**Rocket Race**

**Design: Petr**

**Lesson 3 - Harmony 1**

A Quick Note From Aspen on Prototyping Rocket Race

The goal here is to quickly make a functional prototype that we can start testing with students and iterating on as soon as possible. With that in mind we encourage:

* Using programmer art, premade assets or AI generated assets
* Prioritizing functionality
* Leveraging tools like Github co-pilot or Claude to build quickly
* All pseudo code/functions in the GDD and architecture are suggestions
* Being creative and using your own design sensibilities to build a great game – if you disagree with the current game design that’s okay! Make something even better :)

Summary

One liner: [**Baamboozle**](https://www.baamboozle.com/games)**/**[**Kahoot**](https://kahoot.it/) **meets aural skills**

Music Learning Objective: **helps develop a child's musical ear by teaching them to distinguish between different sounds.**

Game genre: **educational, arcade, trivia**

Themes

Player Emotions (Aesthetics): **competition/teamwork, satisfaction**

Visual/Story Theme: **Cosmic Journey** - **outer space, with the rocket flying through a vibrant, colorful universe filled with stars, planets, and cosmic clouds.**

Player Situation

Platform & Controller: **web based game, laptop/computer, projected in front of a class**

Target Audience: **kids, age 6-12, that signed up for it with a music teacher**

Number of players: **1-20**

Ideal Player & Situation: **Classroom activity - as a part of an after school music education program.**

Total Game Duration: **10 - 20 min**

**Gameplay:**

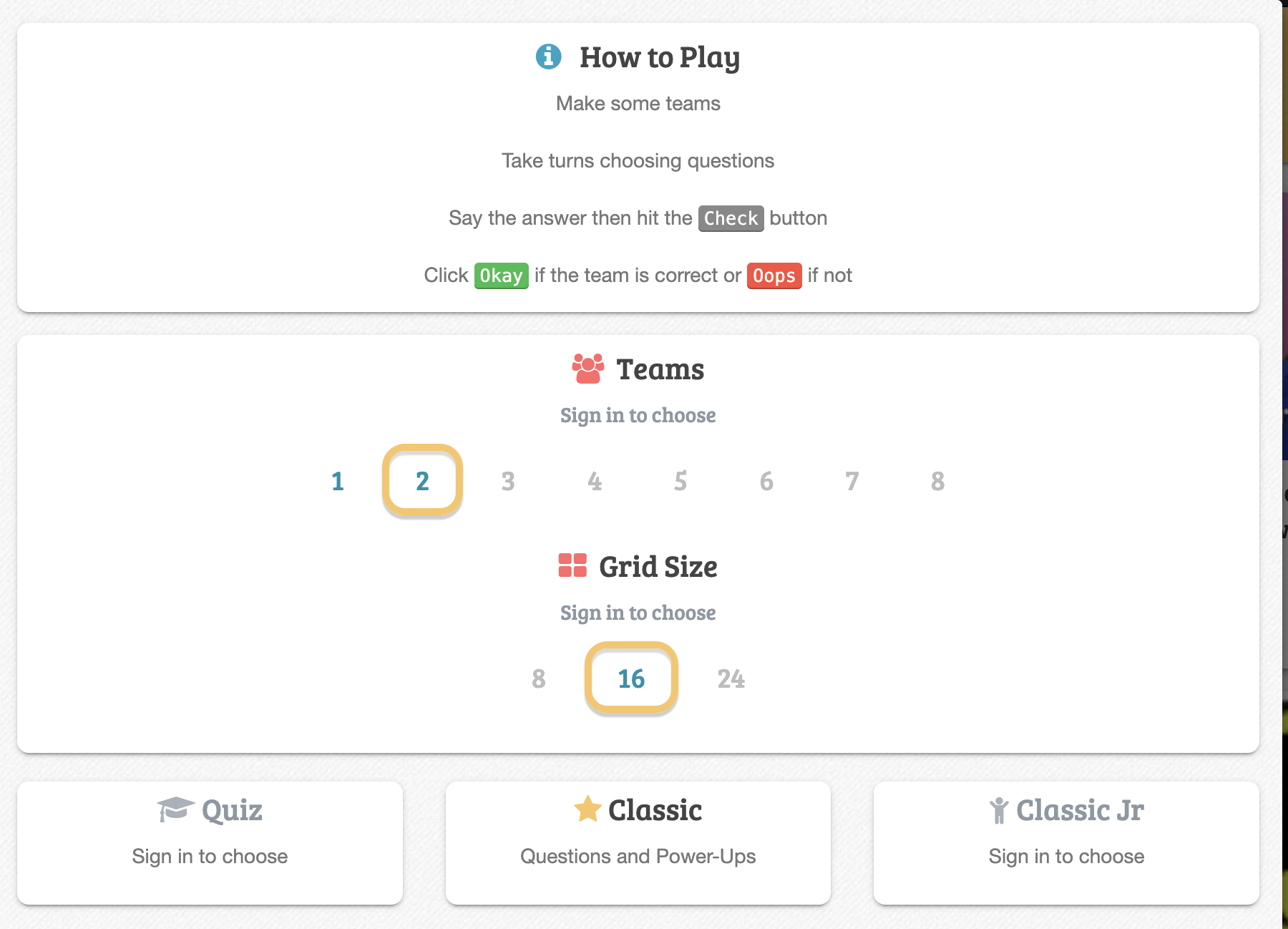
* **Setup**
  + Begins with the start scene where the teacher chooses which topic to teach, timer length, how many teams and other personalizing options. Then they press the start button.
    - Note: Make sure the architecture allows us to extend for
      * More sets of audio files to add different topics.
      * A future ability for teachers to upload their own labeled audio sets.
  + The game begins on Team 1’s turn with all of the rockets at the same height flying next to each other near the bottom of the screen.
* **Outer Loop**
  + The round starts with team 1 and their team name is highlighted. Each team gets a turn in order.
  + **Core Loop**
    1. A team is shown a unique question. The corresponding audio file plays
    2. Multiple choice options appear
    3. Timer starts and counts down. (If it hits zero, move to step 5 with a wrong answer)
    4. The team guesses and the teacher or class leader clicks the chosen option
    5. Game reveals the answer and gives visual/audio feedback about it being right or wrong and the number of units are added to their team’s distance.
  + At the end of the round, after each team had a turn, there is an animation that gets them excited for the next round! Their rocket either has a malfunction animation and moves forward 10 units or the rocket has a boost animation and moves forward 30 units. This depends on if they got the question right or wrong.
* **Game End**
  + The game finishes when the chosen number of rounds have been completed. A gold medal appears above the rocket that got the farthest. If other rockets tied for the win, they also get gold medals.
  + A large “Play Again” button appears in the middle of the screen. This button takes the player back to the main menu.

**Game Objects:**

* First Scene: Start screen

******

* + Title: Rocket Race (Text)
  + “Start” Button - opens the options modal
  + Start screen backing track - mp3 starts on a loop when the game opens and plays until “Play” button is clicked
  + Options (UI Modal Popup)
    - Title: How To Play (Text)
    - Text paragraph:
      * Make Teams
      * Click the answer!
    - Preferences
      * Sets of sounds to teach with (dropdown)
        + Instruments
        + Chords
      * # of teams (dropdown)
      * Display of Teams (equal to # of teams)
        + Team Name (text input)
        + Rocket Design (Dropdown)
        + Rocket Color (Dropdown or other UI)
      * How long the round timer is (dropdown)
        + Default 120 seconds
      * # of rounds
        + Default 5 rounds
    - “Play” Button - starts the main scene
    - Example Screenshot from Baamboozle



* Main Scene: Rocket Race
  + Question Area - top half of the screen

******

* + - Question
      * UI text that says “what do you hear?”
      * Audio source
      * “play sound” Button - timer starts when the button is clicked and potential answers are revealed. The button can be clicked again to hear the audio playback once more.
      * RoundStart() - when a round is starting the question text changes to display the round number (eg: “Round 1”) and then “GO!”
    - Multiple Choice Answers
      * UI image
      * 2 or 4 clickable buttons shown
      * OnClick() - triggers: feedback for correct/incorrect answer, adds to the score, pulls up the next team button
      * Next Team Button - once this button is clicked it triggers the next turn to start. The next team’s name is highlighted and the “play sound” button appears.
        + This allows time for teachers to discuss the previous question with students if needed before moving on.
    - Question Banks
      * Chords
        + 20 mp3 audio files labeled as major or minor
        + 20 strings for the chord names
      * Instrumentation
        + 20 mp3 audio files labeled with instrument names
        + 20 strings for the instrument names
  + Teams Area - bottom half of the screen

******

* + - Team Names (text)
    - Scores (text of distance traveled)
    - The Rocket
      * Sprite - playful, cartoonish with exaggerated features
      * Animations
        + Malfunction/hit by meteor
        + Boost
        + Incorrect answer
        + Correct answer
        + Idle
      * Behavior
        + MoveUp() - moves the rocket up on the screen
        + Malfunction()
        + Boost()
  + Timer
    - A timer just below the question counts down
    - If timer = 0, then current team missed the question
  + Main screen backing track - mp3 starts on a loop when the main screen opens and has half the volume of the audio files that play.

**Visuals:**

* Make the text large
* Make all buttons large and clickable