

---

# AI Sub-Commanders

**Calvin Oktavialdy Setia**

**Department of Computer Science and Creative Technologies**

University of the West of England

Coldharbour Lane

Bristol, UK

[Calvin2.Setia@live.uwe.ac.uk](mailto:Calvin2.Setia@live.uwe.ac.uk)

## Abstract

In this report, we discuss how we can use Artificial Intelligence (AI) to help players of a tower defence game to decide what, where, and whether to upgrade towers. Players can press a button and request the AI's suggestion. The AI will then use the current state of the game to generate a proper response to the player's request.

## Author Keywords

AI; Tower Defense;

## Introduction

With the popularity of tower defence games, more and more people are playing the genre. As not all players have the same skill level when playing the game, a helper could be used to assist lower skilled players to play the game. This project aims to create an AI helper to assist players with their decision making in a tower defence game.

The demo would have a base the player has to protect, multiple different types of enemies that spawns and tries to attack the base, camera control to help the player move throughout the scene, and the ability for the player to place towers on the scene.

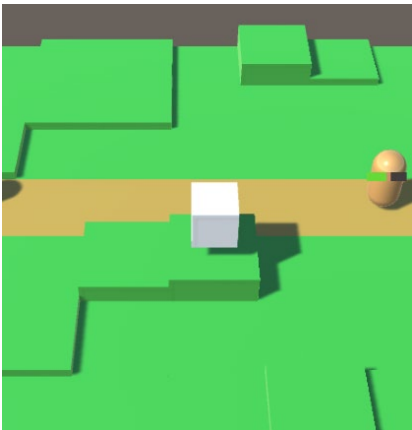
## Background Research

The background research for this project mainly focuses on other tower defence games, and how they





**Figure 1.** Ground blocks have different heights to get player to strategise on tower placements.



**Figure 2.** Indicators are used to let player know their selections.

main inspiration for the game this project is made in is Rogue Tower, (Die of Death Games, 2022). Other games that affect the decisions in the game include Bloons TD 6 (Ninja Kiwi, 2018) and Arknights (Hypergryph, 2019).

### Outcomes Scene

As the project focuses on how AI can help assist players in a tower defence game, a basic tower defence scene needs to be made. Made first in the project, the scene provides basic tower defence functionality.

The map is made of blocks that are grouped into chunks. The blocks have three types; ground blocks, and road blocks. Ground blocks are the blocks where towers are placed, while road blocks are the blocks where enemies walk. At the start of the game, each ground block has a chance to increase its height up to three times. Towers placed on higher blocks have increased range, allowing for players to strategise with the placement of their towers.

Players can place towers on the map, by clicking on a block. When doing so, the tower controller looks up the current selected tower type, and places it on the block. To help visualise where the player is placing, an indicator is shown on top of the currently selected block.

### Towers & Upgrades

All tower types are differentiated by the type of damage they deal. Each tower type has a different base damage status, which are made of three damage types.

Each damage type can only affect a specific defence status. For example, the magic damage can only affect the armour defence type. This is done to make different tower types more effective against different enemy types, while allowing every tower to still damage every enemy.

Players can upgrade their towers and a window will pop up when doing so. All tower types can be upgraded with all damage types, allowing for flexibility in upgrading. When a tower is selected to upgrade, an indicator will show on top of the tower to help the player know which tower they are upgrading, especially when the AI suggests an upgrade, where the player would normally not know which tower it's selecting. The indicator is similar to the tower placement indicator and tower selector indicator to consistently tell the player that when they are selecting something, a white box will appear to show them what they are selecting.

The tower types are as seen below:

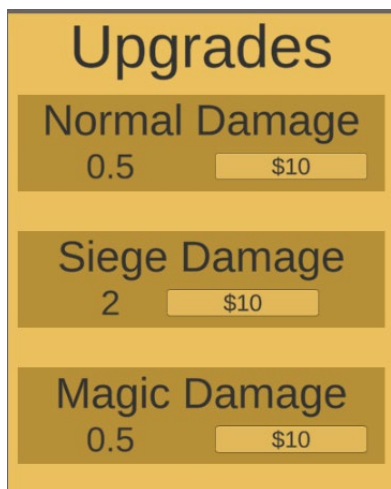
Tower Type	Damage Type	Effective Against
Archer tower	Normal damage	Health
Siege tower	Siege damage	Shield
Magic tower	Magic damage	Armour

**Table 1.** Tower types, damage types, and their effectiveness.

Players can also place platforms on the map. Platforms increase the height of a block, giving towers placed on top of it to have increased range. The platform is treated as a tower, allowing it to be placed on a block,



**Figure 3.** The AI showing which towers the player should buy by moving the suggested button out of line from the others.



**Figure 4.** The AI suggesting to buy an upgrade by shifting the suggested button out of line from the others.

while still allowing other towers to be placed on top of the block.

### AI Suggestion Request

To help the player strategise with the game, an AI helper is implemented. The player can press a button and request for the AI suggestion. The AI will then look up the current state of the game, and give suggestions accordingly. The AI is set to only be able to suggest to the player. This is done to keep the player as the entity who makes the decision, while the AI can help the player by giving them suggestions. The considerations the AI takes are shown below, sorted based on which aspects the AI considers first.

- **Damage Type**

The AI first checks what damage type is needed at the moment. The AI looks up all enemies in the scene, and sums up all their different defensive stats. Then the AI would see which sum is the highest, and return the tower type that is best suited to damage that defensive stat. The AI then shows the suggestion to the player by moving the button of the suggested tower type out of line from the other buttons.

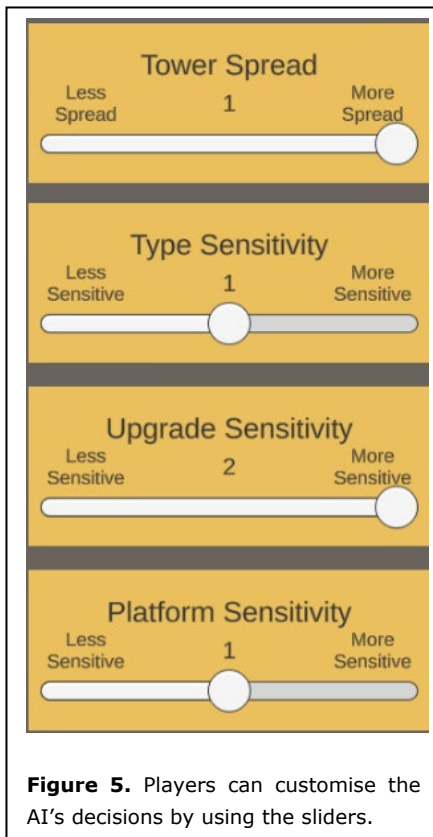
- **Platform**

After the AI knows what damage type it needs, the AI then checks if it should suggest the player to buy a platform instead. The AI first looks up for the highest block in the scene. With that, the AI then calculates the coverage

of the tower if it is placed on the block, and the coverage of the tower if it is placed on the block with an additional height increase. Using that, the AI then calculates the projected price if the tower without a platform has the same coverage as the tower with the platform. The AI then compares the projected price with the sum of the price of the suggested tower, and the price of the platform, suggesting the player to buy a platform if the projected price is higher. If the AI suggests buying a platform, the indicators then change to tell the player to buy the platform, and to show where to put the platform.

- **Upgrade**

If the AI doesn't suggest buying a platform, the AI will then try to suggest buying an upgrade. For every tower in the scene, the AI will calculate the projected price of buying an upgrade when the damage increase is the same as buying a tower. The AI will then find the cheapest tower to upgrade. The AI then compares the projected price of the tower to the price of buying a new tower, suggesting to upgrade if the projected price is lower. If the AI suggests upgrading a tower, the AI will open the tower's upgrade window, and suggest the upgrade by moving the suggested upgrade button out of line from the other buttons. This is done to be consistent and to show the players that whenever the AI suggests a selection, it moves the selection out of line from the others.



- **Tower Placement**

If both the platform and upgrade suggestion is checked to be more expensive than buying a tower, the AI will then suggest buying a tower, and suggests where to place the tower. The AI doesn't already have a tower on top of it, and calculates the coverage value of the block. The coverage value is calculated by first calculating the distance of every road block the tower can reach with its range, to the tower. The AI then sums up the distances and assigns it as the tower's coverage value. Each block's coverage value is also stored in the block object. If the block already has a coverage value stored in it, the value is then used to reduce the total coverage value of the ground block, preventing towers from being bunched up together. The AI will then find the block where the tower has the highest coverage value, and suggest it.

### Controlling AI Suggestion

To fit the different play styles of different players, the AI is implemented to be able to be customised. The player can customise the decisions of the AI by using a UI slider. The customisations are as shown below:

- **Tower Spread**

The tower spread slider controls how spread out each tower is placed. The value of the slider is multiplied to the road block coverage value when calculating each ground block's coverage value, affecting how spread out they are.

- **Type Sensitivity**

The type sensitivity slider controls how sensitive the AI is to the enemies' defensive stats. If the AI is set to be not sensitive, the AI is more likely to suggest the normal damage type. The slider's value is applied to the calculation when the AI sums up all enemies' defensive stats, changing the sensitivity of this aspect of the AI.

- **Upgrade Sensitivity**

The slider for upgrade sensitivity controls how likely the AI suggests to buy an upgrade. Setting the slider to be sensitive sets the AI to be more likely to suggest buying an upgrade. The value of the slider is multiplied to the price of the new tower when comparing it with the projected price of an upgrade, affecting the sensitivity of the AI.

- **Platform Sensitivity**

This slider affects the likelihood of the AI to suggest buying a platform. If the slider is set to be more sensitive, the AI is more likely to suggest buying a platform. The AI takes in the value of the slider and multiplies it with the projected price of buying a tower when comparing it to the sum of the price of the tower and the platform.

### Evaluation

This project showcases the possibility of AI assisting players in a tower defence game. An AI can help players decide different aspects of the game; in this case, the AI helps the player decide what types of

tower the player should buy, whether or not to buy an upgrade instead, or to invest and buy a platform instead, and where to place towers.

While the project successfully showcases the possibility of having an AI helper in a tower defence game, the project is still open for further advancements. To do so,

the game the AI is made in should require more decision making. For example, some peers suggested having more enemy types, especially ones that can heal itself or other enemies. This would force the player and the AI to focus their attacks on that enemy, possibly not attacking “weaker” enemies and letting them pass from towers instead.

## **Acknowledgements**

With this section I would like to thank everyone that has helped and assisted through this project, including peers and lecturers.

## **References**

Bippinbits (2022) Dome Keeper. [Video game]. Raw Fury. Available from:  
[https://store.steampowered.com/app/1637320/Dome\\_Keeper/](https://store.steampowered.com/app/1637320/Dome_Keeper/) [Accessed 1 May 2024].

Die of Death Games (2022) Rogue Tower. [Video game]. Die of Death Games. Available from:  
[https://store.steampowered.com/app/1843760/Rogue\\_Tower/](https://store.steampowered.com/app/1843760/Rogue_Tower/) [Accessed 1 May 2024].

GrizzlyGames (2023) Thronefall. [Video game]. GrizzlyGames. Available from:  
<https://store.steampowered.com/app/2239150/Thronefall/> [Accessed 1 May 2024]

Hypergryph (2019) Arknights. [Video game]. Yostar. Available from:  
<https://play.google.com/store/apps/details?id=com.YoStarEN.Arknights> [Accessed 1 may 2024].

Ninja Kiwi (2018) Bloons TD 6. [Video game]. Ninja Kiwi. Available From:  
[https://store.steampowered.com/app/960090/Bloons\\_TD\\_6/](https://store.steampowered.com/app/960090/Bloons_TD_6/) [Accessed 1 May 2024].