### Technical Design Document Forest Sim

1.0 Revision History

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

2.0 Development Environment

2.1 Game Engine

Proprietary

2.2 IDE

Visual Studio 2019

2.3 Source Control procedures

GitHub(desktop)

2.4 Third Party Libraries

Raylib

2.5 Other Software

Assets – enemy and player bugs(images), map

3.0 Game Overview

3.1 Technical Goals

2D graphics of the players and map

State Machines AI

3.2 Game Objects and Logic

Player character

Moves around, flees when the enemy is within range

Enemy character

Wanders around, chases the player when is within range

Map

3.3 Game Flow

The game is made up of 2 bugs, player bug and enemy bug. The player bug can be moved around using the arrow keys. The player is able to wrap through the screen. The enemy bug wanders around the screen in random direction. If the player gets close to the enemy, the enemy switches to attack and the player switches to flee state. The enemy follows the player until the player is out of reach.

4.0 Mechanics

Movement

The player is moved around using the arrow keys.

Flee

The player will flee once the enemy gets close.

Attack

The enemy will chase the player if in range.

Wander

The enemy wanders around to look for the player.

5.0 Graphics

The game is a top down 2D game with a forest map. The forest has water, scrubs and grass.

6.0 Artificial Intelligence

State Machine

Wander state - The agent roams around the world in random direction when on the wander state

Flee state - The agent runs away from the target if it is within range of the target.

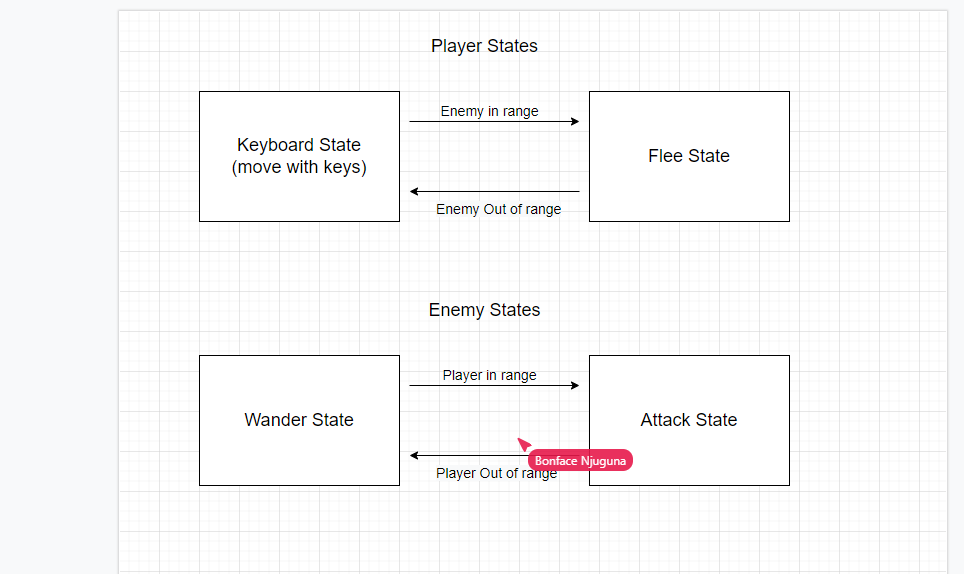
Idle state - The agent does nothing, stands still

Attack state - The agent runs to the target if it is within range.

The player agent is moved around with the arrow keys. If it is within range of 150px from the enemy agent it switches to flee state and automatically runs away from the enemy.

For the enemy agent, it wanders around the world seeking for its target. When the player is within range, it switches state to attack and chases the player until it is out of range.

Diagram



Dijkstra’s Algorithm

Is for finding the shortest paths between nodes in a graph. I used this pathfinding algorithm because it was the easiest to use.

How it works:

The agent has a start point, if a player clicks on a certain position on the map, that is passable, the agent will find the shortest path and move to that point. It also draws a line from one point to the next.

7.0 Interface

7.1 Camera

The camera is top down without movement

7.2 Controls

Arrow Keys – UP arrow for upward movement, DOWN arrow for downward movement, LEFT arrow for left movement and RIGHT arrow for moving to the right.

8.0 Asset List

Bugs.png – 21kb

Forest\_tiles.png – 106kb

9.0 Technical Risks

It is hard to implement a huge system say with 100 agents and enemies