

Scuffedpuolustus

Atte Rätty, 1025068

AIT, 2022

19.4.2023

Yleiskuvaus

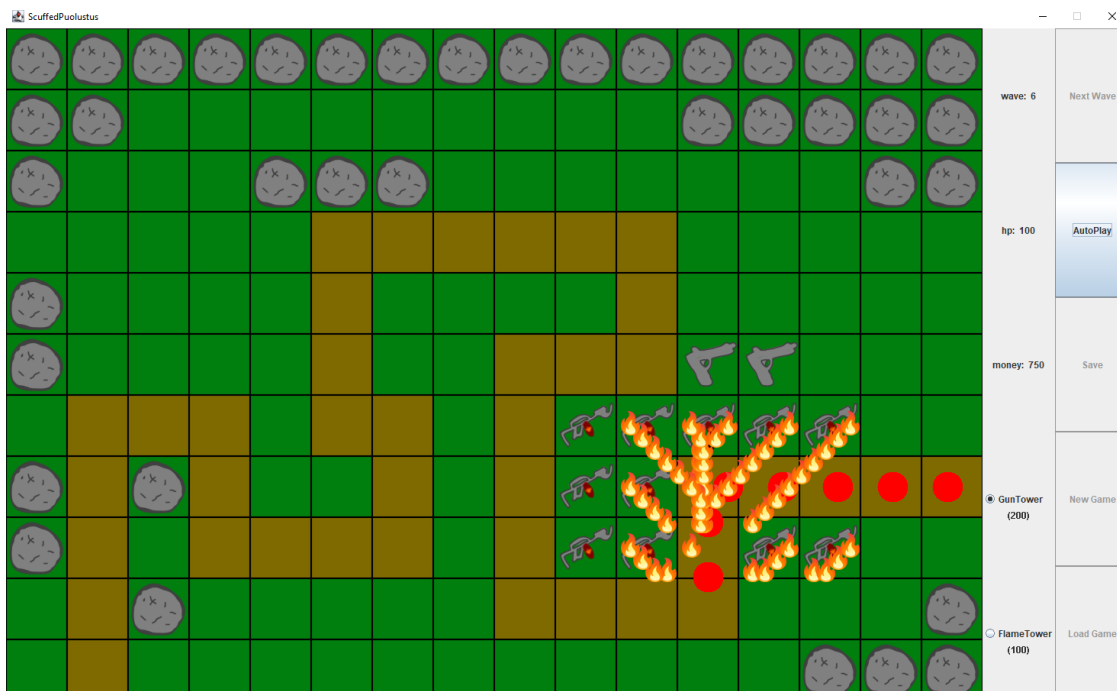
Ohjelma on yksinkertainen tornipuolustuspeli, jossa pelaaja pyrkii estämään vihollisia pääsemästä reitin loppuun. Pelaaja saa rahaa tuhoamalla vihollisia, ja saadulla rahalla pelaaja voi asettaa lisää torneja. Vihollisten määrä ja vahvuus kasvavat jokaisella kierroksella. Peli on toteutettu vaativalla -tasolla.

Käyttöohje

Ohjelma aukeaa uuteen peliin. Pelaaja voi valita, haluaako aloittaa seuraavan vihollisaallon, aloittaa uuden pelin vai ladata tallennetun pelin. Pelaaja voi asettaa uusia torneja kartalle, valitsemalla ensin vasemmalta haluamansa tornityypin, ja sitten painamalla hiiren vasenta näppäintä haluamansa ruudun kohdalla. Pelaaja näkee hp:n ja raha tilanteensa vasemmalla, ja voi laittaa autoplay:n päälle napista, jolloin peli jatkaa automaattisesti seuraavaan vihollisaaltoon.

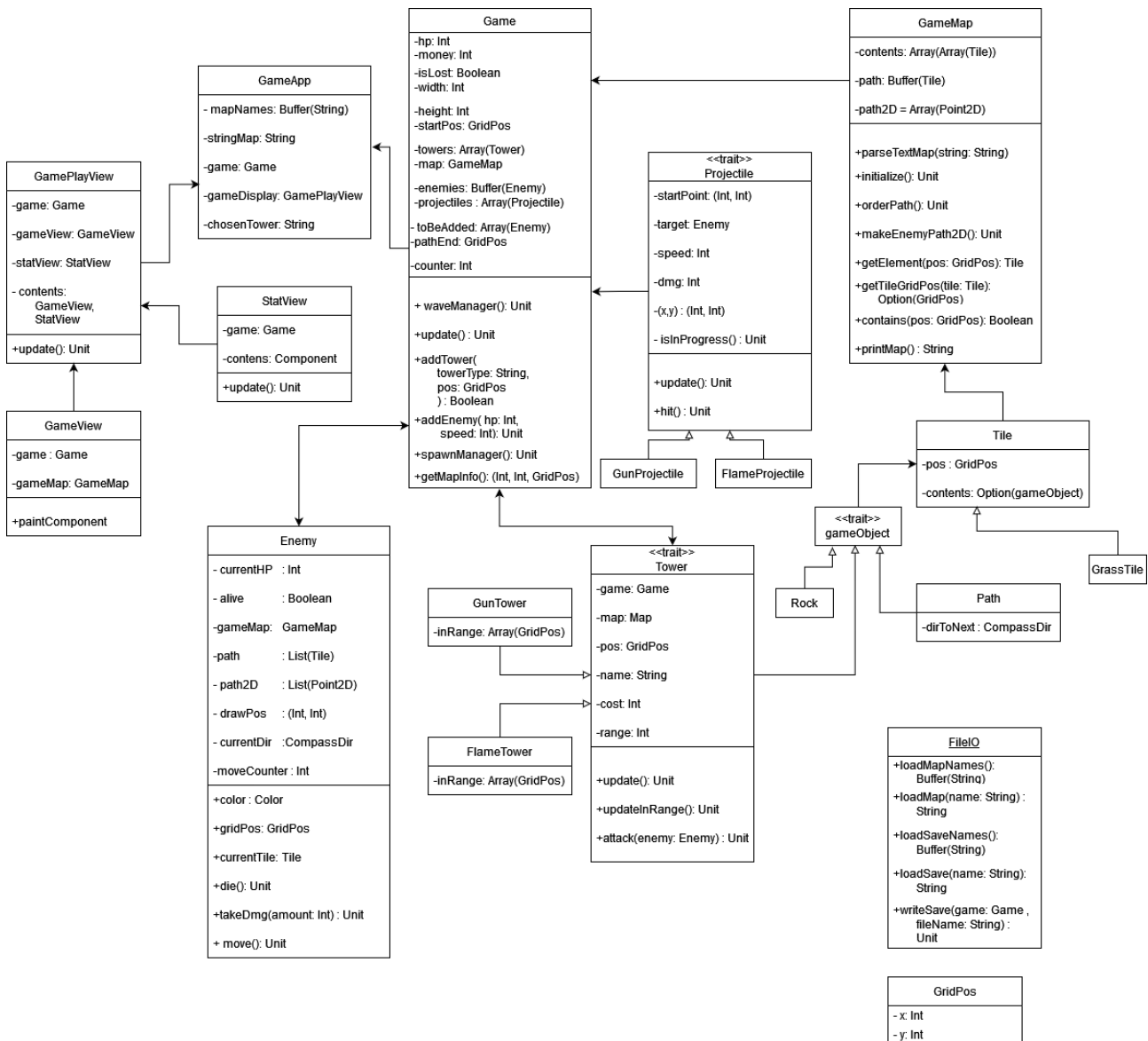
Pelaaja voi tallentaa pelin vihollisaaltojen välissä, painamalla save -nappia, ja kirjoittamalla tekstikenttään nimen tiedostolle.

Pelaaja voi avata tallennetun pelin painamalla load -nappia, ja valitsemalla listasta haluamansa pelin.



Kuvassa peli keskellä kierrosta.

Ohjelman rakenne



Game -luokka kuvaa peli-instanssia ja käsittelee pelilogiikan, vihollisten luomisen sekä tornien hallinnan.

GameApp -Objekti käsittelee sovelluksen graafisen käyttöliittymän.

GameMap -luokka säilyttää tiedon kartan tiilistä ja tiilillä olevista kappaleista.

Enemy -luokka kuvaa vihollista ja käsittelee vihollisen liikkumisen.

Tower -piirreluokka kuvaa torneja.

FileIO -objekti käsittelee tiedostot.

GridPos -luokka kuvaa kordinaatteja kartalla.

Tile -luokka kuvaa yksittäistä tiiltä kartalla.

Projectile -piirreluokka kuvaa projektiilia.

Algoritmit

Tornit etsivät ensimmäisen hyökkäysetäisyydellään olevan vihollisen Game -luokan enemies -listasta, joka on järjestetty Enemy -luokan moveCounterin mukaan, jolloin tornit hyökkäävät aina reitillä pisimmällä edenneeseen viholliseen.

Viholliset liikkuvat GameMap -luokan path2D listan mukaan, jokaisella päivityksellä moveCounter kasvaa, jonka avulla vihollinen valitsee seuraavan pisteen listalta.

Projektiilit liikkuvat laskemalla koordinaatti eron sijaintinsa ja kohteensa välillä, ja muuttamalla sitten paikkaansa nopeutensa verran.

Peli luo uusia vihollisia riippuen kierroksesta. Vihollisten vahvuus riippuu myös kierroksesta.

Kartta alustetaan lukemalla tiedostosta haetusta tekstistä, mitä kartan tiilillä kuuluisi olla. Kartta asettaa tiilien contents -muuttujaan niille kuuluvan objektin. Vihollisten reitti on kirjoitettu kirjaimilla, jotka vastaavat ilmansuuntaa seuraavaan reittipalaan. Tällöin vain reitin alun koordinaatit tarvitsee tallentaa. Tornit on myös tallennettu vastaavilla kirjaimilla.

Tietorakenteet

Sovellus käyttää Scalan valmiita tietorakenteita

Tiedostot

Ohjelma käyttää tekstitiedostoja karttojen ja tallennettujen pelien esittämiseen.

Ohjelma käyttää kuvatiedostoja tiilien ja tornien piirtämiseen.

Karttaa kuvaavassa tiedostossa jokainen tiili saa kirjaimen, joka kuvaa mitä tiilellä on

esim:

```
RAASAAAAARRRR
AAASAAAAARRRR
AAASRAAAAAAR
AAAEESRAAAAA
ARRAASAAAAARR
AASWWAAAAAAR
AASAEESAAAAA
AAEENASAAAAA
AAAAAASAAAAA
AAAAAASAAAAA
#12100401
```

, jossa alimmalla rivillä ovat tiedot kartan koosta ja reitin alkupiste.

R = Kivi

A = Tyhjä

E, W, N, S = Reittitiili, jossa kirjain osoittaa seuraavan reittitiilen ilmansuuntaan. Viimeinen reittitiili osoittaa ulos kartasta.

Tallennettua peliä kuvaavassa tiedostossa ovat tiedot pelaajan hp:sta, rahasta ja kartan tilanteesta.

esim:

```
R10
S
RRRRRRRRRRRRRRRR
RRAAAAAAAAAARRRR
RAAARRRAAAAAAARR
AAAAASWWWWAAAAA
RAAASAAAAANAAAAA
RAAASAAEENGGGGG
ASWWAESANFFFFFFF
RSRNAASANFFSWWW
RSANWWWANFFSFFGG
ASRAAAANWWWGGGR
ASAAAAAAGGGGRRR
#16111608
M960
H100
```

, jossa ylimpänä on tallennushetken kierros, sen jälkeen kartan tilanne, raha ja hp.

G = GunTower

F = FlameTower

Testaus

Ohjelmaa on testattu jatkuvasti pelaamalla peliä, ja kokeilemalla ominaisuuksia mahdollisimman monella eri tavalla, ja eri järjestyksissä.

Testausta suoritettu mm. tulostamalla vihollisten liikkeitä konsoliin ja tarkkailemalla päivittykö graafinen käyttöliittymä oikein. Sekä tarkkailemalla hyökkäävätkö tornit oikeisiin vihollisiin ja onko tornien hyökkäysalue laskettu oikein.

Ohjelman puutteet ja viat

Yksi tornityypeistä jäi toteuttamatta.

Tornien poistaminen puuttuu.

Viholliset piirtyvät joskus erittäin harvoin tuntemattomasta syystä hetken aikaa valkoisina.

Tornien lisääminen jatkossa on suhteellisen helppoa, mutta grafiikoiden ja logiikan toteuttaminen vie aikaa.

Tornien poistamisen lisääminen vaatisi pohtimista kuinka sen toteuttaisi graafisessa käyttöliittymässä, logiikka olisi helppo toteuttaa.

3 parasta ja 2 heikointa kohtaa

Parhaat:

- Grafiikat onnistuneen huonot ja projektiilien yksinkertainen liike sopii hyvin pelin yleiseen tunnelmaan.
- Pelin graafinen käyttöliittymä vaihtaa kokoaan automaattisesti vaihtaessa karttaa.
- Pelin tallentaminen ja lataaminen toimii hyvin, ja välittömästi

Heikoimmat:

- Vähän erilaisia torneja ja torneja ei voi päivittää, eikä poistaa.
- Virhe käsittely voisi olla parempaa.

Poikkeamat suunnitelmasta

Projekti toteutui lopulta lähes täysin aikataulun mukaan. Ensimmäisillä viikoilla kirjoitin graafisen käyttöliittymän alun ja kartan. Sen jälkeen tornit ja niiden asettamisen sekä pelilogiikan. Viimeisillä viikoilla tein parannuksia käyttöliittymään, parannuksia optimointiin, parantelin koodia ja luettavuutta, ja korjailin ilmestyneitä bugeja.

Kokonaisarvio lopputuloksesta

Kokonaisuutena projekti onnistui mielestäni hyvin. Peliä on suhteellisen hauska pelata, ja ominaisuudet toimivat hyvin. Projektin rakenne ja luokka jako onnistui tyydyttävästi. Tornien lisääminen on suhteellisen helppoa, mutta aika ja ideoiden puute rajoittivat niiden toteutunutta määrää.

Peli toimii hyvin, uuden pelin luominen toimii hyvin, tallennetun pelin lataaminen toimii hyvin ja karttojen lataaminen ja piirtäminen toimii hyvin. Peli on suhteellisen hyvin optimoitu, karta voi olla täynnä torneja ja vihollisia ja peli toimii yhä hyvin.

Peliä voisi jatkossa parantaa lisäämällä erilaisia torneja ja lisäämällä torneille päivityksiä. Myös karttoja ja vihollistyyppejä voisi lisätä.

Jos aloittaisin nyt, lisäisin peliin enemmän sisältöä jo aluksi, ja parantelisin pohjalla olevia rakenteita mahdollistamaan sisällön lisäämisen helpommin. Saisin myös tehtyä sisältöä nopeammin, koska ymmärrän käyttämäni kirjastot paremmin ja minulla on parempi idea siitä, miten asioita kannattaa toteuttaa ja mitä rakenteilta toivoisi myöhemmin sovelluksessa.