Luis Ricardo Azuara Maldonado
Student ID: 2415604

Rhythmic Survivor Demo

This project is a rhythmic game, which is a type of game where the main mechanic is for the player to do an action following the music rhythm. The game will have a single-survival level, different waves of enemies with unique patterns that will increase in quantity and endurance, and a single player with three different weapons that will also increase their effectiveness. The game will save the player's personal information to log in and access the progress, in a database also containing the enemies and music data. The main software to build out the game will be Unity for visual and structural development.

I. Features of the selected project

- **Log in**: User creation and credentials validation through SQL.
- **Save and load**: Auto save and load when the user logs in and during the game, using multi-software communication between Unity and SQL.
- **Enemies patterns**: Enemy types with different quantities and velocities that affect the patterns that the player will encounter.
- **Player's movement patterns**: Music implementation under suggested player movements.
- **Weapons**: Three different types of weapons that vary in damage and range, increasing only between games.
- **Menu flows**: Multiple screens from log-in to credits.
- **Themes**: Various types of concepts, varying in song, theme, and maps.

II. End users

This game will have only one player. This type of user is a casual gamer or has an interest in music

III. Integration of the End users with the project

The integration of the end user into the game is limited since it's only to ensure that the progress he makes is saved, loaded and to personalize the themes to his preference.

IV. Areas covered by this project.

- Player's registration: The player will be able to register to save the progress he makes.
- Theme selection: The player will be able to choose a theme to make the game more pleasing to his taste.
- Weapons upgrades: The player will be able to upgrade the weapon he wishes and see the progress he has already made.

Project Users, Actors, Vendors, Actuators

I. User:
   - Player: Casual gamer interested in music and games.

II. Actors:
   - Development team: In-house team in charge of development and actualization of the game, including programmers, artists and game designers. The team delivers new versions of the game to be uploaded to the different platforms and to keep the database up to date.

III. Vendors:
   - Unity: A game engine where the game will be developed. It has a membership subscription to allow the team to publish the game without watermarks and access to exclusive tools.

   - Microsoft Azure: A cloud computing platform created by Microsoft to build, test, deploy and manage applications and services using its data centers.

   - Play Store: A mobile app distribution application made by Google where the game can be published and generate money earnings.

   - App store: A mobile app distribution application made by Apple where the game can be published and generate money earnings.

   - Steam: An app distribution application made by Valve Co. where the game can be published and generate money earnings.

   - Epic Games Store: An app distribution application made by Epic Games Inc. where the game can be published and generate money earnings.

IV. Actuators:

- Rhythmic Survivor Demo: A type of entertaining software with a single player where the main mechanic is for him to defeat the enemies to survive as long as possible, following the music rhythm.

Project Properties

The project will be using C# language for the programming and Microsoft Server SQL 16.0.4115.5 for the database. Unity has the required tools for the front-end and back-end development, allowing C# to be the main language of the game. A repository in Github will be created in order to manage the version control.

- Programming Language: C#
- Unity: Unity 2022.3.30f1
- Database: Microsoft Server SQL 16.0.4115.5
- Source Control: GitHub repository

Plan of Work

- **Week 1 (27 - 31 May):**

  Create the project proposal.
  Define project specifics (scope, requirements, user, vendors, …)
  Define programming language and software version for the development.
  Define game design.

- **Week 2 (3 - 7 June):**

  Design the database and the UML.
  Install the game assets.

- **Week 3 (10 - 14 June):**

  Create the database and connect it to the game.

- **Week 4 (17 - 21 June):**

  Develop the player's movement and attack with the music.
  Weapons upgrades.

- **Week 5 (24 - 28 June):**

Develop saving and loading player data from the database.
Develop the player’s login.

- **Week 6 (1 - 5 July):**

  Develop the different enemies with their movement and attacks.
  Update of the database.

- **Week 7 (8 - 12 July):**

  Develop the gameplay waves and the different levels.
  Update of the database.

- **Week 8 (15 - 19 July):**

  Create the screen flows.

- **Week 9 (22 - 26 July):**

  Test and debug the game.

This project aims to create an experience that allows the player to connect their senses and recreate. The game will have different themes, enemies and music stored in a database created on Microsoft Server SQL, as well as an opportunity for the player to store his progress with the weapons. The main language for the project will be C# and will be developed with Unity, while GitHub manages the versions of the code. This project is designed to be an entertaining rhythmic game, that could be carried out to platforms of gaming consumers such as Steam, Epic Game Store or Play Store.