### Technical Design Document Template

#### 1.0 Revision History

|  |  |
| --- | --- |
| Version | Description |
| 0.1 | add GameManager project and got a (semi)functional button working |
| 0.2 | Added raylib binaries |
| 0.3 | - added buttons to game states  - added pause and play states  - added textures |
| 0.4 | - functional camera  - added nodes  - put in map sprite |
| 0.5 | - added agents and behaviours  - nodes visability is now toggleable with the input 'DBM backspace' |
| 0.6 | - Implemented pathfinding  - release build created |

#### 2.0 Development Environment

**2.1 Game Engine**

Raylib

**2.2 IDE**

Visual Studio 2019

**2.3 Source Control procedures**

Git

**2.4 Third Party Libraries**

Raylib.dll

**2.5 Other Software**

N/A

#### 3.0 Game Overview

**3.1 Technical Goals**

- 2D top-down perspective

- Utilise basic shapes and minor sprite work

- Have an Agent seek to a point

- Have an Agent flee from another agent

- Have agents that harvest berries

**3.2 Game Objects and Logic**

- Farmer: Collects berries that spawn on the map

- Harvest Node: This holds resources and can be destroyed when nodes are lost, Resources are harvested at a rate.

- Heal Node: Heals agents in its range

**3.3 Game Flow**

This is game is actually a simulation (u g0t ba1ted). Basically, there are two agents that rome the field to collect berries. For every berry the agents pick up they earn one point, the agent with the most points is the winner. Some berries are poisonous and will hurt the agent, when the agents are at low hit points (HP) they will return home and heal. When the two agents get close to each other they will run away and when one agent steals a berry from another agent they will break down and cry about how unfair the world is and how they wish they lived in a world without stolen berries (I’m not joking I’m making this bug a feature).

Programming stuff to be considered includes:

* Minimising code
* Minimising files size
* Commenting \*
* Optimisation
* Organising code into correct classes

#### 4.0 Mechanics

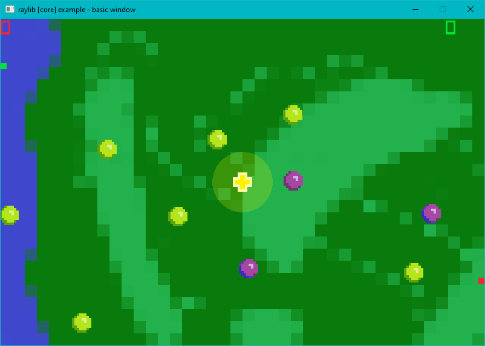
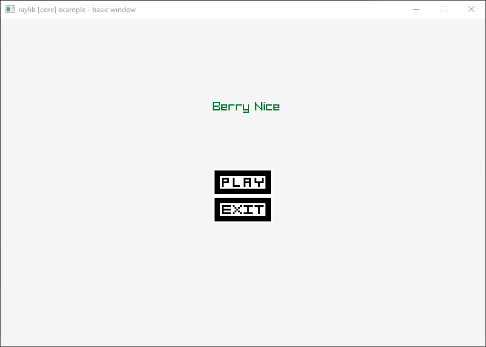
CORE GAME MECHANICS:

* Recourse harvesting
* Scoring system
* Health system
* Agent autonomy

#### 5.0 Graphics

Top-down 2D, minimal sprite work, quaint simplistic style, lots of colour.

Screen Shots:



#### 6.0 Artificial Intelligence

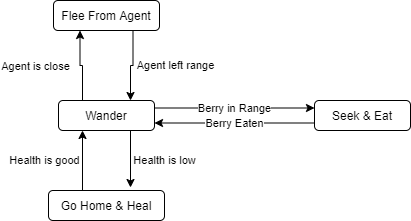
**Behaviour types:**

* **Seeking:** The agent has a target (Vector2\* {x,y} )and will seek toward it
* **Wandering:** The target has no specific direction and will apply force on itself in random directions. To make this feel natural I use a method revolving around a circle and the applied force needs to exist on the circumference of that circle.
* **Fleeing:** When approached by or approaching a target the agent will run in the exact opposite direction
* **Harvesting:** When interacting with an object that is harvestable, and the agent has the correct equipment then the agent will remove resources from the node and add it to their inventory. Resources that do not fit will create a pile of that specific resource.

**The Agents:**

The Tribes people will have the basic behaviours of:

* Feeing: from other agents
* Wandering: around the map
* Path following: To home
* Seek: to berries in sight



<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>

#### 7.0 Items/Resources

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Node | Resources | Storage | Tool | Description |
| Berry | Score  Health | 1  0 | Any  Any | This is merely a humble berry, doing its best for the community. |
| Poison-Berry | Score  Health | 1  -1 | Any  Any | You thought this was a normal berry, but it was a deception. |

#### 8.0 Game Flow

**8.1 ‘Mission’ / ‘Level’ structure**

This game only has the single level in it, that level is the great plains of oodle. Oodle is home to many wonderous creatures and resources. The berries that are saved in this level are the Harvest nodes (position and sprites), pathing nodes (position, cost to traverse), the players starting point and the healing zone (position, range, rate of healing and sprites).

**8.2 Objectives**

The agents objective is to collect as many berries as possible whilst keeping their HP above zero and avoiding each other.

#### 9.0 Interface

**9.1 Menu**

When the player runs the game they will be treated to a splash screen then to the “Main Menu”. The “Main Menu” has a title of the game as well as the play and exit buttons. The Play button transitions the player to the “Play State” and the Exit button closes the window. When hovered over the buttons colour is inverted.

The “Pause Menu” is accessed by pressing the ‘p’ key on the keyboard while in the “Play State”. In this state the game is un-interactable and all updates on the game are stopped. This “Pause Menu”

**9.2 Camera**

The camera is a top-down camera, overlooking the entire map.

**9.3 Controls**

* You can press the “p” button to pause the game

#### 10.0 Asset List

<-45->

<--------99-------->

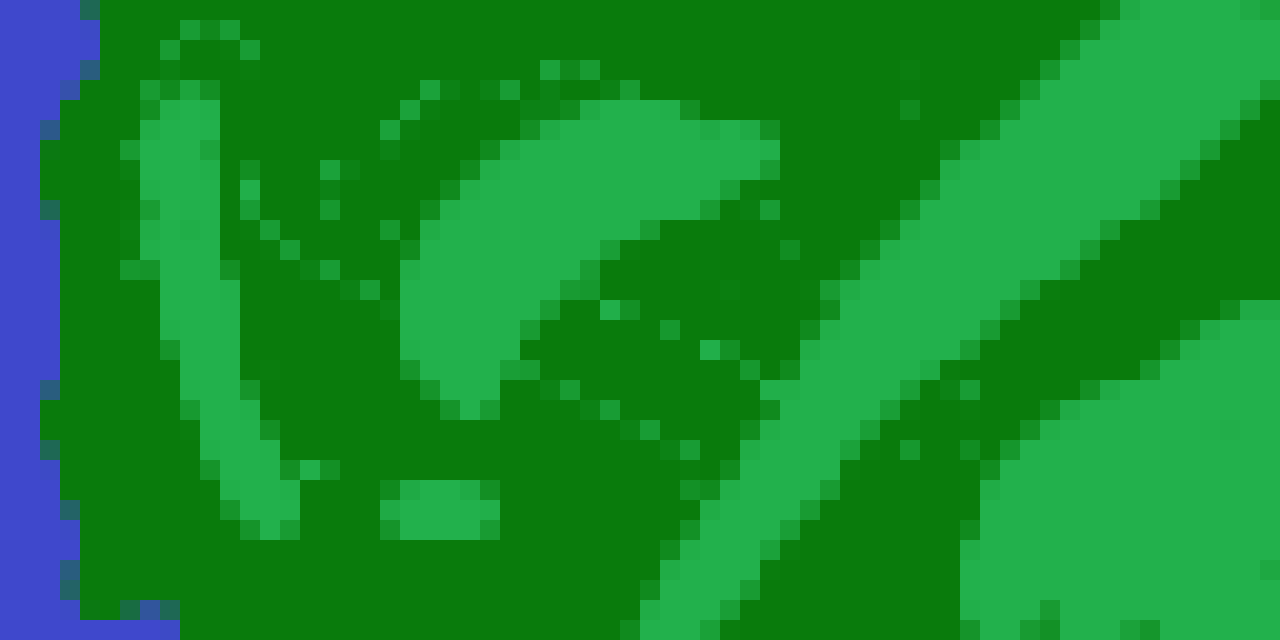
<-32->

Sprite Work: (all png’s)

<-32->







1000x640

#### 11.0 Technical Risks

In this game I want there to be:

* Resource harvesting
* Crafting
* Building
* Agent selecting
* AI Behaviours: (seek, peruse, flee, pathfind)
* Wave system
* Agents have different behaviours at different times
* Agents have different behaviours when equipping different tools