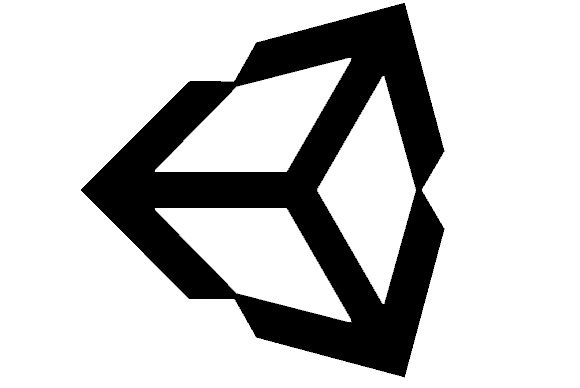
GTO7 – Plan van Aanpak



**Naam:** Frank Severijns  
**StudentNr:** 2383764  
**Klas:** GD7

Datum: 06/10/2015

Inhoud

[Concept 3](#_Toc440400551)

[Gameplay 3](#_Toc440400552)

[Spelmodi 3](#_Toc440400553)

[Schets 4](#_Toc440400554)

[Vijanden 5](#_Toc440400555)

[Drone 5](#_Toc440400556)

[Fighter 5](#_Toc440400557)

[Bomber 5](#_Toc440400558)

[Gunner 5](#_Toc440400559)

[De Speler 6](#_Toc440400560)

[Besturing 6](#_Toc440400561)

[Wapens 6](#_Toc440400562)

[Health/Shield/Experience 6](#_Toc440400563)

[Verplichte Onderwerpen 7](#_Toc440400564)

[Cross-platform 7](#_Toc440400565)

[API 7](#_Toc440400566)

[AI 7](#_Toc440400567)

[Efficient Level Creation 7](#_Toc440400568)

[Persistence 7](#_Toc440400569)

[Vrije Onderwerpen 8](#_Toc440400570)

[GRD7 Onderwerp (Adaptive Difficulty) 8](#_Toc440400571)

[Localization 8](#_Toc440400572)

[Audio Management 8](#_Toc440400573)

# Concept

Het spel valt het best te omschrijven als een shoot ‘em up.

## Gameplay

De speler bestuurt een schip waarmee deze lasers, kogels en raketten kan afvuren. Dit schip bevindt zich aan de onderkant van het scherm en kan over de rest van het scherm bewegen. Het doel is om de vijandelijke schepen kapot te maken door ze te raken met je projectielen. Deze vijandelijke schepen kunnen lasers e.d. terugschieten, die de speler moet zien te ontwijken. Als de speler te vaak geraakt wordt door de vijandelijke schepen gaat de speler af.

De speler kan bij aanvang van het spel kiezen uit 3 verschillende wapens: kogels, raketten en lasers. Kogels zijn onnauwkeurig, maar omdat er veel tegelijk afgeschoten worden kan de speler er potentieel veel vijanden tegelijk mee raken. Raketten gaan richting de dichtstbijzijnde vijand en doen meer schade dan kogels, maar schieten minder tegelijk af en zijn minder snel. Lasers schieten altijd recht vooruit, zijn sneller dan kogels en doen het meeste schade, maar het aantal vijanden die de speler tegelijk kan raken is beperkt.

Naarmate het spel vordert verdient de speler experience punten. Als de speler een level omhoog gaat kan de speler een punt investeren in de drie wapens. Dus als de speler bij aanvang van het spel voor lasers kiest kan de speler bijvoorbeeld ervoor kiezen om de lasers te upgraden waardoor ze meer schade doen en sneller achter elkaar afvuren, of de speler kan ervoor kiezen om de kogels te unlocken, waardoor de speler zowel lasers als kogels kan afschieten. Hierdoor kan de speler ervoor kiezen om zich met een wapen te specialiseren of een soort “jack-of-all-trades” te worden.

## Spelmodi

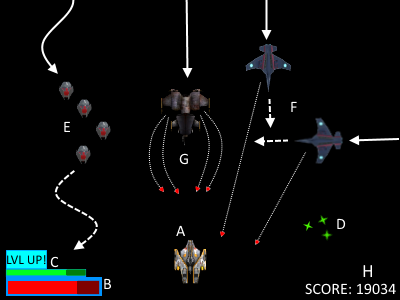
Het spel bevat twee spelmodi: singleplayer en co-op multiplayer.

In beide speelmodi eindigen speelsessies wanneer de speler(s) af is/zijn (te vaak geraakt door vijanden en diens projectielen), of wanneer (een van de) speler(s) zelf de speelsessie afsluit.

Hier is een schets van de game. De speler beweegt het schip over het scherm om vijanden en projectielen te ontwijken. Ondertussen komen er vijanden vanaf de boven- en zijkanten die projectielen afvuren waardoor de speler beschadigd wordt. De speler moet proberen de vijanden te raken om ze te vernietigen.

## Schets

Hieronder staan twee schetsen. Figuur 1 toont een schets van de gameplay features in het spel. Figuur 2 toont het level up menu waar de speler wapens kan upgraden/unlocken.

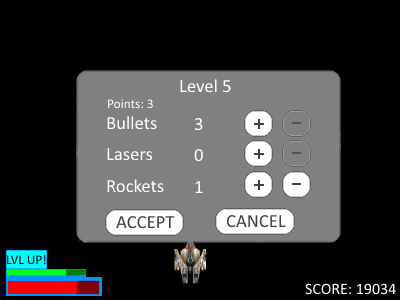
**A**: Het schip van de speler. Dit schip kan iedere kant op bewegen, maar blijft altijd binnen de randen van het scherm.

**B**: De healthbar van de speler. Als de speler beschadigt wordt wordt de healthbar korter. Naast health heeft de speler ook een shield, die in dezelfde bar weergeven wordt. Shield kan herstelt worden door een aantal seconden te stoppen met schieten en geen schade op te lopen. Health herstelt door middel van health pickups. Als de healthbar leeg heeft de speler verloren.

**C**: De experience van de speler. Als de balk vol is komt erboven een knop tevoorschijn waar de speler op kan klikken. Dan krijgt de speler een menu waar punten besteedt kunnen worden aan het unlocken/upgraden van wapens. (Zie figuur 2: Level Up Menu)

1: Gameplay

**D**: Als de speler een vijand kapot maakt laat deze experience punten en health punten achter. Als de speler met zijn schip over de punten vliegt ontvangt hij deze punten.

**E**: Een groep vijandige drone schepen die in een patroon over het scherm bewegen. Deze drones schieten zelf geen projectielen af, maar de speler moet proberen ze te ontwijken om geen schade op te lopen. Drones zijn individueel gemakkelijk om kapot te maken, maar komen altijd in grote groepen tevoorschijn.

**F**: Vijandige fighters en diens projectielen.

**G**: Een vijandige bomber schiet doelzoekende raketten.

**H**: De score van de speler. Omdat het spel endless is, is het behalen van een hoge score het voornaamste doel van het spel.

2: Level Up Menu

## Vijanden

In dit hoofdstuk worden de vijandelijke schepen beschreven. Hierbij ga ik in op het gedrag dat ik de schepen wil laten uitvoeren en voor welk doel ze dienen binnen het spel.

### Drone

De makkelijkste vijand in het spel. De drone beweegt met een bepaald patroon over het scherm. Dit patroon wil ik maken door middel van een spline, zodat ik vervolgens met gemak de drone over de spline kan laten bewegen.  
De drone kan zelf geen projectielen afschieten, dus moet de speler beschadigen door er direct in contact mee te komen. Drones komen altijd in grote aantallen tevoorschijn. Het hoofddoel van de drone is dus meer om bepaalde plekken in het speelveld tijdelijk ontoegankelijk te maken waardoor de speler gedwongen wordt van positie te veranderen.

### Fighter

De fighter is een vijand die altijd snel voortbeweegt en nooit stil blijft staan. De fighter stuurt zolang het off-screen is richting de speler, maar kan geen scherpe bochten maken, waardoor het niet heel nauwkeurig is. Terwijl ze op het scherm zijn vliegen ze alleen rechtdoor en schieten ze projectielen af. Deze projectielen zijn gericht op de positie van de speler, maar kunnen alleen recht vooruit vliegen.  
De fighter dient voornamelijk als simpele vijand waar de speler weinig moeite mee zou moeten hebben.

### Bomber

De bomber is een traag, maar moeilijk te vernietigen schip dat raketsaldo’s afschiet. Deze rakketten volgen de speler en zijn daardoor lastig te ontwijken.  
De bomber beweegt zelf weinig, en kan niet schieten en bewegen tegelijk. De bomber kiest regelmatig een andere positie uit om naar toe te vliegen, om daar vervolgens stil te blijven staan en te schieten.  
Vanwege het hoog aantal health en de doelzoekende rakketten is de bomber een vijand waar de speler vaak zijn aandacht op zal willen focussen om het als eerst uit te schakelen.

### Gunner

De gunner is qua gedrag het meest complex. De gunner schiet snel een groot aantal projectielen af die diagonaal van zich af vuren, zoals de bullets van de speler. Deze projectielen doen weinig schade en gaan alleen rechtdoor (dus niet doelzoekend). De gunner kijkt altijd recht naar beneden.  
De gunner kan, net als de bomber, niet tegelijkertijd schieten en bewegen, en wisselt om een bepaalde tijd af tussen het bewegen naar een willekeurige positie en het schieten.  
Als er meerdere gunners op het scherm zijn kiest de gunner een partner uit. Deze partner spiegelt de bewegingen van de gunner. Bijvoorbeeld: Gunner A gaat naar positie (4,0,0), tegelijkertijd gaat de partner Gunner B naar positie (-4,0,0). Omdat de gunners diagonaal projectielen afvuren onstaat er zo een soort overlappend veld van projectielen die de speler kan insluiten.

# De Speler

## Besturing

De speler bestuurt een enkel schip. Dit schip kijkt ten alle tijden naar de bovenkant van het scherm. De speler kan het schip niet van het scherm af bewegen. Om te bewegen kan de speler gebruik maken van de WASD- of pijltjestoetsen. De speler kan hiermee in alle richtingen bewegen.  
Om te schieten gebruikt de speler de linkermuisknop.

## Wapens

De speler heeft beschikking tot een drietal wapens. Bullets, rockets en lasers.

Bullets vliegen in een rechte lijn in een diagonale richten vanuit de speler. Bullets worden in grote aantallen tegelijk geschoten, maar doen individueel minder schade dan de andere twee wapens. Bij het upgraden van de bullets kan de speler meerdere bullets tegelijk afschieten en doen de bullets meer schade.

Rockets vliegen vanaf de zijkant van de speler richting de dichtsbijzijndste vijand. Rockets zijn trager dan bullets en schieten minder tegelijk af, maar doen veel meer schade. Bij het upgraden van rockets gaan de rockets sneller, doen ze meer schade, worden ze nauwkeurig met sturen richting de vijand en schieten er meer tegelijk af.

Lasers schieten in een rechte lijn uit de voorkant (bovenkant) van de speler. Lasers doen meer schade dan bullets maar minder dan rockets. Bij het upgraden van lasers schiet de speler er meer tegelijk af, schieten de lasers sneller en doen ze meer schade.

## Health/Shield/Experience

De speler heeft health en shield. Zolang de shield niet leeg is wordt schade door de shield geabsorbeerd. Als de shield leeg is en de speler geraakt word verkleint de healthbar. Als de healthbar leeg is heeft de speler verloren. De speler kan zijin/haar health- en shieldbars upgraden waardoor ze vaker geraakt kunnen worden voodat ze game over zijn.  
De shieldbar herstelt zich automatisch zolang de speler niet geraakt word. De healthbar kan herstelt worden door health pickups op te rapen. Deze pickups worden achtergelaten door gedode vijanden.

Net als health pickups laten vijanden ook experience pickups achter. Als de speler deze opraapt (door er overheen te vliegen) krijgt de speler experience. Aan de hand van het aantal experience vult de experience bar. Als de experience bar vol is krijgt de speler een upgrade punt en wordt de experience bar weer leeg gemaakt. Er is steeds meer experience nodig om de volgende upgrade punt te unlocken.

# Verplichte Onderwerpen

## Cross-platform

Het spel moet gebuild kunnen worden op Android, maar zal vanwege de focus op keyboard controls niet speelbaar zijn op andere platforms dan PC.

## API

Ik wil gebruik maken van de GameJolt API. Met de GameJolt API kan ik trophies toevoegen, die de speler kan unlocken door aan bepaalde condities te voldoen (bijvoorbeeld het behalen van 1000 score punten).

## AI

De AI wil ik maken door middel van een Finite State Machine die gebruik maakt van Mecanim (<http://darkgenesis.zenithmoon.com/mastering-unity-2d-game-developmentai-and-state-machines/>). Door gebruik te maken van Mecanim kan ik de state transities beheren met de parameters in Mecanim zelf, hierdoor hoef ik niet voor iedere state in code de condities voor een transitie te controleren. Ook is het hierdoor makkelijker om de State Machine te visualizeren voor eventuele foutopsporing.

## Efficient Level Creation

Omdat een speelsessie theoretisch gezien eindeloos lang kan duren is het niet mogelijk om handmatig vijanden te plaatsen. De vijanden zullen procedureel gegenereerd moeten worden. Hierbij komt een van de vrije onderwerpen ook deels aan bod, omdat bij het genereren van de vijanden de flow van het level in acht gehouden moet worden.  
Omdat er soms een groot aantal projectielen gebruikt kan worden, wil ik gebruik maken van instance pooling om de performance te verbeteren.

## Persistence

Gedurende een speelsessie zal een speler niet kunnen saven of laden. Bepaalde waardes zoals highscores zullen wel automatisch bijgehouden moeten worden zodat de speler deze niet verliest zodra het spel afgesloten wordt.  
Om de score op te slaan wil ik gebruik maken van de GameJolt API. Hiervoor is uiteraard wel een internet connectie nodig, dus de scores van de speler worden ook opgeslagen in de PlayerPrefs zodat de speler ook offline geen vooruitgang verliest. Als de speler weer connectie heeft met het de API zullen de scores in de PlayerPrefs gesynchronseerd worden met de online opgeslagen scores. Instellingen zoals taal en audio volume (zie Vrije Onderwerpen) worden ook in de PlayerPrefs bewaart.

# Vrije Onderwerpen

## GRD7 Onderwerp (Adaptive Difficulty)

Omdat een speelsessie geen vast einde heeft en d.m.v. code gegenereerd wordt moet ik een manier vinden om de moeilijkheidsgraad procedureel te kunnen beïnvloeden. Als het spel tijdens een speelsessie altijd even gemakkelijk of moeilijk blijft zal de speler zich gauw gaan vervelen of juist gefrustreerd raken.  
Hierom wil ik dit onderwerp tijdens GRD7 nader bestuderen zodat dit in het GTO7 project gebruikt kan worden.

## Localization

Om het spel beschikbaar te maken in meerdere talen wil ik een systeem maken dat teksten gemakkelijk kan vertalen. De taal kan door de speler zelf ingesteld worden.

## Audio Management

Ik wil de speler de mogelijkheid geven om het volume van de geluidseffecten en muziek zelf in te stellen. Ook wil ik er voor zorgen dat de waardes van de ingestelde volumes opgeslagen worden zodat de speler niet bij opstarten van het spel de instellingen weer moet veranderen.

Daarnaast lijkt het me handig om een enkel script te hebben wat het afspelen van alle audio beheert. Dit zorgt ervoor dat ik niet in alle scripts waarin ik een geluid wil afspelen functies moet schrijven om het volume op te halen en (indien nodig) andere waardes zoals de pitch te wijzigen.  
Tevens kan ik hierbij gebruik maken van instance pooling, doordat ik een bepaald aantal objecten met audio sources instantieer en opsla. Wanneer er een geluid afgespeeld moet worden kan ik een audioclip instellen en het object activeren voor zolang het geluid afgespeeld wordt.