

Foodfall

Unity Multiplayer Programming Exercise

Concept

Foodfall will be a game where multiple players compete in real-time to catch the most falling food items before time runs out!

User experience

After launching the game, player will land on a Start screen. There is an input field in which the player can enter a name, and a drop list box to select region (if applicable). There is also a large “Start” button.

When the user clicks Start, the game will first determine if there is already an instance which is in “waiting for players” state. If so, the player will be taken to a “waiting to start” screen and the names of the players who have already joined will appear. The player will then wait for the host to start the game. If no instance is in “waiting to start” state, then a new one will be created, and the player will become the host of a new instance. Player will join the new instance on the “waiting to start” screen, but will also have a “Launch” button which triggers the level to begin and enter “gameplay” mode. Only the host (the first one in a new instance) has the Launch button.

When the game is launched, all players will be taken to the play field. Each player will have some character representing them, along with their name and a score (which will start at 0). Players will then all see a synchronized 3-2-1 Go! countdown and message, after which the game itself will start. At the top of the screen will appear a timer counting down for the (configurable) duration of the match.

During gameplay, various different food items will appear in the area above the players and start falling to the ground, where they will then disappear. Characters will be able to move using the keyboard and/or mouse to collect the items by simply colliding their character with the item. Each item can only be claimed by one player. If two players arrive at the item at the same time, the game will need to decide which one to award it to.

After the timer expires, the gameplay will stop and the players will see a sorted list of the scores, and a Finish button which takes them back to the start screen.

Parameters

- Game must be built in Unity using no older than version 2019 LTE.
- Choice of server technology is left to the developer, however it is recommended to use Photon. (PUN/Real-time/Fusion/Quantum/etc.)
- Game must support up to 10 players, and be testable by running multiple clients on the same machine.
- Developer will provide all source project code as well as an executable version of the game (for Windows) or a link to a playable WebGL version.
- Developer will either provide instructions on how to launch the backend server code (if applicable) or make sure that there is a backend service running in the cloud at the time of submission, such that Myria can simply run the game and start a session directly.

Points to note

- All visual design has been left up to the developer. Be as creative as you like! (Hint: creativity is encouraged.) You may create the art and/or use premade assets from other sources.
- Developer may opt to make the game 2D or 3D as desired.
- The game is not required to have bots, but if you opt to add them, they must also be synchronized and conform to the same rules as the human players.
- Developer is free to add features to the game if so inspired. As with the visuals, creativity is encouraged but not required.
- Pay attention to edge-cases (e.g., what if a player drops out somewhere in the flow?).
- Game code will be evaluated as much as gameplay, so make sure that your code is structured well and the game parameters can be easily tweaked/tuned.
- Game is real-time multiplayer, meaning that the synchronization/replication must be as accurate as possible, yet gameplay must be smooth from the users' point of view.

Best of luck with the exercise, and please do not hesitate to contact art.zaratsyan@myria.com if you have any questions.