

# **“Counter Strike AutoPlay Simulator”**

***Mid-Semester Report of  
7<sup>th</sup> Semester Mini Project***

*FOR THE DEGREE OF*  
**BACHELOR OF TECHNOLOGY**  
*IN*  
**INFORMATION TECHNOLOGY**

*BY*

Utkarsh Gupta (LIT2016009)

**UNDER THE SUPERVISION OF**

*DR. VISHAL KRISHNA SINGH*

**INDIAN INSTITUTE OF INFORMATION TECHNOLOGY,  
LUCKNOW**

**September, 2019**

**Supervisor's Signature: \_\_\_\_\_**

# **Abstract**

In order to simulate the computer game named Counter Strike - Global Offensive using Object – Oriented – Methodology ( O.O.M. ), to model the players, their strategies as JAVA classes, encapsulating features and actions of players as member attributes and member functions of JAVA class respectively.

# **DECLARATION**

We hereby declare that the work presented in this project report entitled "**Counter Strike – AutoPlay Simulator**", submitted as mid-semester report of B.Tech. (IT) at Indian Institute of Information Technology, Lucknow, is an authenticated record of our original work being carried out from August 2019 to November 2019 under the guidance of **Dr. Vishal Krishna Singh**. Due acknowledgements has been made in the text to all other material used. The project was done in full compliance with the requirements and constraints of the prescribed curriculum.

Place: Allahabad  
Date: 21 September 2019

Utkarsh  
Gupta (LIT2016009)

# **CERTIFICATE**

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Date: 21 September 2019

Dr. Vishal Krishna Singh

Place: Allahabad

IIIT-Lucknow

# **ACKNOWLEDGEMENT**

This satisfaction that accompanies the successful completion of any task would be incomplete without the mention of few people whose ceaseless cooperation made it possible; whose constant guidance ,inspiration and constructive suggestions that were helpful to me and my group in the preparation of this project.

A very special thanks to **Dr. Vishal Krishna Singh**, our project supervisor, for motivating and inspiring all of us. We are grateful to his for the strong support, guidance, inspiration and constructive suggestions that helped us a lot in the preparation of this project.

# **TABLE OF CONTENTS**

- **Problem Definition**
- **Objectives**
- **Proposed Approach**
- **Class Diagram**
- **Software Requirement**

# **PROBLEM DEFINITION**

There are two teams: "Terrorists" and "Counter-Terrorists". The aim of the terrorists is to go to a special pre-determined site called as "Bomb". The aim of the counter-terrorists is to ensure that none of the terrorists can go to the site called "Bomb". Because playing this game manually is very boring, you have an AI Engine that automatically plays this game for you. There are three kinds of players in both teams possible:

- AggressivePlayers, who tend to run fast. Their energy level reduces by 2 at every step of move. They need to be hit twice to be dead.
- CautiousPlayers, who tend to go very slow. Their energy level reduces by 1 at every step of move. They need to be hit once to be dead.
- BlindPlayer, who run very fast, and do not observe around at all. Their energy level reduces by 3 at every step. They need to be hit 5 times to be dead.

Each player of counter-terrorists selects an opponent player and goes to kill the same. A player can have any of three strategies. It is assumed that everyone knows each other's position.

- Go to nearest terrorist
- Go to a random terrorist
- Go to a terrorist 'ahead' in the map

A terrorist has all the strategies of the counter-terrorists, with one additional strategy: Go to bomb. Any number of terrorists can select any counter-terrorist and vice versa. A terrorist can be the target of any number of counter-terrorist and vice versa. A bomb may be aimed by any number of terrorists. The strategies are constant, however the selected opponent will change as the players move. So the nearest terrorist strategy followed by a player will remain as it is, however the specific terrorist will change as the terrorists move around.

The game is sequential in nature. All players make a move one after the other. Hence the order in which the players move can be critical to the game. The order may be circular (one chance to every player in the same order by which they entered the arena), by energy level (most fit player moves first), by success (the player who killed the maximum opponents moves first). However, first a terrorist moves, then a counter- terrorist, then a terrorist and so on, till both the sets expire, and a new turn starts. There is a single order.



# **OBJECTIVE**

To develop a simulator of the computer game named Counter Strike - Global Offensive using Object – Oriented – Methodology ( O.O.M. ), to model the players, their strategies as JAVA classes, encapsulating features and actions of players as member attributes and member functions of JAVA class respectively.

# PROPOSED APPROACH

Use Object – Oriented – Methodology ( O.O.M. ), to model the players, their strategies as JAVA classes, encapsulating features and actions of players as member attributes and member functions of JAVA class respectively.

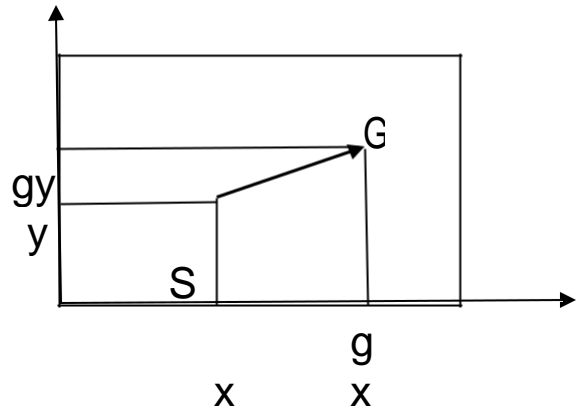
## **AIMove**

Every player has a position in X axis, a position in Y axis and an orientation. The orientation is an angle state governs where the person is looking at ( $\theta$ ).

Suppose the current position is  $S(x,y)$ . The player is moving at a speed  $s$  towards a goal at  $G(gx,gy)$ . The position at the next time step is given by:

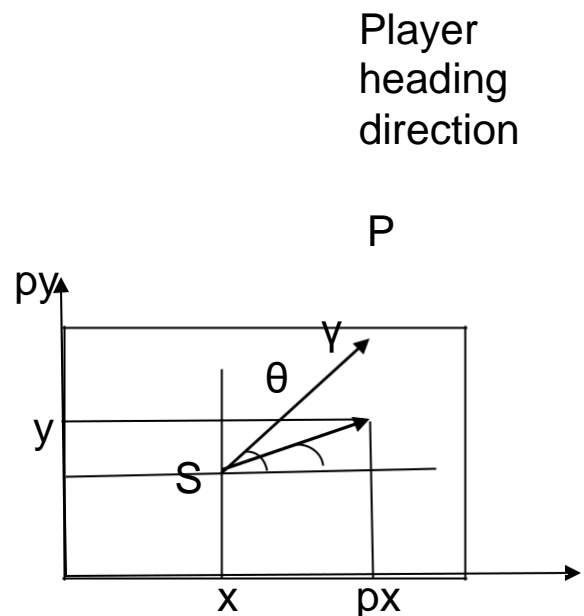
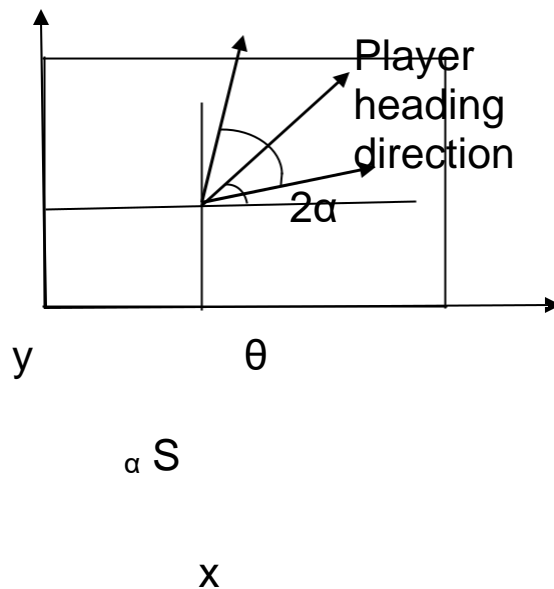
$$(x',y')=(x,y)+s(gx-x,gy-y)/\sqrt{(gx-x)^2+(gy-y)^2}$$

The proof is simple. Consider the vector  $G-S$ , that is  $(gx-x,gy-y)$ . A unit vector in the same direction is  $(gx-x,gy-y)/\sqrt{(gx-x)^2+(gy-y)^2}$ , while a vector at a distance of  $s$  from  $S$  is given by the formula.



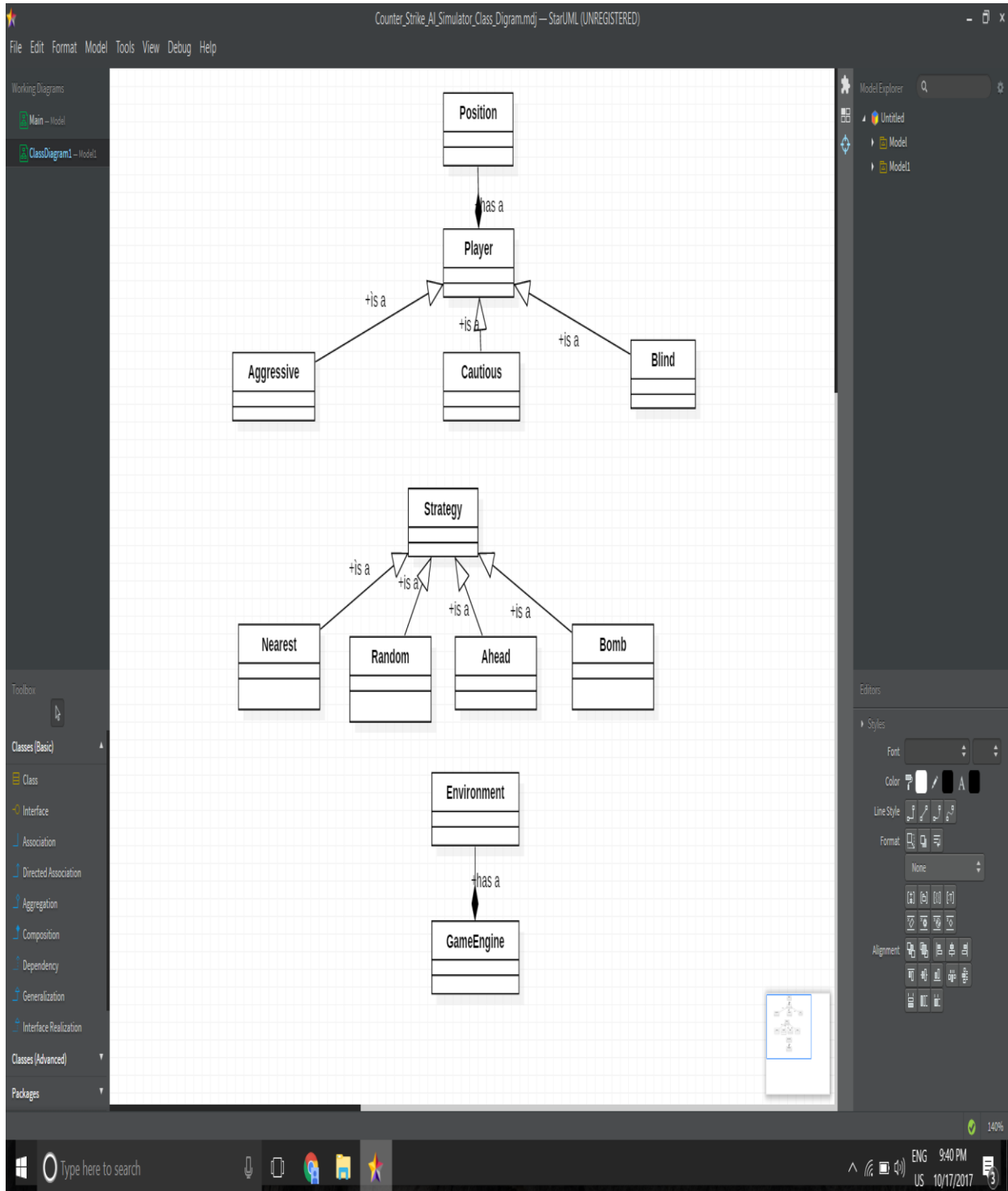
## site()

The function checks if two players are in line of sight to each other. Assume each player can see  $\alpha$  radians around the current orientation ( $\theta$ ). The person is facing at an angle of  $\theta$ , while can look around the angle of  $\pm \alpha$  from the current orientation. The angle  $\alpha$  is different for different players. The angular range of view of the person is hence in the range  $\theta - \alpha$  to  $\theta + \alpha$ .



The angle subtended by a new person at  $(px, py)$  is  $\gamma = \text{atan2}(py - y, px - x)$ . The person is in line of sight if  $\theta - \alpha < \gamma < \theta + \alpha$ . However since angles have a circular property the inequality cannot be directly used. The angle between the heading direction and line SP is given by  $\theta - \gamma$ . Hence for angular coverage,  $\cos(\theta - \gamma) > \cos(\alpha)$ .

The 'ahead' is simply taking  $\alpha = 60$  degrees



# **SOFTWARE REQUIREMENT**

## **Programming Tools:**

Object Oriented Methodology  
Netbeans  
Star UML

## **Programming Language:**

JAVA

## **SUGGESTIONS OF BOARD MEMBERS**



