

SCM Suunnitelma

RTS-peli

Tekijät:
Niko Hämäläinen
Markku Veline
Daniel Suni

Esittely

Markku Velinen, Niko Hämäläinen ja Daniel Suni haluavat kehittää tosiaikaisen strategiapelin. Peli ottaa vaikutteita Warcraft, Starcraft sekä Age of Empire nimisistä peleistä ja sijoittuu fantasiamaailmaan.

Peliä pelataan moninpelinä internetin välityksellä toisia pelaajia tai yksin tietokoneohjattua vastustajaa (tai vastustajia) vastaan. Ideana pelissä olisi ihmisten ja örkkien välinen sotiminen, mutta ”sisällissota” on myös mahdollista. Koko pelimaailma on kolmiulotteisesti toteutettu ja kameraa voi vapaasti pyörittää.

Tarkoituksena pelissä on rakentaa armeija ja tuhota viholliset. Tämä onnistuu keräämällä resursseja (puu ja kulta) ja rakentamalla niistä erinäisiä rakennuksia. Rakennuksissa voidaan sitten esimerkiksi kouluttaa ihmisten puolella jalkaväkeä, jousiampujia, ritareita tai velhoja. Örkeillä jokseenkin samantyylliset joukot, mutta kuitenkin esimerkiksi örkkien jalkaväki olisi huomattavasti voimakkaampaa. Näitä joukkoja johtaa sitten sankari, joka pelaajalla on alussa. Sankarit keräävät roolipelimäisesti kokemusta saaden uusia kykyjä ja taitoja.

Alkutilanteessa pelaajalla on päärakennus, sankari, sekä muutama resurssinkerääjä/rakentaja. Pelaajalla ei itse ole omaa hahmoa pelialueella, vaan voi käskyttää kaikkia ja liikkua vapaasti pelialueella, tosin näkemättä pidemmälle kuin kaikkien yksiköidensä yhteisnäköalue.

Pelissä ohjataan joukkoja pääasiallisesti hiiren vasemmalla valiten ja oikealla antaen osoituspaikasta ja yksiköstä riippuvaisen vakiokäskyn. Esimerkiksi ritarin ollessa valittuna maata klikattaessa ritari komennetaan liikkumaan klikattuun kohtaan, mutta vihollisyksikköä klikatessa ritari lähtee hyökkäämään tähän. Voi myös valita monta yksikköä kerralla ja ohjeistaa niitä yhdellä napin painamisella.

Yksiköt alkavat taistelemaan keskenään niihin hyökättäessä, saadessaan hyökkäyskäskyn tai ollessaan joutilaina vihollisen tullessa näköpiiriin. Taistelussa yksiköt hyökkäävät mahdollisimman nopeasti ja usein kohteena olevan vihollisen kimppuun ja tämän kuollessa valitsevat heti uuden kohteen jos semmoinen on näkyvässä. Kaikilla yksiköillä on tietyn verran kestävyyttä, joka näytetään palkkina yksikön yläpuolella. Yksikkö on kuollut kun sen kesto tippuu nolnaan.

Peliä aloitettaessa valitaan pelin voittoehto ja onko peli rankattu vai ei. Rankatuissa peleissä pelaajat kirjautuvat järjestelmään ja pelin jälkeen pelitulos lähetetään julkista rankingia ylläpitävälle palvelimelle. Palvelin puolestaan tallentaa uuden tuloksen tietokantaan ja päivittää rankinglistan. Oletuksena pelin voittoehto on vihollisen kaikkien rakennuksien tuhoaminen, mutta voidaan esimerkiksi myös pelata niin, että vain sankari täytyy tappaa.

Pelissä on lisäksi karttaeditori, jolla voi tehdä omia karttoja peliin. Karttaeditorilla tehdään piirustusohjelmien tavoin pelialueen maasto, resurssit sekä merkataan alotuspaikat pelaajille käyttäen erilaisia editorissa olevia työkaluja. Eri maastotyytit vaikuttavat yksiköiden liikkumisnopeuteen ja rakennusten rakennusmahdollisuuksiin.

The Plan with a flaw

Kenen vastuulla on mikäkin osa-alue ja kuka tekee mitäkin?

Lähdekoodit buildataan aina merkittävien muutosten jälkeen ja tästä osa-alueesta huolehtii jokainen. Varsinkin, jos muutoksia ollaan päivittämässä pääkehityshaaraan. Viimeisimmän toimivan kokonaisuuden ylläpitämisestä Linux-puolella huolehtii Daniel ja Windows -puolella Markku. Tietovaraston ylläpitämisestä vastaa Niko.

Löydettyessä ohjelmatoiminnallisuus, joka ei täysin toimi halutulla tavalla, tehdään joko: ilmoitus virheraportointijärjestelmään tai, jos toiminnallisuutta ei ole vielä viety pääkehityshaaraan, niin korjaus pyritään tekemään ennen kuin se viedään pääkehityshaaraan.

Mitä projektiin kuuluvaa pidetään tallessa?

Projektin ylläpidon kannalta on oleellista, että kaikista vanhoista versioista pidetään tallessä lähdekoodin lisäksi julkaisuversioiden binäärit. Näiden lisäksi ohjelman dokumentaation kaikki mahdolliset uudistukset pystytään jäljittämään alkutekijöihin asti.

Milloin versionhallintaan viedään ja millä edellytyksillä?

Kaikki kehittävät tuotetta omassa kehityshaarassaan. Kehittäessä pyritään committaamaan riittävän usein, jotta mahdolliset vikatilanteet pystytään helpommin korjaamaan. Pääkehityshaaraan tuotos viedään vasta kuin ollaan varmistettu, että se ei aiheuta ongelmia tuotteessa.

Mitä versionhallintajärjestelmää käytetään ja mitä muita työkaluja ja kääntäjiä mahdollisesti projektissa tarvitaan?

Tuotekehityksessä käytetään uuden sukupolven versionhallintajärjestelmä Bazaaria. Tietovarastoa pidetään julkisella tietovarastopalvelulla nimeltä Launchpad, joka tarjoaa ilmaisen tietovarastopalvelimen. Launchpadin ominaisuuksiin kuuluu muunmuassa tietovarasto ja virheraportointijärjestelmä. Myös sähköpostilista on saatavilla.

Kokoamisessa käytetään Makefileja, jotka ovat generoitu joko automaattisilla Makefile työkaluilla tai tehty käsin. Pääasiallisena kääntäjänä käytetään GCC-4.4:sta. Windows-puolella kääntämiseen ja kokamiseen käytetään Visual Studiota.

Kuinka huolehditaan SCM-suunnitelman ylläpitämisestä?

SCM-suunnitelma viedään tietovarastoon, jotta sitä voidaan tarvittaessa päivittää.