

Quân Đỗ
S3800978

Assignment 2 Documentation

Roulette Game

Introduction	2
Motivation	2
Explanation / How to play	2
Features	2
Feature description	4
Screenshots	6
Video recording link	11

Introduction

This is a casino game about making bets and winning each roulette spin. Captivating millions, roulette has been a mainstay in many casinos across the world. This game let you play a simplified version, with nice jazz music, to get a taste of the real thing.

Motivation

Roulette is my favorite game in a casino. The spinning of the ball around the roulette table and the anticipation of hoping it lands on your chosen bet is very exciting. Plus, the game contains some amount of logic, where you pick how how you're willing to trade per bet, as risker bets nets a higher payout.

Explanation / How to play

- The game already picked a default bet amount for you, but feel free to chose any amount you like. If you pick an amount higher than what you have, the game will automatically chose the highest possible amount for you.
- Click on which pocket you think the roulette would stop on. Bet based on color (red, black), numbers (even, odds), or sections (1st 12, 2nd 12, 3rd 12). Betting on sections nets you a double payout.
- Continue the game for as long as you like. Chose "Finish game" to register your high score. Knowing when to stop makes you a real winner.
- If you run out of money, just head back and try again.
- There's no tricks to win this game, it is fully chance-based.

Features

Main requirements - All done

- At least 4 views:

- Menu view **Done**
 - From this view, users can navigate these below views:
 - Game view **Done**
 - Leaderboard button. **Done**
 - How to play view **Done**
- Leaderboard view:
 - List of high scores (or past scores) **Done**
- Game view:
 - The view where users can play with your game. **Done**
 - Show current score/status of the game. **Done**
 - User can take some actions to progress the game. **Done**
 - Contains some animations during the game. **Done**
- How to play view:
 - Show user rules/how to play this game. **Done**
- Background music for more than two views. **Done**
- Effect sounds when users at least:
 - Take action. **Done**
 - Winning with that action. **Done**
 - Losing with that action. **Done**
 - Result at the end (Ultimate win or lose). **Done**
- User Interface fit in the screens for all iPhone >= 11 **Done**

Extra requirements - 4 extra requirements done

- Users can exit totally from the app during the game, when they open the app again, they can have an option to resume by click on “Continue” button on the menu view. (+5 points). **Done**
- Users can register for different username before playing so the leaderboard will show different names for each score (+5 points). **Done**
- The app also works on iPads and MacOS (+5 points). **Done, macOS using Catalyst**
- UIs works well on the iPhone light mode and dark mode (+5 points). **Done**

Feature description

- **Menu view:** Simple VStack with NavigationView to navigate to each page
- **Game view:**
 - The default bet amount is chosen for the player. Players can change bet amount and chose what to bet on (Red pocket, even numbers, 1st 12 numbers...)
 - Every time players chose their bet, a winning pocket is randomly selected. It is sent to a different view to run an animation that stops on that pocket and shows the result if the player win or lose.
 - The spinning animation is achivied by stacking the list of pockets on top of each other multiple times, then scroll very quickly to the result positioned at the end, creating the illusion of a looped roulette spin.
 - Animation for winning and losing applied from lecture.
 - The games continues until the player chose to finish it or run out of money.
 - Players who chose to finish with money left can register it as a high score. As in real life, the real winner is the one who knows when to stop, as casino games are mostly chance-based.
- **Leaderboard view:**
 - A LeaderboardEntries class with contains all entries in the leaderboard is used as an EnvironmentObject.
 - Leaderboard data is saved and read from UserDefaults. There is sample data for these entries if no data from UserDefaults exist.

- How to play view:

- Simple VStack that shows all the steps to play.

- Background music, sound effect:

- Using AVKit as per lecture.
- Background music attached to View.onAppear and sound effect attached to Button.onAction or View.onAppear where appropriate.

- Resume screen:

- Using UIApplication.willEnterForegroundNotification on .onReceive to set a @State to true, which then render the pause menu instead of the game view.
- Pressing “Resume” on the menu set that @State back to false, thus rendering the game view again.

- Leaderboard username:

- Each LeaderboardEntry object used for containing entry on the leaderboard contains a “username” field.
- Once registered, a username and their score is appended into the LeaderboardEntries EnvironmentObject.

- iOS, iPadOS, macOS:

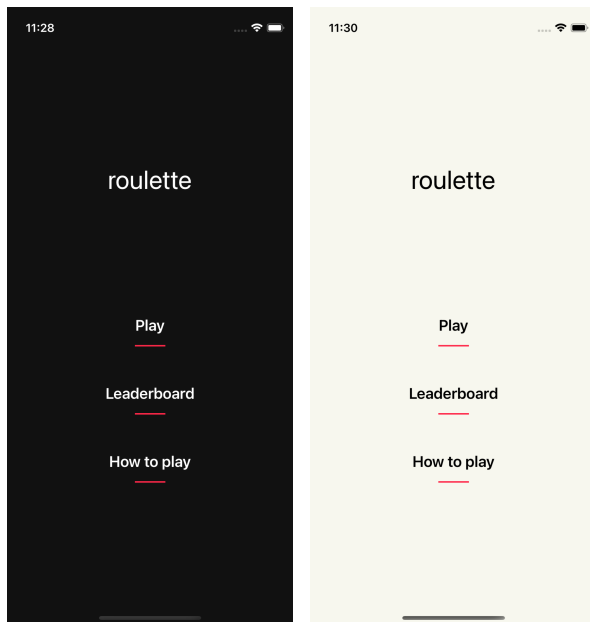
- The game uses native iOS components and views, which can be rendered on iPad without any change.
- macOS made possible using Catalyst, which renders the game with iPadOS layout.

- Light, dark mode:

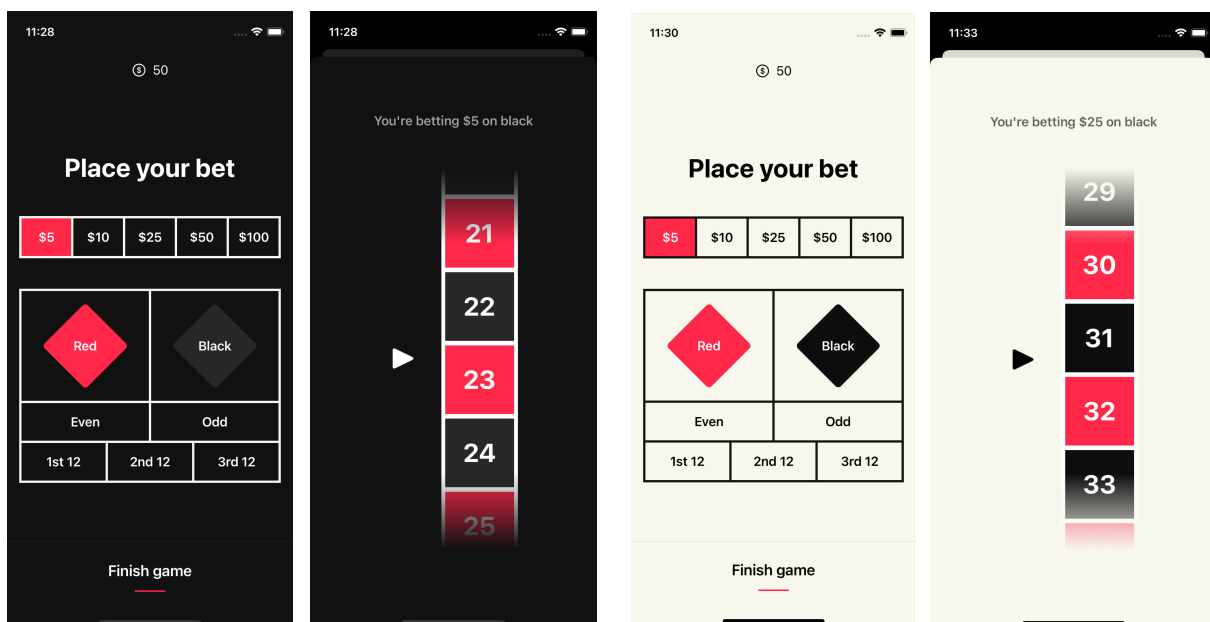
- UI designed beforehand had considered colors for both light and dark mode.
- Using Color asset with both “Any” and “Dark” appearance

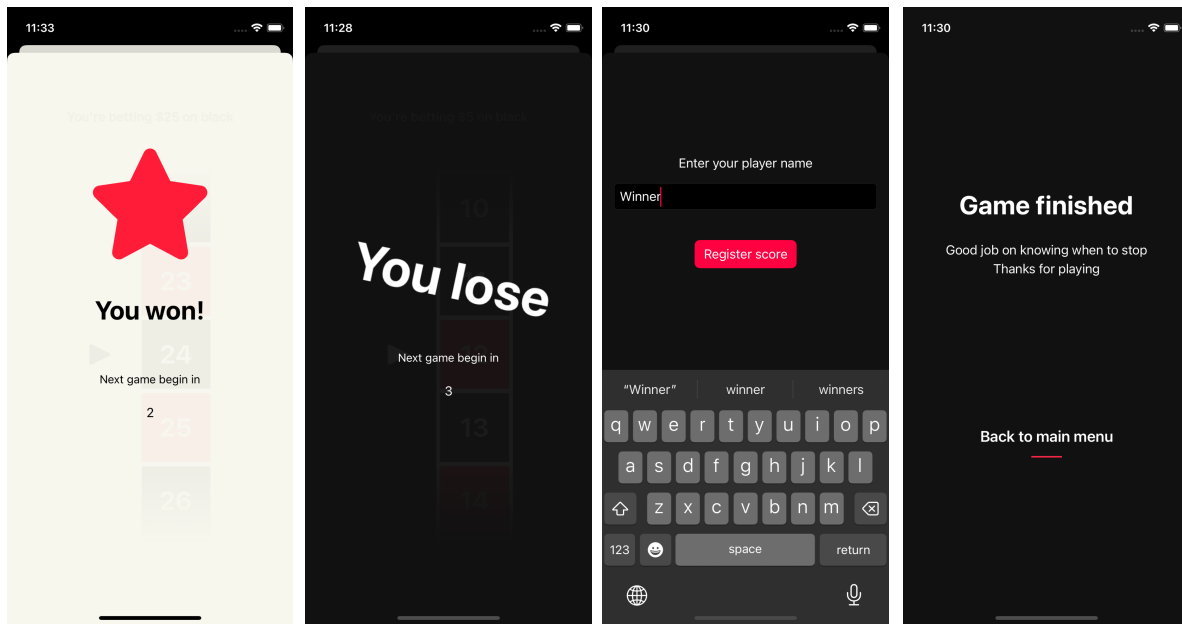
Screenshots

- Menu view:

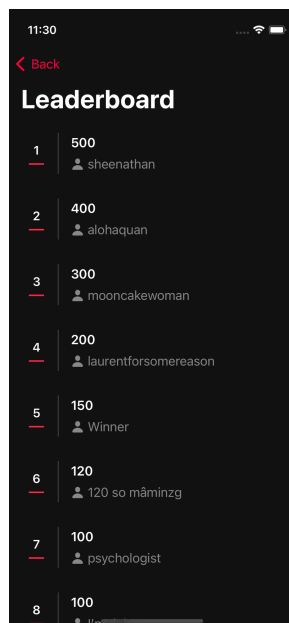


- Game view:

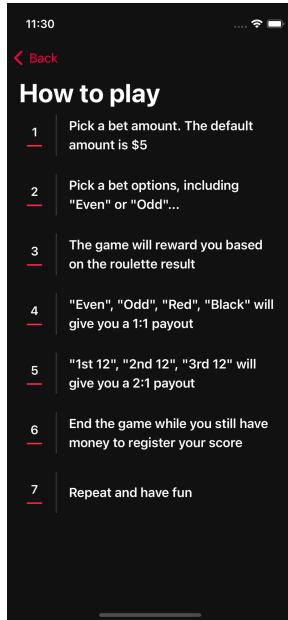




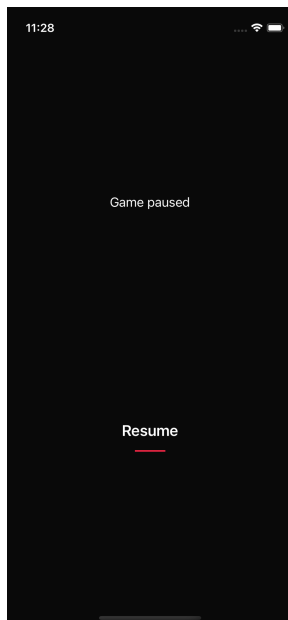
- **Leaderboard view:**



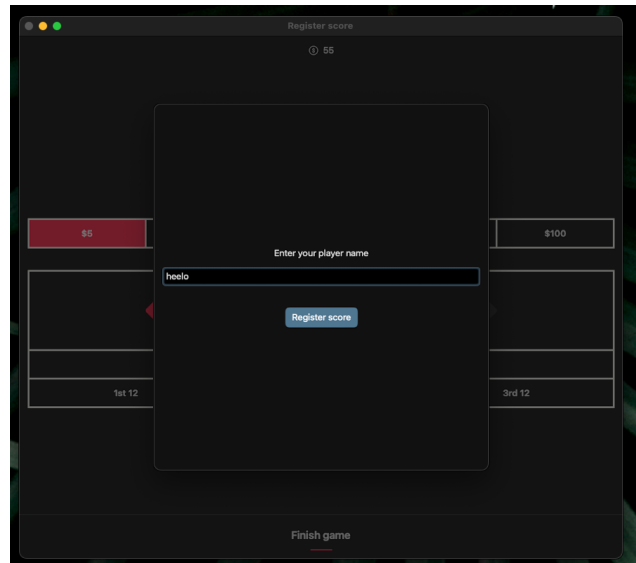
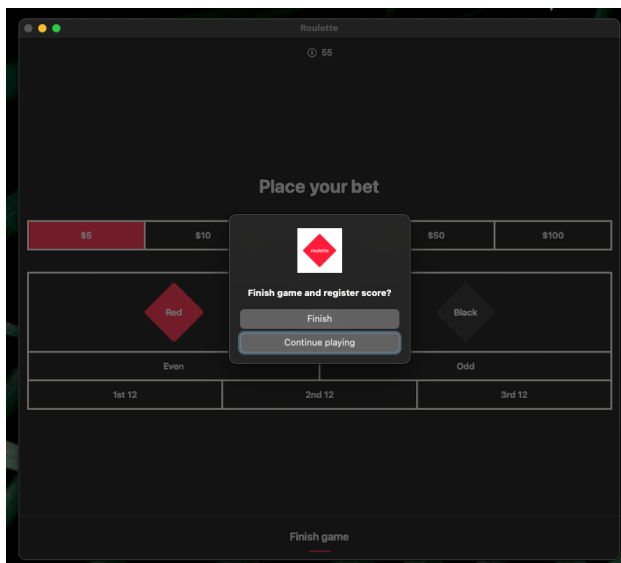
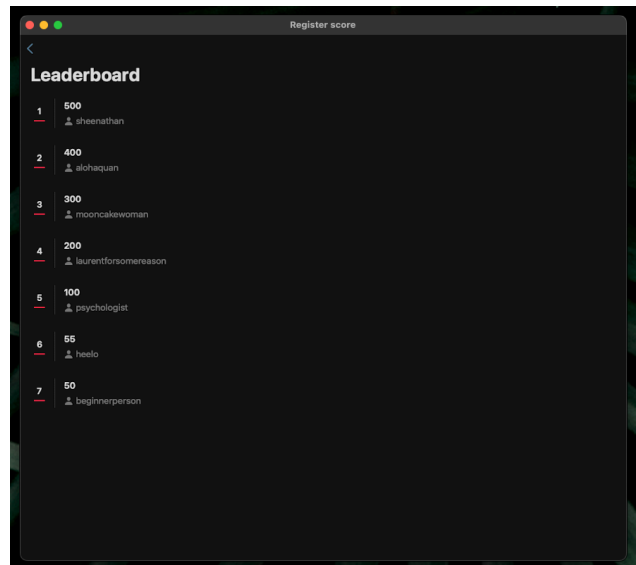
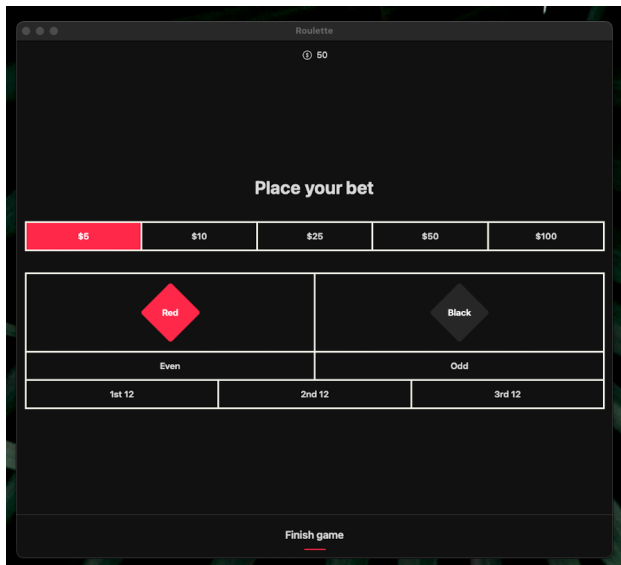
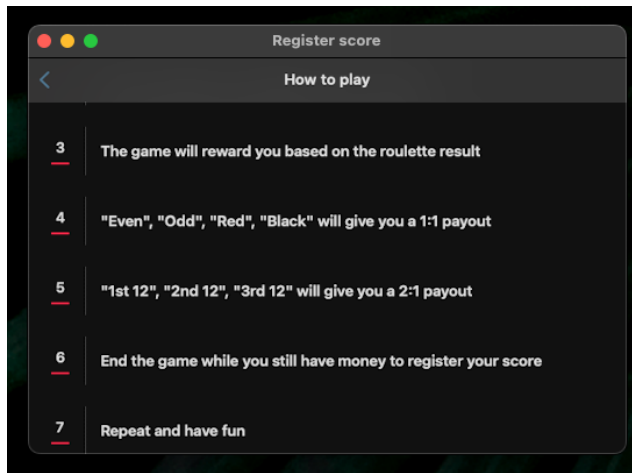
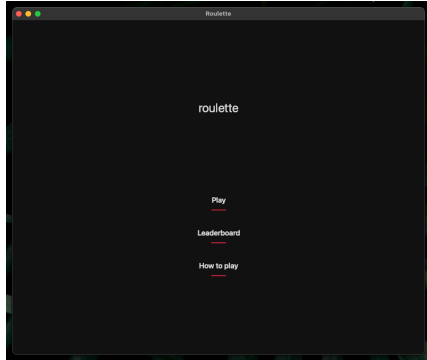
- How to play view:

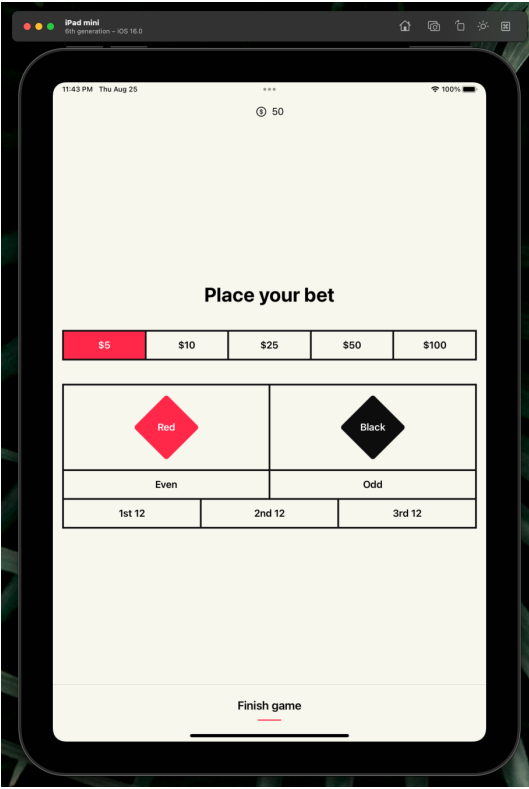
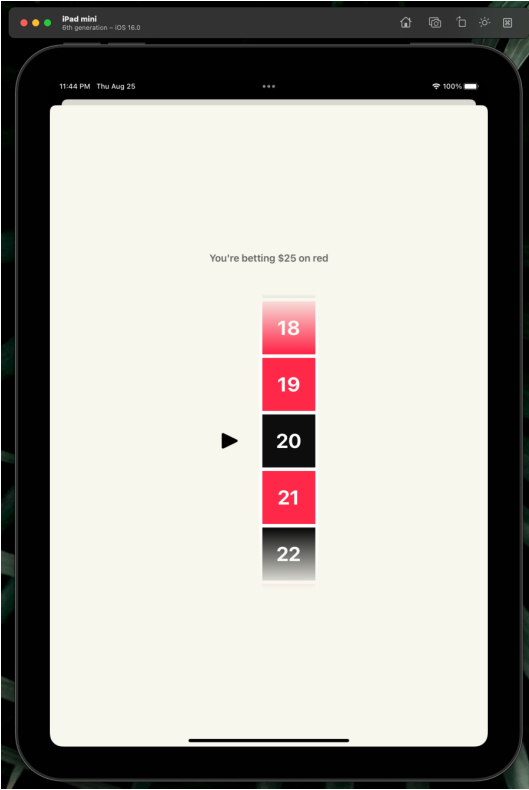
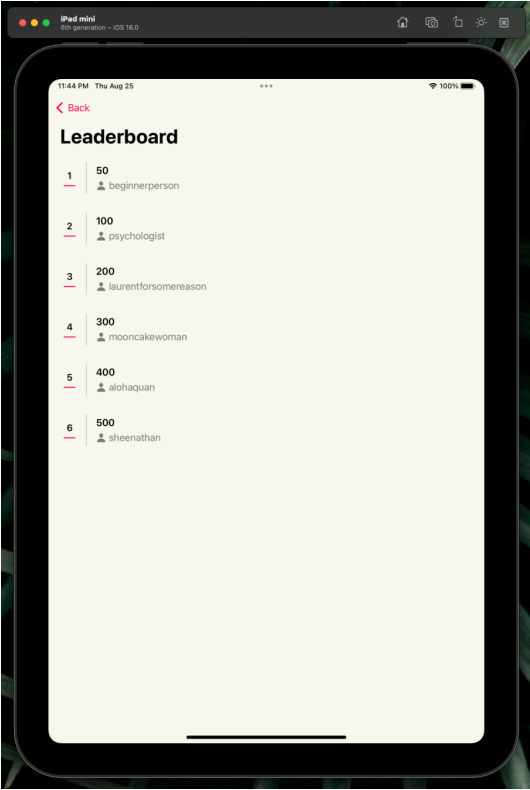


- Resume screen:



- iPadOS, macOS:





Video recording link

<https://1drv.ms/u/s!AkSeWnpnctcTiOlulvB4Z8izBY2vIA?e=sTFJeg>