Ribbit Royale

Baron Baker, Chase Bennett, Adam Bobich, Aidan Caughey, Ryan Dobkin, Luke Garci

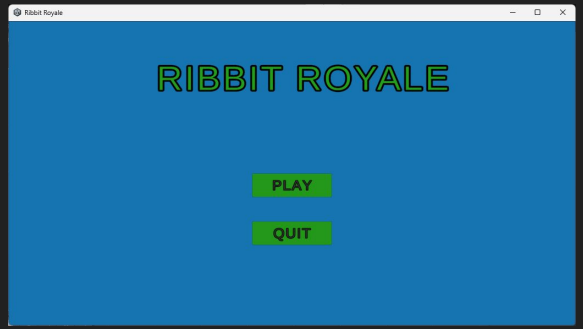
Problems & Motivations

- Lack of free to play party games on the market
- Introduction of unique mechanics and style of minigames
- LAN Multiplayer revival
- Create a casual yet competitive game for everyone to enjoy

Toolset

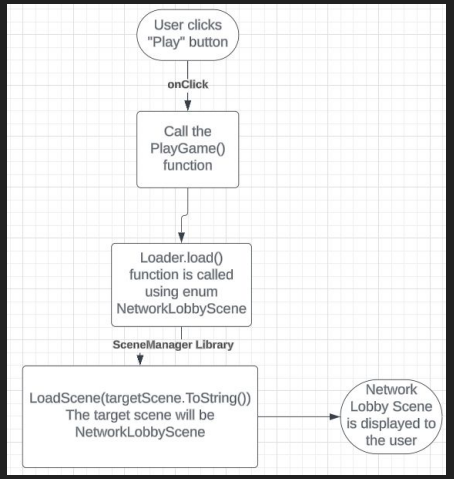
- Unity game engine and C#
- Netcode for GameObjects
  - Handles multiplayer synchronization
- Unity Relay
  - Networking solution that connects players in a peer-to-peer fashion using a server-client environment.
- Unity Lobby
  - A way for players to discover and connect to each other

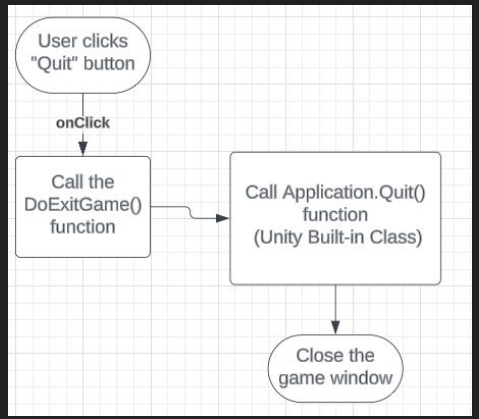
Main Menu Scene



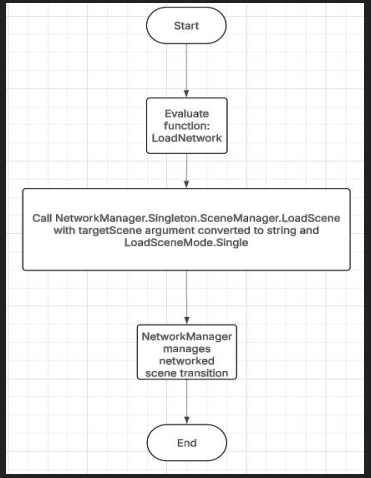
- Main menu scene
- Play button
- Quit button

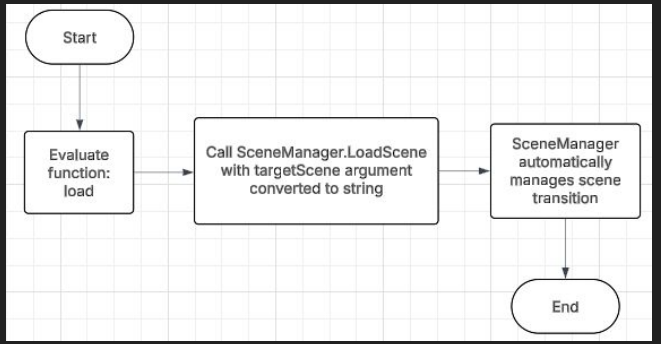
MainMenuUI.cs





Loader.cs

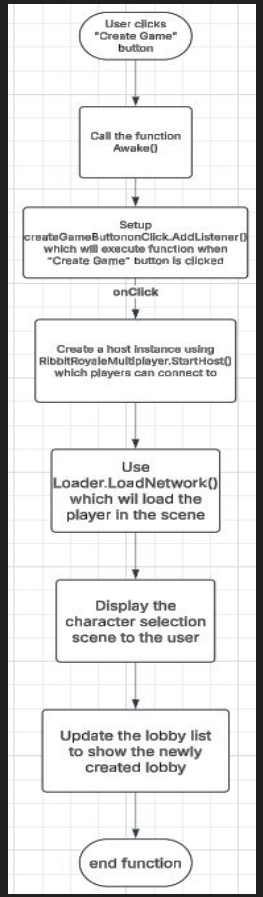


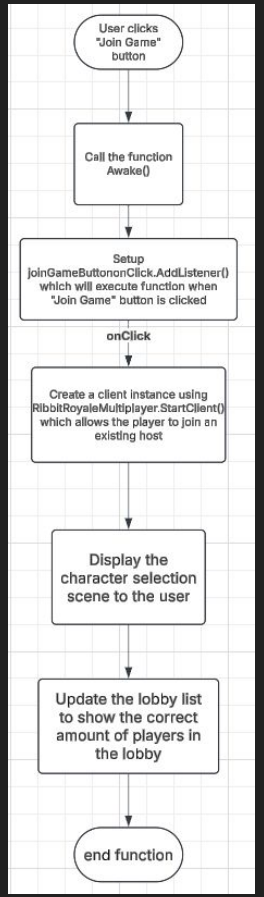


Network Lobby Scene

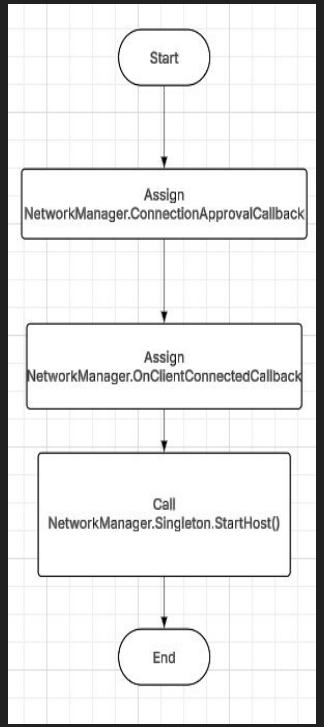
- Create Game button
- Join Game button
- Back button

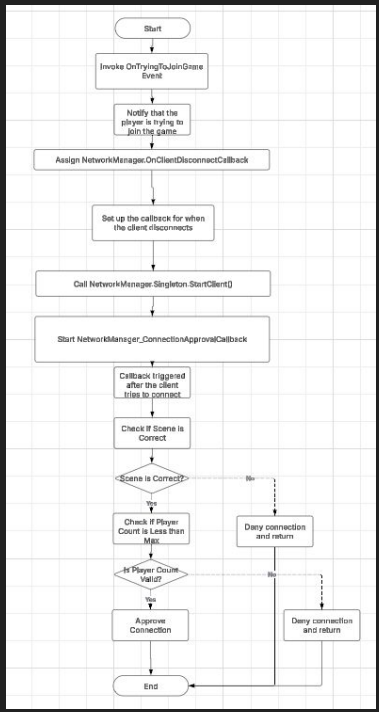
LobbyUI.cs





RibbitRoyaleMultiplayer.cs

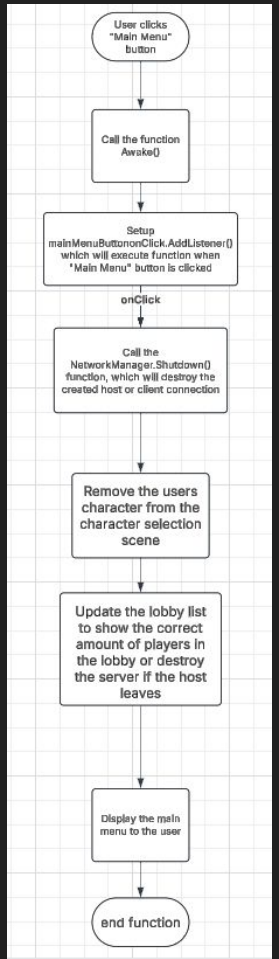


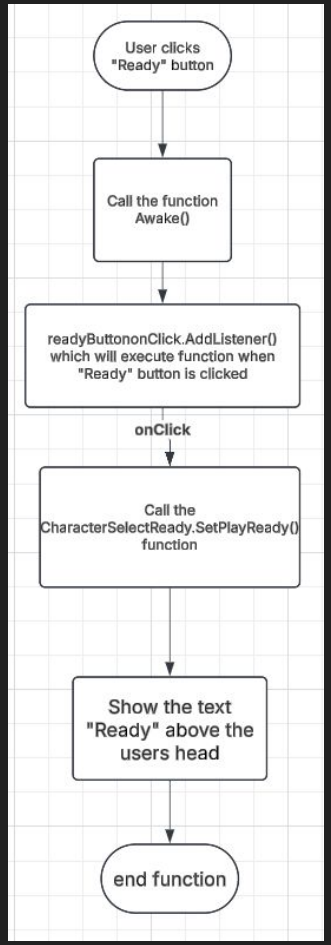


Character Selection Scene

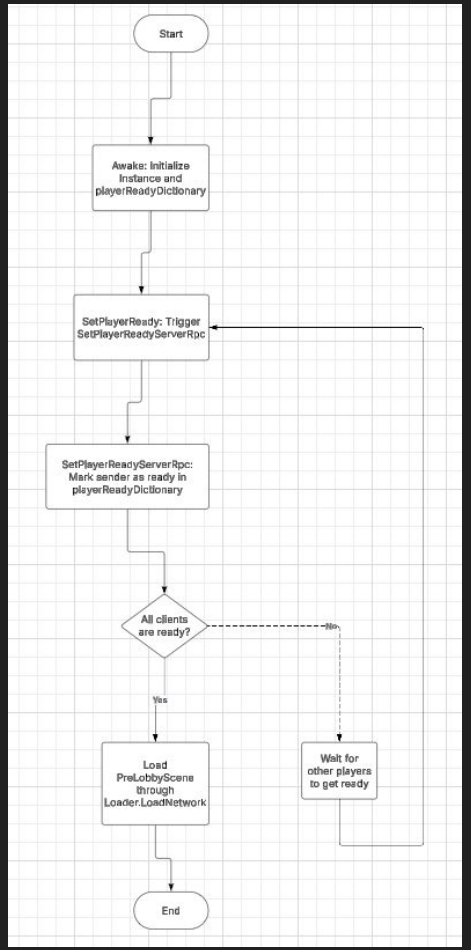
- Display user as a frog in the lobby
- Ready button
- Main Menu button

CharacterSelectionUI.cs

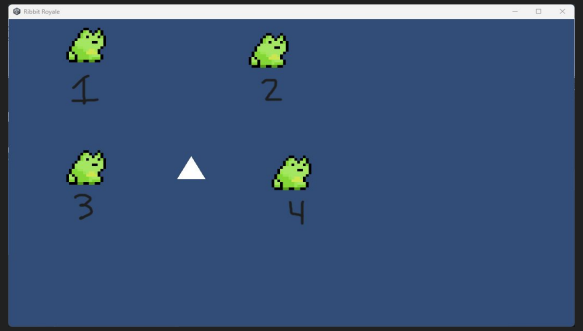




CharacterSelectReady.cs

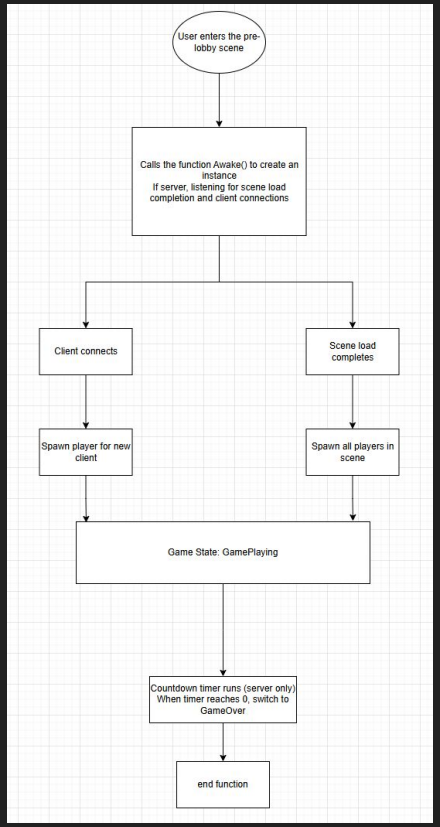


Pre-lobby Scene



- Manages the Game State
- Handles Player Connections

PreLobbyManager.cs



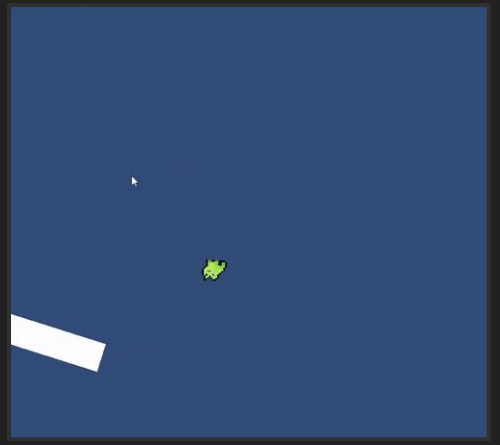
Minigames

- Lickity Split
- Count the Animal
- Snake Escape

Lickity Split

Use your tongue to swing to the finish line faster than your opponents

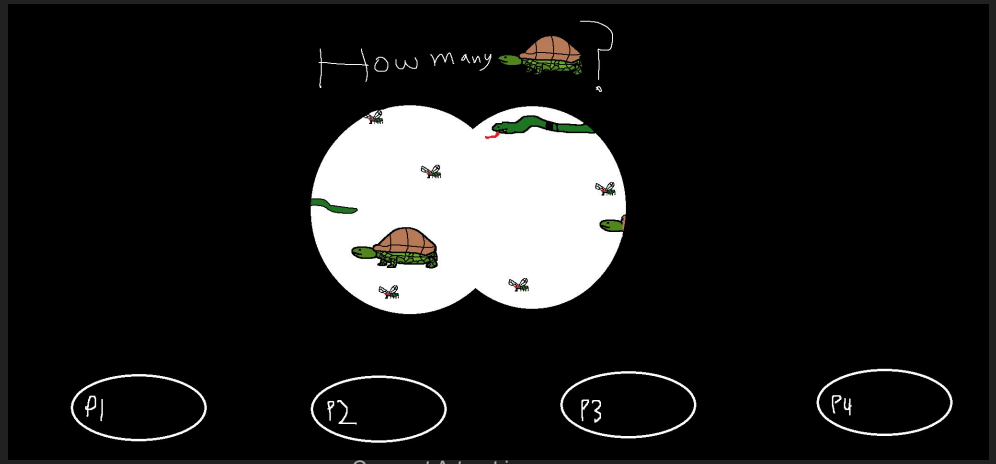
- Movement
  - A / D control horizontal movement
  - If the mouse is clicked, create a tongue object and attach to the nearest node if in range.
  - If mouse released, destroy the tongue connection.
  - If a frog lands, change it’s velocity so it bounces off.
- Swinging
  - Use a Unity Spring Joint to connect the frog to the node (swinging point).
  - Add a small boost of speed in the direction of the swing, giving more satisfying launches.
- Goal - Reach the end platform
  - Use the momentum from your swings to reach the platform at the top of the course
  - Upon reaching the platform, your time will be recorded and compared to the other players



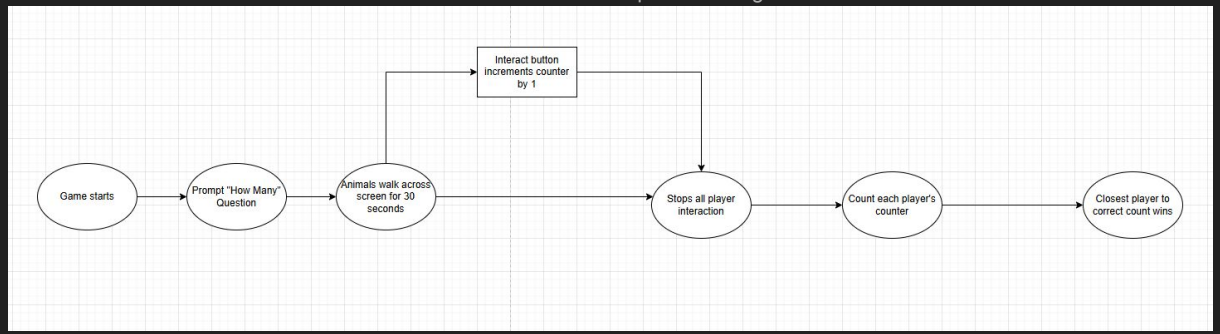
Swamp Spotting

Count the creatures of the swamp

- Objective is to count the specific animal
- Animals walk across screen
  - Players press interact button to count
- Award points based on who's the closest



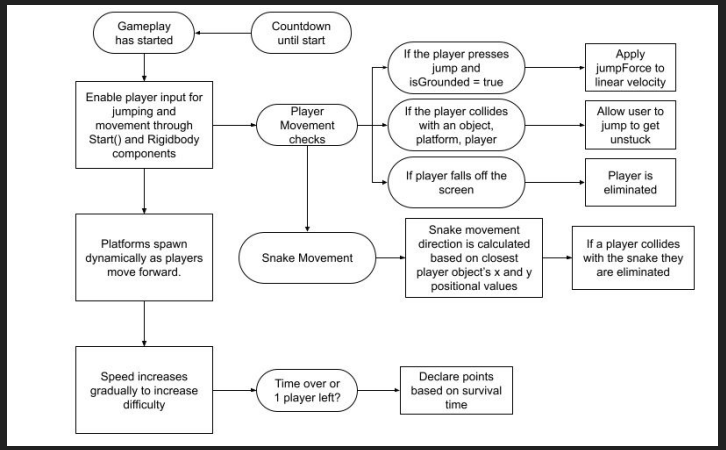
Concept Art not in-game



Snake Escape

Hop for your life, every frog for themselves

- Escape from the snake until you reach the time runs out or you reach the finish line!
- Players can control the frog to move left, right, and up.
- Players will have to jump from platform to platform, across trees or puddles, etc.
- Failure to stay alive until time runs out or falling out of the stage will result in a loss.
- Points are awarded depending on placement among players.



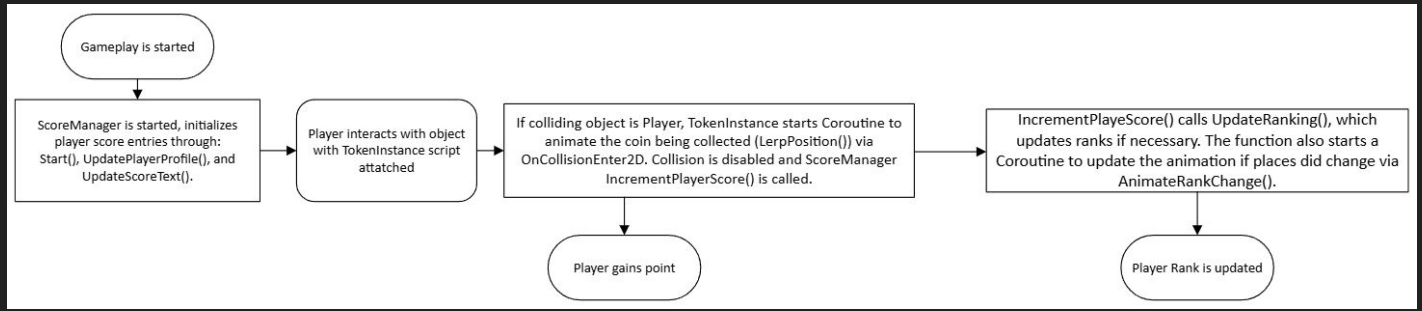
FrogController.cs and SnakeController.cs

Future functionality of SnakeChaseScore.cs included

Sprites still need full animations

Scoring

- Displayed to the user as an overlay
- Populates when players interact
- Lerp animation on place change
- Updates scores and rankings
- Avatars and Colors automatically assigned
- Can be toggled with button press (Tab)



Testing

- Unit Testing (Input Handling, Physics, Game Logic)
- Network Testing (LAN, Latency handling, Synchronization)
- UI Testing (Scoreboard, HUD)
- Performance Testing (Max Players, Network Load, FPS stability)
- Playtesting