

# Galactic defense

Ambrož Klančar, Gašper Rozman, Tim Rekelj

## Abstract

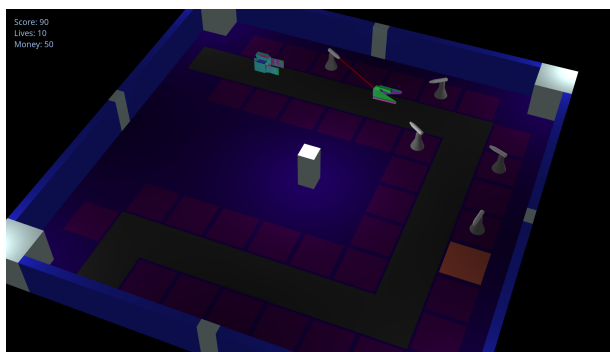
*Za projekt smo se odločili izdelati igro imenovano Galactic defense, ki je žanra 'tower defense'. Idejo za igro smo dobili iz podobne igre imenovane 'Infinitode'. Pri projektu smo uporabili JavaScript, WebGL in Blender.*

## 1 Pregled igre

Galactic defense je igra, ki spada v žanr tako imenovanih 'tower defense' iger. Igra ima omejen svet v katerem lahko najdemo prostor kamor lahko postavljamo tope in pot kjer potujejo sovražne ladje. Cilj igre je strateško postaviti topove, da nobena ladja ne pride do konca poti.

Igra je na začetku enostavna, s časom pa postaja vedno težja. Namreč s časom je na mapi vedno več sovražnih ladij, prav tako pa s časom začnejo prihajati tudi močnejše ladje.

V igri Galactic Defense se znajdemo v galaksiji Zvezdni Svet, kjer nam naš večni nasprotnik grozi z napadom na naš svet. Igralec prevzame vlogo prve obrambe in postavlja topove, da uniči sovražnikove ladje, ki se bližajo. Cilj je preprečiti, da bi sovražnik prodril v naše ozemlje in zaščititi naš planet pred nevarnostjo. Igra prinaša napetost in izzive, saj se mora igralec hitro odločati, kam postaviti topove, da učinkovito zadrži sovražnika.



Slika 1: Izgled igre.

### 1.1 Opis sveta

Naš svet naj bi predstavljal platformo v vesolju kjer igralec postavlja obrambo da se zaščiti pred napadom sovražnih ladij. Svet se ustvari ob začetku igre, kjer iz

JSON datoteke preberemo pozicijo poti in pozicije namenjene obrambi. Vsi elementi so 3d modeli, nekateri se samo vrtijo okoli ene osi (topovi), nekateri se pa tudi premikajo po vseh treh oseh (ladje grejo po poti in se s sinusno funkcijo premikajo gor in dol). Kamera se lahko premika po vseh treh dimenzijah.

#### 1.1.1 Pregled

Uporabnik lahko interaktira s kamero, katero lahko poljubno vrti in spreminja pogled na naš svet.

#### 1.1.2 Ozadje

Privzed pogled na naš svet je ptičja perspektiva, kjer igralec gleda iz vrha na igralno površino. Tako vidi samo zgoraj našete modele. Ko pa le ta premakne kamero pa lahko vidi lebdečo platformo v vesolju. ozadje pa predstavlja črn prostor.

#### 1.1.3 Ključne lokacije

Naše ključne lokacije v igri so pozicije kamor mora uporabnik uporabnik strateško postaviti obrambne topove za karseda učinkovito obrambo.

Možne pozicije za obrambo so obarvane nekoliko svetlejšje kot pot po kateri se premikajo sovražne ladje in so v obliki kvadrata. Eden izmed teh kvadratov je še dodatno poudarjen, s tem igralcu nakažemo katera obrambna pozicija je trenutno izbrana in kam bo postavljen top, če se uporabnik odloči za postavitev le tega.

Naslednja pomembna lokacija pa so poti kjer se premikajo ladje. Igralec na njih nima nobenega vpliva, prav tako pa tudi ne more vplivat na njih. Pot je predstavljena z več kvadrati temnejše barve, ki skupaj predstavljajo povezano pot od začetka do konca sveta.

#### 1.1.4 Velikost

Naš svet je velikosti 100x100 enot, kjer uporabnik interaktira s postavitvijo topov in premikanjem kamere. Poleg sveta pa se vidi še vesolje. Igralni elementi, tj. ladje in topovi pa so velikosti približno 5x5 enot.

#### 1.1.5 Objekti

Vse objekte v naši igri smo naredili sami uporabili pa smo nekaj tekstur iz interneta. Igralne elemente smo naredili v programu Blender. Elemente za menije pa s pomočjo HTML elementov.

Naši topovi so podobni topovom za izdelavo ume-tnega snega, le da so naši topovi nekoliko bolj nevarni in namesto snega proizvajajo močne laserske žarke s katerim lahko stali nevarno ladijo in ji onemogoči napredovanje.

Igra vsebuje tudi dva tipa ladij. Prva je bolj taktično usmerjena saj je hitrejša ampak je manj trpežna in prinaša manj točk igralcu. Drugi tip ladije pa je bolj trpežen saj lahko prenese več udarcev s tem pa tudi prinese več točk igralcu, ko jo le ta uspešno uničena. Izgled objektov lahko vidimo na sliki 1.

#### 1.1.6 Čas

V igri Galactic defense čas nima posebne vloge, razen to da se s časom ladije pojavljajo vedno bolj pogoste ter močnejše.

### 1.2 Igralni pogon in uporabljene tehnologije

Za izdelavo objektov igre smo uporabili program Blender, ki je namenjen tudi modeliranju 3d objektov. Naše modele smo potem izvozili iz programa s pomočjo GLTF tehnologije, ki je namenjena izvažanju in uvažanju podatkov iz modelirnika.

Nato smo v naš projekt s pomočjo JavaScripta uvozili podatke iz programa Blender. Razvijanje smo potem nadaljevali z uporabo JavaScripta, HTML in WebGL.

Za senčilne programe smo uporabili GLSL jezik.

Kot dodatne tehnologije v igri smo uporabili naše lastne senčilnike, ki podpirajo več luči, kjer luč vpliva le na objekte na določeni razdalji, objekti na katere pa svetita dve luči se pa svetloba sešteje.

V igri smo prav tako uporabili bolj napredne tehnike za komunikacijo naših komponent (uporaba lastnih event emitterjev) in delo z stanji igre. V igri imamo stanja, ki predstavljajo igro ali pa menije. S stackom lahko hranimo vsa stanja ter zgodovino kako si sledijo s tem pa imamo lahko aktivno le stanje na vrhu stacka. Primer: ko poženemo igro dobimo "main menu", ki se doda kot stanje na stack, meni potem lahko požene igro tako da doda novo stanje na stack.

### 1.3 Pogled

V igri uporabljamo perspektivno kamero, ki bo svet gledala podobno kot v tretji osebi, kjer bo igralec lahko premikal kamero kot po površini krogle, igralec pa bo lahko videl cel svet.

## 2 Osebek

Igralec ima nadzor nad postavitvijo topov na svet ter obračanjem in premikanjem kamere.

## 3 Uporabniški vmesnik

V igri imamo kar nekaj menijev. Imamo "Welcome / main"meni, meni za navodila za igranje, opis o projektu, "pause"meni, meni ki se prikaže ob zmagi in porazu.

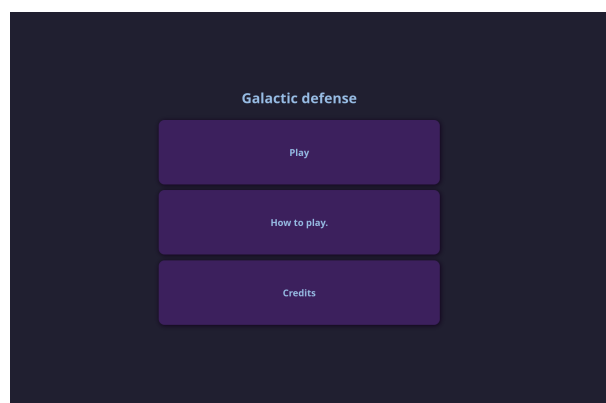
Sestavljen pa je iz HTML elementov in oblikovan s pomočjo CSS. Primer lahko vidimo na sliki 2 in 3 Na začetku smo poskusili implementirati menije kar znotraj

igre ampak smo videli, da za čas, ki smo ga imeli na raspolago to nebi bila dobra odločitev. Zato smo se odločili da menije naredimo z uporabo HTML. Čeprav meniji niso implementirani znotraj igre pa še vseeno delujejo brez težav saj, kot že omenjeno, smo pri implementaciji menijev uporabili stack za nadziranje stanj. S pomočjo le tega imamo možnost, da je v igri aktivno le eno stanje na enkrat, s tem prihranimo nekaj računalniških virov in se izognemo tudi kakšni napaki. Prav tako pa je to nekoliko lepši pristop k implementaciji.

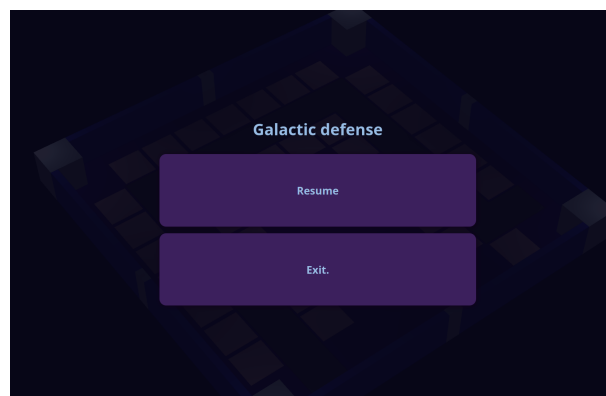
Ko želimo prikazati novi meni ali zagnati igro preprosto dodamo stanje menija ali igre, stack pa bo poskrbel za aktivacijo novega stanja in ustavitev starega stanja.

Uporabnik pa z menijem interaktira s preprostim klikanjem na gumb.

V igri imamo tudi neinteraktiven meni za prikazovanje glavnih podatkov v igri tj. število preostalih življenj, denar in točke. To lahko vidimo na sliki: 1



Slika 2: Glavni meni.



Slika 3: Meni za pavzo.

## 4 Glasba in zvok

V igro smo dodali dva zvočna efekta. Enega za laser, ki se sproži, ko top ustreli, drugi zvočni efekt pa smo dodali za uničenje ladije. Prvi zvočni efekt je podoben zvoku laserja, drugi pa manjši eksploziji.

V igro smo dodali tudi melodijo, ki se predvaja v zanki. Izbrali smo nekoliko bolj živahno glasbo z namenom da bi ustvarili napetost v igri.

Glasbo in efekte smo pridobili na spletni strani Pixabay.<sup>1</sup>

## 5 Gameplay

Naša igra igra se začne z glavnim menijem, kjer ima igralec možnost izbire za prikaz navodil in informaciji o projektu ter možnost za začetek igre.

Ko igralec zažene igro se mu odpre svet, kjer lahko postavlja topove. Takoj pa se začnejo prikazovati tudi ladije, uporabnik mora tako hitro razmišljati kam postaviti prvi top.

Ko topi uničijo ladije uporabnik dobi denar s katerim lahko kasneje kupi dodatne topove.

Igralec med igro lahko pritisne tipko "Escape"ter s tem ustavi igro. V tem meniju ima uporabnik možnost nadaljevanja igre ali pa izhod na glavni meni. Če se uporabnik odloči za izhod iz igre dobi še en meni kjer mora svojo akcijo potrditi.

Igra se lahko zaključi tudi z zmago, igralec postavi obrambo na vsa razpoložljiva mesta, ali pa z porazom, igralec ostane brez življenj. V obeh primerih se prikaže dodaten meni (zmaga ali poraz) kjer ima uporabnik možnost vrnitve nazaj na glavni meni.

Med igro ima igralec možnost premikanja kamere po želji. S smernimi tipkami pa premika trenutno aktivirano polje za postavitev obrambe, ko je zadovoljen z pozicijo kamor želi postaviti novo obrambo, s pritiskom na tipko "enter"postavi nov top.

## 6 Zaključki in možne nadgradnje

Igro bi lahko dodatno nadgradili s tem da bi postavljajanje topov namesto z tipkovnico implementirali z miško in "raycast-anjem". To smo sicer poizkušali in je že skoraj uspelo, na žalost pa nas je prehitel čas, zato smo morali kot alternativno rešitev uporabiti tipkovnico.

V igro bi lahko dodatli tudi več različnih modelov ladij in topov ter omogočili uporabniku izbiro topov. Dodali bi lahko še kakšno mapo. Prav tako pa bi lahko dodali še malo premikanja po tretji dimenziji da bi igra delovala bolj dimenzična.

## Literatura

---

<sup>1</sup><https://pixabay.com/sound-effects/search/game/>