

DISCORDGENIUS

Product Requirements Document: Texas Hold'em Poker Bot

A way to play poker entirely from within a Discord channel.

PRODUCT REQUIREMENTS DOCUMENT • GOLD STANDARD EXAMPLE

PREPARED FOR DISCORDGENIUS ENGINEERING

1. Executive Summary & Objective

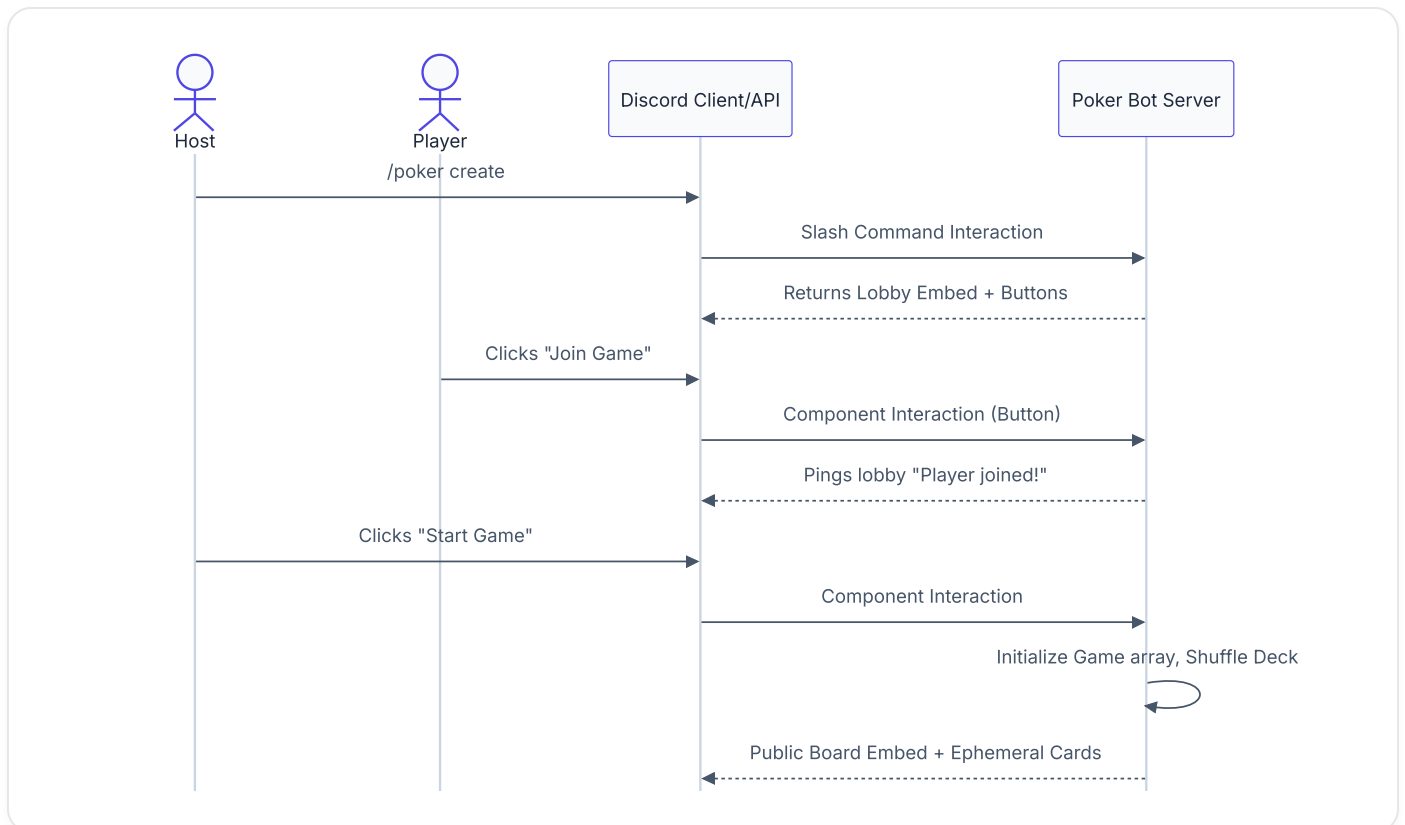
The Objective: To build a fully functional, multiplayer Texas Hold'em poker experience that lives 100% inside Discord channels.

We require a system where server members can host configurable tables, join games, and play out full hands (including betting, side-pots, and fold timers) without ever clicking an external link, logging into a web portal, or downloading an application. The Discord channel *is* the poker table.

2. Platform & Architecture Strategy

- **100% Native Architecture:** The entire user experience must rely solely on native Discord mechanics —Slash Commands, Ephemeral Messages, public Embeds, and Message Components.
- **No External Web Auth:** At no point should a player be directed to a browser to authenticate or play. The application must identify players purely by their Discord User IDs.
- **Isolated Game States:** The bot must cleanly handle multiple concurrent games running simultaneously in the same or different channels. Game state should be mapped strictly to the table host and their active Interaction IDs.

2.1 Architecture Interaction Diagram



3. Visual & Interface Requirements

- **Server-Side Canvas Cards:** All 52 cards in the deck must be fully rendered as high-quality images using a server-side Canvas engine. The bot must strictly utilize the **PNG-cards-1.3** open-source asset pack for all card face imagery.
- **The Public Board:** The community cards (Flop, Turn, River) must be displayed in a public channel embed. The image background should utilize a "felt-green" aesthetic to simulate a real casino table environment.
- **Dynamic Showdowns:** If a hand reaches showdown (or an all-in runout), the bot must generate a combined visual image showing the community cards at the top and the participating players' hole cards at the bottom, explicitly tagging the winner.
- **The Private Hand:** A player's hole cards (their 2 private cards) must *never* be Direct Messaged. They must be sent as an ephemeral image attachment directly in the channel via a `/poker status` command or a "Show My Cards" button.

4. User Experience (UX) & Core Game Loops

The Flow of Play

We want to eliminate the need for players to manually type out slash commands every single time it is their turn. The game embed must heavily leverage Discord's native UI components:



- **Action Buttons:** The active game embed must feature a row of Discord Buttons (Fold, Check, Call, All-In) color-coded for safety (e.g., Fold is Red/Danger).
- **Raise Dropdown:** To streamline betting, include a String Select Menu dropdown with quick-raise options (Min Raise, 1/2 Pot, Pot Size, 2x Pot).
- **The Premove System:** If a player clicks an action button when it is *not* their turn, the bot must queue that action as a "👤 **Premove**" and automatically execute it when the action reaches them, drastically speeding up the pace of play.

The Host Journey

1. **Creation:** A user types `/poker create`. They assume the role of "Table Host".
2. **Lobby Management:** The bot posts a public lobby embed announcing the new table.
3. **Initialization:** When enough players have joined, the host types `/poker start`. The bot shuffles the deck, issues ephemeral hole cards to all seated players, posts the starting public board, and formally pings the first player to act.

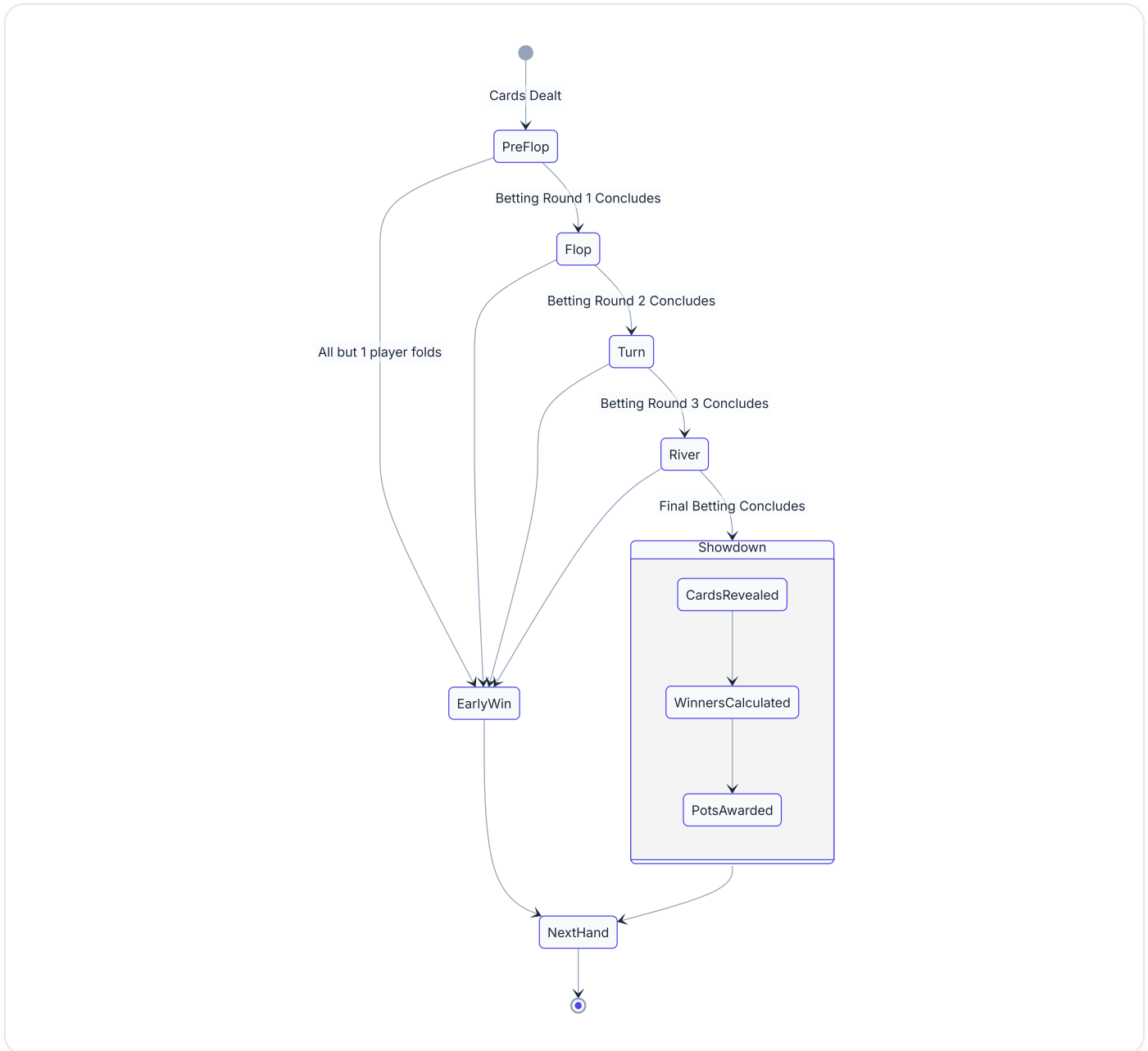
The Player Journey

1. **Joining:** A player clicks the green "Join Game" button on the lobby embed to take a seat.

- *Late Join Handling*: If a game has already started, the late-joining player is placed in a "folded" state and deals in on the subsequent hand seamlessly.
2. **Acting**: When it is a player's turn, the bot actively pings them. If they have less than 10 seconds remaining, the UI prominently flashes a  **HURRY!** warning. They select their action via the buttons or dropdown.
 - The public board instantly re-renders to reflect the action and the new pot size.
 3. **Checking Cards**: At any time, a player can click the " Show My Cards" utility button to privately re-view their hole cards ephemerally and check the math on their remaining chips vs. the current bet.

5. Game Mechanics & State Resolution

5.1 The Hand Lifecycle (State Machine)



- **The Chip Economy:** Players start with a configurable baseline amount of chips.
- **Blind Structures & Antes:** The game must automatically post Small and Big Blinds. The blind amounts must escalate automatically per round, based on configuration, to force action and prevent infinite games.
- **Side Pot Mathematics:** If three players go "all in", but one player has significantly fewer chips than the others, the engine must accurately calculate and separate the main pot from the side pots, ensuring players only win what their chip stacks covered.

6. Configuration Vectors

Table hosts must be able to configure the following parameters via their `/poker create` command (or rely on sensible defaults if omitted):

- `starting_chips` : The number of chips each player begins with (Default: 1000).
- `ante_increase_per_round` : The escalation rate of the blinds (Default: x1.5 increase every 3 hands).
- `turn_timer` : The number of seconds a player has to act before auto-folding (Default: 60s, Max: 300s).

7. Edge Cases & Exception Handling

- **Player Disconnects / AFK:** If a player walks away, the `turn_timer` will expire and the bot will cleanly auto-fold their hand. This is a vital constraint to ensure one AFK player does not freeze a game of 9 other active participants.
- **Mid-Game Quits:** If a player explicitly types `/poker quit` mid-hand, their current chips are forfeited to the main pot, their seat is cleared from the state, and the action skips to the next available player.
- **Insufficient Blind Covering:** If a player goes "all-in" because they cannot cover the mandatory Big Blind, the engine must smoothly force a side-pot creation on the opening pre-flop bet.