

Rock, Paper, Scissors

Create an application that simulates the classic game of "Rock, Paper, Scissors."

Screen Implementation

You will need to decide how to implement the following four screens. One must use Storyboards, another one XIBs, the third one should be done programmatically*. It's necessary to analyze the requirements of each screen and determine which approach best fits each one.

* if you have trouble doing it programmatically, do it using XIB, but don't forget to try first

Requirements

Start Screen

Include a *LaunchScreen* with an image related to the game.

Configuration Screen

On the configuration screen, users will be able to customize their gaming experience, including the following elements and functions:

- **Player's Name:** Provide a text field where the player can enter their name. Entering the name is mandatory to continue.

- Game Options: Users can choose between two game options: by rounds or by points.
 - By rounds: include a slider configured with values from 1 to 5 (no decimals), allowing players to choose the number of rounds needed to win.
 - By points: add three text fields: one for the “win value”, one for the “lose value”, and one for the required score to win.
- Win Value: The value added to the score when winning a game
- Lose Value: The value subtracted from the score when losing a round.
- Required Score to Win: The score the player needs to achieve to win the session.
- "Continue" Button: Present a "Continue" button that allows players to advance to the game after entering a valid name and configuring the winning rounds or points, depending on the case. The button remains **hidden** until all fields are correctly filled out.
- "Information" Button: Add an "Information" button that, when pressed, takes the user to another screen explaining the rules of the "Rock, Paper, Scissors" game. You can use the Info system icon. In the view, Include a section for acknowledgments, which will contain your name.

Game Screen

The main screen displays three buttons, each with an **emoji** hand symbol representing the "Rock," "Paper," or "Scissors" choices. The background of this screen should be gray. When the user selects an option, the application randomly selects the opponent's choice and displays it on the screen.

Determine the game's outcome by comparing the user's and opponent's selections and display whether the user won, lost, or tied.

- If the user wins, change the background of the screen to green to indicate victory.
- If the user loses, change the background to red to indicate defeat.
- In case of a tie, set the background to brown.

In all cases, a label should indicate the result with a string and always mention the player's name.

Game Mode

- If the user selects "by points," the application displays a score label that starts at zero at the beginning of the game. When the user wins or loses a round, the points they input are added or deducted to the score. The points update in real-time and are displayed on the screen for the user to track their progress. When the user reaches the input winning score, a victory alert is displayed. Accepting it removes the alert. Negative scores are not allowed, show zero instead.
- If the user chooses "by rounds," the application displays a round counter that starts at zero at the beginning of the game. Each won round adds 1 point to this counter. When the user reaches the number of total rounds selected on the slider, a victory alert is displayed.

Next Turn:

After a round is played, a "Next Turn" button appears, allowing the user to proceed to the next round. The background of this screen should be gray again, but the score must be maintained according to the selected game mode

Game Reset:

Provide a "Reset Game" button that restores all options and values to their initial states.





