

Software Requirements and Design Document

For

Group 14

Version 1.0

Authors:

Emmanuel Ayala (Student ID: eoa18, GitHub ID: Eoa18)
Ashton Frias (Student ID: agf19e, GitHub ID: AshtonFrias)
Jordan Locke (Student ID: jll19g, GitHub ID: jll19gGH)
Justin Mulder (Student ID: jtm18j, GitHub ID: millingmadly)

1. Overview (5 points)

We will be creating a 2D game through Unity called "Go Home, Hamster!" about a lost hamster trying to get home. Each level will consist of the hamster rolling through various obstacles and avoiding enemies. There will be 4 levels in total, each with its own theme and increasing difficulty.

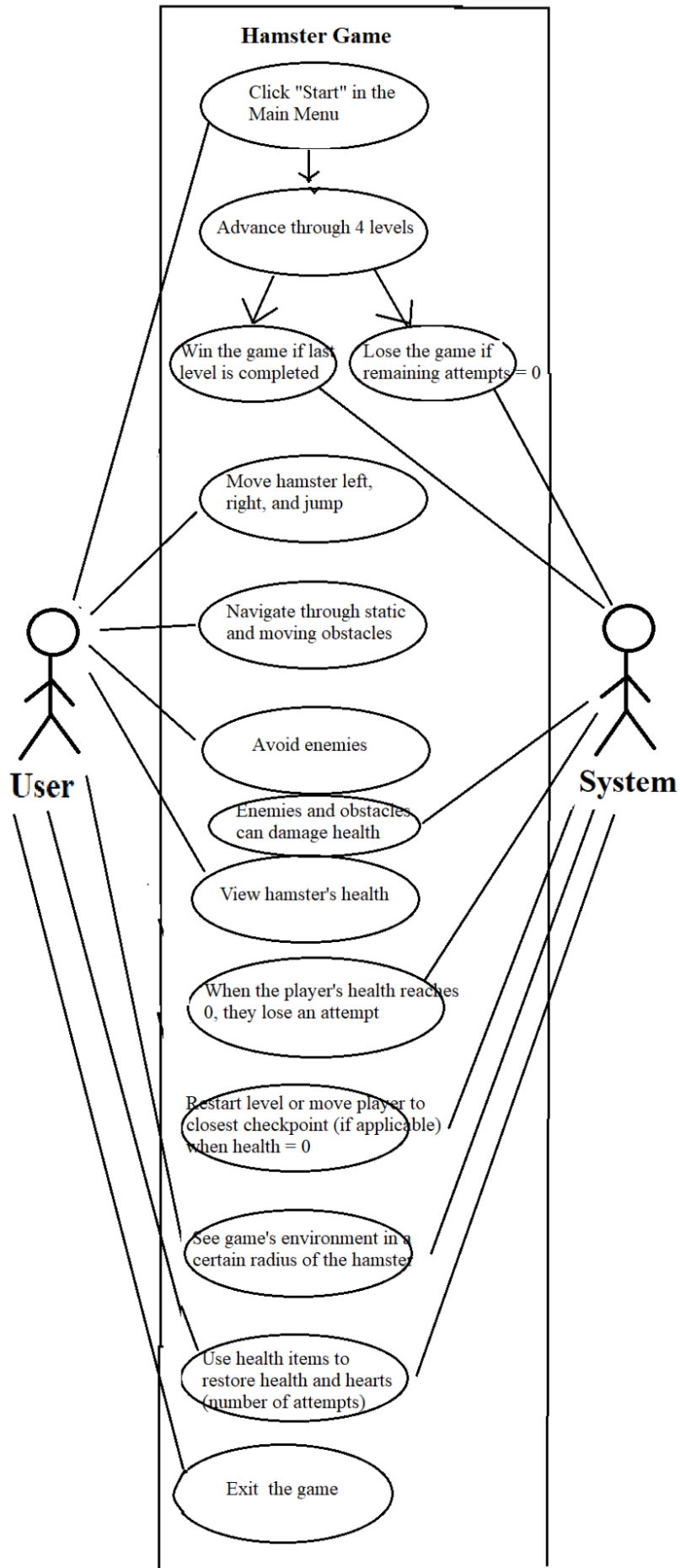
2. Functional Requirements (10 points)

1. (High) The game will allow the user to move the hamster (left, right, jump).
2. (High) The game will have static and moving obstacles that the player must navigate through.
3. (High) The game will have moving enemies that the player will need to avoid.
4. (High) The game will have a health bar for the player's hamster.
5. (High) The game's obstacles and enemies can damage the player's health.
6. (High) The game will have a number of attempts for the player's hamster.
7. (High) An attempt is taken when a player's health reaches 0.
8. (High) The game will restart the level or move the player to the closest checkpoint (if applicable) when the player's health bar reaches 0 and if the player still has remaining attempts.
9. (High) The game will end if the number of attempts reaches 0.
10. (High) The game will only show the environment in a certain radius of the player.
11. (Medium) The game will allow the user to start the game from a main menu.
12. (Medium) The game will allow the user to exit the game at any time.
13. (Medium) The game will have checkpoints.
14. (Medium) The game will have items to restore the player's health bar and number of attempts.
15. (Medium) The game will have four levels.
16. (Medium) The game will have sound effects.
17. (Medium) The game will have music.
18. (Low) The game will have increasing difficulty by progressively adding more obstacles and enemies each level.
19. (Low) The game will allow the user to mute the music.
20. (Low) The game will have a secret level.
21. (Low) The game will have a "You Win!" screen when the last level is completed.

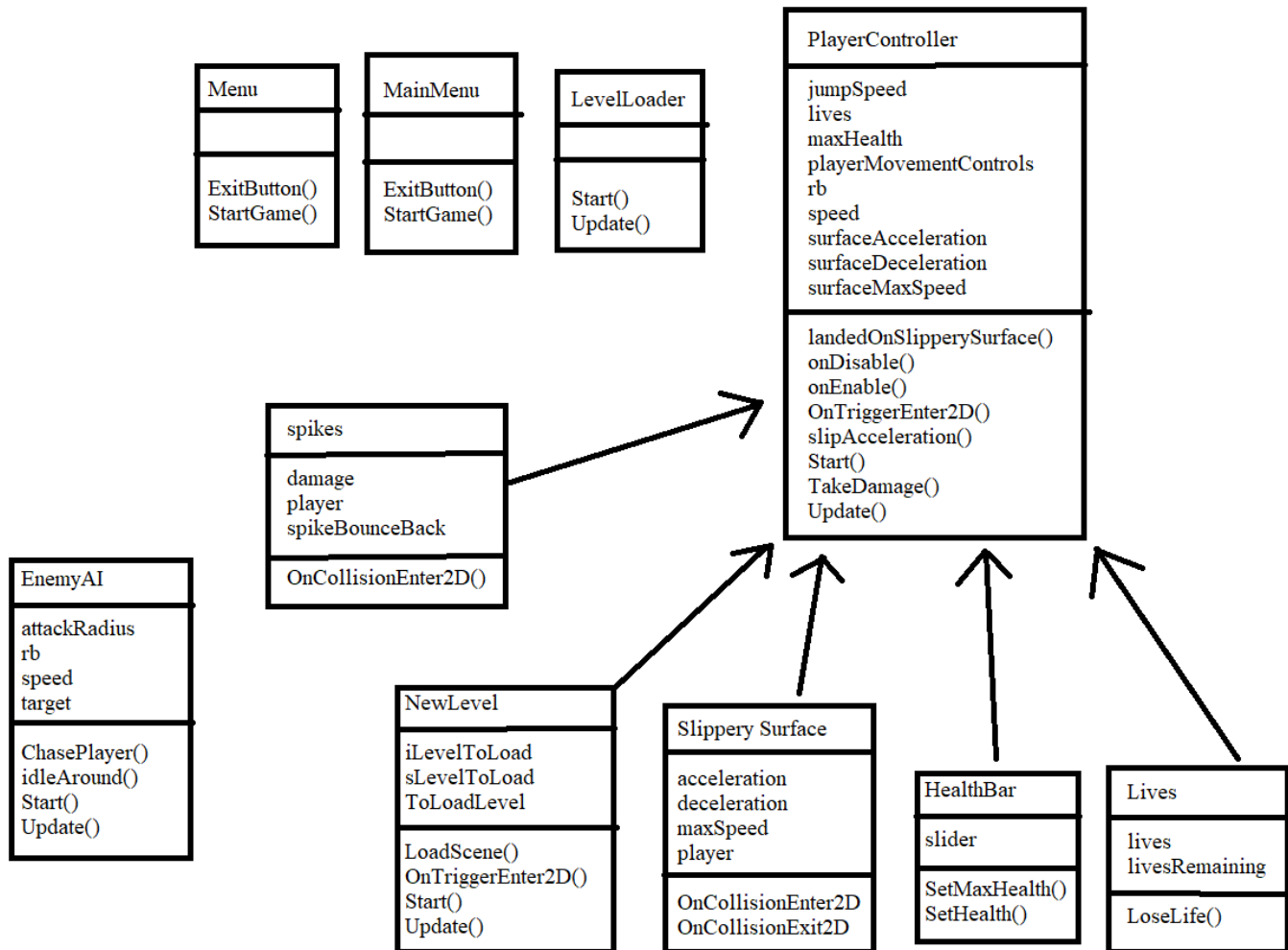
3. Non-functional Requirements (10 points)

1. The game should be easy to use with minimal instructions.
2. The game's frame rate should be decent and allow the game to run smoothly.
3. The game's response time should be high.
4. The game should run with a keyboard in any web browser.

4. Use Case Diagram (10 points)



5. Class Diagram and/or Sequence Diagrams (15 points)



6. Operating Environment (5 points)

We are coding in C# using Unity version 2020.3.19.

7. Assumptions and Dependencies (5 points)

Because we are using custom art, time constraints may affect how much custom art we actually

end up with. This mainly would affect the extra custom art we may end up needing, such as art for assets, backgrounds, enemies, and obstacles.