Poker Game Prototype — Unity

Overview

This project is a poker game prototype built in Unity with a focus on clean architecture and scalable system design.

The goal of the implementation is to demonstrate proper separation of gameplay logic, state management, and UI rather than production-level visuals.

The game includes:

- Full poker round flow (PreFlop → Flop → Turn → River → Showdown)
- Turn-based gameplay with timer
- Chip deduction and pot distribution
- Basic AI opponent
- Match restart without reloading the scene
- Event-driven UI updates

Architecture

The project follows a layered architecture.

Presentation

Handles UI rendering and player input.

Examples:

- SimpleGameUI / Table UI
- Card display
- Action buttons

UI listens to events and does not contain game logic.

Application Layer

Coordinates gameplay flow.

Main components:

- **PokerGameManager** — overall match orchestration
- **GameStateController** — round state machine
- **TurnManager** — turn sequencing
- **TurnTimerService** — turn timer
- **AIDecisionService** — AI actions

Core Domain

Contains all poker logic (independent from Unity).

Models

- GameSnapshot (single source of truth)
- Player
- Card
- PlayerAction

Services

- DeckService
- DealService
- BetService
- PotService
- HandEvaluator

Enums

- PokerRound
- ActionType
- HandRank

Infrastructure

Provides decoupled communication.

- EventManager (event bus)
- GameEvents
- GenericSingleton

Game Flow

1. Match starts → snapshot created
2. Cards dealt in PreFlop
3. Players take turns
4. Actions update pot
5. Betting round completes → state advances
6. Community cards dealt (Flop, Turn, River)
7. Showdown → winner determined
8. Pot distributed → match restarts

Both player and AI use the same action pipeline.

AI

AI decisions are based on hand strength from the evaluator.

The AI generates PlayerAction events just like a real player, keeping gameplay consistent and deterministic.

Design Principles

- Clear separation between UI, orchestration, and domain logic
- GameSnapshot acts as the authoritative state
- Event-driven communication reduces coupling
- State machine controls round flow
- Services encapsulate poker logic

- Architecture prepared for future multiplayer integration

How to Run

1. Open the project in Unity
2. Load the main scene
3. Press Play
4. Use buttons to play against the AI

Possible Improvements

- Full poker hand evaluation
- Hide AI cards until showdown
- Networking integration
- Advanced AI strategy
- UI polish

This prototype focuses on demonstrating scalable architecture and clean gameplay flow suitable for future multiplayer expansion.