

mainmenu-luokka

- Tarjoaa käyttäjälle navigoinnin muihin näkymiin:

- MainWindow
- controls
- highscores

Mainmenun tehtävänä on tarjota pelaajalle päävalikko, josta tämä voi halutessaan aloittaa pelin. Pelin aloittamisen lisäksi päävalikosta voidaan avata controls-Qdialog, joka näyttää käyttäjälle pelin käyttämät näppäimet ja kertoo loppukäyttäjälle sopivassa muodossa pelin toiminnasta ja tavoitteista.

Päävalikosta pääsee myös highscores-Qwidgetiin. Se näyttää pelaajalle leaderboard.txt -tiedostosta haetun top10-pistetaulukon.

MainWindow-luokka

-Tarjoaa käyttäjälle näkymän itse peliin ja oleellisia tilannetietoja:

- GraphicsViewControls, joka näyttää QGraphicsScenen sisältöä käyttäjälle
- useita QLCDNumber-widgettejä jotka näyttävät pelaajalle tämän pisteet, krediitit, onnistuneet ja epäonnistuneet kohtaamiset, sijainnin koordinaatit ja jäljellä olevan ajan
- QListWidgetin, joka näyttää galaksissa ilmenevät distress-signaalit sekä oleellista tietoa niistä
- QPushButton, josta pelaaja voi lopettaa pelinsä ennenaikaisesti

MainWindowilla on hyvin laaja vastuualue pelissä. Se huolehtii graphicsViewin asetuksista ja alustaa displayt oikeisiin arvoihin. MainWindowin vastuulla on myös päivittää GraphicsViewControlsin näkymää tietyillä signaaleilla. Graafiset objektit päivitetään 16 millisekunnin välein, jolloin saavutetaan sulava 62,5 fps. Samalla lasketaan gameTimerin jäljellä oleva aika minuutteina ja sekunteina ja päivitetään se oikeaan widgetiin.

MainWindow osaa myös reagoida distress signaleihin ja esittää niiden sijainnin pelaajalle. Nämä näytetään QListWidgetissä, jossa sijainti esitetään tähtijärjestelmän nimenä ja koordinaatteina.

Tämän lisäksi MainWindowissa yhdistetään player_shipin signaaleja asianmukaisiin slotteihin. Aluksen liikkuesssa pidetään huoli, että alus pysyy näkyvässä näytöllä. Kun taas alukselta saadaan

signaali osumisesta starsystemobjectiin, MainWindow tarkistaa onko tähtijärjestelmässä aluksia ja laukaisee tämän perusteella joko encounterin tai ilmoittaa pelaajalle tähtijärjestelmän olevan tyhjä.

MainWindow sisältää 3 QTimeria. ActionTimer_ on vastuussa NPC-alusten toimintojen kutsumisesta. FrameTimer_ ajastaa ruudunpäivityksen 16 millisekunnin välein. GameTimer_ taas yhdistyy GameOveriin ja huolehtii pelin lopettamisesta ajan loppuessa. GameOveriin yhdistetään myös Quit_Button, josta pelaaja voi lopettaa pelin ennen aikaisesti.

Encounteria luodessa MainWindowin vastuulla on asettaa sille oikeat attribuutit. Encounterin loputtua UI-elementit päivitetään vastaamaan encounterin tulosta. MainWindow myös poistaa GamePropertiesia hyödyntäen tavatun aluksen galaxystä ja luo satunnaisesti uuden aluksen sen tilalle, jolloin alusten kokonaismäärä pysyy samana.

GameProperties-luokka

-Ei tarjoa minkäänlaista näkymää käyttäjälle

GamePropertiesin vastuulla on luoda eventHandler, Galaxy ja GameRunner pelin ajamista varten. Tämä tehdään setPropertiesilla. Jos asetustiedostojen käsittelyssä havaitaan ongelma, siitä ilmoitetaan ja ohjelma suljetaan.

GameProperties säilöö luomansa handlerin, galaxyn ja runnerin private attribuutteina. Pyydettyessä se palauttaa ne MainWindowille get-funktiolla.

MainWindowUtility-luokka

- Ei tarjoa minkäänlaista näkymää käyttäjälle

MainWindowUtility tarjoaa apufunktioita MainWindowille ja Encounterille. Sen vastuuna on luoda galaxyn tietojen perusteella jokaisesta tähtijärjestelmästä starsystemobject ja lisätä se QGraphicsSceneen. Valmis scene palautetaan MainWindowille.

Tämän lisäksi luokka on vastuussa pelaajan aluksen oikeiden asetusten määrittämisestä. Tämä tehdään kutsumalla setupShip-funktiota MainWindowista. Sille annetaan parametrinä viite player_ship -olioon, jolle muokkaukset tehdään.

Luokka on vastuussa myös MainWindowin ajastinten säädöstä. FrameTimerille asetetaan intervaliksi 16. Tämä on invariantti. 16 millisekunnin frameTimer pitää huolen tasaisesta frame ratesta. ActionTimerin intervaliksi asetetaan 2500 millisekuntia. Tämä vastaa vuoropohjaisen toteutuksen vuoroa, jolloin NPC-alusten toimintoja suoritetaan, päivitetään ja lisätään. GameTimeriin asetetaan ajaksi 3 minuuttia ja se säädetään laukeamaan vain kerran.

MainWindowUtility-luokassa on vielä apufunktio getWaifuType. Sille voidaan antaa economy_type, jonka perusteella se palauttaa vastaavan waifu typen.

HighScores-luokka

-UI sisältää listWidgetin, johon päivitetään pelin top10 pisteet leaderboard.txt -tiedostosta, sekä paluu-näppäimen

HighScores-luokka lukee leaderboard.txt -tiedoston, johon parhaat pisteet on tallennettu pisteiden suuruuden mukaiseen järjestykseen. Tämän jälkeen se tallentaa tiedot playerScore-structeihin, jotka sisältävät pelaajan nimen ja pisteet ja laittaa ne vector-rakenteeseen. Tämä vector-rakenne voidaan palauttaa ja tulostaa UI:n tauluun.

HighScores-luokka hoitaa myös leaderboard.txt -tiedostoon kirjoittamisen tarkastamalla pelin päättyessä riittivätkö pelaajan pisteet top10-listalle. Tämä tehdään vertaamalla pisteitä tiedoston sisältöön. Oikean signaalin saatuaan HighScores tallentaa pelaajan nimen ja pisteet leaderboard.txt -tiedostoon ja poistaa aikaisemmin alimpana olleen piste/pelaaja-rivin.

Game over-luokka

-Ilmoittaa pelin loppumisesta pelaajalle ja tarjoaa tilastoja pelatusta pelistä ja mahdollisuuden tallentaa pisteensä, mikäli ne riittivät highscores-listaan

-Erilliset labelit, jotka ilmoittavat onnistuneet ja epäonnistuneet encounterit ja pelaajan saamat yhteispisteet

-QLineEdit, johon pelaajaa pyydetään kirjoittamaan 3 kirjainta nimekseen, mikäli pisteet riittivät highscores-listaan

-QPushButton erikseen lopettamiselle ja annetun nimen tallentamiselle

Game_over-luokan UI aukaistaan kun pelaaja joko painaa Quit-nappulaa mainwindowissa tai aika loppuu. Luokka hakee statistics-oliosta tiedot pelaajan saamista pisteistä ja onnistuneista sekä epäonnistuneista encountereista. Tiedot asetetaan kolmeen eri labeliin, jotka näytetään pelaajalle UI:ssa.

Tämän lisäksi luokka tarkistaa highscores-luokan funktiota hyödyntäen riittävätkö pelaajan pisteet top 10 -listalle. Jos näin on, UI ilmoittaa siitä pelaajalle ja mahdollistaa pisteiden tallentamisen Save-nappulaa painamalla. Save-nappulan painalluksella pisteet ja pelaajan antama nimi välitetään highscores-luokalle joka kirjoittaa tiedot leaderboard.txt -tiedostoon. Tilanteessa, jossa pelaaja jättää osan merkeistä syöttämättä, täyttää highscores loput 3-merkkisestä nimestä A-kirjaimella.

Jos pelaajan pisteet eivät riitä top 10 -listalle, ilmoitetaan siitäkin pelaajalle labelin kautta. Tässä tapauksessa myös save-painike poistetaan käytöstä.

controls-luokka

Näyttää pelaajalle pelin näppäimet. Sisältää tämän lisäksi vinkkejä pelaajille. Ainoa ikkunan toiminto on cancel_button, joka on yhdistetty ikkunan close-funktioon. Luokka on kokonaisuudessaan toteutettu Qt Designerissa.

Encounter-luokka

UI sisältää:

-Label, jossa kuva kohdatusta aluksesta, valitaan tähtijärjestelmän economy_typen mukaan.

-Label, johon muutetaan kohdatun aluksen vastaukset tehtyihin valintoihin

-4 QPushButtonia, yksi on "info", joka antaa lisää tietoa kohdatusta aluksesta. Loput "compliment", "brag" ja "Offer bribe" ovat tapoja, joilla aluksen kanssa pystyy vuorovaikuttamaan. Yksi näistä valinnoista määritetään oikeaksi valinnaksi ja loput tuottavat hylkäyksen

Encounter-luokka saa mainwindowilta tiedon mihin tähtijärjestelmään pelaajan alus on osunut. Tällöin annetun tähtijärjestelmän economy_type haetaan galaxystä. Economyn mukaan asetetaan encounter-oliolle waifu_type_. Tämän perusteella luokka valitsee kohdatun aluksen kuvan ja alustaa kolmesta vuorovaikutus-buttonista yhden oikeaksi vastaukseksi. Lisäksi eri waifu typeille on toteutettu omat vastauksensa info-painikkeen painamiseen.

Mikäli pelaaja valitsee väärän valinnan kolmesta buttonista ilmoittaa UI siitä ja tarjoaa pelaajalle ainoastaan vaihtoehdon poistua. Muut valinnat poistuvat käytöstä.

Jos taas pelaaja valitsee oikein, vaihdetaan kohdatun aluksen vastaukseksi ja buttoneiden tekstiksi uudet valinnat, minkä jälkeen valitaan buttoneista taas yksi oikea vastaus. Jos pelaaja valitsee vielä tämänkin oikein, ilmoitetaan kohtaamisen onnistumisesta.

Encounter-luokka tallentaa outcome_-attribuuttiin tiedon encounterin lopputuloksesta. Tämän jälkeen MainWindow kutsuu statistics-tyyppisen olion viitteelle encounter::setStatistics-funktiota, joka päivittää statistics-olioon oikeat pisteet ja krediitit.

Lisäominaisuudet:

Reaaliaikainen:

NPC-alusten toiminnot tapahtuvat tietyin aikavälein. Galaksissa voi liikkua milloin tahansa.

Tasainen ruudunpäivitys:

viewportin update()-funktiota kutsutaan tasaisin 16 millisekunnin välein. Muulloin käytössä on noViewportUpdate, eli ruudulla ei muutu mikään.

Minimaalinen ruudunpäivitys:

Scenestä päivitetään vain päivityshetkellä viewportissa näkyvä osa. Myöskään HUDin eri countereita ei päivitetä kuin muutoksen tapahtuessa. Aluksen koordinaatit päivittyvät aluksen liikkumissignaalin perusteella.

Vieritettävä kartta:

Pelin karttaa voi siirtää scrollbarilla tai vetää hiirellä eri suuntiin. Hiiren rullalla näkymää voidaan zoomata sisään ja ulos. Zoomaus keskittyy hiiren kursorin sijaintiin. Kun alusta liikutetaan, varmistaa peli että se on näkyvissä ruudulla ensurevisible-funktion avulla.

Pelaajan aluksen tasainen liike:

Pelaajan alukselle on toteutettu ohjaus nuolinäppäimillä. Niillä voidaan kääntää alusta ja käskä sitä liikkumaan eteenpäin.

Top10:

Parhaat 10 tulosta on säilötty leaderboard.txt -tiedostossa. Tiedosto päivittyy, jos pelaaja saa enemmän pisteitä kuin listan huonoin. Tällöin alin jäsen poistetaan listasta ja pelaajan nimimerkki ja pisteet lisätään oikeaan kohtaan.

Pelin skaalautuminen näytön mukaisesti:

MainWindowia voidaan vetää eri kokoiseksi, jolloin sen elementit skaalautuvat ikkunan kokoon. Full screeniin asettaminen skaalaa elementit koko näytölle.