AT Task 2 Report AI Squad Controller

Ryan Manning
18026856

University of the West of England

April 27, 2021

This report outlines the implementation of a AI squad controller

1 Introduction

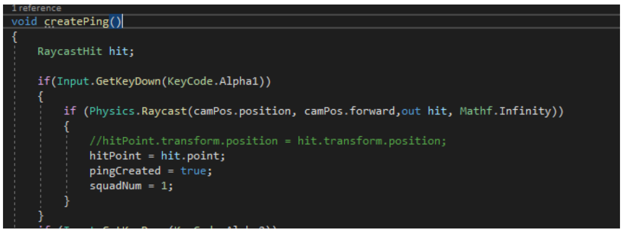
This task was to create a squad AI Controller. The input for this needs to be as minimal as possible but it should give the user full control of the squad still. For this system i want to have a squad of 4 and the player. Some of the actions that i wanted to have in this system is changing the formation that the squad is in. Also some sort of system for the squad to protect each other when they are low.

2 Related Work

In this paper Millington and Funge, 2009 it gives a description of AI in games and different types of AI this is useful to get the grips with how I'm going to make my system. This book Bourg and Seemann, 2004 also does something similar. Some games have really good squad system in them one of these games is Spec Ops the line Keogh, 2013. This game is a single player game where you play with a squad that you can control. In the game you can tell different members of the squad to target certain enemies. This is something id like to add into my system as it will allow a player to focus on different enemies during combat.

3 Method

Firstly i wanted to set up the ping system that will allow the player to press a button and a ping to be created where one or all the squad move to that location. To do this i started by setting up a raycast that comes from the camera to the floor figure 1. This then gives me a position to move the NPC to using set destination.



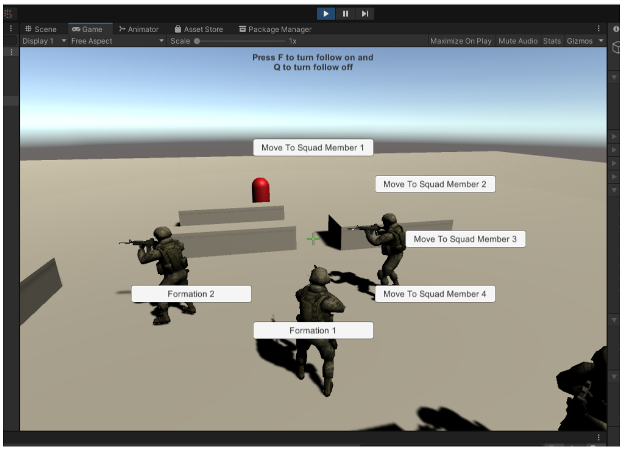
```
void createPing()
{
    RaycastHit hit;

    if(Input.GetKeyDown(KeyCode.Alpha1))
    {
        if (Physics.Raycast(camPos.position, camPos.forward,out hit, Mathf.Infinity))
        {
            //hitPoint.transform.position = hit.transform.position;
            hitPoint = hit.point;
            pingCreated = true;
            squadNum = 1;
        }
    }
```

Figure 1: *Code for the ping system*

Next was creating a way for the squad to stay in a formation to do this i created empty gameobjects and parented them to the player this would mean that all i need to do is to just use the set destination function to constantly move the squad to those gameobjects. This meant that i could just have this on at all times and then later add a button that allows the player to control which state they want to use the squad in.

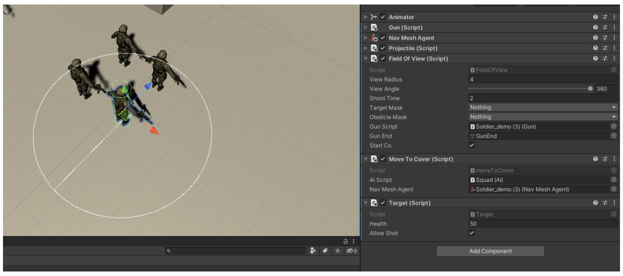
Once this was done i decided to add in some simple UI into the system to allow the player to switch the squad formations and some buttons that controlled the solo move to that i mentioned before. As well as this a crosshair was added to the middle of the screen as it felt hard to aim the move to without it .I did this as i felt that binding all of this to different keys could be too much and a menu wheel UI would allow this to be easier to use figure 2.

Next was adding in a field of view system for both the enemies and the squad members and then actually getting them to automatically to shoot at each other.First was to implement the field of view script. This script



Press F to turn follow on and
Q to turn follow off

Figure 2: *Example of the UI that is in the system*



Script FieldOfView
View Radius 4
View Angle 360
Shoot Time 2
Target Mask Nothing
Obsticle Mask Nothing
Gun Script Soldier_demo (3) (Gun)
Gun End GunEnd
Start Co ✓

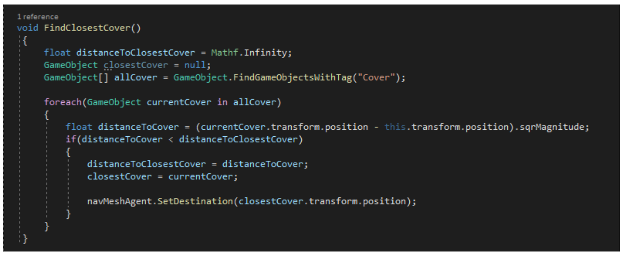
Script moveToCover
Ai Script Squad (AI)
Nav Mesh Agent Soldier_demo (3) (Nav Mesh Agent)

Script Target
Health 50
Allow Shot ✓

Figure 3: *Example of the range of the Field Of View script*

works by using physics.overlapSphere to find which "targets" is is within the range. Once I've found all the targets that's in range i raycast to check the if the enemy has the correct component to be attacked by the squad figure 3. Then the script shoots at the target and applies all the correct damages. Each object that can be shot at has the target script this script carries all the health values and deals with when that objects health reaches below 0. All this works without input so the squad will shoot at any enemies that walks into the radius of the squad members. This also works for the enemies shooting back at each of the squad member as well. Also this system was done using an coroutine this was because other wise the squad member will fire the enemy as fast as possible this way means that there is some delay between each shot.

Finally i added a heal system to the squad members. This system works by adding a healing function in for the squad where when they are out of combat for a certain amount of time they will slowly heal back to full hp. This is done by just storing their health value and then comparing that value after two seconds have passed. This links in with another behaviour that was added in which when a squad member falls below twenty five percent health they will find the nearest cover point and attempt to heal. This works by setting up an array which holds all the gameobjects with the tag cover. Then using foreach to loop through and find the closest piece of cover to that squad member. Once this has been found then move the that squad member to that position using the SetDestination function figure 4. Then the function uses the same system as the healing to check if the squad member has got to full health or not.Once they have they will join back to the squad in formation.



```
1 reference
void FindClosestCover()
{
    float distanceToClosestCover = Mathf.Infinity;
    GameObject closestCover = null;
    GameObject[] allCover = GameObject.FindGameObjectsWithTag("Cover");

    foreach(GameObject currentCover in allCover)
    {
        float distanceToCover = (currentCover.transform.position - this.transform.position).sqrMagnitude;
        if(distanceToCover < distanceToClosestCover)
        {
            distanceToClosestCover = distanceToCover;
            closestCover = currentCover;

            navMeshAgent.SetDestination(closestCover.transform.position);
        }
    }
}
```

Figure 4: *Find cover script which moves the squad member to the closest bit of cover*

4 Evaluation

My prototype is good as it allows the player to control the squad into different formations and allow the squad to shoot at an enemy. As well as this my squad moves to the nearest cover when they are below 25 percent health. However this isn't a great implementation as the squad member currently moves to the nearest bit of cover this doesn't take into account the position of the enemies. This would be something that i could improve on in the system. As well as this reducing the input of the ping system would be good as at the moment the system uses a UI pop up allowing the user to chose which action they want a squad member to do. This could be simplified by using something like voice commands or creating an AI that picks positions automatically based on the enemies positions in the level. However this squad controller system is good as it only has a couple of issues with it.

5 Conclusion

I created a squad controller system that allows the player to control the formation that the squad is in and where the squad are following the player in that formation or not. Also i made it possible for the player to individually move each member of the squad to any position. As well as this I made all the squad members have the ability to shoot using a field of view script that can be changed in the inspector. Finally i made a cover system that moves low health squad players into nearby cover.

Bibliography

Bourg, David M and Glenn Seemann (2004). "AI for game developers". In: *AI for Game Developers*. Vol. 6. 1. O'Reilly Media, Inc.

Keogh, Brendan (2013). "SpecOps AI". In: *Spec Ops: The Line's Conventional Subversion of the Military Shooter*. Vol. 1. 1. School of Media and Communication RMIT University Melbourne, Australia, pp. 1–17.

Millington, Ian and John Funge (2009). "AI In Games". In: *Aritfical Intellegence In Games*. Vol. 1. 2. CRC Press.