Game Overview

Title: Rock Paper Scissors – Web Game
Genre: Casual / Mini Game
Platform: Web (Desktop & Mobile)
Target Audience: Casual gamers and web users looking for a quick interactive experience
Tech Stack: TypeScript, HTML, CSS, LocalStorage, Audio API

Description: A modern version of the classic Rock-Paper-Scissors game. Features responsive design, keyboard and click interactions, AutoPlay mode, sound effects, and persistent score tracking.

Core Gameplay

Mechanics: | Player Action | Outcome | |--------------|--------| | Choose Rock | Wins, loses, or ties against the computer | | Choose Paper | Wins, loses, or ties against the computer | | Choose Scissors | Wins, loses, or ties against the computer | | Auto Play | Automatically plays moves every second | | Reset | Resets scores and clears LocalStorage |

Rules: - Rock beats Scissors
- Paper beats Rock
- Scissors beats Paper
- Tie occurs when player move equals computer move

Score Tracking: - Tracks Wins, Losses, and Ties
- Scores are stored in LocalStorage to maintain state between sessions

Features

Player Interaction: - Click buttons or use keyboard shortcuts (R, P, S)
- Updates result and score on interaction
- Sound feedback plays for manual clicks

Auto Play: - AutoPlay button initiates automatic gameplay every second
- Button text toggles between "Auto Play" and "Stop"
- Manual clicks stop AutoPlay and change button text to "Manual Play"

Sound Effects: - Win: winer.mp3
- Lose: loser.mp3

- Tie: tie.mp3
- AutoPlay does not play sound by default

Responsive Design: - Desktop: Centralized layout with visible controls
- Mobile: Buttons resized and spaced for touch
- Footer visible but unobtrusive

Footer: - Text: "Created with ❤️"
- Links to GitHub: https://github.com/MohsenRahbar
- Full-width, bottom positioned, responsive

Reset: - Resets scores, clears LocalStorage, stops AutoPlay, updates UI

UI Layout

Desktop:

```
[ Rock | Paper | Scissors ]  <- Buttons
[ Result: You win. / You lose. / You tie. ]
[ Move icons: Player vs Computer ]
[ Score: Wins: 0, Losses: 0, Ties: 0 ]
[ Reset button ] [ Auto Play / Stop ]
---------------------------------------
                Footer
        Created with ❤️ (GitHub link)
```

Mobile: - Buttons resized and spaced
- Footer semi-transparent, full-width, text centered
- Result and move icons visible above buttons

Audio & Feedback

Event	Sound File
Player Wins	winer.mp3
Player Loses	loser.mp3
Tie	tie.mp3

• Sounds play only on manual interaction
• AutoPlay does not trigger sounds

Keyboard Controls

Key	Action
R	Play Rock
P	Play Paper
S	Play Scissors
A	Toggle AutoPlay

BDD Scenario Representation

Directory Structure:

```
/Scenario/features
    play_game.feature
    auto_play.feature
    sound_effects.feature
    reset_footer.feature
```

Sample Feature – play_game.feature:

```
Feature: Play Rock Paper Scissors

Scenario: Player clicks Rock
  Given the game is loaded
  When the player clicks the "Rock" button
  Then the result should be displayed
  And the score should be updated
```

- Other features include AutoPlay, sound effects, Reset, and Footer interactions

Technical Notes

- TypeScript for strong typing and state management
- LocalStorage for persistent score tracking
- Audio API for playing feedback sounds

- Media Queries for responsive layout
- Keyboard events attached to window

Future Enhancements

- Animations: Move icons animate when player or computer plays
- Leaderboard: Track multiple players and high scores
- Themes: Change colors and backgrounds
- Mobile Touch Optimization: Larger buttons, better spacing

End of GDD