Alussa ensimmäisestä tapauksessa tuttu osio ohjelman selityksestä, sitä on täydennetty tähän toiseen palautukseen UML-luokkakaaviolla, sen selityksellä, luokkien välisillä vastuilla sekä pelin etenemiskuvauksella.

Tampereen kartta

- klikkiliikkuminen
 - -> kauppaan tai parkkihalleihin
- rahatilanne, Lauran energiataso ja panokset

parkkihalli

- vaihtelevat pohjat, matriisien avulla
- näppäimistö + hiiri liikkuminen
- Lauran health/armor bar
- kyborgien tietoja
- · kierrokselta kertyneet rahat

kauppanäkymä

- rahatilanne
- kyborgien varusteiden taso
- Lauran energiataso, panokset ja varusteiden taso

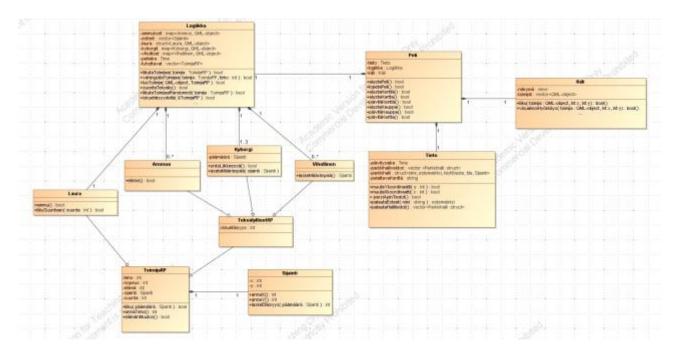
Peli koostuu kolmesta näkymästä, joista ensimmäinen on karttanäkymä. Karttanäkymän voi ajatella olevan pelin päänäkymä, sillä siinä tapahtuu kaikki liikkuminen. Toinen näkymä on kauppanäkymä, jossa pelaaja pääsee ostamaan uusia tarvikkeita tai tarkastelemaan ja vaihtelemaan niitä. Viimeinen pelin näkymä on parkkihallinäkymä. Siinä pelaaja tulee parkkihalliin, jossa tapahtuu kaikki pelaaminen. Kaikki parkkihallit ovat erilaisia, mutta ne on generoitu samalla tavalla, karttamatriisien tms. avulla.

Pelin pelaaminen on yksinkertaisuudessaan sitä, että käyttäjä kiertää parkkihallista toiseen mukanaan kyborgit. Parkkihallista pääsee pois joko voittamalla parkkihallin viholliset tai pakenemalla. Parkkihallissa pelaaja ohjaa näppäimistön avulla Lauraa ja hiirellä voi käskyttää kyborgeja auttamaan mistä ikinä haluaakaan. Kyborgien toiminta perustuu kuitenkin jonkinlaiseen tekoälyyn ja tämä tarkoittaa, ettei niitä pääse täysin ohjaamaan.

Parkkihallin tapahtumissa aiheutuneet vahingot Lauralle jäävät voimaan, mutta kyborgit saa aina hallista poistuttua takaisin käyttöönsä ja täysiin voimiin. Jos kyborgeille haluaa uutta varustusta tulee se hommata ja vaihtaa kauppanäkymässä. Kyborgien varustus on kuitenkin yhtenäinen. Lauralle aseiden vaihtaminen taas toimii myös parkkihallissa pelitilanteen ollessa kesken.

Pärjätessään parkkihallitaistossa pelaaja saa kerättyä tuhoutuneiden autojen romua, jolla hän voi ostaa parempia varusteita. Kun yhdestä hallista on selvitty voittajana, päästään jatkamaan seuraavaan, vaikeampaan halliin. Lopulta peli loppuu, kun kaikista halleista on selvitty voittajana.

Apista saatavien tietojen perusteella luodaan parkkihallit eri vaikeustasoihin. Näissä parkkihalleissa on erimäärä vihollisia ja esteitä sekä niiden koko ja muoto saattavat hieman vaihdella. Jos parkkihallissa on enemmän vihollisia, on myös todennäköisempää, että sinne on löytänyt tiensä myös hieman fiksumpaa tekoälyä omaavia pahiksia.



Kaikki informaatio siirtyy suoraan mallissa päivittymisen jälkeen näkymään. Näkymä muodostuu kolmesta eri tilasta, jotka ovat pelinkartta-, kauppa- ja parkkihallinäkymä ja se koostuu QML osasta ja sitä hallinnoivasta C++ luokasta. Käytämme siis MW-mallia.

Logiikkaan kuuluu vielä funktio, millä kysytään pelin tilaa ja saadaan, että onko parkkihallin pelitilanne loppunut. Tieto-oliolta on myös unohtunut esim. Lauran elämä, ja aseiden tehot.

Lauran liikkumista ja ampumista (näppäimistön perusteella) ei ole vielä tarkemmin mietitty, joten siitä tulee vielä jokunen funktio ja attribuutti. Myöskään pelin pisteitä ja näyttämistä, eikä taistelijoiden ominaisuuksien esittelyä käyttöliittymässä ole esitetty UML-kaaviossa.

Estematriisina esitelty attribuutti voi olla esimerkiksi vektori, jonka sisällä on kaikkien esteiden koordinaatit. Parkkihalli structi sisältää XML-tiedostosta saadun datan.

- Käli-luokka on vastuussa QML-osalta tiedonnäyttämisestä käyttöliittymässä ja C++ taas osaltaan sen tiedon välittämisestä käyttöliittymään.
- Tieto-luokka on vastuussa API-rajapinnan hyödyntämisestä ja sen tietojen käsittelystä.
- Logiikka on vastuussa parkkihallin pelitilanteen aikaisista olioista.
- Peli taas on vastuussa Logiikka, Tieto ja Käli olioista, sekä pelin luonnista ja sen kulusta.

Lopuksi vielä lyhyt käyttötapausesimerkki:

Käyttäjä avaa pelin jolloin kaikki näkymät luodaan, mutta vain karttanäkymä laitetaan näkyville ja XML-datasta otetaan tiedot irti. Tämän jälkeen käyttäjä valitsee jonkun hallin ja pelin näkymä vaihtuu parkkihallinäkymään. Samalla luodaan myös logiikka ja pelilaudalle erilaiset toimijat. Laura kuuntelee näppäimistöä ja tekoäly suorittaa kaiken muun toiminnan. Lopuksi pelaaja joko pääsee kentän läpi tai häviää, jolloin siirrytään takaisin karttanäkymään. Nyt pelaaja voi vielä mahdollisesti käydä kaupassa muuttamassa varustustaan, jolloin tiedot päivitetään Tieto-oliolle. Tätä jatkuu, kunnes peli päästään läpi.

Aluksi olemme ajatelleet jakaa vastuualueet niin, että Miika keskittyy QML-käyttöliittymäpuoleen, Annimari periytettyjen luokkien tekemiseen ja Ilkka muihin luokkiin. Tämän lisäksi se kenellä on pienin työtaakka, rupeaa valmistelemaan XML-tiedoston parsimista.