### ***Technical Design Document***

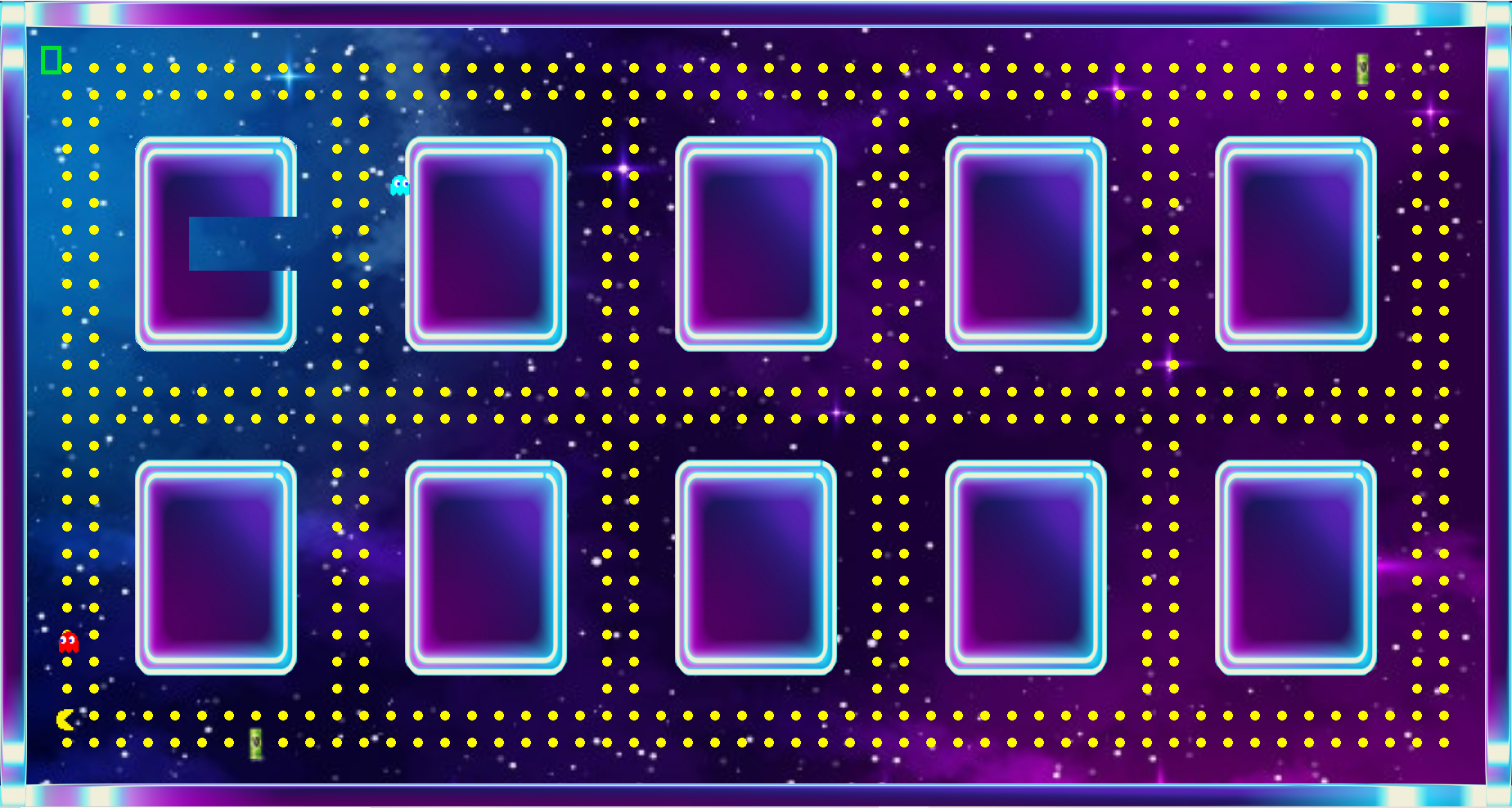
**Game Overview**

**Objectives**

Players objective is to eat as many of the yellow dots as possible to increase their score, whilst avoiding ghosts along the way who are trying to hunt the player down in various ways.

**Technical Goals**

* challenging AI that changes behaviour throughout the game based off AI environment.



**Game Objects and Logic**

**-Player**

* Player Is moved using W, A, S, D keys.
* Player controlled Behaviour whose purpose is to eat dots without being eaten by ghosts.
* Force controlled movement.

**- Red Ghost**

* Astar pathfinding towards Player.
* Flee Astar behaviour away from the player trying to get into his safe spot avoiding the player on its way.

**-Blue Ghost**

* Wander randomly around maze.
* Flee Astar behaviour away from the player trying to get into his safe spot avoiding the player on its way.
* Seek Astar behaviour towards player when player is in range and not powered up.

**-Items**

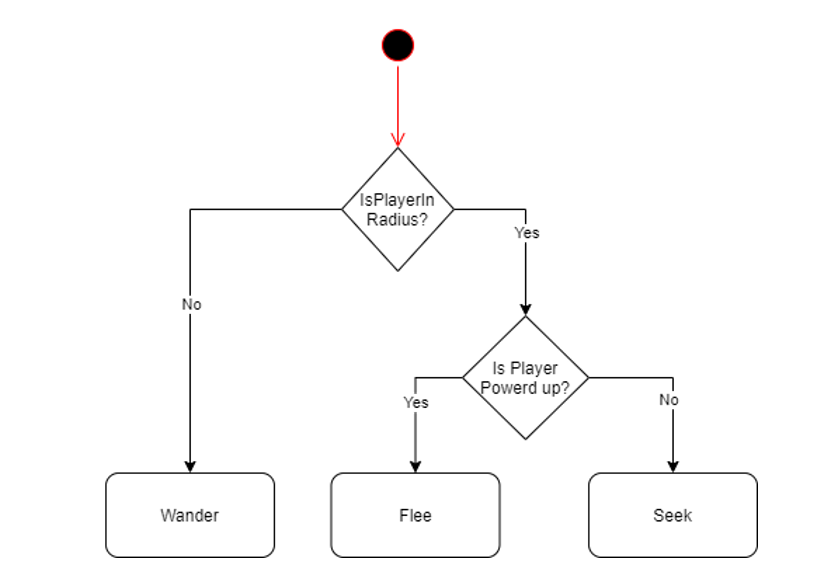
* The V can gives a power up to the player which changes the behaviour of all nearby ghosts to a flee behaviour. Player has the option to chase the ghosts instead for limited time gaining bonus score.

**-Walls**

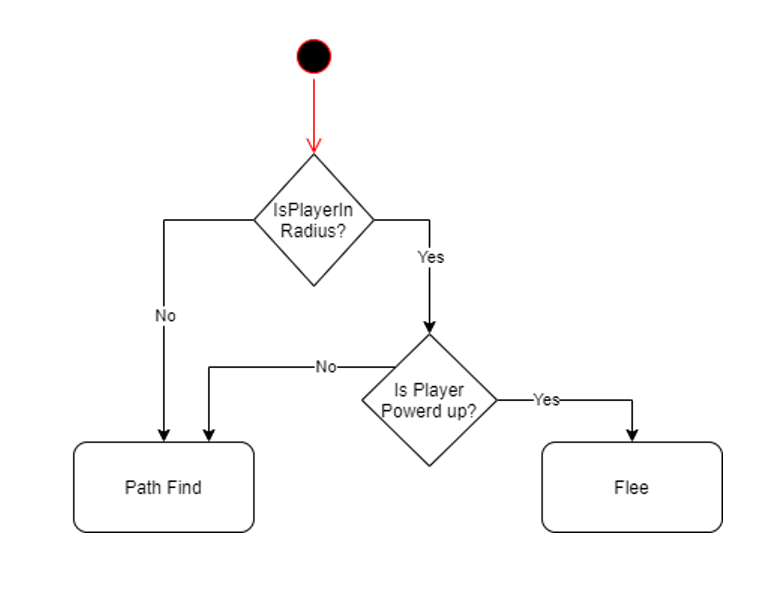
* Boundaries that player and enemy both cannot get past.

**Artificial Intelligence Flow Charts**

***BLUE GHOST***



***RED GHOST***



**Development Environment/Assets**

**-Asset List**

* Pacman art

<http://pixelartmaker.com/art/c5571d953a81efc>

* Wall art

<https://www.vectorstock.com/royalty-free-vector/lowing-electric-square-neon-lamp-vector-8378805>

* V can art

<https://shortysliquor.com.au/v-energy-drink-24-x-250ml-can>

**-Game Engine/Third party libraries**

* Raylib

**-IDE**

* Visual Studio

**-Source Control procedures**

* Git - SourceTree

**-Third Party Libraries**

* Raylib

**-Licences**

|  |  |  |
| --- | --- | --- |
| **Asset** | **Library file location** | **Licence** |
| Raylib | NA | MIT |

**Code suitability/Improvements that could have been made**

* Cases where could have used more get and set functions.
* Little use of const.
* Hard coded collision boxes.
* Hard coded images positions
* Could have improved collision resolution

**Technical impact**

* Searching of visited / nodes on stack could have been made more optimal in pathfinding algorithm.

**-Graphics**

* Top down 2D