Software Implementation and Testing Document

For

Group 19

Version 3.0

Authors:
Miguel Montesinos
Osher Steel
Danny Miranda
Loren Van Malleghem
Joseph White

1. Programming Languages (5 points)

Unity (C++/C#)

- *Used for: Continued development of game mechanics, enhanced visuals, and integrating new assets like portals and wall jump features.*
- *Reason for choosing: Unity's flexibility and robustness continue to be vital for developing our platformer game. The addition of complex platforming elements and new gameplay mechanics benefitted from the efficient scripting capabilities of C++/C# in Unity.*

2. Platforms, APIs, Databases, and other technologies used (5 points)

List all the platforms, APIs, Databases, and any other technologies you use in your project and where you use them (in what components of your project).

- ***Platform:*** *PC*
- ***Unity Engine***
  - ***Used for:*** *Further game development, integrating advanced game mechanics like wall jumping and portals, enhancing visuals, and animations.*
- ***Unity Asset Store***
  - ***Used for:*** *Acquiring additional assets suitable for the new levels and features added in this iteration.*

3. Execution-based Functional Testing (10 points)

- *In this iteration, functional testing focused on the new gameplay features such as wall jump mechanics and portals. We conducted rigorous playtests to ensure these elements functioned as intended and improved the overall game experience. Additionally, functional tests were performed on the three new levels to verify gameplay continuity and challenge progression.*

4. Execution-based Non-Functional Testing (10 points)

Describe how/if you performed non-functional testing for your project (i.e., tested for the ***non-functional requirements*** *listed in your RD).*

- *Non-functional testing this iteration emphasized performance optimization, particularly with the added levels and features. We conducted stress tests to ensure the game maintained stable frame rates and responsiveness under various conditions. User experience testing was also enhanced to assess the game's playability and user engagement with the new platforming elements.*

5. Non-Execution-based Testing (10 points)

Describe how/if you performed non-execution-based testing (such as code reviews/inspections/walkthroughs).

- *Our team performed thorough code reviews and walkthroughs, especially focusing on the newly integrated features and level designs. These sessions helped identify and rectify potential issues in the game's code structure and logic, ensuring a higher quality and more maintainable codebase. We also reviewed the game's design documentation to ensure alignment with the new direction and features.*