

Computer Games Development SE607

Technical Design Document

Year IV

|  |  |
| --- | --- |
| Donal Howe  C00249662 | [Student Name]  [Student Number] |
| [Date of Submission] | |

[Declaration form to be attached]

**Table of contents**

1. Introduction …………………………………………………………………………. 2
2. Technical Design……………………………………………………………………. 2
   1. Helper Classes, Structs………………………………………………………... 2-3
   2. Utilized Classes ………………………………………………………………. 4-5
   3. Algorithm Classes ……………………………………………………………. 5-7
   4. UI and Game Class …………………………………………………………... 8
3. Class Diagram ……………………………………………………………………. 9
4. CRC Cards …………………………………………………………………….... 9-12
5. References ………………………………………………………………………. 12

# 

1. **Introduction**

The objective of this project is to compare the benefits and drawbacks of using commonly used heuristic based guided pathfinding algorithms to the incremental algorithm known as Dstar Lite. This project will discuss the direct benefits of each algorithm in depth, from Astar, Dijkstras search algorithm, Lifelong Planning Astar, and the non-guided algorithm known as Depth First Search when compared to D star Lite within a game’s context.

**2. Technical Design**

The purpose of this document is to effectively communicate the technical details and design decisions of the system/algorithm to the readers.

It could include software architecture, algorithm design, class specifications, pseudo code, etc. with tools such as UML, Class Diagram, CRC Cards.

**2 .1 Class Snippets: Header File**

**Helper classes, Structs:**

|  |  |
| --- | --- |
|  | The Enum class called “WhichAlgorithm” which controls which algorithm is being used a certain time. This Enum class contains the name for every pathfinding algorithm in the project. |
|  | The Enum class called “GridSize” controls the size of the grid which is being used in the program ranging from “small” to “very Large”.  “small” = “10x10” grid  “Large” “50x50” grid  “very Large” = “100x100” grid |
|  | This Enum class called “Race” depicts whether you want to race the algorithms in comparison to dstar lite on a chosen path |
|  | The Enum class called “debug” toggles whether the user wants to see the variable values for Dstar Lite on the screen. This is only available with the small grid size |
|  | The struct called “Screen Size” struct which controls the size of each window |
|  | The Enum class called “Mode” which controls which mode the application is in. behaving differently depending on which one it is in |

**2.2 Utilised Class Snippets:**

|  |  |
| --- | --- |
|  | Public variables of Cell Class  the header file for the Cell(node) which has all the current functions in use.  Private variables to Cell Class |

|  |  |
| --- | --- |
|  | private member variables of the grid class  public members of the Grid class |

**2.3, 2.4 Algorithm’s Classes: + UI and Game Class**

|  |  |
| --- | --- |
|  | Functor used in Dstar Lite  Private members of the Dstar Lite Class  Public members of the Dstar Lite Class |

|  |  |
| --- | --- |
|  | Functor used in “Astar”.  “Astar” class as declared in the header file |
|  | Functor used in “LpaStar”  “LpaStar” class as declared in the header file |
|  | Functor used in “Dijkstra’s” search algorithm.  “Dijkstra’s” Class as declared in the header file |
|  | “Depth First Search” Class as declared in the header file. |
|  | “Menu” class as declared in the header file. |
|  | Public members to class “Game”, as declared in the header file.  Private members to class “Game”, as declared in the header file. |

1. **Class Diagram:**

|  |
| --- |
| Diagram  Description automatically generated |

# CRC Cards:

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Table

Description automatically generated with medium confidence

Graphical user interface, text, application

Description automatically generated with medium confidence

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text

Description automatically generated

Table

Description automatically generated with low confidence

Table

Description automatically generated with medium confidence

Text

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Table

Description automatically generated with medium confidence

Table

Description automatically generated

Text, table

Description automatically generated with medium confidence

A picture containing graphical user interface

Description automatically generated

# References

<https://core.ac.uk/download/pdf/235050716.pdf> - Path Planning Algorithm using D\* Heuristic Method Based on PSO in Dynamic Environment Firas A. Raheema \*, Umniah I. Hameedb

<https://medium.com/@nicholas.w.swift/easy-a-star-pathfinding-7e6689c7f7b2> -Nicholas Swift Feb 27 2017

<http://www.cs.cmu.edu/~ggordon/likhachev-etal.anytime-dstar.pdf> - Maxim Likhachev† , Dave Ferguson† , Geoff Gordon† , Anthony Stentz† , and Sebastian Thrun‡

<https://www.ri.cmu.edu/pub_files/pub3/stentz_anthony__tony__1994_2/stentz_anthony__tony__1994_2.pdf> -Anthony Stentz

Koenig, S. and Likhachev, M. (n.d.). *D\* Lite*. [online] Available at: http://idm- lab.org/bib/abstracts/papers/aaai02b.pdf

encyclopedia.pub. (n.d.). *Jump Point Search Algorithm*. [online] Available at: <https://encyclopedia.pub/entry/24246>