

Muistipeli

Toteutusdokumentti

Jari Haapala

Sisällysluettelo

Harjoitustyön idean kehittäminen	3
Alkuvaiheen haasteet.....	3
Ulkoasu.....	3
Käsiteanalyysi	5
Muistipelin eteneminen	7
Pelin kehittäminen	7
Loppusanat	9

Harjoitustyö – muistipeli

Harjoitustyön idean kehittäminen

Yhtenä valmiina ehdotuksena harjoitustyöksi oli muistipeli. Päätin ottaa sen harjoitustyöni aiheeksi, koska olin aikaisemmin, jo vuonna 1999 saanut päätökseen kaupallisen liikenne-rompun nimeltä ”Lue Liikennettä Jesse”. Siinä yhtenä osa-alueena oli liikennemerkkiaiheinen muistipeli. Toteutuksen tein silloin nimenomaan multimediasovellusten tekemiseen tarkoitettulla Macromedia Directorilla. Sitten Adobe ostettua Macromedian, kokonaisuuteen kuuluneen Director-sovelluksen kehittäminen lopetettiin muutamassa vuodessa. Directorissa ohjelmointikielenä toimi Lingo, joka perustui pitkälti graafisten objektien toiminnan ohjaamiseen ja oli varsin kaukana nykyisestä olio-ohjelmoinnista. Sillä pystyi kuitenkin saavuttamaan toimivia ja sen ajan mukaan riittävän näyttäviä lopputuloksia. Liikenneprojektiin kului aikaa kokonaisuudessaan pari vuotta. Se minkä takia ylipäänsä lähdin sitä toteuttamaan on historiaa ja oma tarinansa, ei siitä sen enempää.

Joka tapauksessa, kun huomasin yhtenä ohjelmoinnin harjoitustyön vaihtoehtona muistipelin palautui mieleeni tämä reilu vuosikymmen aikaisemmin tekemäni projekti. Saisinko hyödynnettyä silloin iltaisin (ja öisin) tekemääni noin kahden vuoden työtä vielä uudelleen? Kävin läpi vanhat varmuuskopioni ja löysin viimein kuvat liikennemerkeistä, jotka olin itse piirtänyt sen aikaisen liikennemerkkilakanan mukaisesti. Tekemiäni vektorikuvia en ole vielä löytänyt, mutta onnistuin löytämään 8-bittiset pic (pct)-formaattissa olevat kuvat. Päätin käyttää niitä ja muokkasinkin ne tähän projektiin sopivaksi. Harjoitustyöhöni otin valikoidusti liikennemerkkejä kolmesta eri merkkiryhmästä, koska kuvat merkkiryhmien sisällä ovat keskenään symmetrisiä ja samankokoisia. Näin ne näyttävät muistipelin kokonaisuudessa asiallisilta.

Alkuvaiheen haasteet

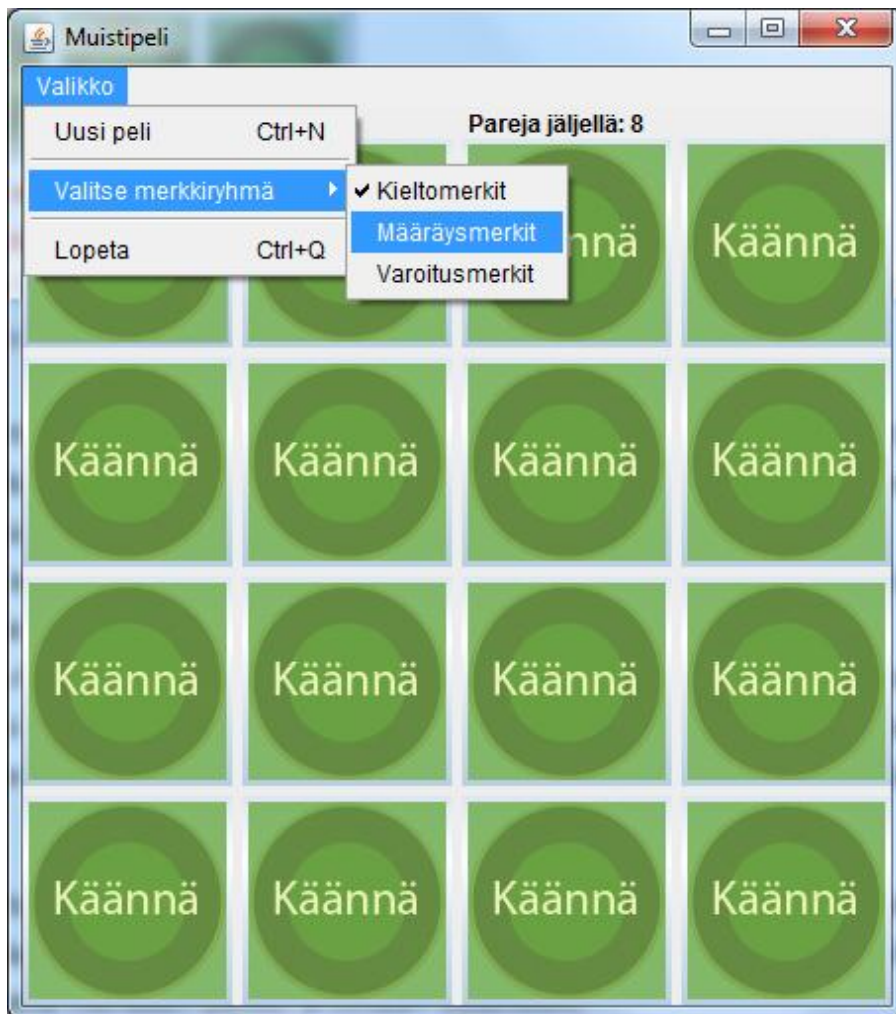
Harjoitustyön ensimmäisenä haasteena oli, kuinka saan ylipäänsä toteutettua graafisen muistipelin Javalla. Miten kuvat saa haettua ulkoisesta hakemistosta niin, että ne saadaan kahdennettua Java-kielillä peliin muistipelin säännöillä toimivaksi? Directorin aikaan se oli suhteellisen yksinkertaista ja random-komennon avulla sai korttien sekoittamisen suoritettua melkoisen vähällä vaivalla. Korttien kääntäminen graafisessa käyttöliittymässä oli sinänsä helppo toteuttaa, mutta miten sitten tapahtuu käännettyjen korttiparien vertailu? Käyttöliittymän tekemiseen NetBeansissa oli vaihtoehtoina joko hyödyntää sen omia sisäänrakennettuja graafisia elementtejä tai sitten tehdä käyttöliittymä täysin koodaamalla.

Ulkoasu

Aluksi kasasin käyttöliittymän ulkoasun lähinnä itselleni malliksi edellä mainituilla NetBeansin graafisilla elementeillä, koska minun on helpompi hahmottaa asioita ja kokonaisuuksia visuaalisesti. Niihin oli kuitenkin vaikea saada siistiä ja samalla oikein toimivaa koodia. Tämä tuntui oudolta ja osin turhauttavaltakin, miksi ihmeessä NetBeans edes tarjoaa siihen mahdollisuuden? Sinänsä tietysti hyvä ominaisuus, mutta vaatii vielä viljalti jatkokehittämistä. Yrityksen ja erehdyksen kautta päädyin loppujen lopuksi tekemään ulkoasunkin koodaamalla, koska muistipelin käsittelyn kannalta taulukkorakenne tuntui selkeältä ja siihen oli luontevaa liittää

GridLayout -asettelu. Näin pystyin tavallaan kuitenkin toimimaan ja tarkastelemaan asioita graafisesta näkökulmasta sekä suunnittelu- että toteutusvaiheessa.

Ulkoasultaan pelin graafinen käyttöliittymä on hyvin perinteinen ja yksinkertainen. Peli avautuu näytölle sen kokoisena kuin tarvitaan. Korttiruutujen ja samalla pelialueen koko määräytyy aluksi kuvien koon mukaan ja ruutujen välissä on viiden pikselin marginaali joka suuntaan. Ruudut asettuvat peli-ikkunaan tasaisesti 4x4 kortin muotoon ja niihin ilmestyy korttien keskenään samanlaiset selkäpuolet. Korttien yläpuolella on ns. normaali valikko, josta voi valita uuden pelin, lopettamisen tai vaihtaa merkkilajin kolmesta eri vaihtoehdosta (alavalikko). Valikkoja ja pelin etenemistä voi ohjata myös tietokoneen näppäimistöltä, mutta peli on kuitenkin parhaimmillaan silloin kun pelataan hiirellä klikkaamalla. Näppäimistöllä pelaaminen on luonnollisesti myös melko työlästä.



Muistipelin graafinen käyttöliittymä ja lopullinen ulkoasu

Käsiteanalyysi

(Tässä käyn samalla läpi koko käsiteanalyysiin liittyvän ajatteluprosessini, lopputulos viimeisenä)

Tarkoitus on ohjelmoida java-ohjelmointikielellä **muistipeli**. **Pelialue** on 4x4 **ruutua** (**kuvakortit** liikennemerkkeistä). Pelin alussa **korttiparit** arvotaan peliin sattumanvaraiseen järjestykseen. **Korttien selkäpuolet** ovat keskenään samanlaisia. **Pelaaja** klikkaa peräkkäin kahta korttia. Mikäli kortit ovat samat, ne jäävät näkyviin ja pelaajan tulee klikata kahta seuraavaa haluamaansa korttia. Jos kortit eivät ole samanlaisia, ne kääntyvät takaisin nurinpäin määrätyn ajan jälkeen (aloitetaan yhdellä sekunnilla, se riittää). Toisiaan vastaavien korttien löydyttyä **ilmoituskenttään** tulee löytyneiden **korttien määrä**, myös tehtyjen **arvausten määrä** kerrotaan.

Kun kaikki kortit ovat löytyneet, annetaan siitä **palaute** pelaajalle. Samalla ilmoitetaan kaikkien käytettyjen **siirtojen määrä**.

Pelaajalla on mahdollisuus aloittaa **uusi peli** sekä **keskeneneräisen pelin** aikana että **onnistuneen ratkaisun** jälkeen. Samaten hän voi lopettaa pelin milloin haluaa.

1-vaihe: Pelin kuvauksesta muodostettu ensimmäinen määrittely
(peliin vaikuttavat substantiivit valittu):

Ehdokas

Analyysi

muistipeli	Peli, jossa etsitään korttipareja.
pelialue	Alue, mihin kortit tullaan sijoittamaan.
ruutu	Kortin sijoituspaikka pelialueella, alussa 4x4 ruutua, voi laajentaa (optio).
kuvakortit	Tiettyyn hakemistoon sijoitetut liikennemerkkikuvat, joita pelissä käytetään.
korttiparit	Kortit kahdennettuna, pelissä etsitään keskenään samanlaisia kortteja.
korttien selkäpuolet	Kortin kääntöpuoli, kaikilla korteilla sama taustakuva.
pelaaja	Pelin pelaaja, käyttää peliin luotuja ominaisuuksia.
ilmoituskenttä	Kenttä jossa kerrotaan pelaajalle onnistumisista/pelaajan tekemisiä.
korttien määrä	Kenttä joka kertoo kuinka monta korttia on löytynyt.
arvausten määrä	Kenttä joka kertoo kuinka monta arvausta pelaaja on tehnyt.
palaute	Palaute pelaajalle ratkaistun pelin jälkeen (kaikki kortit ovat löytyneet).
siirtojen määrä	Palaute pelaajalle kuinka monta siirtoa ratkaisu vaati.

Muuta en näe tarpeelliseksi edes ottaa mukaan, koska ne eivät vaikuta varsinaisesti pelin toimintaan.

2-vaihe: Pelin kuvauksesta muodostettu toinen määrittely (karsittu turhaa ja tehty yhdistelyjä):

Ehdokas

Analyysi

muistipeli	Pelin kokonaismäärittely, tästä tulee pelin graafinen käyttöliittymä
pelialue	Alue, mihin kortit tullaan sijoittamaan (kaikki itse pelialueeseen liittyvä)
kortit	Kaikki kortteihin liittyvät toiminnot

kuvakortit	- sisältyy kortteihin
korttiparit	- sisältyy kortteihin
korttien selkäpuolet	- sisältyy kortteihin
pelaaja	turha (koska tässä pelissä ei kerätä pelaajaan liittyvää tietoa)
ilmoituskenttä	-sisältyy pelialueeseen
korttien määrä	-sisältyy pelialueeseen
arvausten määrä	-sisältyy pelialueeseen
palaute	-sisältyy pelialueeseen
siirtojen määrä	-sisältyy pelialueeseen

3-vaihe: Tarkennusten ja yhdistelyjen jälkeen jää jäljelle pelkästään kolme luokkaa:

Luokat on jaettu kahteen pääryhmään; käyttöliittymä- ja sovellusluokat. Tässä ohjelmassa ei määrittelyni perusteella tarvitse luoda lisää luokkia paitsi että päämetodi tehdään omaksi luokakseen. Tarkoitus on, että Main-luokka pelkästään aloittaa uuden pelin. Mikäli haluttaisiin tehdä vaikkapa pelaajalista tai kerätä tietoa parhaista pelaajista, silloin niille kullekin olisi mielestäni järkevää luoda oma luokkansa. Se selkeyttäisi silloin ohjelman rakennetta.

- Muistipeli

- graafinen käyttöliittymä (pääohjelma olisi mahdollista liittää tähän)
- tässä määritellään myös valikkoihin liittyvät asetukset

- Pelialue

- pelipohja, johon varsinaiseen peliin liittyvät elementit luodaan

- Kortit

- määritellään kortteihin liittyvät ominaisuudet (hakemistot, korttilajit, kortit)

muistipeli	Pelin kokonaismäärittäminen, tästä tulee pelin graafinen käyttöliittymä ja tähän luokkaan kirjoitetaan myös pelin käynnistävä pääohjelma (pelkästään käynnistys)
pelialue	Alue, mihin kortit tullaan sijoittamaan (kaikki itse pelialueeseen liittyvä)
kortit	Kortteihin liittyvät toiminnot

Muistipelin eteneminen

Pelin tarkoituksena on löytää 4 x 4 ruudun kokoiselta pelialueelta toisiaan vastaavat korttiparit perinteisen lautamuistipelin tapaan, ideaalitapauksessa tietenkin mahdollisimman vähillä arvauksilla. Korteissa on liikennemerkkiaiheisia kuvia merkkiryhmittäin jaoteltuna. Merkkiryhmiksi valitsin kielto-, määräys- ja varoitusmerkit, koska merkkiryhmät ovat keskenään erinäköisiä ja ryhmän sisällä olevat merkit ovat puolestaan samanmuotoisia. Näin ollen pelin ulkoasu on koko ajan samantyyppinen eikä pelialueen koko muutu merkkilajia vaihdettaessa. Peli on valmis, kun pelaaja on löytänyt kaikki parit. Pelaaja voi halutessaan kesken pelin vaihtaa merkkilajia tai aloittaa uuden pelin. Kummassakin tapauksessa peli alkaa alusta ja merkkien paikat arvotaan uudelleen.

Pelin alkaessa pelialueella näkyy 16 samanlaista korttia (korttien selkäpuolet ovat näkyvissä). Kun pelaaja klikkaa hiiren avulla jotain korttia, sen kohdalla oleva kuva tulee näkyviin. Tämän jälkeen pelaaja klikkaa hiirellä jonkin toisen kortin näkyviin. Jos kortit ovat samanlaiset, ne jäävät näkyviin. Mikäli kortit ovat erilaiset (eli eivät ole pareja keskenään), ne ovat toisen kortin klikkauksen jälkeen sekunnin ajan näkyvillä ja kääntyvät piiloon (korttien selkäpuolet tulevat näkyviin). Väärin valitut kortit kääntyvät piiloon välittömästi, jos pelaaja klikkaa jotain kolmatta korttia ennen ohjelmaan määritettyä yhden sekunnin aikarajaa. Tällöin kolmas klikkaus kääntää seuraavan vaiheen ensimmäisen kortin näkyviin ja niin edelleen kunnes peli on saatu päätökseen.

Pelin päätyttyä pelaajalle annetaan palautteena tieto kuinka monta siirtoa tarvittiin sekä totta kai tsemppausviesti. Tämän jälkeen pelaaja voi lopettaa pelin tai jatkaa sitä joko samalla tai jollain toisella liikennemerkkiryhmillä.

Pelin kehittäminen

Tässä joitakin ajatuksia/pohdintoja mitä tuli mieleeni muistipeliä tehdessäni.

- liikennemerkkien nimet
 - peliin voisi liittää kuvan lisäksi myös merkin nimen ja tarkoituksen
 - sotkee kyllä pelikentän ja pelaamisen idean jos tulee jokaisen kuvan kohdalla
 - oikein valitun merkkiparin jälkeen?
 - pitäisi laittaa alertina -> sotkee taas pelin?
 - voisi kehittää uuden pelin missä valitaan joko näkyvän liikennemerkkin nimi tai tarkoitus
 - tämä toimii kyllä
- top10-lista
 - ei ehkä muistipelissä tarvita, ehkä jos pelaaja haluaa tietää monellako siirrolla peli on läpäisty
 - täytyy miettiä vielä onko järkevä, koska **aina** on olemassa minimimäärä siirtoja
 - voisi toimia toisaalta suurella vaikeusasteella vaikka moninpelinä
- vaikeusasteen muuttaminen
 - lisää ruutuja, enemmän liikennemerkkejä
- kaikki liikennemerkkiryhvät mukaan
 - myös kaikki mahdolliset perusliikennemerkkit
- merkkivariaatiot (lisäkilvet, ulkomaiden merkit...)
 - turha
- kaksinpeli
 - pelaajien nimet kirjataan jo pelin alussa
 - näytetään pelaajille kummalla on pelivuoro

- pelitulos (kumpi voitti)
- moninpeli
 - verkossa toteutettava monen pelaajan peli, silloin kenttä riittävän suuri
 - toisaalta tästä voisi kehittää edelleen...
 - silloinhan tästä saisi vaikka joukkueiden välisen turnauksen (käytännössä turha, koska ketä kiinnostaa...paitsi esimerkiksi myöhemmin kerrotussa liikennetilanneosiossa, samaan aikaan ajokorttia suorittavien kesken?)
- tarjonta liikenneopetukseen (interaktiivinen liikennesovellus)
 - vastaavaa on jo käytössä, joten turha ellei tähän keksi jotain todella uutta lähestymistapaa (Omassa vuonna 1999 tekemässäni liikennerompun projektisuunnitelmassa esittelin mm. tilannekuvien hyödyntämisen tietokoneavusteisessa liikenneopetuksessa, mutta siltä osin suunnitelma hylättiin mahdottomana ja turhana. Ajokorttikokeissakin käytetään nykyisin tuota silloin esittämääni ideaa.)
 - toisaalta tästä voisi kehitellä vaikkapa netissä toimivan maksuttoman harjoitteluohjelman

Loppusanat

Harjoitustyön tekeminen Javalla alkusuunnittelusta sovellukseksi oli erittäin haastava ja mielenkiintoinen projekti. Tämä työhän oli tavallaan ohjelmoinnin perus- ja jatkokurssin jälkeinen kolmas osa. Harjoitustyön toteuttaminen omatoimisesti vaatii huomattavan paljon, uskallan sanoa jopa liikaakin aikaa. Aikaisemmilla ohjelmointikursseilla saadut esimerkit (ja kurssien sivuiltakin löytyvät) ovat olleet enimmäkseen yksittäisiä toisiaan seuraamattomia tapauksia, joita ei valitettavasti pystynyt kokonaisuutena ainakaan omassa projektissani hyödyntämään monessakaan kohdassa. Olisin toki voinut valita vaikka omaksi harjoitustyökseni aiheen mikä ei olisi ollut itseäni kiinnostava, mutta vähemmän haasteita antava.

Jatkoa ajatellen olisi ehkä hyvä saada ohjelmointikursseille (sekä perus- että jatkokurseille) jokin yhteinen kokonaisuusesimerkki, mitä lähdetään rakentamaan luennoilla (ja luentomateriaaleilla) eteenpäin. Silloin se varmasti auttaa opiskelijaa ymmärtämään paremmin sekä prosessin etenemistä että ohjelmoinnissa tarvittavia syy-seuraussuhteita. Se, että suoritetaan yliopistotason opintoja ei saa mielestäni kuitenkaan tarkoittaa sitä, että istutaan illat tietokoneen ääressä etsimässä käytännön ratkaisuja abstrakteina annettuihin ongelmiin. Ellei sitten ole annettu myös jokin selkeä esimerkki siitä mihin suuntaan ja millä tavoin loppuratkaisuun tulisi päätyä. Tärkeää on myös se miten kykenee analysoimaan omaa tekemistään ja kehittämään sitä kautta osaamistaan ja tarvittavaa lopputuotosta.

Hyvät toisistaan irralliset esimerkit ovat ilman muuta tarpeellisia, että saadaan mahdollisimman laaja yleiskuva ohjelmoinnin perusteista ja mahdollisuuksista. Voisiko kuitenkin olla hyvä ajatus miettiä myös kokonaisuutta jatkumona? Saataisiinko sillä aikaan parempia tuloksia? Vastausta minulla ei nähin kysymyksiin ole, mutta jätänpä kuitenkin nämä kysymykset ilmaan...

Palataanpa takaisin asiaan. Ohjelmoinnin harjoitustyö-projekti oli (kuten alussa mainittu) haastava ja teetti vähintään riittävästi siihen liittyvää työtä. Mielenkiinto ja motivaatio harjoitustyön loppuunsaattamiseen säilyi koko ajan hyvänä. Kurssin aikana omasta työstä saatu palaute oli oikein kohdennettu ja se edesauttoi ongelmanratkaisua.