

Viktori Patalainen

# Tietovisapeli yrityksen markkinointiin

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Mediatekniikan koulutusohjelma

Insinöörityö

21.3.2013

Tekijä Otsikko	Viktori Patalainen Tietovisapeli yrityksen markkinointiin
Sivumäärä Aika	36 sivua + 1 liitettä 21.3.2013
Tutkinto	insinööri (AMK)
Koulutusohjelma	mediatekniikka
Suuntautumisvaihtoehto	digitaalinen media
Ohjaajat	teknologia- ja IT-johtaja Mikko Wirgentius yliopettaja Erkki Rämö
<p>Insinööriyön tavoitteena oli luoda tietovisapeli Flash-tekniikalla markkinointitarkoituksiin. Insinööriyön tilaajana oli IT-alan yritys. Peli suunniteltiin alusta alkaen erittäin monipuoliseksi muokattavaksi ilman, että muokkaajan tarvitsee tuntea Flash-ohjelmointia. Tarkoituksena oli luoda peli, jota asiakas voi helposti muokata vaihtamalla muutamia tiedostoja palvelimella.</p> <p>Erilaiset muokkausoptiot laitettiin helposti luettavaan muotoon käyttäen XML-tiedostoja. Muokattavat tiedostot säilytetään palvelimella, josta ne ladataan pelin alussa. Asiakas voi luoda eri versioita pelistä vaihtamalla pelin tunnuksen ja siihen liittyvät tiedostot. Asiakas voi muun muassa muokata melkein kaikkia pelin sisällä olevia tekstejä, erilaisia pelissä näkyviä optioita ja ulkoasua.</p> <p>Insinööriyön tuloksena valmistui tietovisapeli esimerkkisisällöllä, jota asiakas voi helposti muokata mieleisekseen. Asiakas voi tehdä pelistä useita versioita helposti ja käyttää sitä usean eri asian markkinointiin.</p>	
Avainsanat	multimedia, Flash, markkinointi, tietovisa, Actionscript, olio-ohjelmointi, peli

Author Title	Viktori Patalainen Quiz game for marketing a company
Number of Pages Date	36 pages + 1 appendix 21 March 2013
Degree	Bachelor of Engineering
Degree Programme	Media Technology
Specialisation option	Digital Media
Instructors	Mikko Wirgentius, Chief Technology Officer Erkki Rämö, Principal Lecturer
<p>The goal of the project was to create a quiz game with flash technology. The game can be used for marketing purposes. The project was commissioned by an IT-company. The game was designed to be highly customizable without the person needing to know flash. The goal was to create a game that the client could modify by changing external files on a server.</p> <p>The client is able to customize the settings of the game by modifying an xml file. All external files are located on a server and they are loaded into the flash game at the start of the game. The client can easily create different versions of the quiz game by simply modifying the external files. They can for example change messages, graphic objects and settings inside the game.</p> <p>The finished product was a flash quiz game with temporary content which the client could easily replace with their own content. The client can use the game to easily create different versions of the game and use them to market different things.</p>	
Keywords	multimedia, flash, game, marketing, quiz, actionscript, object oriented programming

## Sisällys

1	Johdanto	1
2	Flash-ohjelma ja Actionscript-kieli	2
2.1	Flash-ohjelma	2
2.2	Actionscript-kieli	4
2.3	Olio-ohjelmointi	7
2.4	Flashdevelop-ohjelmisto	8
2.5	XML-merkintäkieli	9
3	Flash-pelit	10
3.1	Flash-pelien historia	10
3.2	Pelin kehityksen vaiheet	12
3.3	Flash-peliportaalit	17
4	Tietovisapelin suunnittelu ja toteutus	20
4.1	Tietovisapelin suunnittelu	20
4.2	Tietovisapelin toteutus	24
5	Yhteenveto	33
	Lähteet	35
	Liitteet	
	Liite 1. Tietovisapelin valmis asiakkaan muokkaama versio	

## 1 Johdanto

Verkkomainonta on ollut kovassa nousussa internetin yleistyessä. Suosituimpia teknologioita verkkomainonnassa on Flash-tekniikka, jolla voidaan saada aikaan animaatioita ja interaktiivisia sovelluksia, jotka voidaan upottaa internetsivulle. Yleisimmät sovellukset ovat yksinkertaisia bannereita, jotka välittävät halutun viestin käyttäen animaatioita apunaan.

Flash-tekniikka mahdollistaa myös monimutkaisten sovellusten tekemisen. Mainonnan jälkeen Flash-tekniikan suosituin käyttötarkoitus on pelit. Viime aikoina mainonta ja pelit ovat yhdistyneet ja luoneet uudenlaisia tapoja markkinoida tuotteita tai yritystä. Pelit ovat hauska tapa välittää informaatiota, ja useimmat käyttäjät eivät koe pelin muodossa olevaa mainontaa häiritsevänä, koska käyttäjän omat valinnat ohjaavat kokemusta eteenpäin.

Insinööriyön tavoitteena on luoda Flash-tekniikalla tietovisapeli markkinointitarkoituksiin. Pelin tulee olla monipuolisesti ja helposti muokattava, jotta sitä voidaan käyttää useamman kuin yhden tuotteen markkinointiin. Pelistä tulee voida tehdä helposti useita eri versioita, joista jokainen keskittyy yhden tuotteen tai yrityksen markkinointiin. Oletuksena on, että sisällön muokkaajalla ei ole Flash-osaamista, joten pelin muokkauksmahdollisuudet on suunniteltava tämä mielessä pitäen.

Tietovisapelin suunnittelussa on inspiraationa Älypää-tietovisapeli, joka on erittäin suosittu internetissä. Älypää tarjoaa asiakkailleen mahdollisuutta räätälöidä tietovisapelinä asiakkaan toiveitten mukaan. Insinööriyön tavoitteena on rakentaa samankaltainen tietovisapelialusta käyttäen Flash-tekniikkaa. Suoraa kopiota ei kuitenkaan tehdä, vaan peliä tehtäessä se muodostuu omanlaisekseen tietovisaksi, joka on tilaajan omiin tarkoituksiin sopiva.

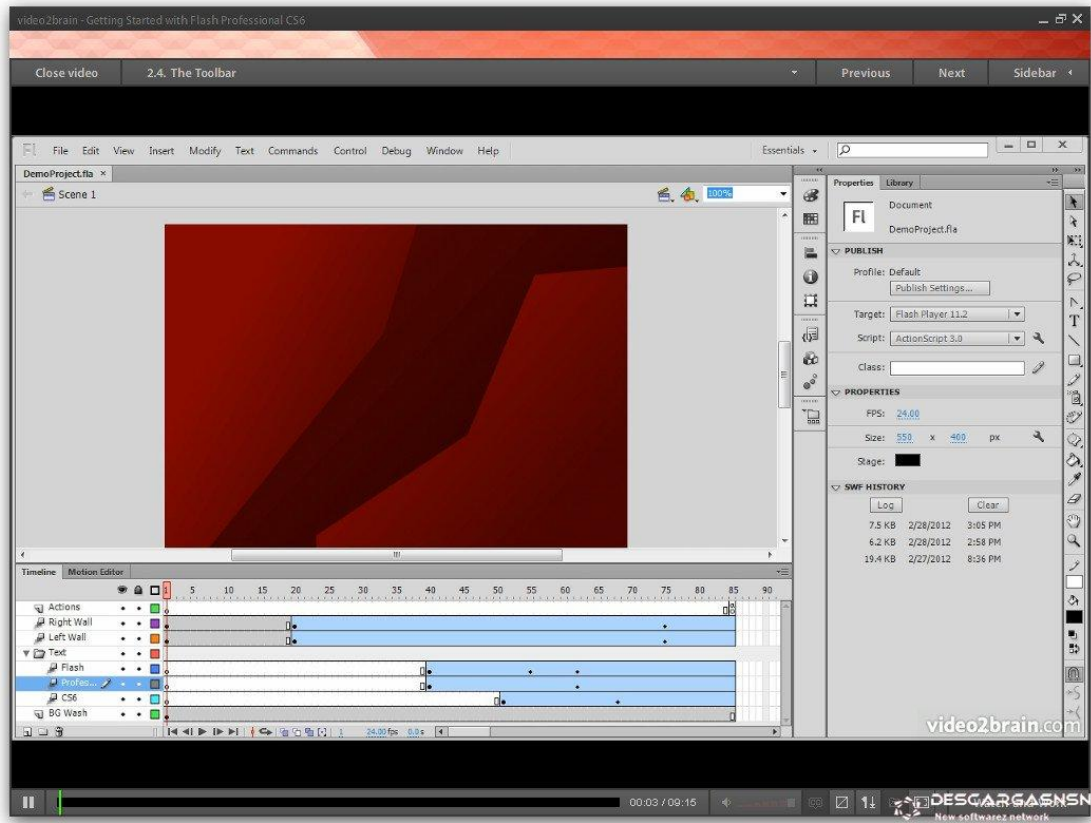
Tietovisa-konsepti on melko yksinkertainen, joten erilaistumista haetaan erilaisilla yksityiskohdilla, ja laajat muokausvaihtoehdot myös varmistavat, että eri sisällön avulla mikään tietovisapelin versio ei ole samanlainen. Pelistä pitää pystyä vaihtamaan muun muassa graafiset elementit, tietovisakysymykset ja vastausvaihtoehdot ja pelin sisältämät kirjoitetut viestit, kuten esimerkiksi tervetulo viesti.

## 2 Flash-ohjelma ja Actionscript-kieli

### 2.1 Flash-ohjelma

Flash on Adoben kehittämä ohjelmistoympäristö, jolla voidaan luoda animaatioita ja interaktiivisia sovelluksia. Flashin kehitys alkoi vuonna 1993, jolloin perustettiin FutureWawe Software -yritys. Ensimmäinen Flash-ohjelma julkaistiin vuonna 1994, jolloin se sai nimen SmartSketch. Ohjelma oli tarkoitettu graafikoille, jotka halusivat piirtää käyttäen elektronista piirtokynää. Ohjelma ei ollut menestys, joten se julkaistiin uudelleen vuonna 1996 nimellä FutureSplash Animator, ja siinä oli nyt mukana mahdollisuus luoda animaatioita. Macromedia osti FutureWawe Softwaren vuonna 1996 ja muutti ohjelman nimen Macromedia Flashiksi. Tästä alkoi nykyisin tunnetun Flashin kehitys. Vuonna 2005 Macromedian osti Adobe, ja nykyisin ohjelmisto tunnetaan nimellä Adobe Flash. (1; 2.)

Animaatioiden teko Flashissa on helppoa. Keskeisenä osana on aikajana, joka koostuu useista ruuduista (frame). Ruudut on asetettu horisontaalisesti peräkkäin, ja aikajana kulkee vasemmalta oikealle. Valmiissa animaatioissa jokainen ruutu näytetään peräkkäin, ja näin saadaan aikaan liikettä. Kuvassa 1 aikajana ja ruudut ovat alhaalla keskellä ja keskellä nähdään valitun ruudun sisältämä näkymä.



Kuvio 1. Adobe Flash CS6 -ohjelman käyttöliittymä (3).

Flash-animaatiot ja sovellukset julkaistaan swf-tiedostomuodossa. Swf-tiedostojen suorittamiseen tarvitaan erillinen ohjelma, Flash Player. Yksi yleisemmistä arvostelun aiheista on nimenomaan riippuvuus erillisestä ohjelmasta. Flash Playerin levinneisyys on yli 98 % tietokoneista, joten suurin osa tietokoneista pystyy toistamaan swf-tiedostoja (4).

Flashissa on myös oma ohjelmointikieli nimeltä Actionscript. Sitä ei tarvitse käyttää animaatioita tehdessä, mutta jos halutaan interaktiivisuutta ja monimutkaista toiminnallisuutta, silloin pitää käyttää joko Actionscript 2- tai Actionscript 3 -ohjelmointikieltä. Actionscriptillä on erittäin helppo lisätä yksinkertaista toiminnallisuutta, kuten esimerkiksi napin painaminen vaihtamaan näkymää, mutta monimutkaisten toiminnallisuuksien tekeminen vaatii paneutumista Actionscriptiin aivan kuten mihin tahansa muuhunkin ohjelmointikieleen.

## 2.2 Actionscript-kieli

Actionscript on ohjelmointikieli, jota käytetään Flash-ohjelmoinnissa. Actionscript perustuu ECMA-ohjelmointikielistandardiin, mikä tarkoittaa, että sen syntaksi noudattaa tiettyjä sääntöjä (5). Flashissa voidaan valita, käytetäänkö vanhempaa Actionscript 2 -kieltä tai uudempaa Actionscript 3 -kieltä. Molempia kieliä ei voi käyttää samassa Flash-projektissa sekaisin.

### Actionscript 2

Actionscript 2 -kieli julkaistiin Flash MX 2004 -ohjelman mukana vuonna 2003. Actionscript 2 on yhä laajimmin käytetty kieli Flash-ohjelmoinnissa, vaikka se onkin jo melkein 10 vuotta vanha. Useimmat käyttäjät, joilla ei ole ohjelmointitaustaa, kokevat Actionscript 2 -kielen helpommaksi käyttää, koska se ei ole niin ankara sääntöjen suhteen kuin Actionscript 3. Actionscript 2 mahdollistaa niin sanotun löyhän kirjoittamisen ohjelmoinnissa, mikä tarkoittaa, että muuttujat voivat sisältää mitä tahansa dataa (5). Tämä mahdollistaa nopean ohjelmistokehittämisen varsinkin yksinkertaisissa sovelluksissa. Huonona puolena siinä on, että mahdollisten ongelmatapausten sattuessa koodin virheiden löytäminen hankaloituu. Tämän vuoksi on suositeltavaa tyypittää muuttujat, jotta virheiden paikannus ja korjaus käy helpommin.

Esimerkkikoodissa 1 nähdään tyypitetyn ja tyypittömän muuttujan ero. Tyypitetty muuttuja voi sisältää vain dataa, joka on tyyppiä String, kun taas tyypitön muuttuja voi sisältää mitä tahansa dataa. Kun koodi suoritetaan, Flash tunnistaa, jos tyypitettyssä muuttujassa on väärää dataa, ja ilmoittaa virheestä. Tyypittämätön muuttuja selviää aina virheiden tarkistuksesta, joten jää ohjelmoijan vastuulle pitää huoli, että muuttujassa on oikeaa dataa ja sitä käytetään oikeassa paikassa. Suurissa sovelluksissa, joissa käytetään useita tyypittömiä muuttujia, on vaarana, että yritetään syöttää vääränlaista dataa paikkaan, johon se ei kuulu. Kun muuttuja on tyypitön, Flash ei osaa ilmoittaa virhetilanteesta, vaan ohjelmoijan on itse etsittävä virhekohta.

```
var myObject = "kissa"; //tyypitön muuttuja
var myObject:String = "kissa"; //tyyppiä String oleva muuttuja
```

Esimerkkikoodi 1. Tyypitön ja tyypillinen muuttuja.



Actionscript 2 -projekteissa koodia voi kirjoittaa melkein minne tahansa, kuten muun muassa objektien sisälle, aikajanalle tai erillisiin Actionscript-tiedostoihin. Koodi voi olla siis erittäin pirstaloitunut eri puolille projektia, mikä hankaloittaa entisestään ongelmien etsintää tai ylipäätään projektin kokonaisuuden ymmärtämistä. Ulkopuolinen henkilö, joka yrittää ymmärtää projektin toiminnallisuutta tai jatkokehittää projektia, on vaikeuksissa, jos projektin koodi on pirstaloitunut useaan eri paikkaan. On suositeltavaa pitää kaikki koodi yhdessä paikassa, kuten aikajanalla, luettavuuden ja sovelluksen kehittämisen parantamiseksi.

Actionscript 2 toi Flashiin mahdollisuuden kirjoittaa luokkia lisäämällä kieleen tuen class- ja extend-sanoille. Luokkatuki mahdollistaa olio-ohjelmointipohjaisten sovellusten kehittämisen. Varsinkin isoissa projekteissa luokkalähtöinen sovelluksen kehittäminen helpottaa kehittämistä ja ongelmien paikantamista. Actionscript 2:n tuki luokille on kuitenkin puutteellinen, ja useat kehittäjät eivät käytä luokkia tämän takia Actionscript 2 -projekteissaan.

### Actionscript 3

Actionscript 3 julkaistiin vuonna 2006, ja se oli huomattava parannus aiempaan, Actionscript 2 -kieleen. Actionscript 3 -kielen rakentaminen aloitettiin tyhjästä, ja tästä syystä ero Actionscript 2 -kieleen on suuri. Huomattavia parannuksia olivat muun muassa uusi Event-järjestelmä, tiukempi koodin tarkistus kirjoitus- ja ajovaiheessa, huomattavasti parannettu suorituskyky luokille ja parannettu tuki XML-tiedostojen käsittelyyn. (5.)

Yksi iso ongelmien aiheuttaja Actionscript 2 -kielessä oli tyyppittömät muuttujat; Actionscript 3 -kielessä suositellaan tyyppittämään kaikki muuttujat. On mahdollista tehdä myös tyyppitön muuttuja, mutta sitä pidetään huonona tapana. Harvoissa tapauksissa voi olla tarpeellista käyttää tyyppitöntä muuttujaa, ja silloinkin kannattaa tyyppittää muuttuja niin sanotuksi villiksi kortiksi, kuten esimerkkikoodissa 2. Villikortti-tyyppiä oleva muuttuja voi sisältää minkä tahansa tyyppistä dataa, aivan kuten tyyppitön muuttuja. Se on tyyppitön, kun se luodaan, ja se muuttuu sille syötetyn datan tyyppiksi.

```
//muuttuja on tyyppiä "villikortti", jolle annetaan MovieClip objekti mikä tekee muuttujasta
tyyppiä MovieClip
var myObject:* = new MovieClip();
```

Esimerkkikoodi 2. Villikortti-muuttujan luominen.

Yksi Actionscript 3 -kielen suurimmista muutoksista, joka vaikuttaa suoraan siihen, miten kehittäjät kirjoittavat koodia, oli uusi Event-järjestelmä. Kun halutaan reagoida johonkin tapahtumaan, kuten esimerkiksi hiiren vasemman napin klikkaus, tapahtumalle pitää rekisteröidä EventListener-objekti.

Esimerkkikoodissa 3 nähdään, miten hiiren klikkaukselle voidaan määritellä EventListener-objekti, ja kun käyttäjä klikkaa hiirellä, siihen voidaan reagoida funktion myHandler sisällä.

```
//rekisteröidään EventListener objekti joka on tässä tapauksessa hiiren klikkaus
addEventListener(MouseEvent.CLICK, myHandler);

//jokaiselle tapahtumalle pitää määritellä funktio joka ajetaan tapahtuman sattuessa
function myHandler(eventObject:MouseEvent):void
{
    //tässä koodia mitä tapahtuu kun klikkaus on tapahtunut
}
```

Esimerkkikoodi 3. Tapahtuman rekisteröinti ja reagointi.

Uusi Event-järjestelmä antaa kehittäjälle paljon valtaa päättää, miten hän haluaa sovelluksen toimivan. Melkein mihin tahansa tapahtumaan sovelluksessa voidaan reagoida, mikä mahdollistaa monimutkaisten interaktiivisten sovellusten kehittämisen. Actionscript 3 on iso harppaus eteenpäin, ja se vaatiikin käyttäjältä vähintään Flash Player 9 -version toimiakseen.

### 2.3 Olio-ohjelmointi

Olio-ohjelmoinnilla tarkoitetaan ohjelmointitapaa, jossa käytettävä data ja siihen liittyvät toiminnot ovat objektien (suomeksi olioiden) sisällä. Objektit voivat periä toistensa ominaisuuksia ja luoda linkkejä toistensa välille. Esimerkkikoodissa 4 myObject on luokasta MovieClip luodun objektin instanssi.

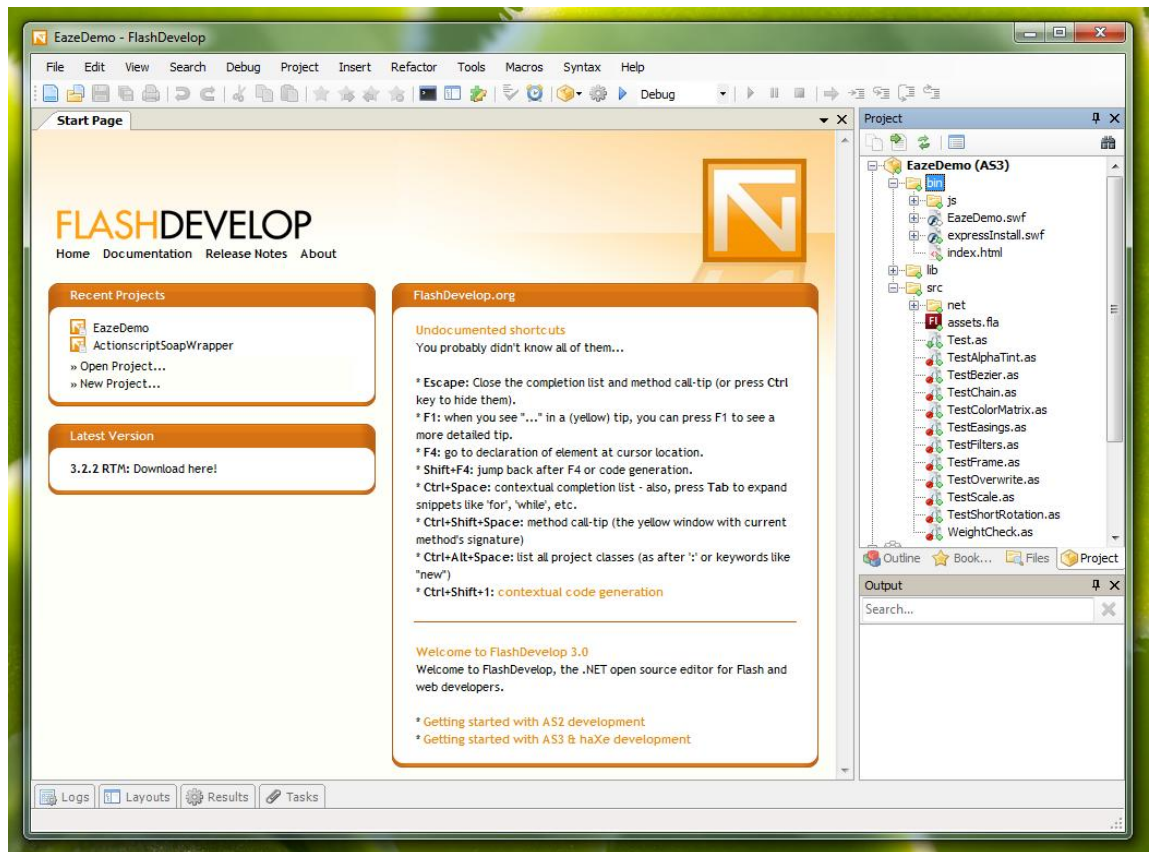
```
var myObject:MovieClip = new MovieClip();
```

Esimerkkikoodi 4. MovieClip-objektin luominen.

Flash-pelin tekemisen aloitan yleensä luomalla luokat Game ja Menu. Menu-luokka sisältää pelin alkuvalikkoon ja sen alavalikoihin liittyvän toiminnallisuuden. Game-luokka sisältää pelimaailman ja siihen liittyvät toiminnallisuudet, kuten esimerkiksi objektien törmäysten hallinta. Molemmista objekteista on vain yksi instanssi kerrallaan luotuna. Näitä objekteja kutsutaan yleisesti "container" -objekteiksi, koska ne sisältävät paljon muiden objektien instansseja ja hallinnoivat niitä.

## 2.4 Flashdevelop-ohjelmisto

Actionscript 3 -kieli soveltuu olio-ohjelmointiin paremmin kuin Actionscript 2, koska se on paljon kehittyneempi kieli. Adobe'n Flash-ohjelmisto ei ole kovin käyttäjäystävällinen mitä tulee olio-ohjelmointiin, joten ulkopuolisen ohjelman käyttö on suositeltavaa. Tämän insinööriyön tekemiseen käytettiin Flashdevelop-ohjelmistoa (kuva 2), joka on ilmainen vapaan lähdekoodin ohjelmisto (6).



Kuvio 2. Flashdevelop-ohjelman yleisnäkymä (6).

Flashdevelopin rakensi vuonna 2005 ryhmä intohimoisia Flash-kehittäjiä. Flashdevelop on vapaan lähdekoodin projekti, joten vuosien mittaan eri kehittäjiä on ollut monia ja Flashdevelop on muovautunut yhdeksi parhaista ohjelmistoista olio-ohjelmointiin Flashissa. Flashdevelop tarjoaa muun muassa koodin täydennyksen, ja se on nopea ja kevyt. (6.)

## 2.5 XML-merkintäkieli

XML on lyhenne sanoista Extensible Markup Language. XML-kieltä käytetään yleisesti tiedon varastointiin järjestelmälliseen muotoon. Ihmisen ja tietokoneen on helppo lukea XML-muotoista tietoa. XML auttaa jäsentämään suuriakin tietomassoja, ja se suunniteltiin muun muassa tiedon välitykseen internetissä. XML-kieli muistuttaa HTML-kieltä, mutta sitä ei ole tarkoitettu sivunkuvauskieleksi. XML-kieli suunniteltiin välittämään ja tallentamaan tietoa päämääränä kuvata, mitä data on, kun taas HTML suunniteltiin näyttämään, miltä tieto näyttää. (16.)

Ensimmäisen rivin XML-tiedostossa pitäisi aina olla XML-määrittely, kuten esimerkikoodissa 5. Sen jälkeen voidaan kirjoittaa, mitä tietoa halutaan XML-tiedoston sisältävän. Tiedon pitää noudattaa XML-standardeja siinä, miten se on jäsennelty. Tieto kirjoitetaan niin sanottujen ”tagien” sisään, kuten esimerkikoodissa 5, jossa on käytetty viittä eri tagia, root, data, person, name ja age.

```
<?xml version="1.0" encoding="UTF-8" ?>
<root>
    <data>
        <person>
            <name> Jaska </name>
            <age> 18 </age>
        </person>
    </data>
</root>
```

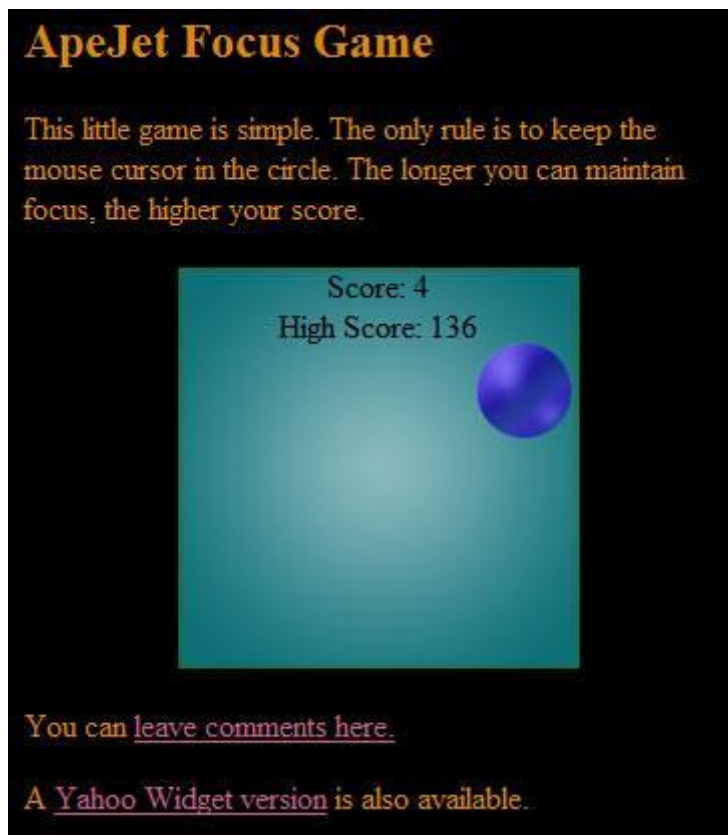
Esimerkkikoodi 5. XML-tiedoston perusrakenne, jossa esimerkisisältö.

Jokainen tagi pitää sulkea käyttämällä vinoviiva-merkkiä, ja sen jälkeen merkitään tagin nimi, joka suljetaan. Tiedon pitää aina olla jonkin tagin sisällä tai muuten XML-tiedosto on virheellisesti koottu ja sitä koneellisesti luettaessa voi tulla ongelmia. (16.)

### 3 Flash-pelit

#### 3.1 Flash-pelien historia

Flashin tulo mullisti tavan käyttää internetiä. Aluksi Flashia käytettiin tuomaan yksinkertaisia animaatioita ja interaktiivisuutta internetsivuille, mutta kehittäjät huomasivat nopeasti Flashin mahdollisuudet. Ensimmäisiä Flash-pelejä, jotka saavuttivat suuren suosion, olivat muun muassa avaruusräiskintäpeli Asteroidsista tehdyt lukuisat kopiot. Pelit olivat yksinkertaisia (kuva 3), mutta äärimmäisen koukuttavia. Flash-pelien suosion kasvaessa niiden kehittäjien määrä kasvoi nopeasti. (8.)



Kuvio 3. Yksi varhaisia Flash-pelejä (8).

Ensimmäiset Flash-pelit tehtiin käyttäen Actionscript 1 -kieltä tai sitä edeltävää Flashin omaa yksinkertaista ohjelmointikieltä. Tämä rajoitti suuresti sitä, mitä Flashilla voitiin tehdä peleissä. Oli suuri vaatimus saada lisää toiminnallisuutta Flashiin, ja Actionscript 2 oli vastaus toiveeseen, kun se julkaistiin vuonna 2003. Flash-pelien kehittäjät pystyivät nyt tekemään erittäin monimutkaisiakin pelejä. Internetiin alkoi tulla erilaisia Flash-pelejä, kuten muun muassa Mario-klooneja ja muita suosittujen pelien kopioita. (8.)

Actionscript 2 oli iso askel eteenpäin Flash pelien kehittämisessä, mutta internetin kehittyessä ja pelaajien tarpeen kasvaessa jopa Actionscript 2 koettiin rajoittavaksi. Seuraava kehitysaskel olikin Actionscript 3, joka toi paljon lisää mahdollisuuksia siihen, minkälaisia pelejä voitiin tehdä. Actionscript 3 -kielen myötä voitiin nyt tehdä muun muassa 3D-tehosteita Flash-peleihin (kuva 4), ja uudet ominaisuudet mahdollistivat vieläkin monimutkaisempien ja suurempien pelien kehittämisen.



Kuvio 4. Flashilla tehty 3D:tä hyödyntävä peli (9).

Flash-pelit voivat nykyään olla samanveroisia kuin konsolien pelit.

### 3.2 Pelin kehityksen vaiheet

#### Idean kehittäminen

Valmiin pelin kehittäminen alusta loppuun vie paljon aikaa. Ennen itse pelin tekemistä kannattaa ensin suunnitella hyvin, mitä aikoo tehdä. Hyvä peli vaatii onnistuakseen hyvin mietityn konseptin. Kaikki lähtee ideasta (11). Useimmat ideat eivät pääse kehitysvaiheeseen saakka. Monilla on ideoita, mutta vain hyvät ideat kannattaa kehittää valmiiksi peliksi. Kun saa mielestään hyvän idean, ei kannata heti alkaa tehdä siitä peliä vaan ideaa kannattaa ensin tarkastella kriittisesti. Ideaan ei kannata rakastua, koska se voi heikentää kykyä tarkastella sitä objektiivisesti.

Ensiksi tulee määritellä, mihin peligenreen idea liittyy (12). Jos idea on vaikka pallomainen hahmo, joka hyppii pelimaailmassa eteenpäin vältellen esteitä ja yrittäen päästä paikasta A paikkaan B, genre vaikuttaisi olevan tasohyppely. Idea on tässä vaiheessa hyvin raaka. Seuraavaksi ideaa aletaan miettiä tarkemmin. Tärkeitä kysymyksiä tässä vaiheessa ovat muun muassa, onko peli-idea uniikki, onko peli hauska pelata, onko se koukuttava, onko peli pitkäkestoinen ja miten se eroaa muista saman genren peleistä. Idea on tässä vaiheessa helposti muokattavissa, koska mitään konkreettista ei ole vielä tehty.

Onko peli-idea uniikki? Täysin uniikin idean saaminen on hankalaa, ellei mahdotonta, mutta se ei tarkoita, ettei voisi keksiä ideaa, joka lähestyy jo keksittyä ideaa erilaisesta näkökulmasta. Pelistä voi tehdä uniikin myös lisäämällä siihen ominaisuuksia, joita ei ole muissa samantyyppisissä peleissä. Tasohyppelypeli, jossa pallomainen hahmo yrittää hyppiä tiensä paikasta A paikkaan B, ei ole uniikki idea, joten miten siitä saataisiin uniikimpi? Hahmolle voi esimerkiksi antaa erikoiskykyjä, kuten superkimmoisuus tai vaikka kyky tarrautua seiniin kiinni. Peli-idea ei ehkä ole vielä kukaan kovin uniikki, mutta erottuu jo muista tasohyppelypeleistä. Mitä useimpia erilaisuuksia peliin laittaa, sitä uniikimmaksi se tulee ja erottuu muista ideoista. Asioita ei kuitenkaan kannata lisätä vain tuomaan erilaisuutta.

Onko peli hauska pelata? Kun lisää uusia asioita peli-ideaan, kannattaa myös miettiä, tekeekö lisäys pelistä hauskemman pelata. Uniikkikin idea voi olla täysi floppi, jos peliä ei ole hauska pelata. Mikä sitten tekee pelistä hauskan? Yhtä vastausta on vaikea antaa, koska eri ihmiset pitävät eri asioista. Peli-idea miettiessä kannattaakin unohtaa



kaikkien miellyttäminen, koska se on mahdoton tehtävä. Ensimmäinen kriitikko sille, onko jokin idea hauska vai ei, on tietenkin idean keksijä. Kannattaa miettiä peli ensin hauskaksi ja kysyä sitten muitten mielipiteitä ja päättää, tarvitseeko peli-ideaa muuttaa palautteen perusteella. Jos ollaan tekemässä esimerkiksi räiskintäpeliä, jossa lahdetaan hirviöitä ja saadaan ehdotukseksi, että päähahmo kannattaa muuttaa sateenkaaren väriseksi hevoseksi, kannattaa miettiä tarkasti, onko se hyvä idea pelin kannalta. Kaikkia ei voi miellyttää, joten alkuperäiselle peli-idealle kannattaa olla uskollinen ja tehdä hillittyjä muutoksia muilta saadun palautteen perusteella.

Onko peli-idea koukuttava? Jokainen pelin tekijä haluaa, että hänen peliään pelataan. Jos halutaan myös, että peliä pelataan useammin kuin kerran, tarvitaan peliin jokin koukku. Hyvin suunniteltu koukku saa pelaajan addiktoitumaan peliin ja pelaamaan sitä useamman kerran. Koukkuna toimii usein jokin pelaajan suorittama teko, jota toistetaan useasti, kuten esimerkiksi hirviön tappaminen. Tärkeää koukuttumisessa on, että teosta saadaan palkinto, kuten hirviön tappamisesta saadaan esimerkiksi uusi ase, mutta jos sen saa jokaisen hirviön tappamisesta, pelaaja kyllästyy nopeasti. Hän tietää, että jos hän haluaa uuden aseensa, hänen tarvitsee vain tappaa yksi hirviö. Tämän vuoksi palkinto, kuten uusi ase, kannattaa antaa satunnaisesti hirviön tappamisesta. Kun pelaaja huomaa, että uusi ase ei putoa jokaiselta hirviöltä, hän haluaa tappaa lisää hirviöitä, kunnes uusi ase putoaa. Salakavala koukku on asetettu, ja pelaaja on niellyt sen kokonaan. Peli-ideaa miettiessä kannattaa siis miettiä, mikä on tämän pelin koukku ja saako siitä vielä paremman. Hyvä koukku vaikuttaa siihen, saako peli sata vai miljoona pelaajaa. (10.)

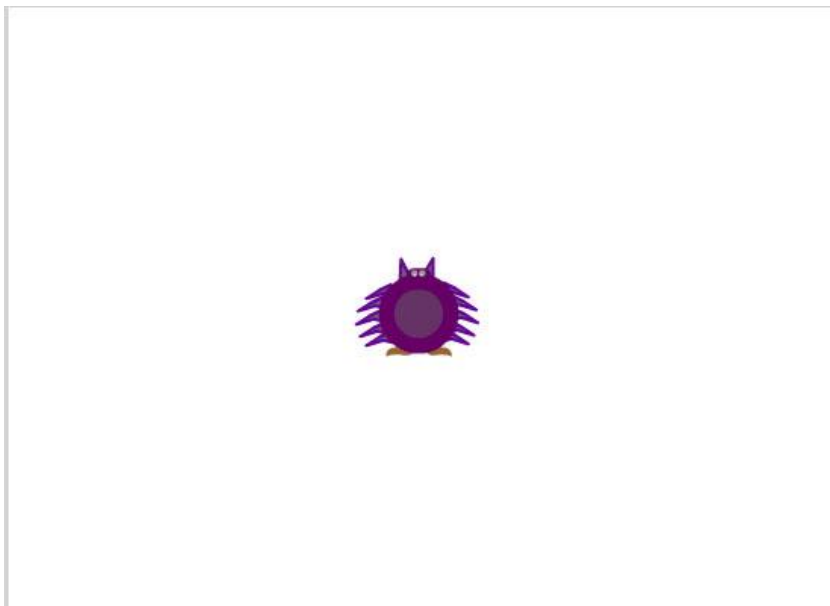
### Prototyypin kehittäminen

Kun idea on hyvin mietitty, seuraava vaihe on prototyypin tekeminen. Prototyyppi on karkea versio pelistä ja havainnollistaa pelin päämekaniikat. Kun ideaa on muokattu ja siihen on lisäilty uusia asioita, voi olla hankalaa tunnistaa pelin päämekaniikat. Prototyyppiä suunnitellessa kannattaa karsia kaikki ominaisuudet pelistä, ja miettiä, mitkä ovat kaikkein tärkeimmät asiat, jotka pelissä pitää olla, jotta sitä voi pelata. (11.)

Seuraavassa on yksi esimerkki-idea. Pelin päähahmo on karvainen pallohahmo. Pelin genre on tasohyppely. Hahmo pystyy hyppäämään ja liikkumaan sivusuunnassa eteenpäin ja taaksepäin. Pelimaailma koostuu tasanteista, joita hyväksikäyttämällä hahmon pitää päästä alkupaikasta maaliin. Hahmo voi muuttaa kimmoisuuttaan napin

painalluksella, joka mahdollistaa suuremman hyppykorkeuden. Pelimaailma on täynnä ilmassa leijuvia kolikoita, joita hahmo voi kerätä koskemalla niitä. Joillakin tasanteilla on myös hirviöitä, joita hahmon pitää välttää koskemasta tai hahmo kuolee. Pelimaailmassa on myös satunnaisesti ilmestyviä timantteja, jotka toimivat samoin kuin kolikot mutta antavat kaksi kertaa enemmän pisteitä. Peli koostuu kymmenestä tasosta, ja jokaisen tason jälkeen hahmo voi käyttää keräämiään pisteitä pelin sisällä olevassa kaupassa, joka myy muun muassa hattuja hahmon ulkonäköä muuttamaan.

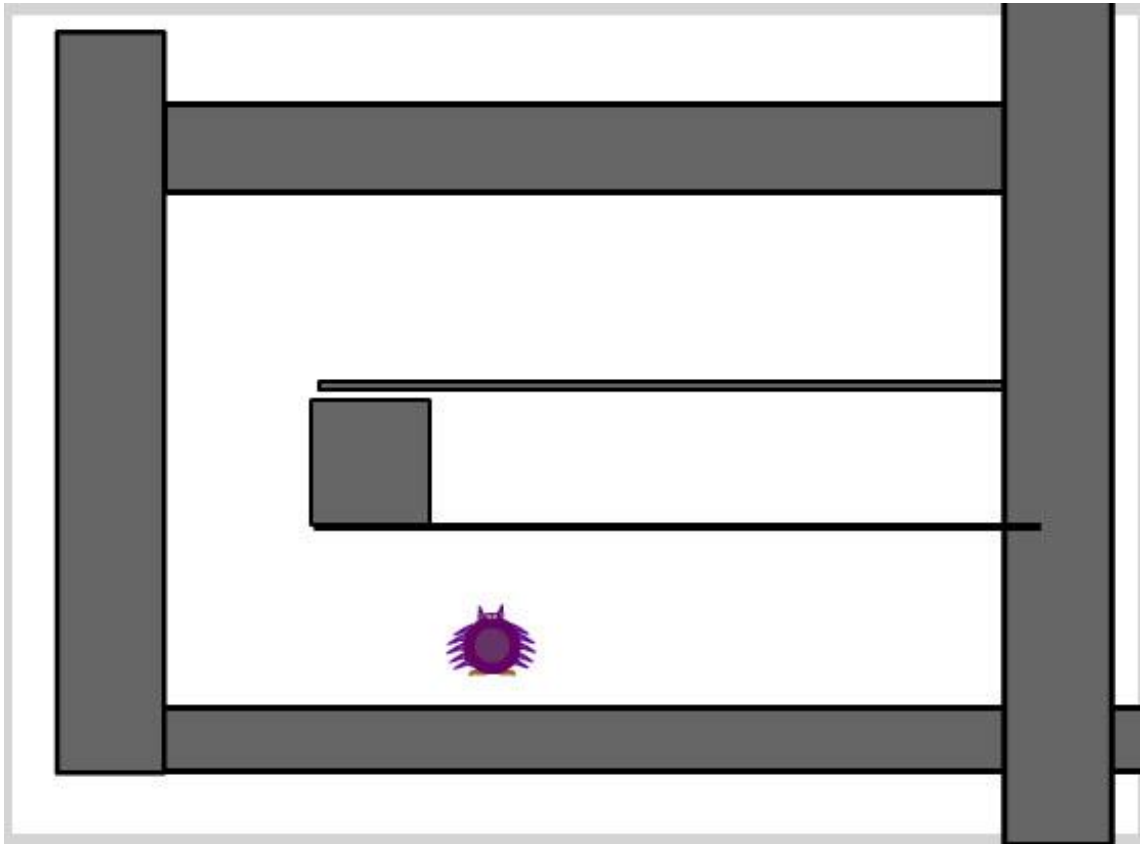
Prototyyppejä tehtäessä edellä mainitusta ideasta pitää ensin poimia pelin päämekaniikat. Koska peli keskittyy karvaisen pallohahmon ympärille, ensimmäisenä pitää siis luoda tämä hahmo. Koska pelimaailmaa ei vielä ole, hahmo on tässä vaiheessa pelkkä graafinen objekti (kuva 5). Seuraavaksi hahmolle annetaan sen vaatimat perustoiminnallisuudet. Hahmon pitää pystyä hyppäämään ja liikkumaan sivusuunnassa eteenpäin ja taaksepäin.



Kuvio 5. Pelihahmo tyhjässä maailmassa.

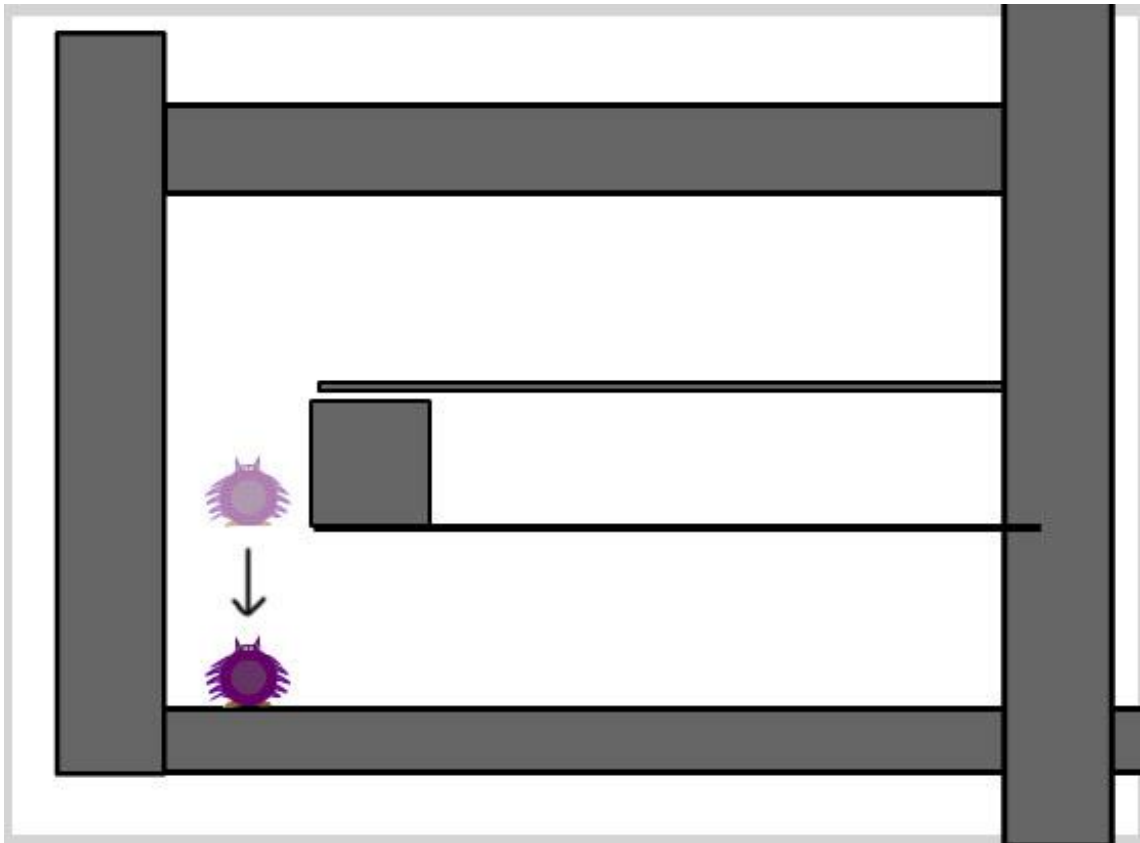
Seuraavaksi tarvitaan pelimaailma, jossa hahmo pystyy liikkumaan. Pelimaailma koostuu tasanteista, joiden päällä hahmo pystyy seisomaan ja liikkumaan. Jotta hyppääminen onnistuu tasanteelta toiselle, tarvitaan myös painovoimaa, muuten hahmo leijuisi vain pois.

Jos prototyyppi suoritetaan nyt, hahmo putoaisi ulos ruudulta suoraan alaspäin (kuva 6).



Kuvio 6. Pelihahmo pelimaailmassa.

Tasanteista pitää tehdä kiinteitä, jotta hahmo ei mene niiden läpi. Tasohyppelypeleissä tärkein osa on törmäystentunnistamisjärjestelmä. Tämä järjestelmä on prototyypin aikaavievin osa-alue: siinä määritellään pohja sille, miten objektit pelimaailmassa reagoivat toisiinsa. Kun hahmo osuu tasanteeseen, sen pitää pysähtyä tasanteen päälle eikä mennä siitä läpi (kuva 7).



Kuvio 7. Hahmo pysähtyy osuessaan tasanteeseen.

Kun nämä asiat ovat valmiina, on erittäin karkea ensimmäinen versio prototyypistä valmis. Ennen valmista prototyyppiä on tehtävä paljon töitä. Hahmon kimmoisuuden muuttaminen suuremman hyppykorkeuden mahdollistamiseksi on myös yksi pelin peruspilareista. Jotta prototyyppi pystyy havainnollistamaan, miltä peli vaikuttaisi valmiina, se on myös lisättävä peliin. Asiat kuten hattujen ostaminen pelikaupasta ovat toissijaisia prototyypin kannalta. Hatut eivät vaikuta itse pelin toiminnallisuuteen, joten hattukauppa jätetään lisäämättä prototyyppivaiheessa.

#### Beta-vaihe ja viimeistely

Prototyyppi etenee edellä mainituissa vaiheissa, kun pelin vaatimia toiminnallisuuksia lisätään peliin. Kun peli sisältää kaikki perustoiminnot, jotka havainnollistavat, miltä peli vaikuttaisi valmiina, peli voidaan siirtää beta-vaiheeseen. Beta-vaiheessa peli voidaan esitellä muille pelaajille, jotta heiltä saataisiin rakentavaa palautetta pelistä. Tämä on tärkeä prosessi, koska muut pelaajat ehtivät vielä osoittamaan selkeitä virheitä pelissä ja antamaan tärkeää palautetta, jota kannattaa käyttää hyväksi pelin parantamiseksi. Beta on pitkälle kehitetty prototyyppi, joten suuriakin muutoksia voidaan vielä tehdä

tässä vaiheessa, mutta jos peli-idea on alusta asti hyvin mietitty, ei suuriin muutoksiin ole yleensä tarvetta.

Beta-vaihe voi kestää kauemmin, kuin siihen pääsemiseen on kulunut aikaa. Viimeistelyssäkin voi mennä todella pitkään, ja pelin kehittäjän pitää itse osata sanoa, milloin peli on tarpeeksi viimeistelty. Kun viimeistely on valmis, seuraava vaihe on pelin julkaisu. Internetissä on useita eri Flash-peliportaaleita, joille käyttäjät voivat ladata oman pelinsä muitten nähtäväksi.

### 3.3 Flash-peliportaalit

Newgrounds.com

Newgrounds on tunnetuin Flash-peliportaali (kuva 8). Kuka tahansa voi ladata sivustolle oman Flash-pelin tai -animaation muitten nähtäväksi. Se on heti nähtävillä sivustolla latauksen jälkeen, ja muut käyttäjät voivat arvioida sen asteikolla yhdestä viiteen ja jättää kommentin.



Kuvio 8. Newgrounds.com-peliportaalin etusivunäkymä (13).

Newgrounds perustettiin harrastelijavoimin 1990-luvun loppupuolella. Se on sallinut muiden käyttäjien lataamia Flash-pelejä ja -animaatioita vuodesta 1999 asti, ja sen suosio on kasvanut paljon siitä lähtien. 2000-luvun aikana Newgrounds nousi maailman suosituimmaksi Flash-pelien portaaliksi. Sivustolle ladataan nykyäänkin useita kymmeniä Flash-pelejä ja -animaatioita päivittäin. (13.)

Newgrounds on erittäin hyvä paikka ladata oma pelinsä muitten nähtäväksi. Muitten, pienempien peliportaalien edustajat voivat ottaa pelin Newgroundsista omille sivuilleen, ja näin peli leviää internetissä hyvin. Käyttäjien korkealle arvostamat pelit saavat myös enemmän näkyvyyttä sivustolla, ja tämä entisestään parantaa pelin levinneisyyttä.

#### Kongregate.com

Kongregate on yksi suurista peliportaaleista, jotka sallivat kaikkien käyttäjien ladata omat Flash-pelinsä sivustolle. Muut käyttäjät voivat arvioida pelejä asteikolla yhdestä viiteen ja kommentoida pelejä. Kongregate tarjoaa pelinkehittäjille, jotka lataavat oman pelinsä sivustolle, osan pelin kehittämistä mainostuloista. Suosittu peli voi siis tuottaa kehittäjälle hyvän summan rahaa mainostuloista. Kongregate on panostanut myös sosiaalisiin ominaisuuksiin, muun muassa laittamalla chat-mahdollisuuden, jossa muitten pelaajien kanssa voi keskustella samalla, kun pelaa jotain peliä sivustolla. (14.)

#### Fgl.com

Flashgamelicense.com eli fgl.com on sivusto, johon kannattaa tutustua, ennen kuin julkaisee pelinsä internetissä. Fgl tarjoaa käyttäjille mahdollisuuden löytää omalle Flash-pelille sponsorin. Sivusto toimii huutokauppaperiaatteella, jossa pelien kehittäjät voivat ladata oman pelinsä sivustolle ja yrittää myydä peliportaalien edustajille mainostilaa omasta pelistään. Useat eri peliportaalien edustajat käyttävät sivustoa, joten hyvällä pelillä on suuri mahdollisuus tulla huomatuksi ja saada hyvä sponsorisopimus.

Parhaassa tapauksessa prosessi on seuraavanlainen:

- Pelin kehittäjä lataa pelinsä sivustolle ja täyttää perustiedot, kuten pelin kuvauksen.
- Pelin kehittäjä laittaa pelin myytäväksi huutokauppaan.
- Sponsorit alkavat tehdä tarjouksia ja kilpailla pelistä.
- Pelin kehittäjä voi lopettaa huutokaupan 72 tunnin varoitusajalla, kun hän tuntee, että hyvä tarjous on tullut.
- Varoitusajan umpeuduttua pelin kehittäjä hyväksyy mielestään parhaan tarjouksen.
- Pelin kehittäjä ja valittu sponsori tekevät sopimuksen pelistä.
- Pelin kehittäjä lisää sponsorin vaatimat mainokset ja muut asiat peliin.
- Kun molemmat osapuolet ovat tyytyväisiä, peli julkaistaan internetissä.

Huippupelit ovat saaneet useiden kymmenien tuhansien dollareitten arvoisia sponsorisopimuksia. Fgl kannattaa siis tarkistaa, ennen kuin julkaisee pelinsä internetissä. Kaikki pelit eivät saa sponsorisopimusta, mutta hyvät pelit eivät usein jää huomiotta. Sopimusehdot kannattaa myös lukea tarkasti.

## 4 Tietovisapelin suunnittelu ja toteutus

### 4.1 Tietovisapelin suunnittelu

Insinööriyönä tehdyn tietovisapelin suunnitteluun osallistui myös asiakkaan edustajana toiminut teknologia- ja IT-johtaja Mikko Wirgentius. Suurimmaksi osaksi sain kuitenkin vapaat kädet suunnitella tietovisapeliä, mutta Mikko Wirgentiuksen piti tietenkin hyväksyä tekemäni asiat.

Tietovisapelin suunnittelu aloitettiin käyttämällä jo olemassa olevaa tietovisapeliä, Älypäättä, suunnittelun pohjana. Älypää-tietovisa sisälsi monia haluttavia ominaisuuksia, joten sitä käytettiin tavoiteltavana päämääränä pelin yleisilmeessä, ulkoasussa ja halutuissa ominaisuuksissa. Tärkeää oli myös pitää mielessä, että tietovisapelin piti olla helposti muokattavissa ilman, että muokkaajan tarvitsisi osata käyttää Flashia. Peli tulisi myös koostumaan kolmesta vaiheesta: alkuvalikosta, pelinäköymästä ja loppuvalikosta.

Suunnittelussa Älypäään käyttämä alkuvalikko todettiin selkeäksi ja toimivaksi (kuva 9).

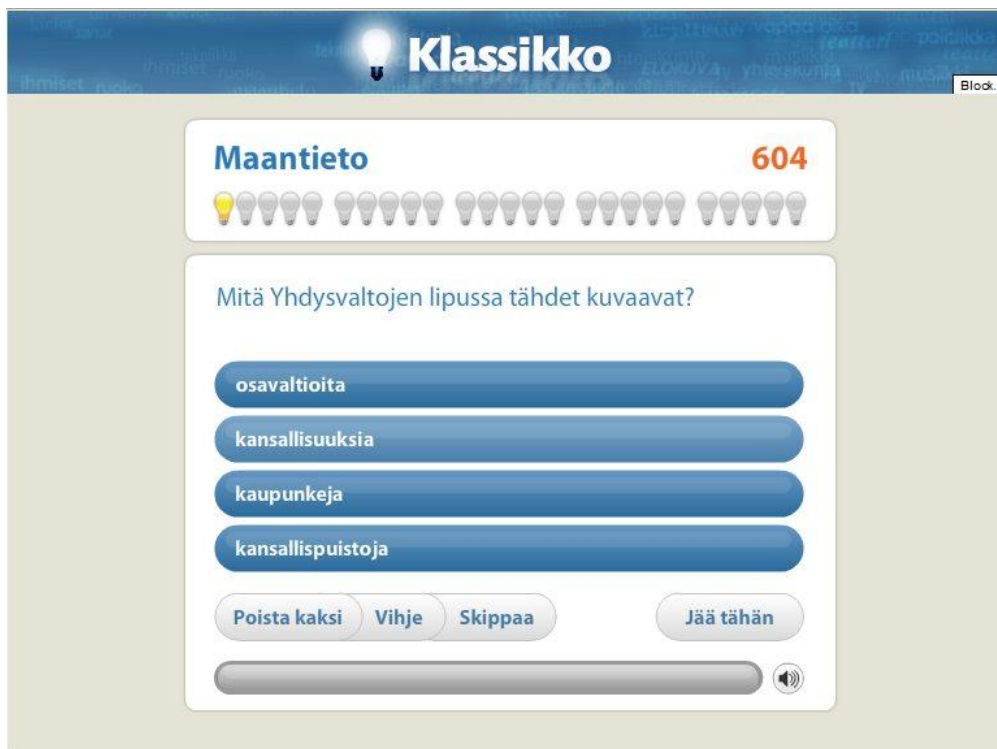


Kuvio 9. Älypää-tietovisapelin alkuvalikkonäkymä (15).



Taustakuvan oli tietenkin oltava vapaasti vaihdettavissa, ja myös tervetuloviestin, joka kertoo pelistä perustietoja, kuten kuvassa 9. Pelin muokkaajan piti pystyä vapaasti vaihtamaan alkuvalikossa näkyvää viestiä ilman Flash-osaamista, joten päädyttiin käyttämään XML-tiedostoja informaation pitäjinä. Tauskuvan ja äänten piti myös olla vapaasti vaihdettavissa, joten nekin pitää ladata palvelimelta. XML-tiedostoja päätettiin käyttää myös ilmaisemaan, mistä tarvittavat komponentit voitiin ladata Flashin sisään. Pelaajan nimimerkin valitseminen todettiin myös tärkeäksi. Nimimerkkiä käytettäisiin pelin myöhemmässä vaiheessa, kun pelaaja voi lähettää saamansa pistemäärän palvelimelle, joten pelaajan nimimerkki on tärkeä ilmaisija, kuka pistemäärän on kerännyt. Ruudulla näkyvä informaatio ladataan peliin siis ulkoisesti, ja Flash-sovellus määrittelee muun muassa käytettävän fontin, fontin koon, tekstin sijainnin ja muita tyyliasetuksia sekä pelaamisen aloittamiseen vaadittavan toiminnallisuuden.

Älypään käyttämä pelinäkömä todettiin myös hyvin suunnitelluksi, ja siinä olevat ominaisuudet olivat samoja, jotka suunniteltavaan peliin haluttiin (kuva 10). Kysymykset päätettiin rajoittaa tiettyyn lukumäärään, ja pelissä piti pystyä seuraamaan, missä kysymyksessä ollaan menossa, aivan kuten Älypään pelinäkömässä ylhäällä olevat lamput, jotka syttyvät, kun vastaa oikein (kuva 10).



Kuvio 10. Älypää-tietovisapelin pelinäkömä (15).

Älypäässä on aina neljä vastausvaihtoehtoa kysymyksestä riippumatta. Tässä projektissa kuitenkin vastausvaihtoehtoja ei haluttu rajoittaa neljään, sillä kysymyksissä saattaisi olla vähemmän vastausvaihtoehtoja. Vastausvaihtoehtojen määrä on siis muuttuja, joten peli piti koodata järkevästi se mielessä pitäen.

Älypää tarjoaa pelaajalle myös kolme oljenkortta, joita voi käyttää apuna missä kysymyksessä tahansa. Kun oljenkorsi on käytetty kerran, sitä ei voi enää käyttää toisessa kysymyksessä. Tämä ominaisuus todettiin hauskaksi, ja se haluttiin mukaan projektiin. Taas piti pitää mielessä, että kysymykset eivät välttämättä noudattaisi samaa kaavaa kuin Älypäässä, eli esimerkiksi jos vastausvaihtoehtoja olisi alle neljä, ”poista kaksi” -oljenkorsi ei voisi olla käytössä.

Jossain kysymyksissä saattaisi puuttua ”Vihje”-oljenkorsi. Projektia tehtäessä pitäisi oljenkorsijärjestelmä tehdä helposti muokattavaksi ja helposti mukautuvaksi erilaisiin kysymyksiin. Pelin keskeyttäminen pelaajan toimesta, ”Jää tähän” -nappula, todettiin turhaksi. Pelin tulisi päättyä, kun pelaaja vastaa väärin kysymykseen, mutta pelaajaa ei rangaista väärästä vastauksesta, toisin kuin Älypäässä, jossa väärin vastattaessa peli päättyy ja pelaaja menettää jo kerättyjä pisteitä.

Kuvassa 10 alhaalla oleva aikajanapalkki, joka kuvastaa kysymyksen vastaamiseen varattua aikaa animaation avulla, todettiin myös hyväksi ideaksi. Antamalla pelaajalle aikaraja pystytään välttämään mahdolliset huijausyritykset, kuten vaikka oikean vastauksen ”googlettaminen”.

Kysymykset päätettiin myös laittaa XML-tiedostoon, josta ne parsittaisiin Flashissa ja näytettäisiin pelaajalle. Pelinäkömän staattisia objekteja olisivat muun muassa oljenkortet, ”lamput”, pistemäärä ja aikajanapalkki. Muu informaatio tulee XML-tiedostoista.

Älypään käyttämä loppuvalikkonäkymä (kuva 11) sisälsi haluttuja tietoja, mutta ulkoasu voisi olla paremmin järjestetty. Älypää antaa pelaajalle palautetta suorituksen perusteella kertomalla pelaajalle hänen tasonsa, kuten kuvassa 11 annettu taso ”kisälli”.



Kuvio 11. Älypää-tietovisapelin loppuvalikkonäkymä (15).

Tasojen ja palautteen antaminen pelisuorituksen perusteella todettiin hyväksi ideaksi ja halutuksi ominaisuudeksi. Loppunäkymässä pitäisi myös näkyä perustiedot, kuten pistemäärä ja se, mihin kysymykseen peli loppui. Uudelleen pelaaminen kuuluisi myös perusominaisuuksiin. Tuloksen jakaminen kaverin kanssa tai Facebookissa jakaminen olivat myös haluttuja ominaisuuksia, mutta eivät välttämättä pakollisia.

## 4.2 Tietovisapelin toteutus

Pelin toteuttamiseen päätin käyttää Flashdevelop-ohjelmaa, koska se soveltuu olio-ohjelmointiin paremmin kuin Adoben Flash-ohjelmisto. Kieleksi valitsin Actionscript 3:n, koska se on kehittyneempi kuin edeltävät Actionscript-versiot ja tarjoaa parempia työkaluja ohjelmiston kehittämiseen.

Ensimmäiseksi tavoitteeksi asetin prototyypiversion, jossa ovat mukana keskeiset ominaisuudet ja kaikki kolme pelin vaihetta: alkuvalikko, pelinäköymä ja loppuvalikko. Aluksi päätin tehdä prototyypistä niin sanotun offline-version, mikä tarkoittaa, että mitään komponentteja ei ladata palvelimelta, vaan ne ovat kaikki samassa kansiossa koneella. Myöhemmissä versioissa muokkaisin sitten niin, että tarvittavat komponentit ladattaisiin palvelimelta. Tämä menettelytapa on järkevä, sillä samasta kansioista ladattaessa ei tarvitse huolehtia mistään yhteys- tai palvelinpuolen ongelmista. Ylimääräistä koodausta ei myöskään aiheudu, sillä prosessi on sama molemmissa tapauksissa ja palvelinlataukseen siirtyminen vaatii vain latausosoitteen vaihdon: esimerkiksi C-levyllä oleva latausosoite vaihdetaan nyt www-osoitteessa olevaan osoitteeseen.

Keskeiset ulkopuoliset tiedostot ovat seuraavat:

- config.xml-tiedosto.
- kysymykset.xml-tiedosto.
- sisalto.png-tiedosto.
- tausta.png-tiedosto.
- oikein.mp3-tiedosto.
- vaarin.mp3-tiedosto.

Kaikki mainitut tiedostot piti luoda, jotta pystyttäisiin heti alusta asti tekemään järkevä pohja pelille. Tiedostot olisi voinut myös upottaa Flashiin ilman, että tarvitsee tehdä mitään latauksia, mutta myöhemmässä vaiheessa koko koodi pitäisi tehdä alusta asti, koska valmiissa versiossa tiedostot kuitenkin ladataan palvelimelta, joten on järkevää koodata latausprosessi alusta asti, vaikka tiedostot ladataankin aluksi omalta koneelta samasta kansioista. Aluksi tiedostot sisältävät tilapäistä tietoa, ja myöhemmässä vaiheessa ne korvataan todellisella tiedolla, esimerkiksi oikeat kysymykset syötetään kysymys.xml-tiedostoon.

Config.xml-tiedosto sisältää pelin käyttämiä asetuksia ja muuta tärkeää tietoa. Se sisältää esimerkiksi tyyliasetuksia teksteille, kuten fontin värin ja tervetuloa-tekstin sisällön. Siinä säädetään myös pelin kannalta tärkeitä asetuksia, kuten kysymysten määrää ja kysymyksistä saatua pistemäärää. Config.xml kertoo myös, mistä ladata muut ulkopuoliset tiedostot.

Kysymykset.xml-tiedosto sisältää kysymykset ja niihin liittyvät tiedot. Koodiesimerkkissä 6 nähdään kysymyksen rakenne XML-tiedostossa.

```
<kysymys>
    <teksti></teksti>

    <options>
        <option oikein="true"></option>
        <option></option>
        <option></option>
        <option></option>
    </options>

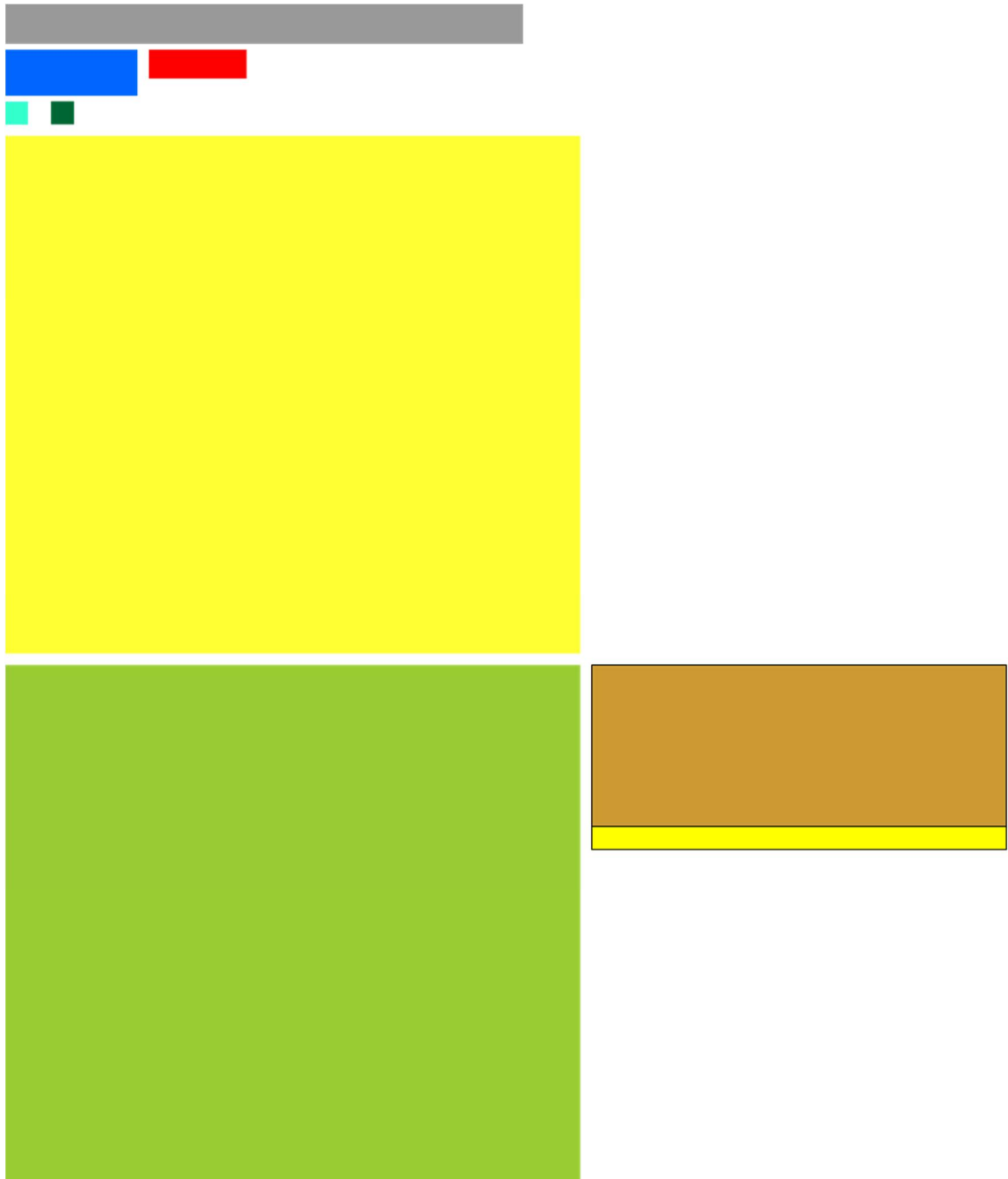
    <vihje></vihje>

    <vaikeus></vaikeus>
</kysymys>
```

Koodiesimerkki 6. Kysymyksen XML-rakenne.

Teksti-tagi sisältää itse kysymyksen. Options-tagi sisältää vastausvaihtoehdot option tagien sisällä. Yksi vastausvaihtoehtojen tagi sisältää myös parametrin "oikein", joka ilmaisee, että tämä vaihtoehto on oikea vastaus ja näin ollen muut vaihtoehdot ovat väärin. Vihje-tagin sisällä on vihje-oljenkorteen liittyvä teksti, joka näytetään pelaajalle, kun hän käyttää kyseisen oljenkorren. Vaikeus-tagi määrittelee kysymyksen vaikeustason muihin kysymyksiin verrattuna. Vaikeus voi olla mikä tahansa positiivinen kokonaisluku, kun ykkönen on matalin vaikeustaso.

Sisalto-kuvatiedosto sisältää pelissä käytettyjä graafisia komponentteja (kuva 12). Kaikki pelin grafiikka on yhdessä tiedostossa lukuun ottamatta pelin taustakuvaa.



Kuvio 12. Sisalto.png-tiedosto.

Kuvassa 12 nähdään pelissä olevat graafiset komponentit. Ylhäällä ensimmäisenä on vastausvaihtoehdoissa käytettävä nappula-grafiikka, ja sen alapuolella kaksi pienempää nappula-grafiikkaa. Pienet neliöt niiden alapuolella ovat oikein- ja väärin-kuvakkeet, jotka vastaavat Älypäässä käytettyjä lamppeja. Kaksi isoa laatikkoa ovat

pelialueen taustakuvia, eli se alue, jossa ovat alkuvalikon komponentit ja pelinäkömänn komponentit. Ruskea-keltainen laatikko on vihje-laatikon grafiikka.

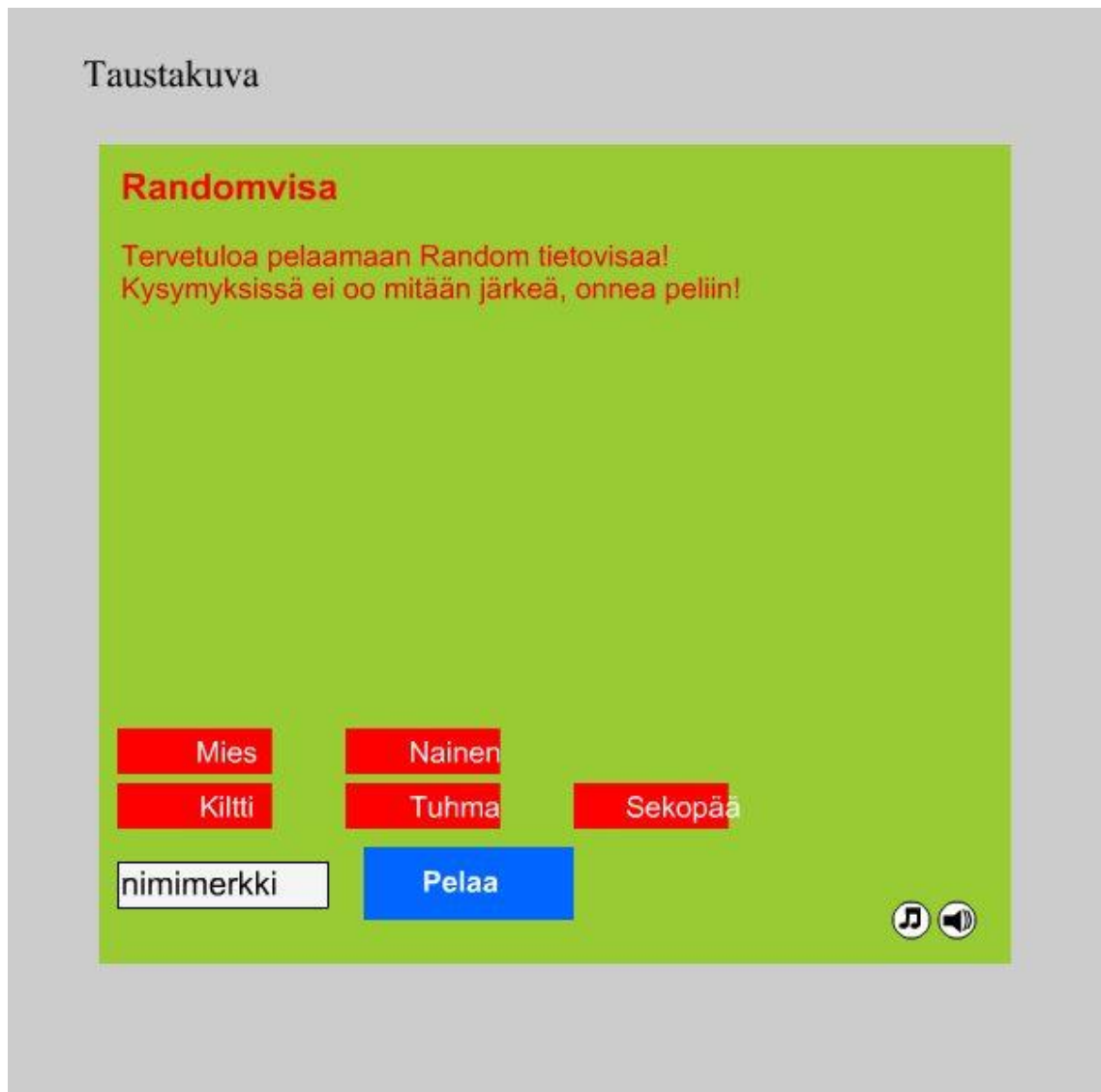
Kun sisalto-kuvatiedosto on ladattu Flashiin, siitä leikataan eri graafiset objektit irti. Tämä tehdään käyttämällä Actionscript-koodia, joka nähdään koodiesimerkissä 7.

```
private function sliceContentImage():void
{
var bm:Bitmap = loaded[2];
var bmd:BitmapData = bm.bitmapData;
//big button
var bmd1:BitmapData = new BitmapData(450, 35);
bmd1.copyPixels(bmd, new Rectangle(0, 0, 450, 35), new Point(0, 0));
pics.push(bmd1);
//small button
var bmd2:BitmapData = new BitmapData(115, 40);
bmd2.copyPixels(bmd, new Rectangle(0, 40, 115, 40), new Point(0, 0));
pics.push(bmd2);
//etc....
}
```

Koodiesimerkki 7. Sisalto.png-kuvan objektien leikkaaminen Actionscriptissa.

Eri graafisten objektien sijainti kuvassa on erittäin tärkeää, koska ne leikataan kuvasta irti aina samasta paikasta pikselin tarkkuudella. Koodiesimerkissä 7 nähdään, että "big button" eli vastausvaihtoehdoissa käytetty iso nappula -grafiikka on määritelty tarkasti koordinaattien kohtaan (0,0) ja sen leveys on 450 pikseliä ja korkeus on 35 pikseliä lähtien sen alkukoordinaateista (0,0). Leikattu graafinen objekti tallennetaan muuttujan sisään, joten siitä voidaan helposti luoda kopioita Flashin sisällä. Leikkausprosessia ei tarvitse toistaa ensimmäisen kerran jälkeen, vaan jo leikatuista objekteista tehdään kopioita ja niitä voidaan käyttää tarvittavissa paikoissa. Graafisen objektin pitää olla määriteltyjen parametrien sisällä, mutta sen ei tarvitse täyttää koko pinta-alaa. Png-tiedostomuoto valittiin sen takia, että siinä voidaan määritellä läpinäkyvyyttä. Graafiset objektit voivat siis olla muunkin näköisiä kuin rumia laatikoita, koska läpinäkyvyyden avulla käyttäjät eivät näe, että graafinen objekti vie laatikon kokoisen tilan.

Kuvassa 13 nähdään pelin prototyypiversion alkuvalikon näkymä, jossa graafiset komponentit on leikattu sisälto-kuvatiedostosta irti ja sijoitettu paikoilleen.



Kuvio 13. Tietovisapelin prototyypiversion alkuvalikkonäkymä.

Tervetuloa-viesti tulee config.xml-tiedostosta. Flashissa määritellään kaikkien komponenttien sijainnit ja toiminnallisuus, kuten nimimerkin valinta ja pelaa-nappula. Kuvassa 13 alhaalla oikeassa kulmassa on myös nappulat, joista saa musiikin ja äänet pois päältä. Punaiset nappulat ovat valintoja, joita pelaaja voi tehdä ennen pelin alkua. Pelaaja voi valita sukupuolen, mies tai nainen, ja luonteenpiirteen, kiltti, tuhma tai sekopää. Tehdyt valinnat vaikuttavat pelin loppuvalikossa annettavaan arvonimeen. Nimi-merkkiä käytetään, jos pelaaja haluaa pelin lopussa lähettää pisteensä muitten nähtäväksi tulostaulukkaan.



Kuvassa 14 nähdään prototyypin pelinäkömää. Ylhäällä vasemmalla keltaisella taustalla nähdään viisi neliötä, jotka kuvastavat, kuinka monta kysymystä pelissä on yhteensä ja missä kysymyksessä nyt mennään. Vihreäksi muuttunut neliö kuvastaa oikeaa vastausta, eli ensimmäiseen kysymykseen on vastattu oikein ja nyt ollaan toisessa kysymyksessä. Vaikka sen yläpuolelle on kirjoitettu erikseen vielä "kysymys 2", on visuaalinen esitys silti tarpeellinen, koska se auttaa paremmin hahmottamaan, kuinka pitkällä pelissä ollaan ja kuinka paljon kysymyksiä on vielä jäljellä. Näitten alapuolella on harmaa aikajanapalkki, jossa kuvan 14 tilanteessa aika on kulumassa loppuun, joten aikajanassa sisällä oleva animaatiopalkki on muuttunut punaiseksi ja pienenee.



Kuvio 14. Tietovisapelin prototyypiversion pelinäkömää.

Kuvassa 14 oleva kysymys tulee kysymykset.xml-tiedostosta, jossa prototyypin tässä versiossa on tilapäistä tietoa, ja siksi kysymys ja vastausvaihtoehdot ovat hölmöjä. Vastausvaihtoehtojen alapuolella on kolme oljenkorsi-nappulaa. Ne ovat itsestään selviä eli "poista kaksi" -oljenkorsi poistaa kaksi väärää vastausvaihtoehtoa, "vihje" antaa vihjeen ja "skippaa" ohittaa kysymyksen ja antaa toisen sen tilalle.

Kysymykset valitaan sattumanvaraisesti kaikkien kysymysten joukosta. Prototyypiversiossa on yhteensä kymmenen kysymystä, joista valitaan viisi sattumanvaraisesti. Sen jälkeen valitut kysymykset järjestetään vaikeustason mukaan ja esitetään pelaajalle helpoimmasta alkaen. Samaa kysymystä ei siis esitetä uudelleen yhden pelikerran aikana. Uudella pelikerralla sama kysymys voi tietenkin tulla, ja prototyypin tapauksessa samoja kysymyksiä tulee koko ajan. Jotta pelaaja ei voi kuitenkaan opetella ulkoa vastausta riippuen vastausvaihtoehdon paikasta, esimerkiksi oikea vastaus on aina ylimäinen vaihtoehto, peli vielä sekoittaa vastausvaihtoehtojen paikat aina, kun kysymys esitetään. Pelaaja siis pakotetaan miettimään hetki ennen vastausta, vaikka kysymys olisi sama ja pelaaja tietää vastauksen. Tämä on helppo keino pitää peli mielenkiintoisena useamman pelikerran ajan. Tämän helpon keino koodaus oli kuitenkin hieman hankalaa, kun oikea vastaus tahtoi hukkaa vastausvaihtoehtojen sekoittamisen jälkeen. Muutaman yrityksen jälkeen sain kuitenkin ongelmat ratkaistua.

Kuvassa 15 nähdään pelin prototyypiversiion loppuvalikko. Heti ylhäällä vasemmalla keltaisella taustalla nähdään pelin tilanne sen loputtua, eli viisi neliötä, jotka kertovat, että pelaaja vastasi yhteen kysymykseen oikein ja vastasi väärin toisessa kysymyksessä. Sen alapuolella peli kertoo pelaajalle, kuinka paljon pisteitä hän keräsi ennen väärää vastausta, ja antaa hänelle tason, joka määräytyy kerätyn pistemäärän perusteella. Tasot on löytyvät config.xml-tiedostosta, jossa on muun muassa määritelty tason titteli ja siihen vaadittava minimipistemäärä. Taso määräytyy myös alkuvalikossa tehtyjen valintojen perusteella. Prototyypiversiossa on tasot nimetty kuvaavasti valintojen perusteella, eli kuvassa 15 annetusta tasosta voidaan helposti päätellä, että pelaaja on valinnut sukupuolekseen mies ja luonteekseen tuhma.



Kuvio 15. Tietovisapelin prototyypin loppuvalikko.

Asiakas halusi mahdollisuuden järjestää kilpailuja pelissä, joten pelaaja voi halutesaan antaa sähköpostiosoitteensa ja lähettää sen palvelimelle. Mukana lähetetään myös pelaajan keräämä pistemäärä, ansaittu taso ja nimimerkki.

Pelin näkyvyyden parantamiseksi ja uusien pelaajien houuttelemiseksi päätettiin myös lisätä kaksi toimintoa. Ensimmäinen on "kutsu kaverisi" -nappula, joka avaa tekstikentän, johon pelaaja voi syöttää kaverin sähköpostiosoitteen, joka sitten lähetetään palvelimelle. Palvelin luo viestin ja lähettää sen annettuun sähköpostiosoitteeseen. Toinen toiminto on "jaa tulos FB:ssa" -nappula, joka ottaa yhteyttä Facebookin omaan palveluun. Palveluun lähetetään tietoa pelistä, kuten pelaajan pistemäärä, taso, nimimerkki ja lyhyt viesti, jossa mainostetaan peliä ja kutsutaan viestin lukijaa pelaamaan. Tämä

viesti jaetaan käyttäjän Facebook-seinällä. Toiminto perustuu tietenkin käyttäjän vapaaehtoisuuteen, eli käyttäjän pitää olla kirjautunut Facebookiin ja hyväksyä viestin lähettäminen seinälle. Sosiaalinen media on hyvä paikka mainostaa tuotteita, ja jos saa tavalliset ihmiset tekemään mainostamisen puolestaan, se on kaikkein parasta.

Loppuvalikossa on myös tietenkin ”pelaa uudelleen” -nappula, josta pelaaja lähetetään takaisin alkuvalikkoon ja hän voi halutessaan vaihtaa jo tehtyjä valintoja ja pelata uudelleen. Jo valitut valinnat ovat valittuina automaattisesti, joten pelaajan ei tarvitse halutessaan muuttaa mitään, vaan hän voi vain aloittaa uuden pelikerran.

Kaikkien ominaisuuksien lisäämisen jälkeen alkoi pitkä hiomisvaihe. Peleissä viimeistelyvaihe saattaa kestää kauemmin kuin itse pelin tekeminen siihen asti. Hiomisella tarkoitan esimerkiksi tekstikenttien koon muuttamista, värin vaihtamista, paikan vaihtamista, eli pieniä mutta tärkeitä asioita, että saadaan hyvännäköinen ja huoliteltu lopputulos. Kun peli oli hiottu täydelliseksi, se luovutettiin asiakkaalle, josta tämä teki oman version vaihtamalla graafiset komponentit ja muokkaamalla XML-tiedostoja (liite 1).

## 5 Yhteenveto

Flash on erittäin suosittu tekniikka ja erittäin laajasti käytössä. Se soveltuu mainiosti pelien tekemiseen, ja internet on täynnä erilaisia pelejä, jotka on tehty käyttäen Flashia. Flash-pelien suuren suosion johdattamana myös mainostajat ja markkinoijat ovat kiinnostuneet niistä. Flash-pelit ovat mainio tapa mainostaa jotain tuotetta tai yritystä tavalla, jota käyttäjät eivät pidä häiritsevänä.

Flash-pelien tekeminen on myös suhteellisen helppoa. Niitä voi alkaa tehdä kuka tahansa vaikka yksin, kunhan on perehtynyt Flash-ohjelmaan ja Actionscript-kieleen. Pelien tekemisen helppouden ansiosta Flashilla on lukuisia kehittäjiä, ja joka päivä tulee uusia Flash-pelejä internetiin.

Insinööriyössä tehtiin Flash-tekniikalla tietovisapeli käytettäväksi yrityksen mainonnassa. Suunnittelun pohjana käytettiin jo olemassa olevaa tietovisapeliä, jonka avulla muodostettiin omanlainen lopputulos. Yritykselle tehty tietovisapeli koostuu kolmesta päävaiheesta. Ensimmäisenä pelaaja näkee alkuvalikon, jossa hän voi valita itselleen nimimerkin, sukupuolen ja vapaaehtoisen pelin säätöihin vaikuttavan option. Toinen päävaihe on itse tietovisapelin pelaaminen, jossa pelaajalle esitetään kysymys ja vastausvaihtoehdot. Pelaaja voi käyttää pelissä valmiina olevia apukeinoja, joilla hän voi poistaa vääriä vaihtoehtoja, saada vihjeen tai ohittaa kysymyksen. Oikein vastaamalla pelaaja saa uuden kysymyksen uusilla vastausvaihtoehdoilla. Peli jatkuu, kunnes pelaaja vastaa väärin tai hän vastaa oikein kaikkiin esitettyihin kysymyksiin. Kolmas päävaihe on loppuvalikko, jossa pelaajan suoritus arvostellaan. Pelaajan on mahdollista tallentaa pelissä saatu pistetulos palvelimelle tai pelata uudelleen.

Flash-pelit voivat olla erittäin monipuolisia, ja niiden sisältöä pystytään muokkaamaan ulkopuolelta. Tätä ominaisuutta käytettiin tässä projektissa hyväksi. Tietovisapeli suunniteltiin helposti muokattavaksi ilman, että muokkaajan tarvitsee koskea Flashiin tai edes tuntea Flash-tekniikkaa ollenkaan. Peli toimii eräänlaisena kehyksenä tai tyhjänä tauluna, johon voidaan vaikuttaa ulkopuolelta. Pelin sisältö ladataan palvelimelta ja näytetään oikeassa muodossa pelin sisällä. Flash huolehtii pelin logiikasta ja toiminnallisuudesta. Pelkkiä kuvatiedostoja ja XML-tiedostoja vaihtamalla pelistä saadaan aivan erinäköinen ja erituntuinen kokemus.

Projektin tekeminen oli miellyttävä kokemus. Tavallisesti Flash-pelit tehdään itsenäisesti, eli kaikki on Flash-tiedoston sisällä valmiina eikä Flash lataa mistään ylimääräisiä tiedostoja. Tässä projektissa kaikki sisältö piti ladata palvelimelta, joten se oli erilaista kuin mitä olen tehnyt aiemmin Flash peleissä. Palvelinyhteyksien ja eri ladattavien tiedostojen parsiminen toi mielenkiintoista haastetta projektiin.

## Lähteet

- 1 Grandmasters of flash an interview with the creators of flash. 2008. Verkkodokumentti. Coldhardflash. <<http://coldhardflash.com/2008/02/grandmasters-of-flash-an-interview-with-the-creators-of-flash.html>>. 2008. Luettu 14.1.2013.
- 2 Tasawr Interactive. 2010. Adobe flash history and basics. Verkkodokumentti. <<http://www.slideshare.net/teksymmetryan/adobe-flash-history-and-basics>>. 2010. Luettu 14.1.2013.
- 3 Getting started with flash professional cs6. 2012. Verkkodokumentti. Video2brain. <[http://www.descargasnsn.com/uploads/posts/2012-07/1342040813\\_2012.07.12\\_01h04m17s\\_004\\_.jpg](http://www.descargasnsn.com/uploads/posts/2012-07/1342040813_2012.07.12_01h04m17s_004_.jpg)>. 2012. Luettu 14.1.2013.
- 4 Adobe flash player PC penetration. 2009. Verkkodokumentti. Adobe. <[http://www.adobe.com/products/player\\_census/flashplayer/PC.html](http://www.adobe.com/products/player_census/flashplayer/PC.html)>. 2009. Luettu 17.1.2013.
- 5 Actionscript. 2013. Verkkodokumentti. Wikipedia. <<http://en.wikipedia.org/wiki/ActionScript>>. 2013. Luettu 21.1.2013.
- 6 Flashdevelop. 2013. Verkkodokumentti. Flashdevelop. <<http://www.flashdevelop.org/>>. 2013. Luettu 24.1.2013.
- 7 Object oriented programming OOP. 2013. Verkkodokumentti. Webopedia. <[http://www.webopedia.com/TERM/O/object\\_oriented\\_programming\\_OOP.html](http://www.webopedia.com/TERM/O/object_oriented_programming_OOP.html)>. 2013. Luettu 24.1.2013.
- 8 Hanson, Jamie. 2009. History of flash games and how it has evolved. Verkkodokumentti. <<http://www.articlesbase.com/internet-articles/history-of-flash-games-and-how-it-has-evolved-747482.html>>. 2009. Luettu 5.2.2013.
- 9 3D rally racing. 2013. Verkkodokumentti. Freegamesaccess. <<http://www.freegameaccess.com/images/3d-rally-racing-sml.gif>>. 2013. Luettu 6.2.2013.
- 10 Wong, David. 2010. 5 creepy ways video games are trying to get you addicted. Verkkodokumentti. <[http://www.cracked.com/article\\_18461\\_5-creepy-ways-video-games-are-trying-to-get-you-addicted.html](http://www.cracked.com/article_18461_5-creepy-ways-video-games-are-trying-to-get-you-addicted.html)>. 2010. Luettu 6.2.2013.
- 11 Stages of game development. 2012. Verkkodokumentti. Over00. <<http://www.over00.com/?p=2060>>. 2012. Luettu 7.2.2013.
- 12 Kraft, Gina. 2011. The stages of game development. Verkkodokumentti. <<http://ezinearticles.com/?The-Stages-of-Game-Development&id=5658605>>. 2011. Luettu 7.2.2013.
- 13 Fulp, Tom. 2013. Newgrounds. Verkkodokumentti. <[www.newgrounds.com](http://www.newgrounds.com)>. 2013. Luettu 11.2.2013.
- 14 Kongregate. 2013. Verkkodokumentti. Kongregate. <[www.kongregate.com](http://www.kongregate.com)>. 2013. Luettu 11.2.2013.

- 15 Älypää. 2013. Verkkodokumentti. Älypää. <[www.alypaa.com](http://www.alypaa.com)>. 2013. Luettu 14.2.2013.
- 16 Extensible Markup Language. 2008. Verkkodokumentti. W3C. <<http://www.w3.org/TR/xml/>>. 2008. Luettu 19.2.2013.



## Tietovisapelin valmis asiakkaan muokkaama versio

**kaalimato.com**  
**Tietovisa**

**Kaalimato.com -tietovisa**

Anna alle nimimerkki, jolla haluat pelata. Nimimerkki tulee näkyviin tuloslistaan.

Valitse myös oletko tuhma vai kiltti, ja oletko mies vai nainen. Valinnat vaikuttavat ansaitsemaasi "arvonimeen".

Onnea peliin!

Mies Nainen  
Kiltti Tuhma

nimimerkki  Pelaa

Like Send 326 people like this. Sign Up to see what your friends like.

Pelin pelaamisen tarvitsel selaimen Flash-tuen. Sen voit ladata [täältä](#).

**kaalimato.com**  
Tästä verkkokauppaan

**Kunniataulu**

**Viikon parhaat** **All-time best**

kul...	16960
Mestariarakastaja	
veijo	12920
Kuuma pakkaus	
NiGo	12160
Playboy	
mr n1ce guy	10000
Lisenssi-seksologi	
Kuuma	9240
Lisenssi-seksologi	
emm	9160
Kuuma pakkaus	
kissimirri	8400
Lisenssi-seksologi	
anom79231	8080
Lisenssi-seksologi	
k	8000
Lisenssi-seksologi	
hotti	6840
Seksinoviisi	
huorraaa	5920
Aloitteleva nymfomaani	

**kaalimato.com**  
**Tietovisa**

**Kysymys 3** **880**

Minkä eläimen orgasmi kestää pisimpään?

Hevosen  
Lampaan  
Sian  
Kukon

Poista kaksi Vihje Skippaa

Like Send 326 people like this. Sign Up to see what your friends like.

Pelin pelaamisen tarvitsel selaimen Flash-tuen. Sen voit ladata [täältä](#).

**kaalimato.com**  
Tästä verkkokauppaan

**Kunniataulu**

**Viikon parhaat** **All-time best**

kul...	16960
Mestariarakastaja	
veijo	12920
Kuuma pakkaus	
NiGo	12160
Playboy	
mr n1ce guy	10000
Lisenssi-seksologi	
Kuuma	9240
Lisenssi-seksologi	
emm	9160
Kuuma pakkaus	
kissimirri	8400
Lisenssi-seksologi	
anom79231	8080
Lisenssi-seksologi	
k	8000
Lisenssi-seksologi	
hotti	6840
Seksinoviisi	
huorraaa	5920
Aloitteleva nymfomaani	


**kaalimato.com**  
**Tietovisa**

**880**

♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥

Vastasit väärin ja peli loppui siihen.  
Keräsit 880 pistettä.  
Olet tasoltasi **Aloittelija!**

Lähetä pisteesi tuloslistalle klikkaamalla  
"Lähetä". Osallistuaksesi viikkopalkintojen  
jakoon, anna myös sähköpostiosoitteesi!

**Kunniataulu** 

Viikon parhaat	All-time best
<b>kul...</b> <i>Mestariarakastaja</i>	<b>16960</b>
<b>veijo</b> <i>Kuuma pakkaus</i>	<b>12920</b>
<b>niGo</b> <i>Playboy</i>	<b>12160</b>
<b>mr n1ce guy</b> <i>Lisensiseksologi</i>	<b>10000</b>
<b>Kuuma</b> <i>Lisensiseksologi</i>	<b>9240</b>
<b>emm</b> <i>Kuuma pakkaus</i>	<b>9160</b>
<b>kissimirri</b> <i>Lisensiseksologi</i>	<b>8400</b>
<b>anom79231</b> <i>Lisensiseksologi</i>	<b>8080</b>
<b>k</b> <i>Lisensiseksologi</i>	<b>8000</b>
<b>hotti</b> <i>Seksinoviisi</i>	<b>6840</b>
<b>huorraa</b> <i>Aloitteleva nymfomaani</i>	<b>5920</b>

**kaalimato.com**  
Tästä verkkokauppaan

Like Send 326 people like this. Sign Up to see what your friends like.

Pelin pelaamisen tarvitset selaimen Flash-tuen. Sen voit ladata [täältä](#)

