

# Python Y2 projektin dokumentti

## Strategiapeli

### 1. Yleiskuvaus

Peli on vuoropohjainen strategiapeli tekoälykkäällä tietokonevastustajalla, joka on toteutettu PyQt5-kirjaston avulla. Käyttöliittymä on täysin graafinen ja pelikentät ja joukkueet ovat konfiguroitavissa ulkoisten tekstitiedostojen avulla.

Peliä voi pelata joko kahdella ihmispelaajalla tai yhdellä ihmispelaajalla ja tietokonevastustajalla. Tietokonevastustajan oletetaan aina olevan pelaaja numero 2.

Pelissä komennetaan joukkoa erilaisia hahmoja, joilla on erilaisia vuoron aikana suoritettavia toimintoja. Joukkueet koostuvat neljästä erilaisesta hahmotyypistä, joista jokaisella on erilaiset liikkumistavat ja hyökkäykset. Pelaajan vuoro koostuu ensin vapaavalintaisen hahmon liikuttamisesta ja sen jälkeen vapaavalintaisen hahmon hyökkäyksestä. Liikutettu hahmo ja hahmo, jolla hyökätään voivat olla keskenään erit. Jokaisella hahmotyypillä on omanlaisensa HP (health points), sallitut liikkumisruudut, sekä hyökkäys. (Hahmojen ominaisuuksista tarkemmin myöhemmin) Pelikenttä on tiili-pohjainen suorakaide, jonka rakenne luetaan tekstitiedostosta (formaatin yksityiskohdat myöhemmin). Tiilityyppejä on kolmea erilaista: vapaa, puu ja kivi. Pelaajat voivat liikkua vain vapaisiin tiiliin eikä puiden tai kivien läpi voi hyökätä. Puita voi tuhota tulihyökkäyksellä, mutta kivet ovat pysyviä. Pelin tietokonevastustaja on jokseenkin tekoälykäs (yksityiskohdat myöhemmin), päliä voi halutessaan pelata myös kahdella ihmispelaajalla. Peli päättyy, kun jommankumman joukkueen kaikki hahmot on tuhottu. Pelin jälkeen sen voi halutessaan lopettaa, tai aloittaa uudestaan.

### 2. Käyttöohje

Pelin pelaaminen vaatii PyQt5-kirjaston. Peli aloitetaan ajamalla game.py -tiedosto. Jos käyttäjä haluaa tehdä asetustiedostoihin muutoksia (karttaan tai joukkueisiin), ne tulee tehdä ennen uuden pelin aloittamista.

Muuten peli opastaa käyttäjäänsä hyvin: pelinäkymän alareunassa olevaan tekstikonsoliin tulostuu alussa mikä väri kuvastaa kumpaakin pelaajaa, kumman pelaajan vuoro on milläkin hetkellä, mitä toimintoa peli ihmiskäyttäjältä odottaa, sekä aina jos jokin hahmo ottaa vahinkoa tai kuolee.

Ihmispelaaja valitsee liikuteltavan tai hyökkäävän hahmon aina sitä klikkaamalla, jonka jälkeen ruudulla näytetään sallitut liikkumis- tai hyökkäysruudut. Ruutu valitaan myös klikkaamalla.

Ruudun alareunassa on kaksi painiketta: "Skip" ja "Undo select". "Skip"-napilla voi halutessaan skipata meneillään olevan vuoronsa vaiheen. Jos esimerkiksi et halua liikuttaa ketään hahmoa, "Skip"-napin painaminen siirtää pelaajan vuoronsa hyökkäysvaiheeseen. Jos pelaaja ei halua tai voi hyökätä vuorollaan, "Skip"-napin painaminen siirtää vuoron toiselle pelaajalle. "Undo select"-illä voi taas perua äsken valitsemansa hahmon valinnan. Jos ihmispelaaja on esimerkiksi liikkumisvaiheessa valinnut hahmon, mutta tajuaa ettei voikaan liikkua järkevään paikkaan, pelaaja voi painaa "Undo select"-nappia ja valita toisen hahmon.

Liikuttamalla cursorinsa hahmojen päälle voi nähdä kyseisen hahmon tiedot eli nimen ja HP:n. Alotus-HP:t ovat hahmokohtaisia, hahmojen nimet voi päättää itse asetustiedoston avulla.

Hahmojen ominaisuudet:

Tankki, eli Tank, on väriltään keltainen. Tankin aloitus-HP on 30 ja se tekee hyökkäyksellään 10 vahinkoa. Se voi liikkua vuorollaan kahden ruudun etäisyydelle kyseisestä sijainnistaan, mutta voi hyökätä vain viereisiin ruutuihin.

Maagi, eli Mage, on väriltään vihreä. Maagin aloitus-HP on 10 ja sen hyökkäys on muista poikkeuksellinen. Valitun hyökkäysruudun ympärystä (3x3 ruudukko) syttyy tuleen kolmen vuoron ajaksi. Tuli tekee siinä olevaan hahmoon joka vuorolla 5 vahinkoa. Tuli myös tuhoaa puut, joihin se osuu. Maagi voi liikkua vuorollaan yhden ruudun etäisyydelle kyseisestä sijainnistaan, ja se voi hyökätä x- ja y-suunnissa maksimissaan neljän ruudun päähän sijainnistaan pois lukien itsensä viereiset ruudut.

Ninja on väriltään violetti. Ninjan aloitus-HP on 20 ja se tekee hyökkäyksellään 5 vahinkoa. Se voi liikkua vuorollaan kuten shakin lähetti, eli vinosti niin pitkään, kunnes vastaan tulee este. Ninja voi hyökätä vain viereisiin ruutuihin.

Tarkka-ampuja, eli Sniper, on väriltään vaaleansininen. Tarkka-ampujan aloitus-HP on 10 ja se tekee hyökkäyksellään 5 vahinkoa. Se voi liikkua vuorollaan viereisiin ruutuihin, ja voi hyökätä x- ja y-suunnassa niin pitkälle, kuin se näkee (eli ei puiden tai kivien läpi).

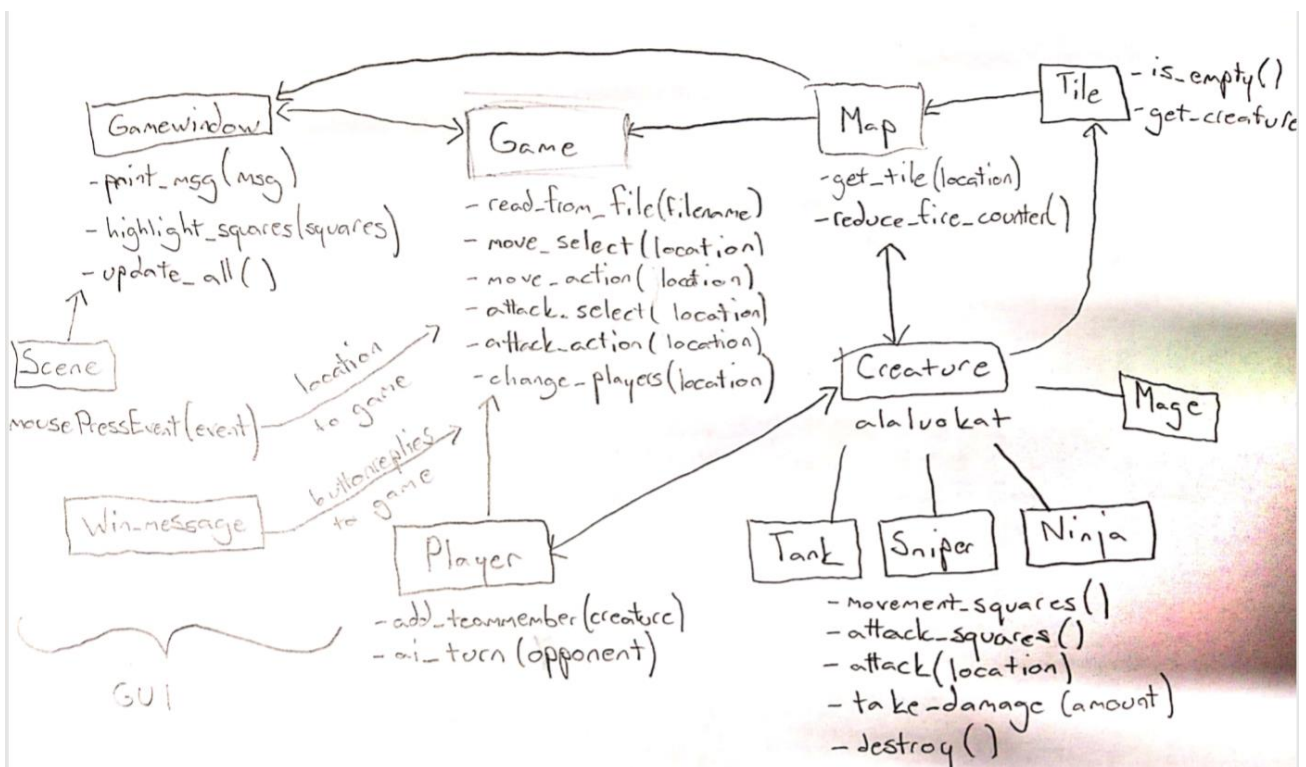
Hahmojen kuollessa niiden ikonit katoavat peliruudukosta. Peli päättyy, kun jommankumman pelaajan joukkue on tuhottu. Silloin ilmestyy info-ikkuna, joka

onnittelee voittanutta pelaajaa, ja ikkunasta voi valita joko lopettamisen tai uuden pelin.

### 3. Ulkoiset kirjastot

Ohjelma ei käytä muita ulkoisia kirjastoja PyQt5:n lisäksi.

### 4. Ohjelman rakenne



Yllä olevassa kaaviossa on havainnollistettu ohjelman luokkajako niin havainnollisesti kuin osasin. Siitä puuttuu vain luokat Location, Creature\_graphitem ja Tile\_graphitem. Lokaatio sen vuoksi, että se on luokka lähinnä helpottamaan lokaation käyttämistä ja kaikki muut luokat ovat siitä riippuvaisia. Grafiikkaobjektien omat luokat jätin pois sen vuoksi, että ne olivat lopulta ohjelman kannalta aika turhia, sillä päädyin tekemään kaikelle yhteisen klikkihandlerin.

Pyrin rakentamaan ohjelman niin, että peli toimisi taustalla myös ilman grafiikoita, jotta esimerkiksi vaihtaminen toiseen grafiikkakirjastoon olisi kätevää ja että pelilogiikkaa voi päivittää itsenäisesti. Grafiikat siis koostuvat peli-ikkunasta, jossa on

kaikki pelin aikana näkyvä. Pelikentän grafiikkascene on oma luokkansa, sillä toteutin sceneen oman klikkihandlerin, joka informoi pelille pelikentän koordinaatteina mihin käyttäjä on klikannut. Win\_message-luokka on QMessageBoxin perivä luokka, jonka tarkoituksena on lähinnä kertoa pelille, suljetaanko se vai aloitetaanko uusi peli.

Game-luokka suorittaa pelin alustamisen ja pelin kulusta huolehtimisen. Sen tärkeimmät metodit ovat neljä metodia, jotka suorittavat vuorossa tapahtuvat neljä toimintoa. Move\_select, move\_action, attack\_select ja attack\_action. Tämän syklin välissä vaihdetaan pelaajaa metodilla change\_players.

Map ja Tile ovat aika yksinkertaisia luokkia, niiden tarkoituksena on vain kuvata karttaa ja tiiliä, joista se koostuu. Kartta pitää huolen kartassa tapahtuvista muutoksista, eli tällä hetkellä käytännössä vain tulen vuorolaskurista ja puiden poistamisesta. Tiilet taas tietävät oman lokaationsa ja tyyppinsä lisäksi myös niissä mahdollisesti olevan hahmon tiedonkulun helpottamiseksi.

Hahmoja kuvataan luokalla Creature, jolla on neljä kutakin hahmotyyppiä vastaavaa alaluokkaa. Hahmot suorittavat esimerkiksi sallittujen liikkumis- tai hyökkäysruutujen laskennan sekä hyökkäämisen ja vahingoittumisen.

Pelaajia taas kuvataan Player-luokalla. Pelaaja-luokan on tarkoitus pitää kirjaa kummankin pelaajan omista hahmoista pelin puolesta. Toteutin kyseiseen luokkaan myös tekoälyn toiminnan pelaajaluokan metodeina, joka olisi ehkä kannattanut tehdä omaan luokkaansa myöhemmän muokkaamisen ja parantelun helpottamiseksi.

Main-funktiossa on hoidettu ensimmäisen pelin alustaminen ja eri metodien kytkeminen toisiinsa, minkä olisi ehkä voinut siirtää pelin alustukseen.

## 5. Algoritmit

Pelissä ei ole tekoälyn lisäksi muita algoritmeja.

Tekoäly toimii seuraavasti:

1. Liikkuminen. Liikkuminen toimii seuraavalla prioriteettijärjestyksellä. Jos edellinen toiminto ei ollut mahdollista suorittaa, suoritetaan seuraava.
  - Ensisijaisesti liikutetaan tulella oleva hahmo pois tulesta, jos hahmo pystyy poistumaan siitä yhdellä vuorolla.
  - Seuraava prioriteetti on siirtää heikommat pelaajat (maagit ja tarkka-ampujat) pois jos vastustajan tankki on hyökkäyskäytävällä, eli jos vastustaja pystyy tappamaan sen yhdellä iskulla.

- Käydään läpi hahmot, jotka eivät ole hyökkäysetäisyydellä vastustajan pelaajasta ja katsotaan, pystyykö joku niistä liikkumaan hyökkäysetäisyydelle.
  - Käydään läpi oman tiimin tankit ja ninjat ja pyritään liikuttamaan joku niistä mahdollisimman lähelle vastustajan heikointa pelaajaa.
  - Käydään läpi oman tiimin maagit ja tarkka-ampujat ja pyritään liikuttamaan joku niistä mahdollisimman lähelle vastustajan heikointa pelaajaa.
2. Hyökkäys. Käydään ensin läpi tiimin jokainen hahmo ja tehdään sanakirja, jossa hahmoa vastaa lista mahdollisista hyökkäysruuduista, missä on vastustajan hahmo.
- Hyökätään maagilla vastustajan heikoimpaan pelaajaan mihin voi osua
  - Hyökätään tankilla vastustajan heikoimpaan pelaajaan mihin voi osua
  - Hyökätään tarkka-ampujalla vastustajan heikoimpaan pelaajaan mihin voi osua
  - Hyökätään ninjalla vastustajan heikoimpaan pelaajaan mihin voi osua

Molempiin voisi lisätä parametreja ja suunnittelua useamman vuoron eteenpäin. Liikuttamisen viimeinen osa ei myöskään ole jälkeensä ajateltuna kovin fiksu, sillä molemmat hahmotyypit pystyvät ampumaan kaukaa. Tekoäly ei osaa myöskään taktikoida tulen vaikutuksilla pelikenttään, eikä ota huomioon tulen koko vaikutusalueita, pelkästään Creature-luokan laskeman hyökkäysruudun.

Tekoälyn algoritmi on kyllä helposti paranneltavissa, sillä suurin osa vaiheista on omana funktionaan.

## 6. Tietorakenteet

En tehnyt omia tietorakenteita, käytin pythonin omia. Hyödynsin paljon listoja ja sanakirjoja, eli muuntuvatilaisia rakenteita, sillä ne sopivat parhaiten käyttötarkoituksiini, missä hahmojen määrä, tiilien määrä tms voi vaihdella.

## 7. Tiedostot

Pelikenttä luetaan tekstitiedostosta. Ensimmäisellä rivillä ilmoitetaan kentän ulottuvuudet kokonaislukuina, ensin x ja sitten y. Seuraavilla riveillä kerrotaan kentän rakenne formaatissa, jossa yksi merkki vastaa pelikentän yhtä ruutua. Eri tiilityyppejä ilmaistaan numeroilla. Esimerkiksi

000100  
002010  
000000  
000200

Vastaisi 4 x 6 pelikenttää, jossa 0 tarkoittaa vapaata tiiltä, 1 puuta ja 2 kiveä.

Tällä hetkellä ohjelma lukee kartan maps-kansiossa olevasta "map2.txt"-tiedostosta, eli jos haluaa muokata pelikenttää, se tulee tehdä kyseiseen tiedostoon ellei halua muokata lähdekoodia.

Joukkueen kokoonpano luetaan myös tekstitiedostosta. Ensin kerrotaan pelaaja numero 1:n tyyppi (Human tai Ai, isoilla ja pienillä kirjaimilla ei ole väliä). Sen jälkeen luetellaan pelaaja 1:n joukkue, jokainen hahmo omalla rivillään. Hahmosta kuuluu kertoa hahmon tyyppi (ei väliä isoilla ja pienillä kirjaimilla), nimi ja aloituspaikan x- sekä y-koordinaatti, kyseisessä järjestyksessä ja kaikki toisistaan vähintään yhdellä välilyönnillä erotettuna. Jos annettu sijainti on virheellinen (kartan ulkopuolella tai varattu) kyseistä hahmoa ei lisätä kartalle. Pelaaja 2:n asetukset noudattavat samaa formaattia ja niiden tulee olla aina tiedostossa toisena. Tietokonevastustajan tulee aina olla pelaaja 2. Joukkueiden kokoonpanot luetaan tiedostosta "settings.txt".

## 8. Testaus

Pelin testaus on toteutettu ainoastaan pelaamalla peliä paljon käytännössä. Olen testannut sitä itse ja koittanut huomioida mahdollisimman monia eri tilanteita ja olen antanut pelin myös muille testattavaksi, jotta voisin katsoa, tuleeko normaalissa käytössä vastaan ongelmatilanteita. Suunnittelemani rakenne toimi hyvin, en joutunut lennossa muuttamaan rakennetta juurikaan ilmestyneiden ongelmien takia. Testasin ohjelman toimivuutta myös yleensä heti lisättyäni jotain uutta.

## 9. Ohjelman tunnetut puutteet ja viat

En ole havainnut projektissa vikoja, esim. kaatumiseen johtavia vikoja, kunhan tiedostoissa noudattaa määrittelemääni formaattia. Toisaalta en ehtinyt testaamaan tekoälyä niin pitkään kuin muita ominaisuuksia, joten siihen saattoi jäädä mahdollisesti joitakin vikoja.

Tekoäly on jokseenkin tyhmä, se ei esimerkiksi ota huomioon onko maagin hyökkäyksen vaikutusalueella omia pelaajia. Sitä vastaan pystyy kuitenkin pelaamaan eikä se ole aivan liian helppo vastus. Tulossa oleva hahmo ei välttämättä näytä olevan

tulessa, sillä molemmat kuvat ovat saman kokoisia neliöitä, ja hahmot on piirretty kartan tiilien päälle.

Tiedostojen lukemiseen ei ole toteutettu kovin kattavaa virheen käsittelyä, siihen ei jäänyt aikaa joten oletan, että käyttäjä osaa noudattaa annettua formaattia. Se, että tietokonevastuksen tulee aina olla pelaaja numero 2, on puute joka johtuu ohjelman vuororakenteesta. Jos minulla olisi ollut enemmän aikaa, olisin halunnut korjata sen.

Ohjelmassa ei ole myöskään graafista asetus-ikkunaa tai tapaa kysyä käyttäjältä mistä tiedostoista asetukset halutaan lukea, joten asetukset tulee kirjoittaa ennalta määrättyihin tiedostoihin.

### 10. 3 parasta ja 3 heikointa kohtaa

Parhaat:

- Käyttäjäystävällisyys. Mielestäni pelaaminen on intuitiivista ja onnistuu myös lukematta kaikkia ominaisuuksia etukäteen. Kaikki tarvittavat ohjeet tulostetaan teksti-ikkunaan ja hahmojen liikuttelu ja niillä hyökkääminen on helppoa korostettujen ruutujen vuoksi. Skip- ja Undo-napit helpottavat myös pelaamista, esimerkiksi uuden pelaajan ei tarvitse tietää hahmojen liikkumistapoja, vaan voi esikatsella miten hahmot pystyvät liikkumaan ja sen jälkeen perumaan valinnan.
- Pelimekaniikat. Kahdella ihmispelaajalla peli on yllättävän tasaväkinen ja mielenkiintoinen. Hahmotyypit ovat selkeän erilaisia toisistaan ja luotu eri tarkoituksiin, joten pelin voittamiseksi joutuu oikeasti taktikoimaan.
- Konfiguroitavuus. Oman kartan lisäksi peliä voi muokata paljon mielensä mukaan. Voi valita pelaako yksin tietokonetta vastaan vai kaksin, oman joukkueen voi koota ja nimetä haluamallaan tavalla ja hahmojen aloituspaikat voi määrätä itse. Tällä tavoin pelin haastetasoa voi muokata itse tai kokeilla miten erilaiset asetukset vaikuttavat taktikointiin. Jos esimerkiksi tietokonevastus tuntuu liian helpolta, voi joukkuerakenteita muuttamalla luoda itselleen lisää haastetta.

Heikoimmat:

- Tekoäly. Tekoäly on toteutettu kiireessä ja rakennettu vain muutaman funktion ympärille, joten se on helposti jallitettavissa ja saattaa tehdä tyhmiä päätöksiä. Olisin mielelläni hionut sitä monipuolisemmaksi.
- Graafinen toteutus. En nähnyt hirveästi vaivaa graafisen ilmeen suunnitteluun ja toteutukseen, joten se ei varmasti ole paras mahdollinen. Nappien kokoa ja sijaintia olisi voinut parannella ja tällä hetkellä ikkunan

koko ei ole suhteutettu pelikentän kokoon, joten sitä saattaa joutua mahdollisesti kasvattamaan. Lisäksi hahmojen tiedot voisivat näkyä jossain ruudulla jatkuvasti, tooltip ei ole yhtä kätevä käyttäjälle mutta valitsin sen siksi että se oli helppo toteuttaa.

- Siisteys, tai siis sen puute. Ohjelman tiedostoissa on paljon ylimääräistä tai kommentoitua koodia sillä jouduin tekemään paljon pieniä muutoksia kesken koodailun eikä lopulliseen siistimiseen jäänyt paljon aikaa. Eri tiedostojen ja luokkien riippuvuudet ovat myös hiukan sekavat ja jouduin sijoittelemaan peliä alustaessa paljon metodeja toisista luokista toisiin. Main-funktion olisi myös voinut siirtää omaan tiedostoonsa.

## 11. Poikkeamat suunnitelmasta

Jouduin tekemään muutamia muutoksia alkuperäiseen suunnitelmaan.

Yksinkertaistin vuororakennetta yhden hahmon liikuttamiseen ja yhden hahmon hyökkäämiseen, koska tajusin projektia toteuttaessa ja testaillessa, että se toimii pelimekaniikallisesti paremmin ja tekee vuoroista nopeampia. Suunnitteluvaiheessa en ollut päättänyt kovin tarkasti, miten pelaaja antaa syötteet ohjelmalle, mutta suunnitelmassa mainittiin eri näppäimiä. Päätin kuitenkin muuttaa kaikki syötteet hiirellä tehtäviksi pelikokemuksen yksinkertaistamiseksi. Tämä valinta yksinkertaisti myös pelin rakennetta, koska peli saa vastaan (nappeja lukuunottamatta) vain yhdenlaista dataa, eli klikkauksen koordinaatit.

Jätin myös ajanpuutteen vuoksi paljon lisäominaisuuksia pois, kuten esimerkiksi pelitilanteen tallentamisen ja graafiset valikkoikkunat. Tallennustoiminnon hylkäsin jo aika aikaisessa vaiheessa, mutta aika pitkään vielä oli ajatuksena tehdä graafinen valikko, mistä saa esimerkiksi valita yksin- tai kaksinpelin, mitä karttaa käytetään. Se oli tosin prioriteettilistallani viimeisenä, joten jätin sen tekemättä.

Aikaisemmin suunnittelemani toteutusjärjestys ja karkea luokkajako toimi kyllä hyvin työnteon tukena, enkä poikennut niistä juurikaan. Olisin tosin voinut aloittaa tekoälyn kirjoittamisen aiemmin.

## 12. Toteutunut työjärjestys ja aikataulu

Työn etenemisen näkee aika hyvin git-repositorystä, mutta kuvailen sen lyhyesti tässä.



Aloitin siitä, että saisin perusgrafiikat toimimaan. Ne olivat ensimmäisessä muodossaan valmiit ja toimivat 28.3. Sen jälkeen oli luonnollinen jatkumo implementoida kenttää alustaessa sen lukeminen tiedostosta, jonka sain tehtyä samana päivänä. Samoihin aikoihin sain myös valmiiksi hahmotyypit ja toiminnon hahmojen tietojen tarkastelemiseen. Tässä muodossa projekti oli checkpointissa.

1.4. lisäsin tekstikonsolin ja aloitin toteuttamaan pelin kulun toteuttamista ja kommentojen vastaanottamista pelaajalta. Tämä vaihe vaati hieman aikaa ja pohdintaa ja oli valmis vasta 10.4.

15.4. lisäsin tulihyökkäykset jonka jälkeen aloitin pelitestauksen. Tämän jälkeen lähinnä korjailin ja hioin testatessa vastaan tulleita ongelmia, esimerkiksi lisäsin painikkeet ja korjasin muutamia bugeja.

19.4. yksinkertaistin main-funktiota ja toteutin joukkueiden asetuksen lukemisen tekstitiedostosta. Tämän ja viimeisten committien välissä ei ole mitään, mutta lisäsin tähän aikaan myös pelin lopputtua tulevan teksti-ikkunan. Pelin uudelleen aloittamisessa oli kuitenkin vielä ongelmia.

22.4. aloitin tekoälyn tekemisen ja sain sen valmiiksi ja toimimaan seuraavana päivänä. Tällöin sain myös pelin uudelleenaloittamisen toimimaan.

### 13. Arvio lopputuloksesta

Lopulta olen aika tyytyväinen lopputulokseen. Sain kaikki pyydetty toiminnallisuudet valmiiksi ja peliä on jopa hauska pelata. Siinä ei tosin ole juuri mitään pyydettyä enempää ja pyydetty tekoäly ei ole paras mahdollinen, joten ehkä kokonaisuudessa projektini on hieman keskinkertaista parempi, muttei kuitenkaan mitenkään erinomainen. Lisäksi palautus on myöhässä, joten hyväksyn täysin kaikki siitä johtuvat pistevähennykset. Olen lähinnä iloinen, että sain projektin valmiiksi.

### 14. Viitteet

Ohjelmoinnin tukena käytin lähinnä Qt5:n dokumentaatiota

<https://doc.qt.io/qt-5/>

ja seuraavaa yksinkertaista PyQt5-tutorialia

<https://pythonspot.com/pyqt5/>

Kaikki grafiikka on lisensoitu vapaaseen ei-kaupalliseen käyttöön ja löydetty osoitteesta, mikään ei ole itse piirtämään

<https://opengameart.org/>

## 15. Liitteet

Kuva pelistä käytössä

