Clock Blockers

Start Date: 18.02.2020

Last Edited: 25.02.2020 02:34:20

Developers: Kenneth Hoff

Contents

Working Title

Clock Blockers

The name was taken directly from this game's main <Word for basing something heavily on something...> of the same name, Clock Blockers video by Corridor Digital

Concept Statement

It's a Multiplayer FFA FPS where you control a character, and after completing a 'game round', a new round starts where your previous actions are being replicated by an NPC. Any serious deviation from that course, and an AI will take control over that character.

Genres

This is a FPS that involves a lot of strategy; You might encounter a scenario where you don't want to save your teammate, as it would put you in a spot where you hinder yourself in a future round.

Target Audience

The target audience for this game is teenage/adult, primarily male.

Unique Selling Point

It's an FPS game where your actions have serious repercussions for future rounds, and where you might go against a normal FPS standard of 'better aim = better rank'. Of course, aim still helps.

Player Experience and Game POV

What is the fantasy the game grants the player?

This game uniquely focuses on being able to undo your previous mistakes, while setting up yourself in the future for an advantage. It focuses a lot on the ideas of time and manipulating past- and future actions.

The setting is TBD. Preferably chronal. (Relating to time)

Player Goal

The goal is to be the last man standing on the final act of a round. This is done by carefully planning your actions, and not be rash. Every action you make will last the entirety of the round, unless you actively work towards undoing it.

It does not inherently matter if you lose every single act of the round. If you're able to win the final act, you will win that round.

First to win X rounds wins the match.

Platforms, Technology, and Scope

This game would primarily launch on Steam but could easily be ported to consoles in the future. I don't see phones ever becoming a thing.

Created almost exclusively in Unity. Currently there is no concept of how long it will take, although it will take months.

Core Loops

The game is composed of matches, where players fight off against each other.

Every match is composed of multiple rounds. Whomever wins the most rounds wins the match.

A round is composed of multiple acts. The winner of the acts is whomever has the most remaining characters at the end of the final act. Most often there will only be one player alive at the end, but there are time limits to each act.

In an act, you acquire currency by winning rounds, and doing certain actions, like killing the enemy, or defending your teammates. You can spend this currency at the start of an act to acquire equipment such as weapons, armour, and other utility, or you can save it for future acts.

Objectives and progression

[Objective is mentioned multiple times already ??]

Game Systems

This game relies on these main systems.

- Action replay and storage
    - Being able to record actions, and then replay those actions in future rounds.
    - This was originally naïvely (and, as of writing this, still is) implemented as "Record every action the player makes and replay those actions from a list". This will later be rewritten to (And the entire idea was acquired in the making of this document – Creating this document helped way more than I had intended, or ever expected):
        - Occasionally (Say every 0.2 seconds), record the transform location of the character (Position and rotation). Whenever something more event-driven is taken place (Shooting mostly), that will also be recorded *with intent* (What target was I aiming at?)
        - When replaying you must then interpolate the difference in position by some method and move accordingly, as well as attempt to shoot at the target you originally intended.
            - If any action failed, you "unlink" from the action replay, and an AI takes over.
- Character AI
    - Predict intention of an action, and correctly execute them.
        - If a character hit an enemy, store the enemy it hit in the action, and when replaying, aim towards that enemy and shoot. Conceptually quite easy.
        - What if you're shooting at a wall to bait an enemy? Were you trying to hit an enemy, or just shooting at a wall? It needs to know if it should be smart or dumb.
    - Create new actions that have some logic.
        - After "unlinking", there needs to be a system in place to make the character do something. Conceptually quite easy.

Interactivity

The player can move their character freely in 2D space, as well as somewhat in 3D space by jumping. The player is actively looking for the enemy characters and killing them, as well as making sure their teammates survive (or don't, if the player thinks that's their purpose).

There will be normal 3D physics, as well as (potentially) destroyable terrain.

Milestones

Development

- 18.02.2020 – Started the project, with no real plan

Design

- 25.02.2020 – Created this document.
    - This drastically improved the design of the game and helped me visualize a future possibility of how the game could work. After a week of development, there were still significant core issues with action replay that I could not understand how to fix, especially in its accuracy. It often lost a few micro degrees of movement, or rotation, and it had no way of correcting this mistake. After a few seconds of replays, it wasn't even able to hit a still target from 10 meters with a RayCast