

## Projectverslag: Tower Defense

Dit project werd grotendeels samen gemaakt. Bij sommige onderdelen had de ene een groter aandeel dan de andere. Dit was ongeveer de takenverdeling voor die onderdelen:

### Thomas Dhondt

- Targeting
- Wavelogica
- Inlezen van de wereld

### Bart Middag

- Pathfinding en blueprint
- Beweging
- Rendering

### Ontwerpsbeslissingen

#### Pathfinding (+datastructuren)

Bij de datastructuren hebben we ervoor gekozen om van de eerste/laatste elementen van elke datastructuur lege elementen te maken die verwijzingen hebben naar de echte eerste/laatste elementen die zich in die datastructuur bevinden. Dit is bij de priority queue `head` en `tail`, en bij de hashset het eerste element van elke bucket.

Bij pathfinding hebben we de hulpfunctie `find_path` aangepast, zodat die het `Path`-object meekrijgt als argument. Het is immers veel gemakkelijker om direct in de `nodes`-array van dat object te schrijven.

Het was ook een stuk gemakkelijker en efficiënter om bij het vrijmaken van het geheugen van de datastructuren, de inhoud hiervan – nodes van een pad – ook vrij te maken. De datastructuren worden toch enkel bij pathfinding gebruikt, en alle gealloceerde nodes bevinden zich toch in één van de twee datastructuren.

#### Beweging

In de methode `update_movement` besturen we niet alleen de beweging van de vijanden en de kogels, maar ook de effecten van de spells en alles wat met damage en health te maken heeft. Dit doen we hier omdat de vijanden hier toch al allemaal overlopen worden, wat we anders meerdere keren zouden moeten doen.

#### Plaatsen van torens (blueprint)

Voor het plaatsen van torens controleren we elke keer dat een vijand een node verdergaat op zijn pad, of als de muis van tile verandert, of het blueprint geldig is. Dit zorgt ervoor dat we geen torens op vijanden kunnen plaatsen of op plaatsen waar er geen toren mag staan. Er zal ook geen toren kunnen gezet worden als deze het pad van een vijand blokkeert.

Doordat we dit vóór het plaatsen controleren in plaats van (zoals in de opgave beschreven staat) enkel als er geklikt wordt, kunnen we direct zien of de toren werkelijk geplaatst kan worden of niet.

### Extra's

Toen we klaar waren met het programmeren van het spel, hebben we om het leuker te maken zelfgemaakte muziek (die verandert naargelang wat er gebeurt in het spel) toegevoegd en enkele sprites veranderd. Zeker niet vergeten het geluid aan te zetten dus!

Om muziek toe te voegen en ons daarbij nog steeds aan de opgave te houden, moesten we allegro initialiseren in `init_game_state`.

Ook hebben we een afsluitschermd toegevoegd als je game over behaalt.