

TIE-02402 Ohjelmointi 3: Tekniikat

Pirkanmaan valtaus

Projektin dokumentaatio

Linnea Viitanen, Aleksi Myllyniemi
1.12.2019

Pelin säännöt

Peli on tarkoitettu kahdelle pelaajalle. Pelissä pyritään valtaamaan mahdollisimman monta laattaa kartalta, joten voittaja on se, joka ehtii valtaamaan yli puolet pelissä vallattavissa olevista laatoista. Pelaajien värit ovat sininen ja punainen, ja pelaajien valtaamat alueet erottuvat toisistaan näillä väreillä.

Laattoja on pelissä viisi erilaista: ruoho, vesi, kivi, metsä ja marjapuska. Näistä laatoista marjapuska on sellainen, jota kumpikaan pelaaja ei voi vallata. Se on tarkoitettu puolueettoman resurssien keräämiselle. Laatan valtaaminen tapahtuu rakentamalla siihen. Rakennukseen pelaaja tarvitsee erilaisia resursseja, joita pelissä on seuraavanlaisia

- Kulta (GOLD), Malmi (ORE), Ruoka (FOOD), Puu (WOOD), Kivi (STONE)

Molemmille pelaajille annetaan pelin alussa tietty määrä resursseja, joilla peli lähtee käyntiin. Resursseja kerätään lisää rakentamalla erilaisia rakennustyypppejä, joille on mahdollista lähettää työläisiä keräämään resursseja.

Rakennustyytit ja rakentaminen

Pelissä on kolmenlaisia rakennuksia, joita pelaaja voi rakennuttaa.

- Talo (HOUSE), Kaivos (MINE), Laituri (DOCK)

Talon pystyy rakentaa vain ruoho- tai metsälaatalle. Metsälaatalle rakennettaessa pelaaja ei menetä puuresursseja. Kaivoksen pystyy rakentaa vain kivelle. Laiturin taas pystyy rakentaa vain veden äärelle. Yhdelle laatalle pystyy rakentaa vain yhden rakennuksen. Alla olevassa taulukossa näkyy rakennusten ostamiseen tarvittavat resurssit.

	Kulta (GOLD)	Malmi (ORE)	Ruoka (FOOD)	Puu (WOOD)	Kivi (STONE)
Talo (HOUSE)	50	-	25	100	50
Kaivos (MINE)	50	-	25	150	-
Laituri (DOCK)	50	-	50	100	-

Resurssien kerääminen

Resurssien kerääminen tapahtuu pelissä ostettavien työläisten avulla. Työläisiä on kahdenlaisia, perustyöläisiä (worker) ja kaivajia (miner). Molempia tyypppejä saa ostettua pelaajan omaa tukikohtaa klikkaamalla. Pelaaja voi lähettää työläisen kaivokselle, laiturille tai marjapuskille (berrybush). Työläisistä vain kaivaja pystyy menemään kaivokselle, mutta ei mihinkään muualle.

Kun pelaaja lähettää työläisen resurssien keruuseen, on työläinen poissa (unavailable) kahden pelaajan oman vuoron ajan. Työläisen tullessa takaisin pelaajalle automaattisesti lisätään työläisen tuomat resurssit. Aktiviteetin omaavalle laatalle saa lähettää korkeintaan kolme työläistä kerrallaan. Työläisten lukumäärä laatalla ilmaistaan laatassa näkyvän numeron avulla. Puolueettomalla marjapuskalla voi samanaikaisesti olla molempien pelaajien työläisiä, mutta laatan maksimiluku on silti kolme.

Pelaajan ollessa valmis, hän luovuttaa vuoron seuraavalle pelaajalle (END MOVE).

Tärkeimpien luokkien vastuujako

1. GameEventHandler (gameeventhandler.hh / gameeventhandler.cpp)

Tämän luokan vastuulla on työläisten toiminnan kontrolloiminen pelin aikana, sekä työläisten aktiviteeteista muodostuneiden resurssien muuttaminen. Lisäksi luokan vastuulla on pelaajan tekemien pelinsisäisten ostotapahtumien kontrollointi (rakennuksen ja työläisen ostaminen).

2. ObjectManager (objectmanager.hh / objectmanager.cpp)

Tämän luokan vastuulla on varastoida pelin laatat sekä rakennukset, ja palauttaa niitä käytettäväksi muissa luokissa. Tämä luokka myös kutsuu jokaiselle laatalle QGraphicsSceneä, jolle laatat piirretään pelikartaksi.

3. MapWindow (mapwindow.hh / mapwindow.cc)

Tämän luokan vastuulla on initialisoida koko peli, sekä sen käyttämät resurssit. Luokassa rakennetaan pelaajat, sekä tarkistetaan pelitilannetta jokaisen vuoron jälkeen. Luokka on myös vastuussa vuoron vaihtumisesta, sekä käyttöliittymän oikeellisuudesta vuoronvaihdon jälkeen (Asianmukaiset tekstit, esim. Game Over näytetään vasta kun peli loppuu).

4. Gamescene (gamescene.hh / gamescene.cpp)

Tämän luokan vastuulla on pelin kartalla tapahtuvat muutokset. Kartan painamiseen liittyvät toiminnot, kuten erilaisten dialogien avaaminen, sekä asianmukaiset vasteet (pelaajat eivät voi operoida kartalla toisen omistamia objekteja). Lisäksi luokka on vastuussa kartalle piirrettävistä objekteista sekä niiden ulkonäöstä. Luokan tulee päivittää kuvia sitä mukaa, mitä pelaajat pelissä tekevät.

Tiedossa olevat ongelmat tai puutteet

1. Pelaajien tulisi pystyä päivittämään HeadQuarters (sille on oma nappulansa dialogissa, joka avautuu, kun klikataan HQ:ta), mutta aika ei riittänyt toiminnallisuuden valmistumiseen.
2. Includet cpp-tiedostoissa. Näitä alettiin viime hetkellä siirtelemään .hh tiedostoihin, jota jarrutti se, että includet olivat usein ristiriidassa keskenään.
3. Kovakoodatut arvot ympäri ohjelmakoodia.
4. Yksikkötesteissä kahden funktion testaaminen epäonnistuu, vaikka kyseiset funktiot toimivat pelissä.
5. Käyttämättömät funktiot: jotkin toiminnallisuudet jäivät vajaiksi, kun aika loppui kesken. Ei kuitenkaan raaskittu poistaa niitä projektista, koska ajateltiin että saatetaan jatkaa pelin kehitystä myös kurssin jälkeen.
6. Graafinen käyttöliittymä saattaa näyttää erilaiselta virtuaalityöpöydällä tarkasteltaessa (testatessa huomattiin). Dokumentin lopussa olevassa liitteessä näkyy, miltä pelin kuuluisi näyttää. (Johtuneeko erikoisesta fontista)
7. Yhdelle laatalle pystyy rakentaa vain yhden talon, mutta laatan buildingeja kutsuvassa funktiossa palautuu vektori. Ei ehditty vaihtaa tähän järkevämpää ratkaisua, vaikka se olisikin ollut helppoa.

Sovittu ja toteutunut työnjako

Sovittu työnjako

1. Linnea
 - Graafinen toteutus: Kartan piirtäminen sekä alussa että pelin aikana (rakennusten sekä erilaisten toimintojen piirtyminen kartalle), käyttöliittymän toteutus (mapwindow sekä mahdolliset popup-ikkunat), jokaisen laatan graafinen suunnittelu
 - GameEventHandlerin ja ObjectManagerin toimintaan saaminen (periyttäminen jne), jotta kartan piirto onnistuu
2. Aleksi
 - Pelinmekaniikan toiminta (iso osa GameEventHandlerin funktioista) ja laatta ja rakennus ja pelaajapohjaluokkien periytyminen ja toiminnallisuus StudentResourcemaps luominen
 - Housing resurssi toiminnallisuuden luominen
 - oma resurssiluokka

Toteutunut työnjako

1. Linnea

- Graafinen toteutus (kaikki sovitut kohdat), `gamescene.cpp/hh`
- Rakennusten ostoon tarkoitettu dialog sekä siihen liittyvä toiminta
- Dialog, joka aukeaa, kun pelaaja klikkaa sellaista laatkaa, jolla voi suorittaa aktiviteetin (berrybush tai omat laatat, joissa aktiviteetti): työläisen lähettäminen laatalle, ja laatan kuvan päivittyminen sen perusteella, kuinka monta workeria laatalla on.
- Starting screen
- Mapwindow ja siinä esiintyvät toiminnot: mm. voiton tarkistus, pelivuoron vaihtuminen, pelaajien alustus

2. Aleksi

- Pelimekaniikan toiminta (kaikki sovitut kohdat), `gameeventhandler.cpp/hh`, sekä osa `objectmanagerista`
- Workerin toiminnan pohjafunktiot
- Laattatoiminnallisuuden pohjafunktiot
- Pelaajaresurssien managerointi
- Yksikkötestien tekeminen
- Housing luokka päätettiin jättää pois, vaikka luokka olikin suurin piirtein toteutettu

Ohjelman toiminta

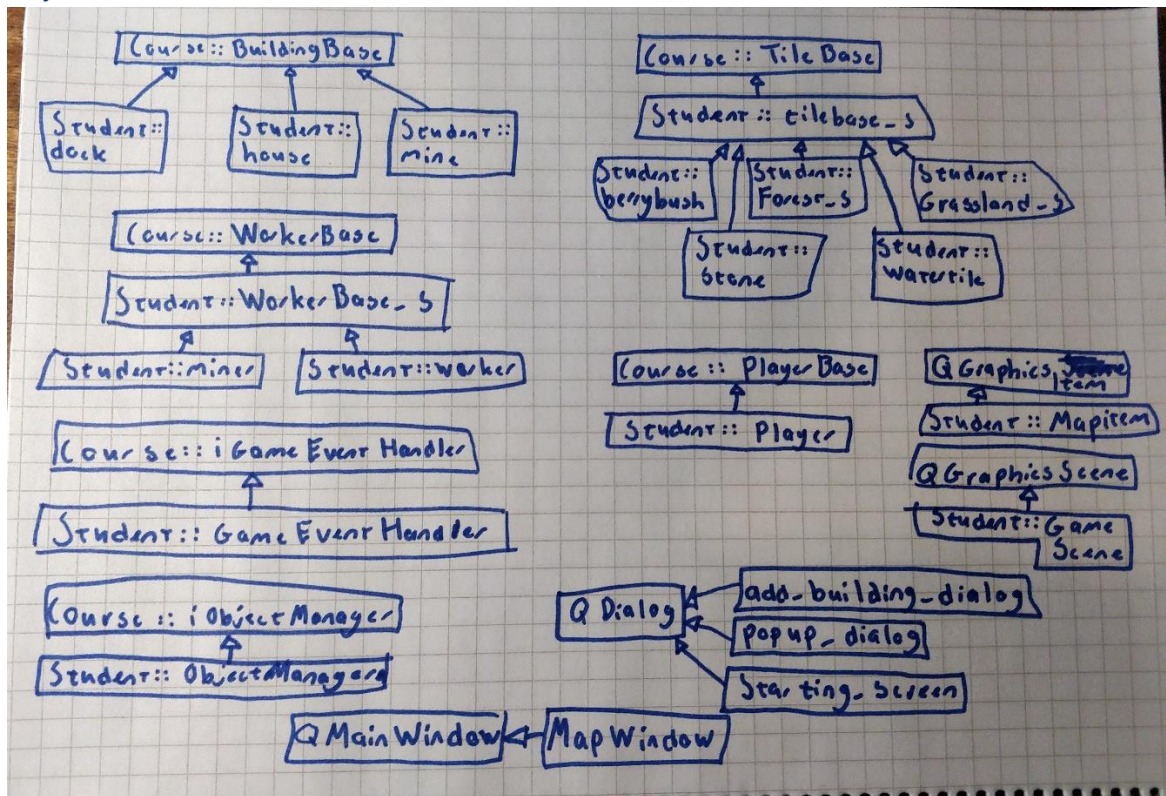
Ohjelma käynnistyy `mapwindowin` kautta, jossa pelaajille luodaan omat oliot `Player-` luokan avulla. `Mapwindowissa` luodaan myös `GameEventHandler` olio pelin tapahtumien kontrollointia varten, `ObjectManager` olio pitämään kirjaa pelissä esiintyvistä laatoista sekä rakennuksista, sekä `GameScene` olio vastaamaan pelin kartalla tapahtuvista asioista.

`ObjectManagerissa` olevaan tietorakenteeseen tallentuvat laatat saadaan luomalla niistä jokainen `TileBase` olioksi `WorldGenerator` olion avulla. `GameScenessä` `TileBase` oliot piirretään kartalle, kun ne yksi kerrallaan sidotaan niitä vastaavaan `MapItem` –olioon. Laattaolioille annettaville rakennuksille luodaan `BuildingBase` oliot, jotta laatta tietää, millainen rakennus sillä on.

`MapWindowiin` ja `GameSceneen` on yhteydessä kaksi `Dialogia`, jotka initialisoidaan pelin erilaisten tapahtumien mukaan.

Liitteet

Ohjelman rakenne



Käyttöliittymän graafinen ilme

