	Starting server visualization to better show cards and turn order	Satyam Singh
10/20	Testing PCB + addressing any issues	Darren Liao
	Simple server visualization done (can see turn + cards)	Satyam Singh
	Starting app with both android/iOS support	Satyam Singh + K.B. Erdene
10/27	Breadboard demo #2	Everyone
	Train computer vision on different chips and player movements	K.B. Erdene
	PCB #2 pass audit and finalize fixes	Darren Liao
	Camera feed and cards show in app (emulator)	Satyam Singh
11/3	Order PCB #2	Darren Liao
	Integrating computer vision logic with the card detection	Satyam Singh
	Get poker table from machine shop and start planning assembly	Everyone
	Test PCB for any new issues + address issues in final revision	Darren Liao
11/10	PCB #3 pass audit + order (if needed)	Darren Liao
	Work on completing the MVP on the app	Satyam Singh + K.B. Erdene
11/17	Final assembly of the poker table	Everyone
	Clean up app and prepare for final demo	Everyone
11/24	FALL BREAK	

12/1	Demo	Everyone
12/8	Final presentation	Everyone

Figure 8: RFID Poker Board Design Timeline by week

4 Ethics and Safety

The RFID Poker Table raises important ethical considerations, particularly regarding gambling. While the project is intended for recreational and educational purposes, there is a risk it could be misused in unregulated gambling environments. According to the IEEE Code of Ethics and ACM Code of Ethics, engineers must prioritize public welfare and ensure that technology does not promote harmful or illegal behavior [6], [7]. To address this concern, we will clearly document that the system is not designed for commercial or profit-driven gambling, but rather as a learning tool to enhance casual gameplay. Additionally, we will implement security safeguards, such as controlled access, to prevent users from gaining unfair advantages (e.g., viewing opponents' cards).

Beyond gambling-related concerns, our ethical responsibility also extends to fairness, transparency, and honesty in development. We will accurately report the limitations of our system, including potential RFID read inaccuracies, and avoid overstating the system's capabilities, such as chip tracking precision or potential AI-based strategy suggestions. Ensuring fair play is an ethical obligation; thus, the system's interface will be designed to show only information each player is entitled to see, preserving the integrity of the game. By aligning with IEEE ethical principles of honesty, fairness, and responsibility, we aim to foster trust and accountability in both the design and use of our system [6].

From a safety perspective, our project must comply with relevant electrical and campus laboratory safety standards. Since the system will now be powered through a wall outlet rather than lithium-ion batteries, our design will incorporate safe AC-to-DC conversion, overcurrent protection, and proper insulation of high-voltage components in accordance with UL and IEC safety standards

[8], [9]. This reduces risks associated with battery overheating or short-circuiting while introducing new considerations for electrical shock prevention and power supply reliability.

All RFID modules will operate within FCC Part 15 power limits, ensuring minimal electromagnetic exposure to users [5]. Prototyping and assembly activities, including soldering, wiring, and handling exposed electronics, will strictly adhere to campus lab safety protocols, such as using protective eyewear, anti-static wrist straps, and ventilated soldering areas. By proactively addressing both ethical and safety aspects, this project demonstrates a balance of innovation, responsibility, and compliance, ensuring a positive and safe experience for all users.

5 References

[1] K. Browne, "Home Poker and Informal Gambling: Social Play and Rule Negotiation," Journal of Leisure Studies, 2018.

[2] J. Palomäki, M. Laakasuo, and M. Salmela, "Expertise and Emotion in Poker Decision-Making," Journal of Gambling Studies, vol. 37, no. 2, pp. 457–474, 2021.

[3] Skadi, "Misconception 14: RFID Signals Can Pass Through Any Material," *RFID Label*, 11 months ago. [Online]. Available: https://www.rfidlabel.com/an-analyzing-guide-of-misconception-in-rfid-what-you-need-to-know/#:~:text=best%20read%20performance.-,Misconception%2014%20:%20RFID%20Signals%20Can%20Pass%20Through%20Any%20Material,different%20applications%20with%20careful%20planning. [Accessed: Oct. 12, 2025].

[4] University of Illinois Urbana-Champaign, "Computer Engineering," *The Grainger College of Engineering*, 2025. [Online]. Available: https://grainger.illinois.edu/academics/undergraduate/majors-and-minors/computer-engineering

[5] ZipRecruiter, "Machinist Salary in Champaign, IL: Hourly Rate," *ZipRecruiter*, 2025. [Online]. Available: https://www.ziprecruiter.com/Salaries/Machinist-Salary-in-Champaign,IL. [Accessed: Oct. 11, 2025]. ziprecruiter.com

[6] IEEE, "IEEE Code of Ethics," IEEE, 2020. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html

[7] ACM, "ACM Code of Ethics and Professional Conduct," Association for Computing Machinery, 2018. [Online]. Available: https://www.acm.org/code-of-ethics

[8] UL Standards, "UL 60950-1: Safety of Information Technology Equipment," UL Standards, 2014.

[9] IEC, "IEC 62368-1: Audio/Video, Information and Communication Technology Equipment – Safety Requirements," International Electrotechnical Commission, 2018.

[10] Federal Communications Commission, "Title 47 CFR Part 15 – Radio Frequency Devices," FCC, 2021.