

Team E8: Smart Poker Table Design Review

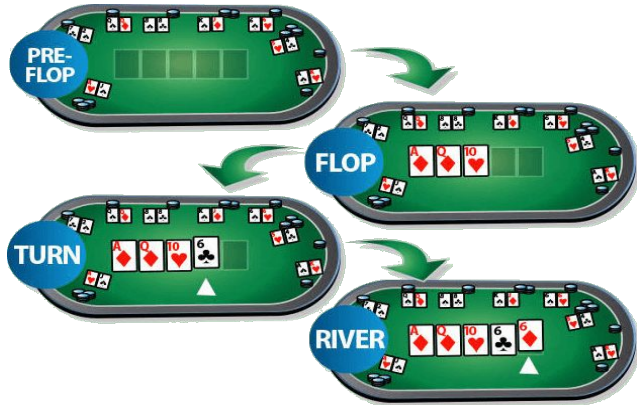
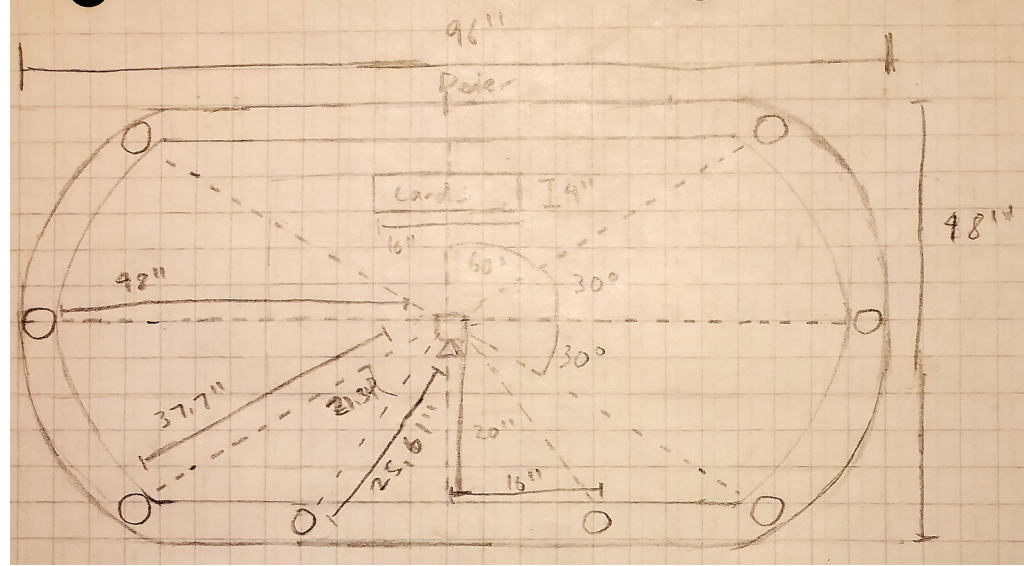
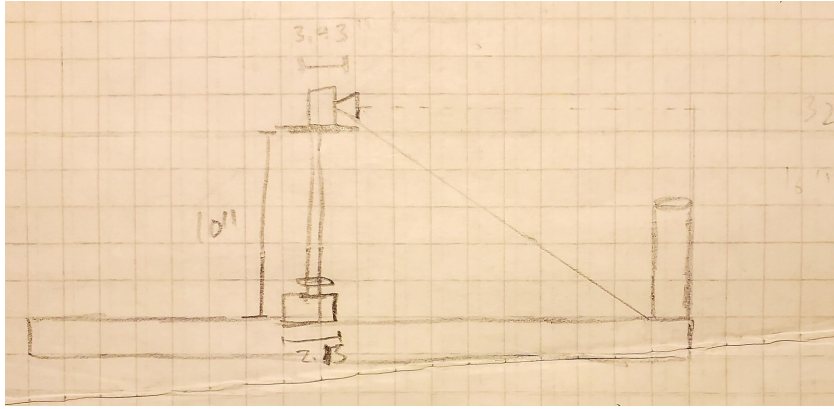
Zongpeng (Steve) Yu, Brandon Hung, Patrick Kollman

Application Area

- Display poker statistics (pot size and player stack sizes), that are available in online poker games, to real life poker players in a casino
- CV provides relatively robust and fast stack height/color reading
- Intuitive Software for dealer to keep track of the game state

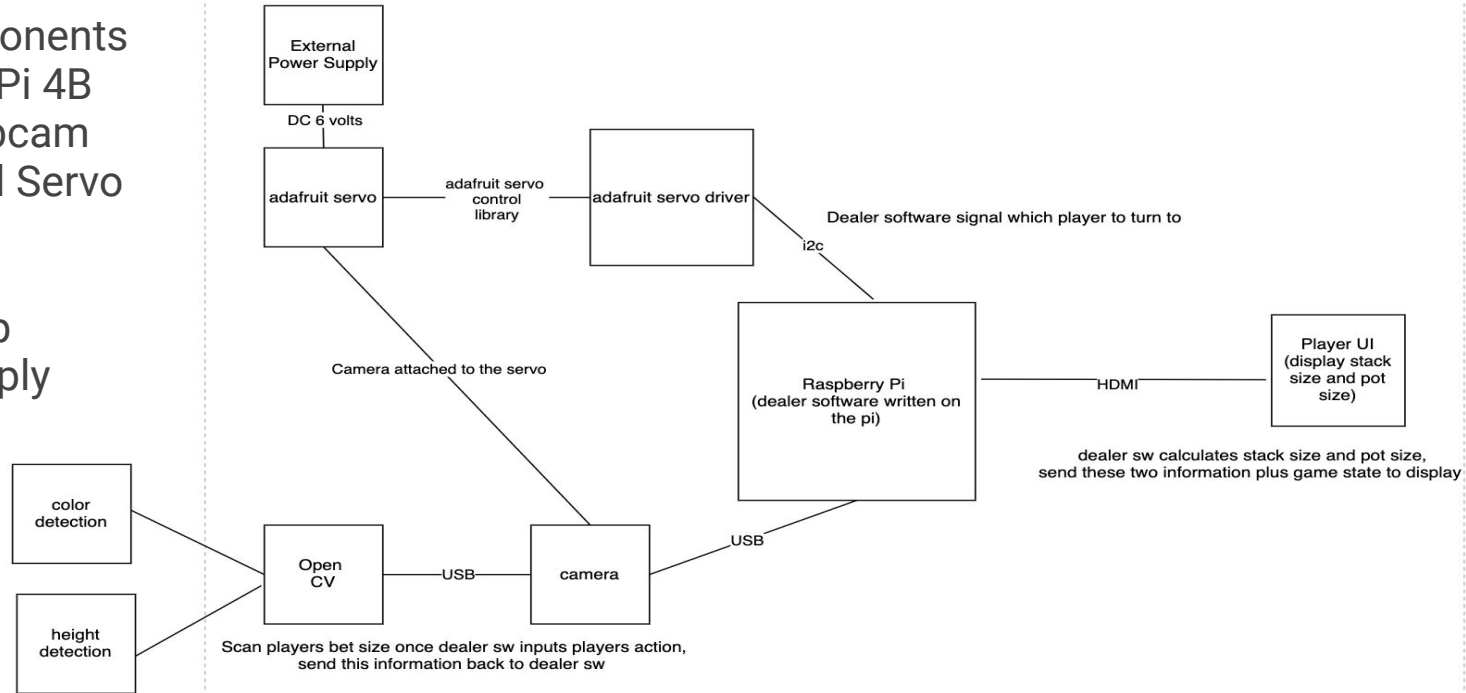


Solution Approach



Hardware, Systems, and Subsystems

- Hardware components
 - Raspberry Pi 4B
 - IFROO Webcam
 - 16 Channel Servo Driver
 - Servo
 - Power Strip
 - Power Supply Adaptor

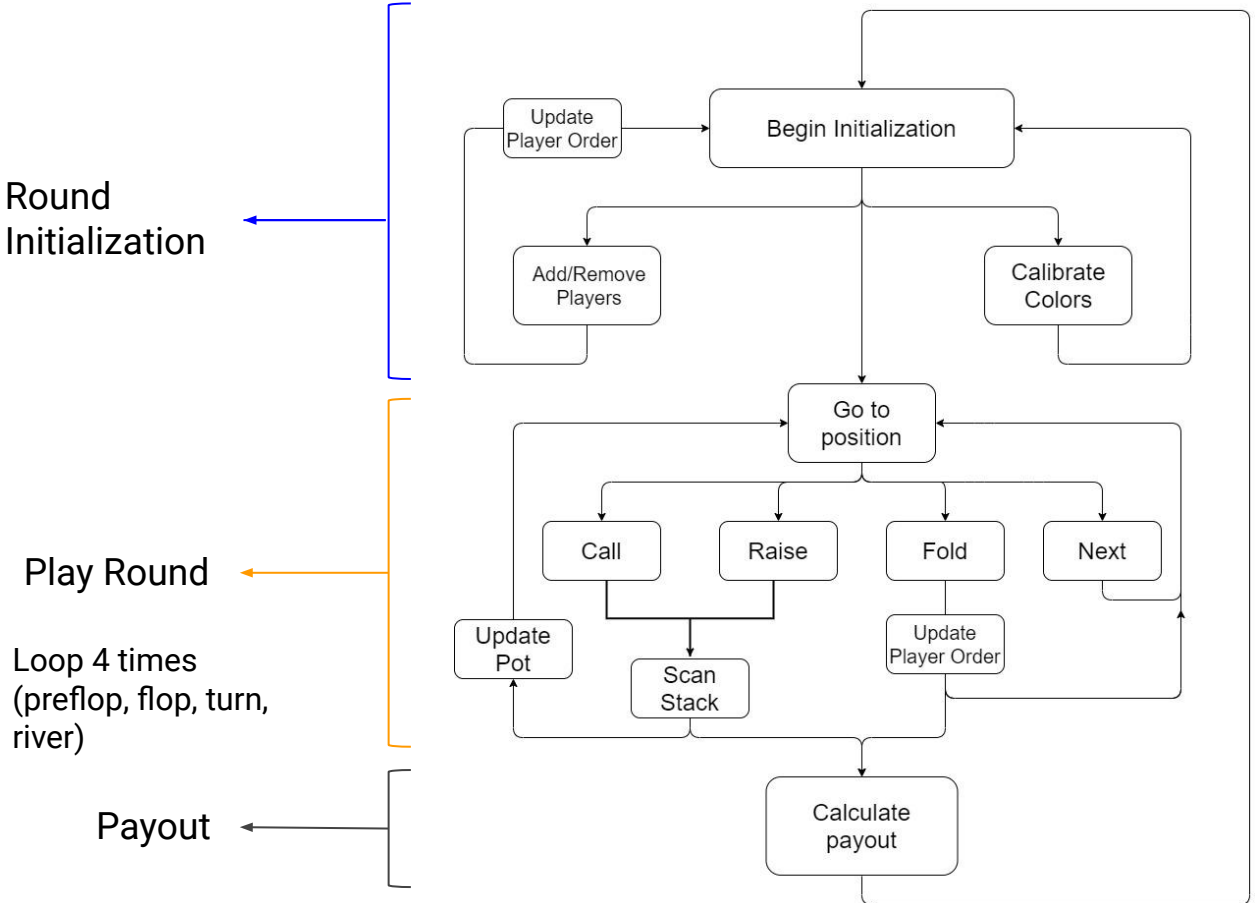


Implementation Plan

- Purchased
 - Hardware
- Designed
 - Servo-driven turntable, dealer/player UI, stack detection algorithm, game state representation algorithm
- Downloaded/Open-Source
 - Servo library, hardware communication protocols, OpenCV

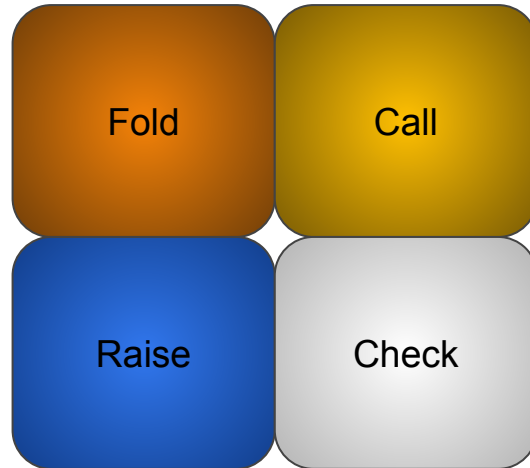
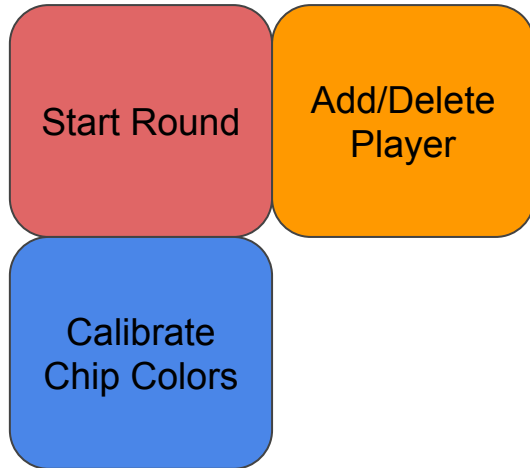


Implementation Plan for SW



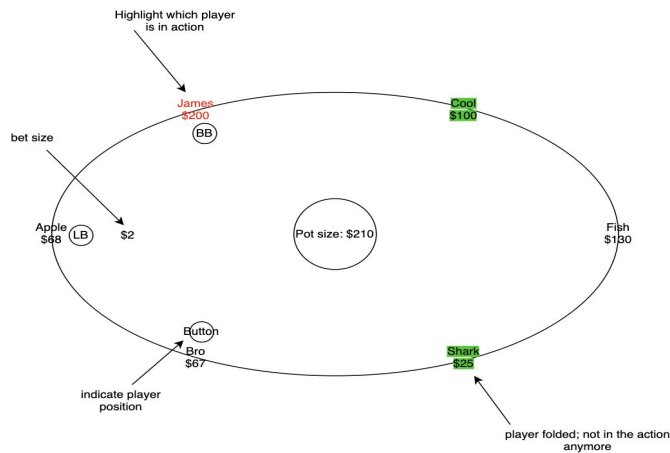
Dealer UI

- Simple button interface
- Round Start: add/remove players, calibrate chip colors, start round
- Play round: fold, call, raise, check
- Payout: N/A



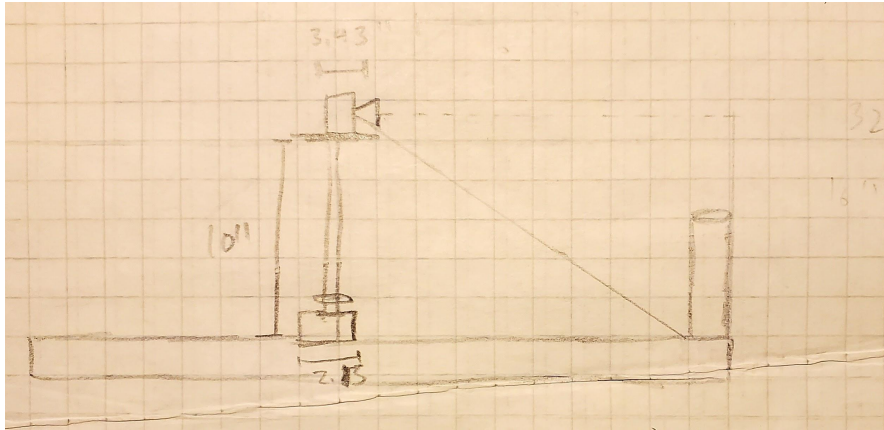
Player UI

- Monitor displays pot and stack sizes
- Modelled from online poker games
- Updates when players bet or are added/removed



State Tracker

- Round start: update player order, calibrate chip color
- Play round: go to position, scan stack/update pot, update player order
- Payout: reward winners with correct amount



Metrics and Validation

Component	Testing Strategy	Metrics	Challenges
Servo Motor	Input: unique ID assigned to each player Output: servo rotating to the specified position	Camera rotate within +/- 4.5 inches of target; camera should rotate to the position within 5 seconds	Motor provides enough torque; power supply
Computer Vision	Input: unoccluded picture of bet Output: total value of bet	Verify +/-10% of total stack value	Different distances/lighting can lead to detection challenges
Software	Input: Game state inputs entirely simulated in software Output: game state, stack sizes, and pot size update correctly in the simulation	100% accurate simulation required; the simulation should update correctly given the inputs it receives	Providing versatility for game initialization and accurately maintaining the game state
Overall System	Input: dealer software controls Output: servo position and updated game state, stack, and pot size to monitor display	After integration, the previous subsystem metrics hold and the display is updated within 5 seconds	Successful integration

Project Management

