### Technical Design Document Template

1.0 Revision History

<As you revise the document, list what was changed and when it was changed>

|  |  |
| --- | --- |
| Version | Description |
| 1.0 | Initial document |

2.0 Development Environment

2.1 Game Engine

<Visual Studio 2019>

2.2 IDE

<Visual Studio 2019>

2.3 Source Control procedures

<Github>

2.4 Third Party Libraries

<Raylib>

2.5 Other Software

<2D Pixel art>

3.0 Game Overview

3.1 Technical Goals

<60fps, Working AI>

3.2 Game Objects and Logic

<A list of logical elements in the game, i.e. eat, drink, protect territory (stretch goal), claim more territory (stretch goal)>

3.3 Game Flow

<The player will simply watch a simulation of AI surviving. The player doesn’t do anything except watch, and the simulation ends when all but one AI is dead>

4.0 Mechanics

4.1 Definite Mechanics

- The AI moves for the closest food when it’s hungry

- The AI moves for the nearest source of water when it’s thirsty

- The AI wanders around when not eating or drinking

4.2 Stretch Goal Mechanics

- The AI has territory that they control

- The AI protects their territory and the food inside it

- The AI becomes stronger and braver after eating another AI

- The AI becomes more desperate when their territory shrinks due to food shortage

- The defending AI has an intimidation radius that makes other AI run unless they have enough bravery or desperation.

5.0 Graphics

<Top-down pixels>

6.0 Artificial Intelligence

<Describe how AI works, i.e. state machine, fuzzy logic, GOAP. Describe the various behaviours and how they change behaviour, how do the ‘creatures’ in the game evaluate the world> <include diagrams/flowcharts showing decision making processes>

7.0 Items

<List of items you can pick up that can affect the player, and what they will affect, like ‘picking up the hammer (refer collisions above) adds 5 to the players attack attribute’. Include details on how items influence gameplay or AI logic.>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Item | Parameter | Parameter | Parameter | Description |
| Default | 6 | 6 | 6 |  |
| Weapon | 5 | 7 | 7 |  |
| Weapon | 8 | 5 | 5 |  |
| Weapon | 5 | Possible 10 | NA | text |
| Weapon |  |  |  |  |

8.0 Game Flow

9.1 ‘Mission’ / ‘Level’ structure

<Are all levels stored in memory? what data is saved across levels, are levels loaded synchronously to prevent pauses?>

9.2 Objectives

<What does the player try to accomplish on each level/mission? How is the players progress evaluated?>

9.0 Levels

<If any of the Levels require specific behaviours, describe those here>

10.0 Interface

10.1 Menu

<What are the menu options and what do they do?>

10.2 Camera

<Describe the camera, how it moves, perspective/orthographic, can it switch? How? Does it need to render-to-texture? does it prevent itself going through walls, use flowcharts to document behaviour>

10.3 Controls

<Keyboard, tablet touch/swipe/tilt, joystick, mouse etc. record double taps, multi touch, use mouse smoothing/ scale mouse for aiming etc.>

14.0 Asset List

<List all files needed, along with known attributes >

16.0 Technical Risks

<if you want your game to be a 1000 player pvp battle royale with 4k 120fps graphics, you need to say if this is doable and how you intend to do it>