Software Requirements Specification

Cat & Mouse Game

Version 1.0

1/12/23

Table of Contents

1.Introduction

1.1 Purpose
1.2 Scope
1.3 Definitions, Acronyms & Abbreviations

2.Overall Description

2.1 Game perspective
2.1.1 ‘Cat’ Perspective
2.1.2 ‘Mouse’ Perspective
2.2 Game System

3.Interface

3.1 Interface screen
3.2 Controls for the game

References

1. Introduction

1.1 Purpose

This document specifies and lays out the assumptions, constraints, and requirements of the cat & mouse game. This document is intended to be used as a reference by the current project developers, and the users.

1.2 Scope

The scope of this system is to build a game wherein the graphs are generated via a random algorithm discussed in the report [1]. The graphs are generated with complete randomness and by ensuring that the probability of both the agents of the game namely the ‘cat’ and the ‘mouse’ have a probability of winning nearly equal to half.

The user inputs will be on the parameter ‘number of nodes’ as a result of which the map of the game will be generated with that many number of nodes. It will be using an aspect of adversarial searching method wherein the opponent will be using the [2] algorithm to try and beat the player. In this game players can choose to either become the ‘cat’ or the ‘mouse’ and accordingly the [2] algorithm will work. The system won’t support saving games.

1.3 Definitions, Acronyms & Abbreviations

UI - user interface
AI - artificial intelligence
Pygame - Library in python for making games

Sprite - the art that are used in the game

2. Overall Description

2.1 Game Perspective

The users when entering the game will be asked how many nodes they want and upon giving a number the map is generated according to the algorithm discussed in [1]. Then the player is given a choice to either play as the ‘cat’ or the ‘mouse’ where the conditions are different for both the entities.

2.1.1 ‘Cat’ Perspective

In the ‘cat’ perspective the player’s main objective is to catch the ‘mouse’ entity which is an AI that works on [2] algorithm. The ‘cat’ entity can move to any adjacent node on the map. The map will be generated in such a way that the probability of the ‘mouse’ winning will be close to 0.5.

2.1.2 ‘Mouse’ Perspective

In the ‘mouse’ perspective the player’s main objective is to run away from the ‘cat’ entity till a set number of moves after which there will be a doorway on one of the nodes in the map , upon reaching which the ‘mouse’ will be declared the winner. In this mode the ‘cat’ entity will use [2] algorithm for catching the ‘mouse’ entity.

The restriction on this entity is that it can traverse the adjacent nodes but it cannot go back to the node where it came from i.e it can’t go back to the node from where it reached the present node.

2.2 Game System

The system is developed so that the player will give the first two inputs once which are as follows: a) number of nodes and b) ‘cat’/’mouse’ mode. Thereafter the input and output to the player will be displayed on the UI where it will be constantly updated.

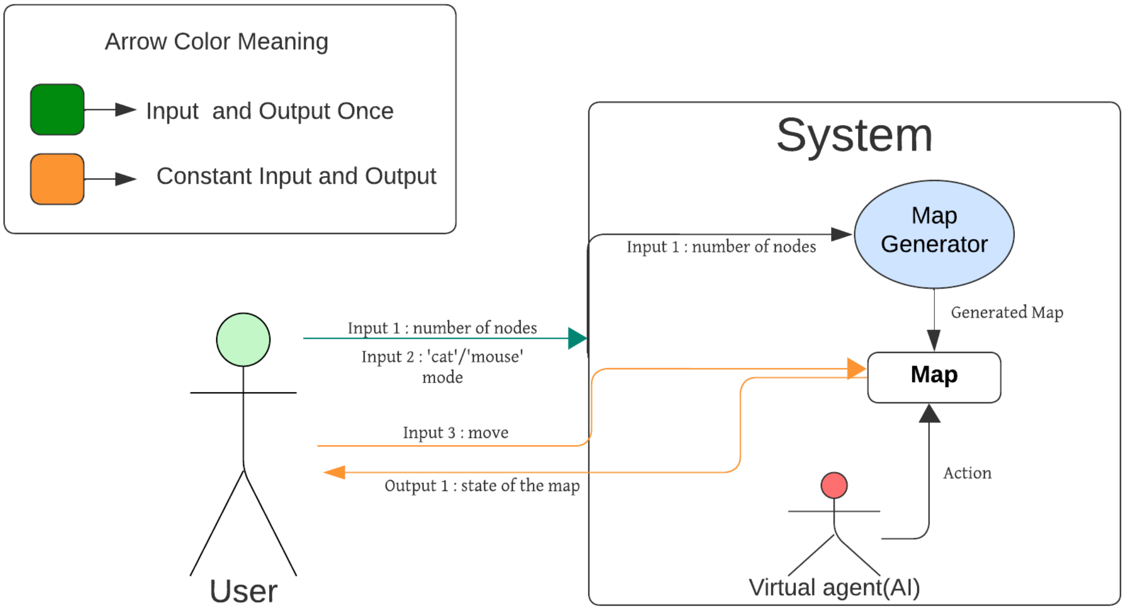

Arrow Color Meaning
Input and Output Once
Constant Input and Output
System
Map Generator
Input 1 : number of nodes
Generated Map
Map
Input 1 : number of nodes
Input 2 : 'cat'/'mouse' mode
Input 3 : move
Output 1 : state of the map
Action
Virtual agent(AI)
User

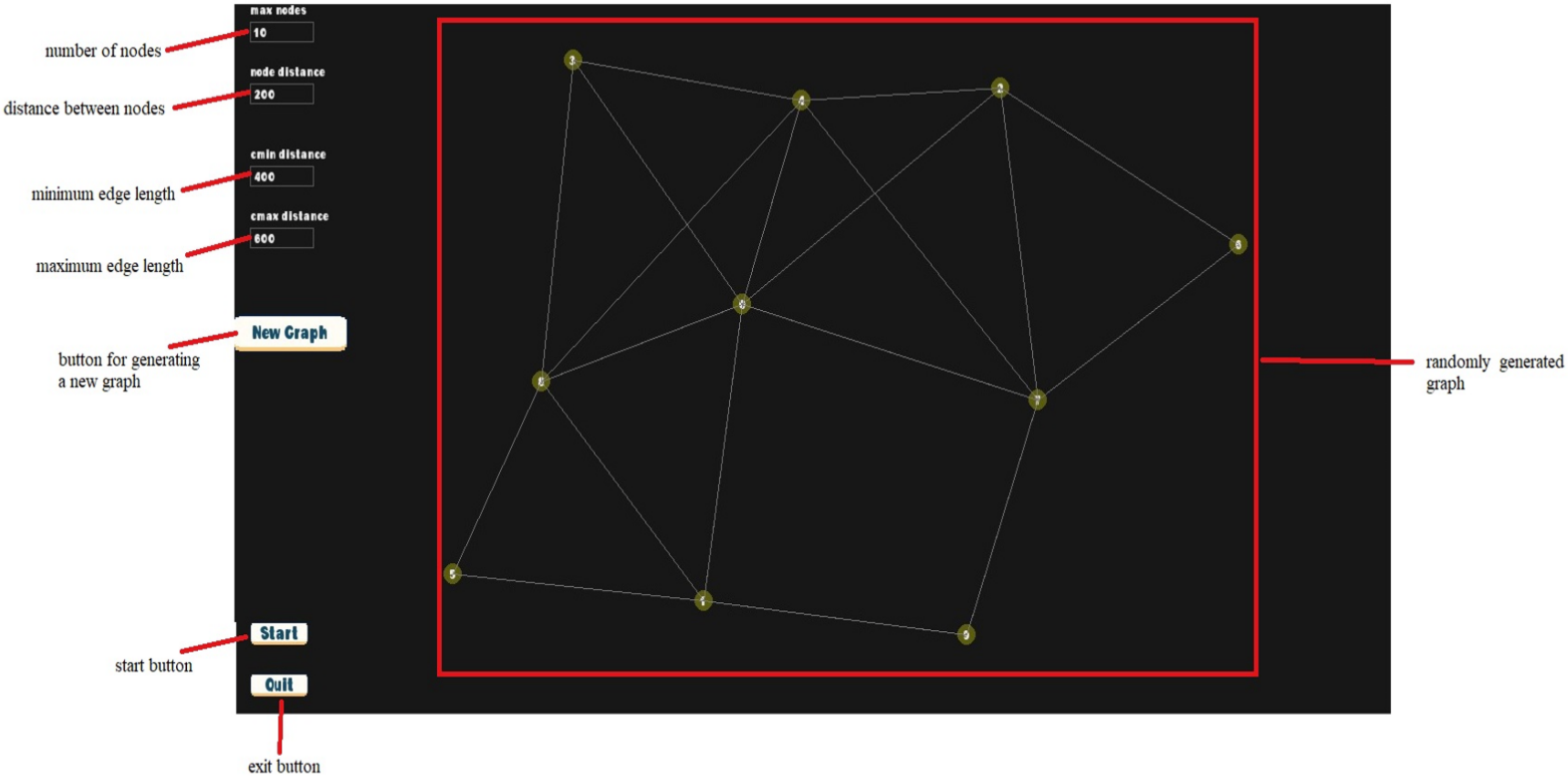
Diagram 1. Interaction of user with the system

The game will continuously update the environment according to the player’s decision and the AI’s decision and show the updated map in every instance to the user.

3. Interface

3.1 Interface Screens

The game has 2 interface screens, one is the home screen and the other is the game screen.



max nodes: 10

node distance: 200

cmin distance: 400

cmax distance: 600

number of nodes

distance between nodes

minimum edge length

maximum edge length

button for generating a new graph

start button

exit button

randomly generated graph

Diagram 2. Home screen

The home screen has these following details : -
i) box for entering number of nodes
ii) box for entering node distance
iii) box for entering minimum edge distance
iv) box for entering maximum edge distance

v) button for generating the new graph
vi) start game button
vii) exit button
viii) randomly generated graph

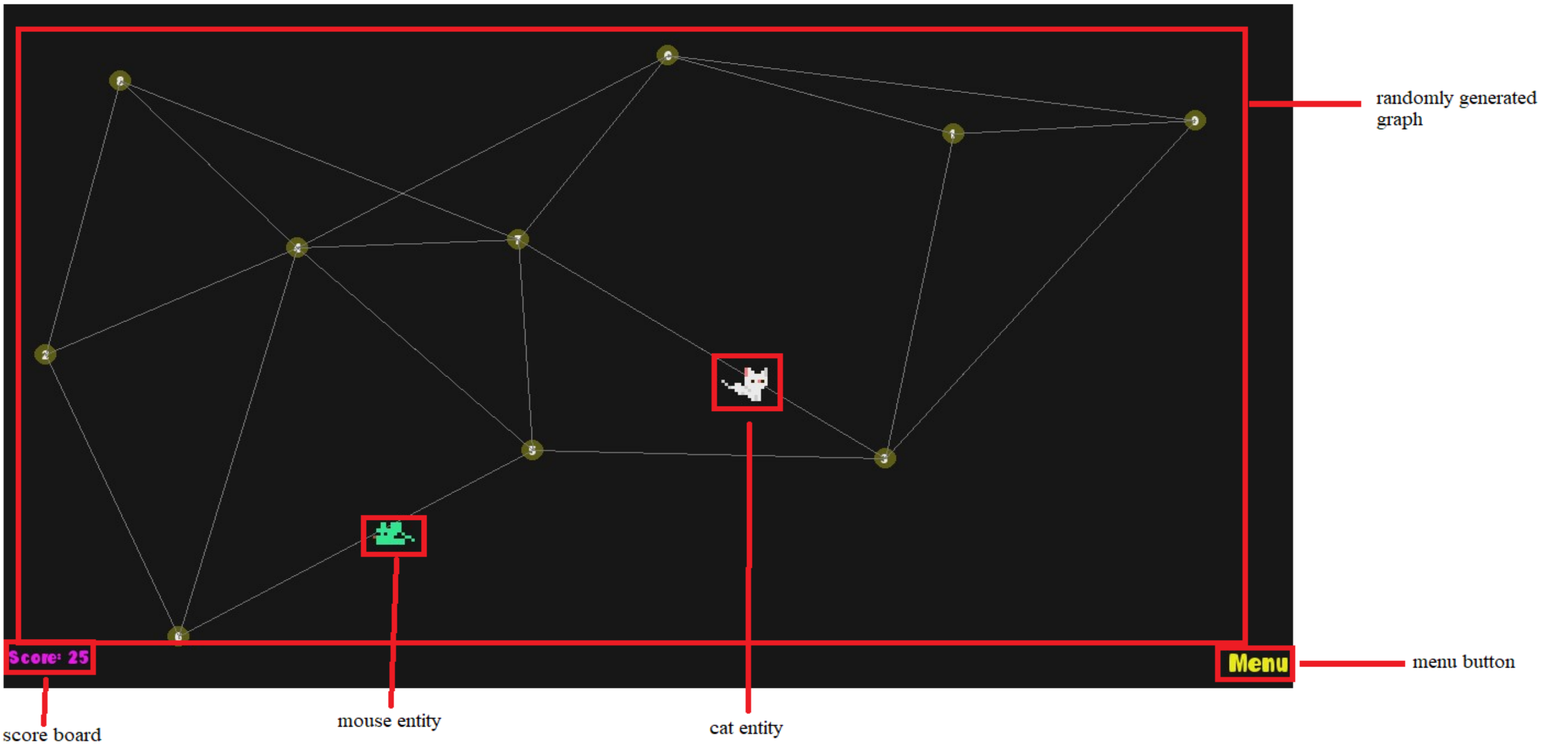


Diagram 3. Game screen

The game screen contains the following details : -

i) randomly generated graph
ii) score board
iii) mouse entity
iv) cat entity
v) menu button

3.2 Controls for the game

The game controls are only with the mouse button. MouseClick1 is used for clicking on the nodes where we want our entity to be led.

References

[1] Our Report
[2] https://en.wikipedia.org/wiki/Alpha%E2%80%93beta_pruning