



# HTML5 PELIKEHITYKSESSÄ

Juho Ryhänen

Opinnäytetyö  
Maaliskuu 2015  
Tietojenkäsittelyn koulutusohjelma  
Digimedia

## TIIVISTELMÄ

Tampereen ammattikorkeakoulu  
Tietojenkäsittelyn koulutusohjelma  
Digimedia

RYHÄNEN, JUHO:  
HTML5 Pelikehityksessä

Opinnäytetyö 31 sivua  
Maaliskuu 2015

---

Opinnäytetyön tavoitteena oli tutkia, miten HTML5 soveltuu pelikehitykseen sekä selvittää uusien verkkostandardien merkitys, käyttömahdollisuudet ja hyödyt pelikehityksen näkökulmasta. Tarkoituksena oli tuottaa siihen liittyvää (itse)opiskelumateriaalia, joka auttaisi muita pelikehityksestä kiinnostuneita opiskelijoita ymmärtämään HTML5:n hyötyjä ja mahdollisuuksia pelikehityksen näkökulmasta.

Oppaassa keskitytään HTML5-pelikehityksessä käytettäviin verkkostandardeihin, niiden taustoihin ja pelikehityksessä tarvittaviin ohjelmistoihin sekä (2D) pelikehittämiseen.

Opinnäytetyön ohessa suunniteltiin ja toteutettiin yksinkertainen HTML5-selainpeli, The Jumper. Peli toteutettiin tutkitusta aineistosta opituilla asioilla sekä käyttäen apuna Quintus-pelimoottoria. Peli on tasoloikkapeli, jossa pelin sankarihahmo kerää pisteitä ja yrittää selvitä pelin tasojen läpi kuolematta.

## **ABSTRACT**

Tampereen ammattikorkeakoulu  
Tampere University of Applied Sciences  
Degree Programme in Business Information Systems  
Option of Digimedia

JUHO RYHÄNEN:  
HTML5 in Game Development

Bachelor's thesis 31 pages  
March 2015

---

The object of this thesis was to research how HTML5 suits for game development and to find out about the new web standards along with their possible uses and benefits regarding game development. The purpose was to produce (self-)study material, which would help other students who are interested in game development to understand the benefits and possibilities of HTML5 game development.

The guide was focused on the web standards used in HTML5 game development and their backgrounds, the programs that are needed in game development, and (2D) game development itself.

Alongside with the thesis, a simple HTML5 browser game The Jumper was designed and implemented. The game was implemented on basis of the information gained from the research materials and with the aid of the Quintus game engine. The game is a platformer game, in which the player collects points and tries to survive through the levels without dying.

---

Key words: HTML5, game development, game, 2D

## SISÄLLYS

1	JOHDANTO.....	7
2	HTML, CSS JA JavaScript .....	8
2.1	HTML .....	8
2.2	CSS .....	9
2.3	JavaScript.....	9
3	HTML5 JA CSS3 .....	10
3.1	HTML5 .....	10
3.1.1	Canvas .....	10
3.1.2	WebGL.....	10
3.1.3	Paikallinen varastointi (Local Storage).....	11
3.1.4	WebSocket .....	12
3.1.5	Audio.....	12
3.2	CSS3 .....	12
3.2.1	CSS3 Transition .....	13
3.2.2	CSS3 Transform.....	13
3.2.3	CSS3 Animation .....	13
4	HTML5-PELIKEHITTÄJÄN OHJELMAT .....	14
4.1	HTML5-yhteensopiva internetselain .....	14
4.1.1	Google Chrome .....	15
4.1.2	Mozilla Firefox.....	15
4.2	Tekstieditori .....	15
4.2.1	Komodo Edit .....	15
4.2.2	Notepad++.....	16
4.3	Kuvankäsittely- ja vektoriohjelmat.....	16
4.3.1	Adobe Photoshop .....	16
4.3.2	GIMP.....	18
4.3.3	Adobe Illustrator .....	19
4.3.4	Inkscape.....	20
4.4	Audioeditori: Audacity .....	20
5	HTML5-PELIKEHITTÄMINEN .....	22
5.1	Quintus-pelimoottori.....	22
5.2	Pelielementit .....	23
5.2.1	Sprite .....	23
5.2.2	Ääni .....	24
5.2.3	Törmäysten havaitseminen.....	25
5.3	Pelin rakentaminen .....	25

6	HTML5-PELI – THE JUMPER.....	27
6.1	Tausta ja idea .....	27
6.2	Testaaminen .....	28
7	POHDINTA.....	29
	LÄHTEET.....	30

**LYHENTEET JA TERMIT**

HTML	HyperText Markup Language
CSS	Cascading Style Sheet
XML	Extensible Markup Language
API	Application Programming Interface
DOM	Document Object Model
IDE	Integrated Development Environment

## 1 JOHDANTO

Opinnäytetyön aiheen valinta oli minulle vaikeaa, sillä en oikein tiennyt mikä olisi sellainen aihe, josta olisi minulle hyötyä tulevaisuuden kannalta. Pitkän päähkäilyn jälkeen päätin, että halusin aiheen liittyvän jollain tapaa pelikehittämiseen. Pelikehittämistä on niin monenlaista, että seuraava pulma olikin keksiä minkä alustan pelikehittäminen olisi miellyttävä valinta. Päädyin lopulta HTML5-pelikehittämiseen, sillä HTML5-pelit toimivat alustasta riippumatta lähes millä tahansa selaimella.

Opinnäytetyön tavoitteena on tutkia, miten HTML5 soveltuu pelikehitykseen sekä selvittää uusien verkkostandardien merkitys, käyttömahdollisuudet ja hyödyt pelikehityksen näkökulmasta. Opinnäytetyön tarkoituksena on tuottaa siihen liittyvää (itse)opiskelumateriaalia. Oppaan on tarkoitus auttaa muita pelikehityksestä kiinnostuneita opiskelijoita ymmärtämään HTML5:n hyötyjä. Oppaassa keskitytään HTML5-pelikehityksessä käytettäviin verkkostandardeihin ja niiden taustoihin ja pelikehityksessä tarvittaviin ohjelmistoihin sekä (2D) pelikehittämisen saloihin.

Opinnäytetyön ohessa suunnitellaan ja toteutetaan yksinkertainen HTML5-selainpeli; peli toteutetaan aineistosta opituilla asioilla. Peli on tasoloikkapeli, jossa pelin sankarihahmo kerää pisteitä ja yrittää selvitä pelin 3 tasoa läpi kuolematta. Hahmo voi tuhota vihollisia hyppäämällä näiden päälle.

Pohdinta-kappaleessa käyn lävitse mitä opinnäytetyön aikana opin, millaisia ongelmia tuli vastaan ja mitä olisin voinut tehdä eri tavalla.

## 2 HTML, CSS JA JavaScript

### 2.1 HTML

HTML on tietokonekieli, joka on kehitetty verkkosivujen luomiseen. Näitä verkkosivuja voi katsella kuka tahansa, jolla on toimiva yhteys internettiin. HTML-kieli on suhteellisen helppo oppia, perusteet voikin oppia jopa yhdellä istumalla. Kieli on jatkuvassa kehityksessä, jotta se saavuttaa kasvavan internetyleisön tarpeet ja vaatimukset. Kehitystä johtaa W3C-organisaatio, jonka vastuulle on jätetty kielen suunnittelu ja ylläpito.

HTML määritelmä tulee sanoista: *HyperText Markup Language*.

- *HyperText* on metodi, jota käytetään verkossa liikkumiseen – klikkaamalla erikoistekstiä (hyperlink), pääsee seuraavalle sivulle. Hyper vain tarkoittaa sitä, että se ei ole lineaarinen – toisin sanoen voi mennä minne vain internetissä klikkaamalla linkkejä – ei ole määritelty järjestystä, jossa asiat pitäisi tehdä.
- *Markup* tarkoittaa, mitä HTML tagit tekevät tekstile niiden sisällä. Ne merkkäavat sen tietyn tyyppisenä tekstinä (esimerkiksi **lihavoitu** teksti).
- *Language* – HTML on kieli, koska se sisältää koodisanoja ja syntaksin kuten mikä tahansa muukin kieli.

HTML koostuu sarjoista lyhyitä koodeja kirjoitettuna tekstitiedostoon – nämä ovat tageja. Teksti tallennetaan sitten HTML-tiedostona, jota voidaan katsella selaimella, kuten Internet Explorer tai Google Chrome. Selain lukee tiedoston ja kääntää tekstin näkyvään muotoon.

## 2.2 CSS

CSS (*Cascading Style Sheet*) on verkkosivujen esitystavan kuvailuun suunniteltu kieli. Tämä sisältää sivuston värit, ulkoasun ja fontit. Se mahdollistaa verkkosivujen esitystavan sopeuttamisen erilaisille laitteille, kuten suurille näytöille, pienille näytöille tai tulostimille. CSS on riippumaton HTML:stä ja sitä voidaan käyttää minkä tahansa XML-pohjaisen merkintäkielen kanssa. Kun HTML:ää pidetään erillään CSS:stä, tämä yksinkertaistaa sivujen ylläpidon, tyylien jakamisen muihin sivuihin ja sivujen räätälöinnin eri ympäristöihin.

CSS luotiin säästämään aikaa ja vaivaa, kun internetin suosio alkoi kasvamaan ja sivustojen ulkoasulla alkoi olemaan tärkeä rooli osana sivuston menestymistä. Ilman CSS:ää HTML koodaajat joutuivat kirjoittamaan uudelleen ja uudelleen samoja tageja samalle sivulle. Tämä johti suuriin HTML-tiedostoihin ja ajan hukkaan.

## 2.3 JavaScript

JavaScript on ohjelmointikieli, jolla on monia eri käyttötarkoituksia. Sen on kehittänyt Netscape Communications ja se on tullut käyttöön ensimmäisen kerran vuonna 1995. JavaScript on todella laajasti käytetty ohjelmointikieli, sillä jokainen yleisesti käytetty internetselain tukee sitä, näitä ovat mm. Internet Explorer, Mozilla Firefox, Google Chrome, Safari.

JavaScriptiä käytetään internetsivujen interaktiivisten ominaisuuksien luomiseen. Sitä voidaan käyttää valikoiden luomiseen, lomakkeiden validoimiseen, kuvien vaihtamiseen ja oikeastaan mihin tahansa, mitä voit ajatella internetsivujen tarvitsevan. Sitä käytetään myös palvelimien verkko-ohjelmoinnissa, pelikehityksessä, mobiili- ja työpöytäsovellusten luomisessa. Enimmäkseen JavaScriptiä kuitenkin käytetään verkkoympäristössä. Hyviä esimerkkejä JavaScriptin tehokkaasta käytöstä ovat Google Maps ja Google Gmail.

JavaScriptin kirjoittamisen aloittaminen on yksinkertaista, siihen tarvitsee oikeastaan vain tekstieditorin ja internetselaimen. HTML:n ja CSS:n tunteminen auttaa huomattavasti, jos haluaa käyttää JavaScriptiä internetsivuillaan.

## 3 HTML5 JA CSS3

### 3.1 HTML5

HTML5 on yksinkertaisesti HTML:n 5. inkarnaatio. HTML5:n edeltävää inkarnaatiota on käytetty vuodesta 1997. HTML5 on kokenut suuria parannuksia edeltävään HTML:ään verrattuna. Se on suunniteltu tuottamaan lähes kaiken mitä halutaan tehdä verkossa, ilman erinäisiä ohjelmia ja selainlaajennuksia, kuten Flash ja Silverlight. Se kykenee mm. animaatioon, sovelluksiin musiikista elokuvaan ja sillä pystyy luomaan monimutkaisia ohjelmia ja pelejä, jotka toimivat selaimella. HTML5 hallitsee jopa web-sovelluksien käytön ilman yhteyttä verkkoon.

HTML5:tä saa käyttää kuka tahansa ilmaiseksi. Se toimii loistavasti eri alustoilla, käytettiinpä sitten tablettia, älypuhelinta, tietokonetta tai Smart TV:tä. Pääasia on, että selain tukee HTML5:tä, minkä suurin osa selaimista tekee jo hyvin.

Seuraavaksi käydään läpi muutamia uusia HTML5:n ominaisuuksia, joita voidaan käyttää, kun luodaan HTML5-pelejä.

#### 3.1.1 Canvas

Canvas on HTML5 elementti, joka mahdollistaa yksinkertaisten muotojen piirtämisen ja bittikarttojen manipulaation. Canvas-elementtiä voisi kuvailla dynaamisena kuva (<img>) -tagina. Canvasin sisään voidaan ladata kuvia, piirtää muotoja sen sisään ja olla vuorovaikutuksessa sen kanssa Javasriptia käyttäen. Canvas on oleellinen osa HTML5-pelikehittämistä. (Mazkan 2011, 9.)

#### 3.1.2 WebGL

WebGL laajentaa Canvas-elementtiä tarjoamalla joukon 3D-grafiikka ohjelmointirajapintoja (API) internetselaimessa. WebGL mahdollistaa 3D-renderoinnin

3D HTML5-peleille ja muille sovelluksille. Tästä hyvä esimerkki on ZygoteBody 3D-anatomia työkalu (ks. Kuvio 1).



KUVIO 1. Kuvakaappaus WebGL:ää käyttävästä ZygoteBody 3D-anatomia työkalusta (zygotebody.com 2015.)

Ikävä kyllä, kaikki selaimet eivät vielä tue WebGL:ää. Pöytätietokoneen tai kannettavan tietokoneen käyttäjällä ei ole ongelmia, mutta jos käyttää tabletin tai älypuhelimien selainta, ongelmia mahdollisesti tulee eteen.

### 3.1.3 Paikallinen varastointi (Local Storage)

HTML5 tarjoaa erinomaisen tietojen varastointiratkaisun internetiselaimille. Ennen HTML5:tä tiedot tallennettiin evästeisiin, joihin selain pystyi tallentamaan pienen määrän tietoa, yleensä tekstiä. Paikallinen varastointi on turvallisempi ja suuria määriä tietoa voidaan tallentaa paikallisesti selaimen ilman vaikutusta sivujen toimivuuteen.

Paikallinen varastointi varastoi tietoa jatkuvasti. Tieto on tallella senkin jälkeen, kun selain suljetaan. Paikallisen varastoinnin ansiosta on mahdollista tallentaa pelitietoja

paikallisesti internetselaimeen, kuten pelin edistyksen, ansaitut saavutukset ja pisteet (Seidelin 2011, 13).

### 3.1.4 WebSocket

WebSocket on osa HTML5:ttä. Se yhdistää verkkosivun palvelimelle. Se tarjoaa tapahtumapohjaisen yhteyden selaimen ja palvelimen välille. Tämä tarkoittaa sitä, että päätteen ei tarvitse kysellä palvelimelta uutta tietoa vähän väliä. Palvelin työntää päivityksiä selaimelle aina, kun tietoja päivitetään. HTML-pelikehittämisen kannalta tämä tarkoittaa sitä, että pelaajat voivat olla vuorovaikutuksessa toisiinsa lähes reaaliajassa. Kun pelaaja tekee jotain ja lähettää tiedon palvelimelle, palvelin lähettää tapahtuman muille yhdistetyille selaimille. Toisin sanoen tämä mahdollistaa HTML5 moninpelien luomisen. (Mazkan 2011, 10.)

### 3.1.5 Audio

Taustamusiikki ja ääniefektit ovat tärkeä osa pelisuunnittelua. HTML5 sisältää musiikkia ja ääntä tukevan audio-tagin. Tämän ominaisuuden ansiosta ei tarvita selainlaajennusta (Flash) toistamaan musiikkia tai ääniä.

## 3.2 CSS3

CSS3 on CSS:n viimeisin standardi. CSS3 on täysin yhteensopiva aikaisempien CSS-versioiden kanssa. Se on myös erittäin hyödyllinen HTML5-pelejä luodessa. Uudet CSS3:n ominaisuudet mahdollistavat DOM:n (*Document Object Model*) animoinnin eri tavoilla. Tämä mahdollistaa monipuolisten DOM-pohjaisten HTML5-selainpelien luomisen. Näitä ominaisuuksia ovat mm. transition, transform ja animation.

### 3.2.1 CSS3 Transition

Normaalisti tyyli muuttuu heti, kun lisätään uusi tyyli elementtiin. CSS3 transition lisää tweeningin, kohde-elementin tyyli muutoksien aikana (Mazkan 2011, 12). Tweening on prosessi, joka generoi välikuvan kahden kuvan väliin. Tämä antaa vaikutelman, että ensimmäinen kuva kehittyy tasaisesti toiseksi kuvaksi.

Esimerkiksi voidaan haluta sinisen pallon muuttuvan punaiseksi, kun hiiri vie sen päälle. Ilman transitionia väri vaihtuu välittömästi, mutta transition mahdollistaa värin vaihtamisen tasaisesti punaisesta (purppuran kautta) siniseksi.

### 3.2.2 CSS3 Transform

CSS3 transform mahdollistaa elementtien liikuttamisen, skaalaamisen, pyörittämisen, kääntämisen ja venyttämisen. Se siis on efekti, joka antaa elementin vaihtaa muotoaan, kokoaan ja sijaintiaan. Transform on jaettu kahteen alakategoriaan: 2D ja 3D.

### 3.2.3 CSS3 Animation

CSS3 Transition on eräänlainen animaation muoto. CSS3 Animation on transitionista seuraava askel. Sen avulla on mahdollista määritellä animaation avainkuvat (key frame). Jokainen avainkuva sisältää joukon ominaisuuksia, joiden pitäisi muuttua sillä hetkellä. Se on kuin joukko CSS3 transitioneita asetettuna toinen toisensa jälkeen kohde-elementille.

W3schools.comin mukaan CSS3 Animaatiot voivat korvata Flashilla ja Javascriptillä luodut animaatiot olemassa olevilla verkkosivuilla (W3Schools 2015).

## 4 HTML5-PELIKEHITTÄJÄN OHJELMAT

Kun on aloittamassa pelikehittämistä HTML5:n parissa, tarvitaan moderni tietokone ja muutamia ohjelmia, joista monia jo luultavasti löytyy koneelta. Tarvitaan HTML5-yhteensopiva selain, hyvä tekstieditori, kuvankäsittely- ja vektoriohjelma sekä mahdollisesti audioeditori.

Seuraavassa on muutamia esimerkkejä ilmaisista ja myös maksullisista vaihtoehdoista.

### 4.1 HTML5-yhteensopiva internetiselain

Internet-selaimella on tärkeä rooli HTML5-pelikehityksessä. HTML5-pelit pyörivät selaimella, eli ilman selainta pelin pelaaminen on mahdotonta. On myös tärkeää, että selain tukee HTML5:sta niin hyvin kuin mahdollista. Kaikki selaimet eivät tue HTML5:n kaikkia ominaisuuksia. Html5test.com-sivuilla on mahdollista testata käyttämänsä selaimensa HTML5-toimivuutta (ks. Kuvio 2).

	Chrome	Firefox	Internet Explorer	Opera	Safari
Upcoming			TP 343		
Current	39 501	35 449	11 336	26 497	8.0 396
Older	38 499	34 449	10 297	24 485	7.0 352
	37 494	33 444	9 113	23 462	6.0 326
	35 482	32 444	8 33	22 473	5.1 250
	34 475	31 446		20 461	
	26 458	30 434		18 453	
	10 269	28 416		12.10 338	

KUVIO 2. Kuvakaappaus HTML5 tukevien selaimien pisteytyksestä. Täysi pistemäärä on 555. (html5test.com 2015.)

### **4.1.1 Google Chrome**

Google Chrome on Googlen kehittämä ilmainen internetselain, joka tuli ensimmäisen kerran yleisön käyttöön vuonna 2008. Chrome on nopea, yksinkertainen ja turvallinen internetselain, rakennettu modernia internetiä varten (Google 2015). Itse suosittelen Google Chromea käytettäväksi HTML5-pelikehittämisessä. Testien mukaan Chrome tukee tällä hetkellä parhaiten HTML5:sta (ks. Kuvio 2).

### **4.1.2 Mozilla Firefox**

Mozilla Firefox tai paremmin tunnetusti vain Firefox on Mozillan kehittämä ilmainen internetselain. Firefox tuli julkiseen käyttöön vuonna 2002. Html5test.comin mukaan Firefox on listalla tällä hetkellä 3. parhaiten HTML5 tukeva selain (ks. Kuvio 2).

## **4.2 Tekstieditori**

Tekstieditoria käytetään HTML5:n, JavaScriptin ja CSS:n kirjoittamiseen. Tarvitaan siis tekstieditori, joka mielellään tukee kaikkia näitä.

### **4.2.1 Komodo Edit**

Komodo Edit on ilmainen tekstieditori, joka on erityisesti tarkoitettu ohjelmointikielille, mukaan lukien Python, PHP, Ruby, Perl, Tcl, NodeJS, HTML, CSS ja JavaScript. Se on kehittänyt ActiveState ja se on julkaistu ensimmäisen kerran vuonna 2007. Se toimii Windows-, Mac OS X- ja Linux-käyttöjärjestelmillä. Komodo Editistä on myös maksullinen versio Komodo IDE, joka sisältää enemmän ohjelmointiympäristötoimintoja (IDE), mukaanlukien virheiden jäljitystoiminto (debugging). Itse suosittelen Komodo Editiä aloittelevalle HTML5-pelikehittäjälle. Se on monipuolinen ja selkeä käyttää.

## 4.2.2 Notepad++

Notepad++ on ilmainen lähdekoodi- ja tekstieditori. Sen on kehittänyt Don Ho ja se on julkaistu vuonna 2003. Notepad++ tukee useita eri ohjelmointikieliä, kuten PHP, Java, HTML, CSS ja JavaScript. Se toimii vain Windows-käyttöjärjestelmällä. Notepad++ on yksinkertainen ja helppokäyttöinen, mutta ei niin monipuolinen kuin Komodo Edit.

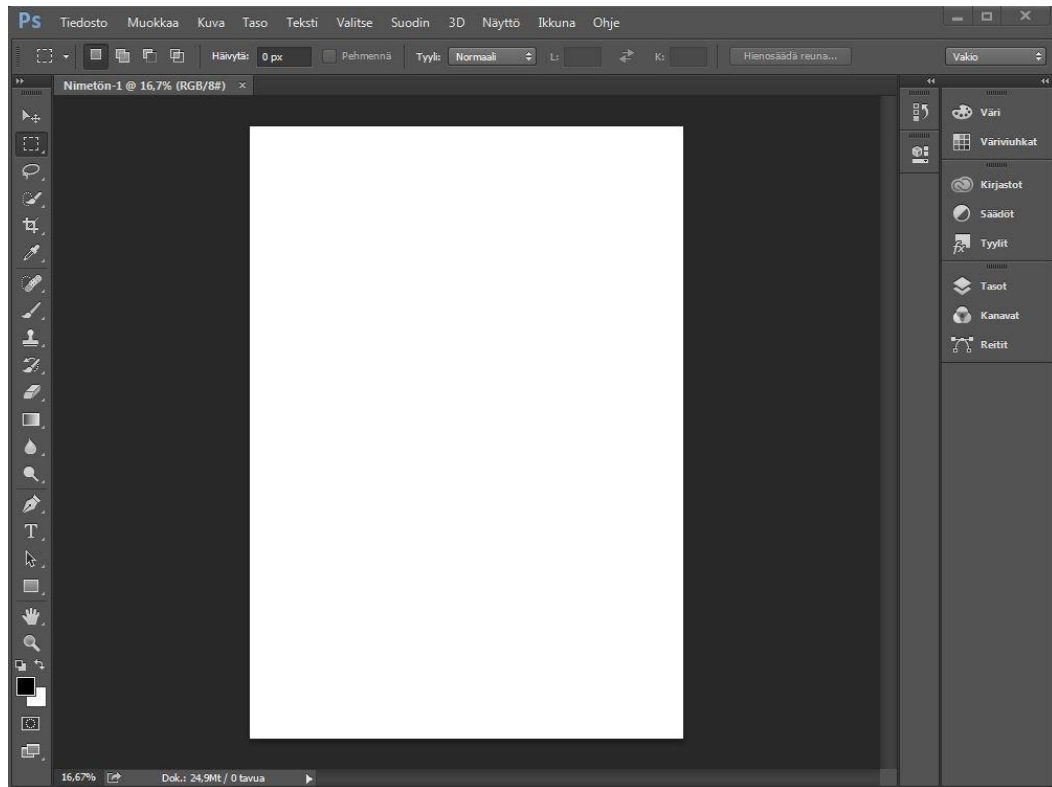
## 4.3 Kuvankäsittely- ja vektoriohjelmat

Suurin osa peleistä käyttää grafiikoita jossain muodossa. On mahdollista löytää kaikki tarvittavat grafiikat internetistä, mutta yleensä graafikoita pitää muokata tai pitää luoda alusta alkaen täysin omia grafiikoita. Tähän tarvitaan hyvä kuvankäsittely- ja vektoriohjelma.

### 4.3.1 Adobe Photoshop

Adobe Photoshop on maksullinen bittikarttagrafiikkaeditori eli kuvankäsittelyohjelma, jonka on kehittänyt Adobe Systems. Photoshop on kuvankäsittelyohjelmien parhaimmistoa, ja onkin tällä hetkellä markkinoiden suosituin kuvankäsittelyohjelma. Ohjelmaa on päivitetty vuosien varrella useaan otteeseen ja siihen on lisätty ominaisuuksia, jotka ovat tehneet siitä yleisen välineen graafisille suunnittelijoille. Photoshop toimii Windows- ja Mac OS X-käyttöjärjestelmillä.

Photoshop on bittikarttagrafiikkaeditori eli toisin sanoen sillä voi luoda ja muokata piirroksia tai valokuvia manipuloimalla pikseleitä. Tämä on verrattavissa vektorigrafiikkaeditoreihin, joita vuorostaan käytetään linjojen tai viivojen ja muotojen manipuloimiseen. Photoshopin käyttöliittymä on monipuolinen ja helposti muokattavissa (ks. Kuvio3).



KUVIO 3. Photoshop käyttöliittymä (Adobe Photoshop CC 2015, kuvakaappaus.)

Photoshopilla on oma tiedostomuoto PSD (Photoshop Document). PSD-tiedostot sisältävät kuvan, joka tukee useimpia Photoshopin kuvanmuokkaus ominaisuuksia. Näihin ominaisuuksiin kuuluu mm. layerit eli tasot, joihin kuvan eri osat ja tehosteet voidaan lajitella, jotta näiden käsittely on helpompaa. Näitä tasoja voi halutessaan piilottaa ja lukita paikoilleen. Näin alkuperäinen kuva säilyy ja näin voidaan tehdä useita muokkauksia uusiin tasoihin. Photoshopilla on tietenkin mahdollista tallentaa useisiin eri tiedostomuotoihin, kuten JPEG-, PDF- ja GIF-tiedostomuotoihin, mutta tämän jälkeen kuvaan tehdyt muokkaukset tallennetaan ja layerit poistetaan.

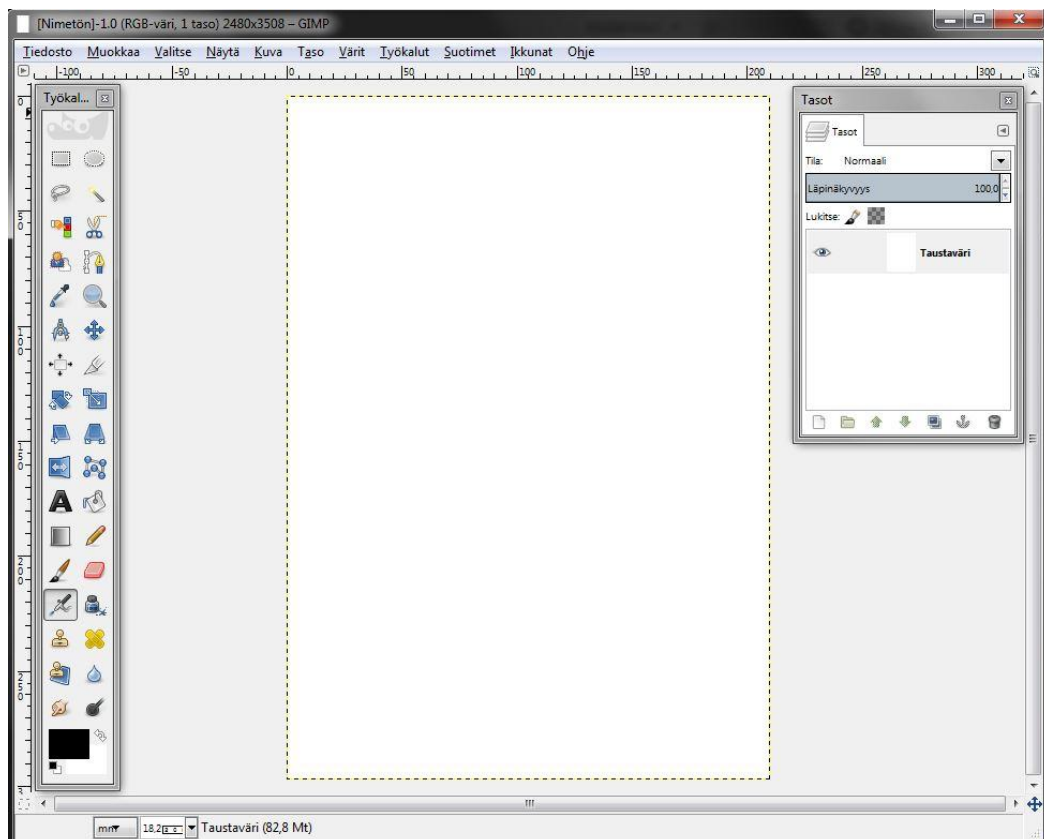
Photoshopin käyttämisen opettelu voi viedä oman aikansa, mutta onneksi internet on täynnä oppaita, joita käyttämällä aloittelijastakin on mahdollista tulla taitava kuvankäsittelijä lyhyessäkin ajassa.

### 4.3.2 GIMP

GIMP (GNU Image Manipulation Program) on ilmainen kuvankäsittelyohjelma, jonka on kehittänyt The GIMP Development Team. GIMP on toiminnallisuudeltaan hyvin samankaltainen Adobe Photoshopin kanssa. Mikäli ei ole mahdollisuutta käyttää Photoshopia tai ei ole varaa siihen, GIMP on todella hyvä vaihtoehto. GIMP on alustariippumaton eli se toimii Windows-, Mac OS X- ja Linux-käyttöjärjestelmillä.

GIMP:llä on myös oma tiedostomuotonsa XCF. Se on hyvin samankaltainen kuin Photoshopin PSD. Se siis sisältää kaiken tiedon, jota GIMP tukee. XCF on nimetty eXperimental Computing Facilityn mukaan, missä GIMP sai alkunsa.

GIMP on käyttöliittymältään hyvin yksinkertainen (Kuvio 4), mutta sitä on mahdollista muokata lisäämällä valikosta uusia työkalu- tai ominaisuikkunoita, kuten tasot tai toimintohistoria.

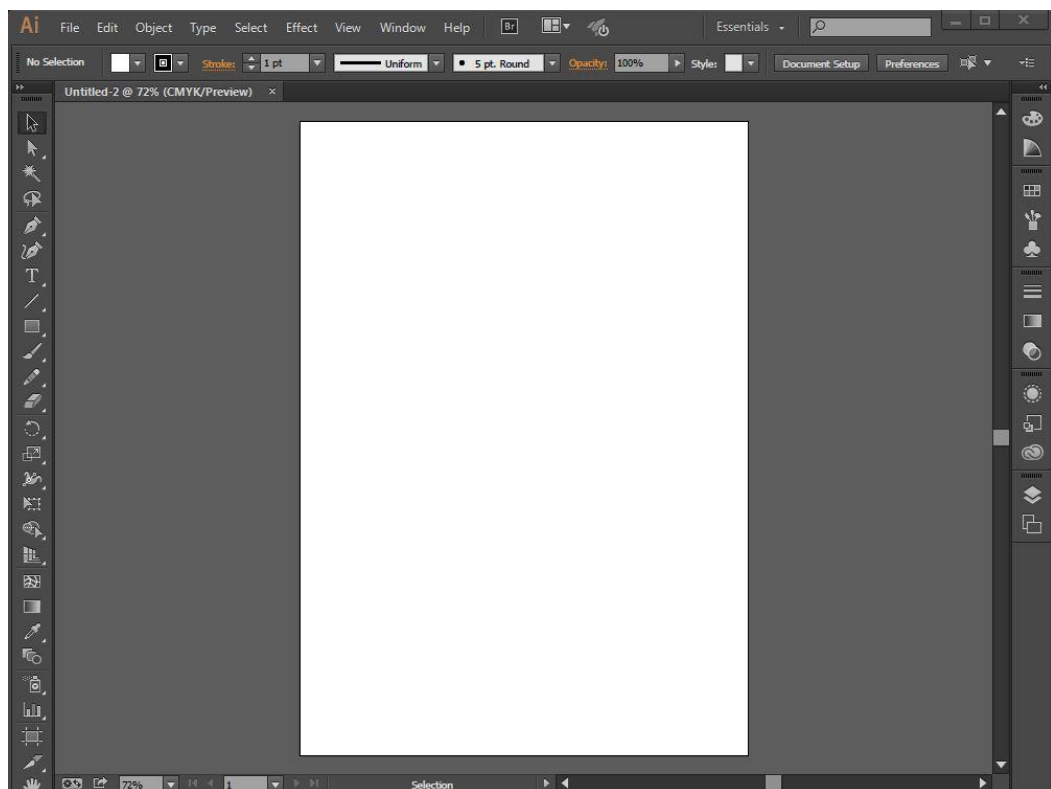


KUVIO 4. GIMP käyttöliittymä (GIMP 2 2015, kuvakaappaus.)

### 4.3.3 Adobe Illustrator

Adobe Illustrator on Adobe Systemsin kehittämä maksullinen vektorigrafiikkaohjelma. Se on ollut graafisen suunnittelun standardi-ohjelma vuosien ajan, printtituotannosta digitaaliseen. Illustratorilla luodut vektorigrafiikat ovat helposti skaalattavissa ilman että kuvien laatu kärsii, mikä tekee siitä erityisen hyödyllisen mm. internetgrafiikoiden luomiseen. Illustrator toimii Mac OS X- ja Windows-käyttöjärjestelmillä.

Illustrator voi vaikuttaa todella monimutkaiselta käyttää ja työkaluiltaan laajalta, toisin kuin sen kuvankäsittelyyn tarkoitettu sisarusohjelma Adobe Photoshop. Illustratorin käyttäminen voi olla hieman vaikeaa aluksi, mutta pienen harjoittelun jälkeen sen käyttäjäystävällisiä ominaisuuksia alkaa arvostamaan. Käyttämistä helpottaa, jos tuntee ennestään Photoshopin, koska Illustratorin käyttöliittymä (Kuvio 5) muistuttaa kovasti Photoshopin käyttöliittymää (ks. Kuvio 3).

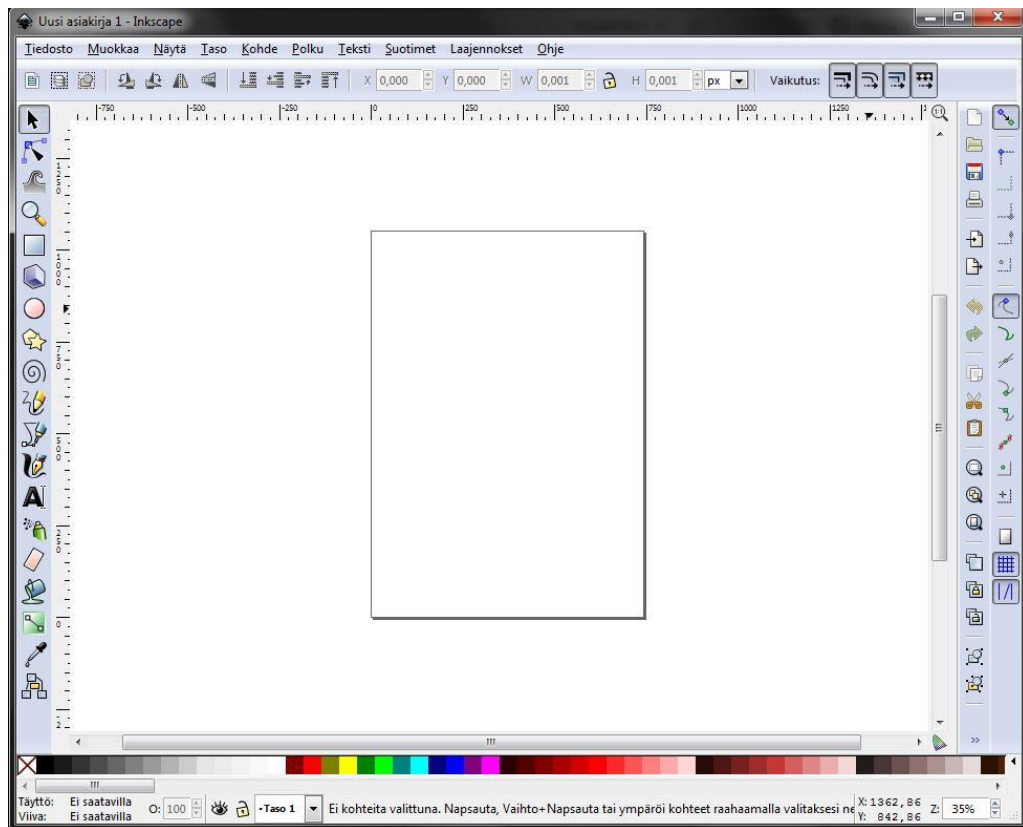


KUVIO 5. Illustrator käyttöliittymä (Adobe Illustrator CC 2015, kuvakaappaus.)

### 4.3.4 Inkscape

Inkscape on ilmainen vektorigrafiikkaohjelma, jonka on kehittänyt The Inkscape Team. Se on Illustratorin ilmainen vastine, aivan kuten GIMP on Photoshopin ilmainen vastine. Inkscape on myös alustariippumaton eli se toimii koneella kuin koneella, käyttöjärjestelmästä riippumatta.

Inkscapen käyttöliittymä on suhteellisen yksinkertainen (Kuvio 6), mutta sen opettelu voi viedä jonkin verran aikaa. Sen muokkaaminen ei ole mahdollista, toisin kuin Adoben ohjelmistoissa.



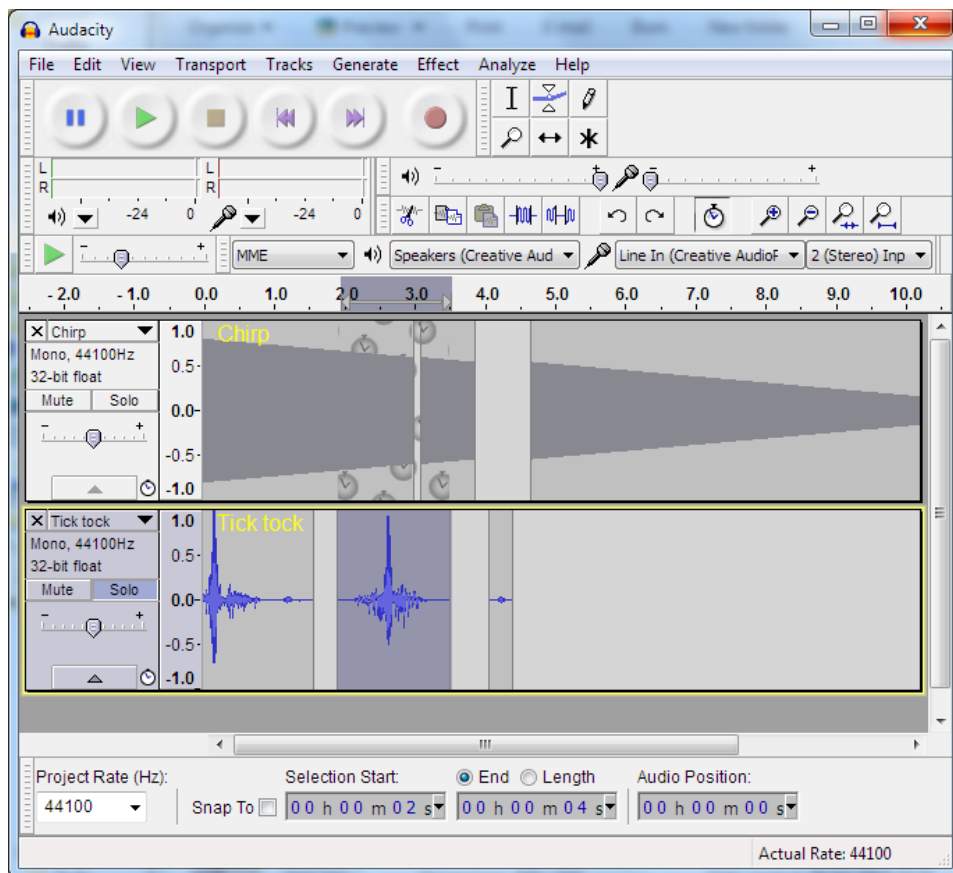
KUVIO 6. Inkscape käyttöliittymä (Inkscape 2015, kuvakaappaus.)

### 4.4 Audioeditori: Audacity

Audioeditori ei ole aivan välttämätön osa HTML5-pelikehitystä, sillä internet on täynnä valmiita ilmaisia peliääniä ja musiikkia, jotka ovat vapaasti käytettävissä. Audioeditori tulee tarpeeseen, jos haluaa käyttää omia ääniä tai musiikkia tai muokata jo olemassa olevia.

Audacity on ilmainen avoimen lähdekoodin audioeditori ja nauhotin. Se on saatavilla Windowsille, Mac OS X:lle, Linuxille. Käyttöliittymä (Kuvio 7) on käännetty 51 kielelle, mukaan lukien suomen kieli.

Audacity on erittäin monipuolinen ohjelma: sillä voi nauhottaa live-ääntä, muuttaa kassetinauhat ja levytallenteet digitaaliseen muotoon ja CD-levyiksi, editoida ääntä ja paljon muuta.



KUVIO 7. Kuvakaappaus Audacityn Windows käyttöliittymästä. (audacity.sourceforge.net 2015.)

## 5 HTML5-PELIKEHITTÄMINEN

### 5.1 Quintus-pelimoottori

Pelimoottori on ohjelmistokehys, joka on suunniteltu pelien luomiseen ja kehitykseen. Se on tärkeä osa pelikehittämistä, sillä se laittaa pelin liikkeelle. Pelimoottori tiivistää peleihin liittyvät yleiset tehtävät yhdeksi kirjastoksi, jotta pelikehittäjät voivat keksittyä pelin yksityiskohtiin, jotka erottavat pelin muiden joukosta.

Pelimoottorit tarjoavat uudelleen käytettäviä komponentteja, joita voidaan manipuloida tuomaan pelin eloon. Elementtien lataaminen, näyttäminen ja animointi, objektien välisen törmäilyn havaitseminen, fysiikka, tiedon syöttäminen, graafinen käyttöliittymä ja jopa osat pelin tekoälystä voivat kaikki olla komponentteja, jotka muodostavat pelimoottorin (Harris 2013, 114).

HTML5-pelimoottoreita on monenlaisia; maksullisia ja maksuttomia. Kaikki HTML5-pelimoottorit eivät tue HTML5:n kaikkia ominaisuuksia, kuten WebGL:ää tai Audiota. Osa pelimoottoreista saattaa sisältää karttaeditorin, joka on erittäin hyödyllinen pelien taso suunnittelun ja toteutuksen kannalta. Karttaeditorin voi myös hankkia erikseen, mutta tässä tapauksessa pelimoottorin täytyy pystyä tukemaan karttaeditorin tiedostomuotoa, joka on usein karttaeditori-kohtainen. Pelimoottori kannattaa siis valita tarkkaan, jotta löytää omaan tarkoitukseensa sopivan.

Quintus on ilmainen ja helppokäyttöinen pelimoottori, joka on kehitetty nimenomaan HTML5-pelejä mielessä pitäen. Se on siis erittäin hyvä pelimoottorivalinta, kun on aloittelemassa HTML5-pelikehittämistä. Sen avulla voi helposti kehittää 2D mobiili- ja työpöytäpelejä. Internetistä löytyy useita tutoriaaleja ja oppaita, kuinka Quintus-pelimoottoria voidaan hyödyntää.

Quintus ei sisällä karttaeditoria, joten sen pariin kannattaa hankkia sellainen, jos haluaa helpottaa pelin tasojen luomista. Karttaeditori kannattaa valita toteutettavan pelin genren mukaan, sillä kaikki editorit eivät sovellu kaiken tyyppisten pelien tasojen luomiseen.

Quintus-moottori on HTML5-pelimoottori, joka on suunniteltu modulaariseksi ja kevyeksi. Se sisältää suppean JavaScript-mäisen syntaksin. Quintuksessa ei ole tarkoituksena tuoda HTML5-JavaScriptmoottoriin standardia oliopohjaista pelimoottorin rakennetta, vaan siinä hyödynnetään joiltain osin jQueryn piirteitä. Se tarjoaa myös liitännäiset, tapahtumat ja valitsinsyntaksin. ”Vain yksi perijä”-mallin sijaan Quintus tarjoaa perinteisen perinnän lisäksi joustavan komponenttimallin, jotta olisi helpompaa käyttää uudelleen toiminnallisuutta ja jakaa sitä pelien ja objektien kanssa. (Quintus 2015.)

## 5.2 Pelielementit

Peleissä on paljon esineitä ja asioita, jotka liikkuvat ympäriinsä törmäillen toisiinsa ja pitäen jonkinlaista ääntä (Harris 2013, 133). Nämä ovat elementtejä, joita ilman peliä ei voi tehdä. Nämä elementit ovat tärkeä osa mitä tahansa pelikehittämistä, oli sitten kyseessä HTML5-pelikehittäminen tai konsolipelikehittäminen.

### 5.2.1 Sprite

Sprite on yksi graafinen kuva (Kuvio 8), joka on integroitu osaksi isompaa pelin kohtausta, jotta se näyttäisi olevan osa sitä. Spritet ovat suosittu tapa toteuttaa HTML5-pelien grafiikka, sillä ne ovat kevyitä käyttää.



KUVIO 8. 2D-pelielementti eli sprite (Quintus 2015.)

Spritet ovat suosittu tapa luoda suuria, monipuolisia kohtauksia, koska niitä on mahdollista manipuloida erillisenä muusta kohtauksesta. Tämä mahdollistaa paremman kontrollin siihen, kuinka kohtaus (scene) renderoidaan, sekä siihen kuinka pelaajat voivat olla vuorovaikutuksessa kohtauksen kanssa.

On normaalia, että pelissä on käytössä useita, jopa satoja spritejä. Näiden yksittäisten kuvien lataaminen kuluttaisi paljon koneen muistia ja laskentatehoa. Spritejen hallinnoin helpottamiseksi ja niin monien spritejen käyttämisen välttämiseksi, monet pelit käyttävät sprite-arkkeja (Kuvio9).



KUVIO 9. Sprite-arkki eli spritesheet (Quintus 2015).

Sprite-arkkeja käytetään nopeuttamaan kuvien näytölle tuomisen prosessia. On paljon nopeampaa noutaa yksi kuva ja näyttää vain osa tuosta kuvasta, verrattuna siihen että noutaisi montaa kuvaa ja näyttäisi niitä.

Jos halutaan tehdä hieman interaktiivisempi sprite, kannattaa lisätä siihen animaatiota. Tämä on helppoa toteuttaa sprite-arkin -animaatiolla. Sprite-arkin animaatio on toiminnaltaan melko yksinkertainen: otetaan sprite-arkki ja vaihdetaan nopeassa tahdissa renderoitua spriteä, jotta saadaan aikaan illuusio liikkeestä.

Kun spriteen lisätään animaatiota, ääniefekti ja törmäyksen havaitseminen, saadaan aikaan toimiva ja interaktiivinen pelielementti.

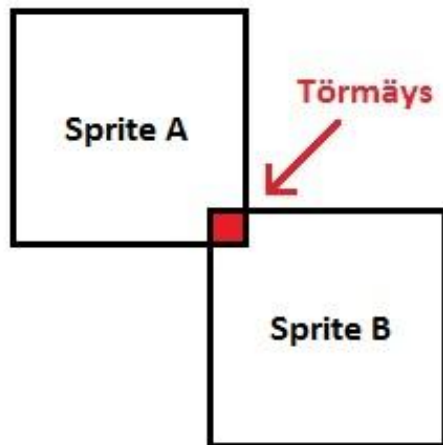
## 5.2.2 Ääni

Ääni on myös tärkeä pelielementti. Ääniefektit ja taustamusiikki lisäävät peliin paljon, kuten tunnelmaa, realistisuutta ja mielenkiintoa. Näiden lisäksi ne yleensä myös antavat pelaajalle tärkeää palautetta pelin tapahtumista.

Ääniefektit ovat olleet aiemmin selainpelien suurimpia heikkouksia. Internetselaimilla oli hankaluuksia niiden ääniominaisuuksiensa kanssa. HTML5 onneksi ratkaisee tämän ongelman Audio-tag -ominaisuutensa ansiosta, ainakin jollain tasolla (Harris 2013, 141).

### 5.2.3 Törmäysten havaitseminen

Pelien mielenkiintoisimmat asiat tapahtuvat, kun spritet törmäävät toisiinsa (ks. Kuvio 10). Pelimoottoreilla on normaalisti jokin väline, jolla testataan ovatko spritet päällekkäin. Tätä kutsutaan törmäyksen havaitsemiseksi ja se on mahdollista toteuttaa monella eri tavalla (Harris 2013, 144).



KUVIO 10. Törmäyksen havaitseminen.

Törmäyksen sattuessa sen yhteyteen yleensä lisätään jokin toiminto. Tällainen toiminto on esimerkiksi toisen spriten tuhoutuminen.

## 5.3 Pelin rakentaminen

Kun hallitsee spritejen luomisen, on helppo jatkaa varsinaisen pelin rakentamiseen. Jotta pelissä olisi jotain järkeä, on tärkeää, että peli sisältää seuraavat asiat:

- **Juoni:** Juonen ei tarvitse olla erityisen monimutkainen, mutta siinä on oltava jonkinlainen teema tai asetelma. Hyvä esimerkki tästä on PAC-MAN.
- **Päämäärä:** Pelaajilla tulee olla päämäärä, jonka he yrittävät saavuttaa. Jos pelissä on suuri päämäärä, se vaatii pienempiä päämääriä matkalle. Pelaajalla täytyy olla aina jokin tavoite, oli se sitten vihollisen kaappaaminen, tason loppuun pääseminen tai kisan ensimmäisen sija.

- Este: Jotta pelistä saataisiin mielenkiintoinen, pelaajan tiellä täytyy olla jokin este tai esteitä matkalla. Se voi olla jokin esine, johon pelaaja yrittää olla osumatta, vihollinen joka ampuu pelaajaa tai yksinkertaisesti aika.

Näiden asioiden lisäksi peliin tulee lisätä muutamia mekanismeja, jotka hyvä pelimoottori yleensä tarjoaa:

- Törmäyksen havaitseminen: Peleissä on usein, ellei lähes aina, esineitä jotka törmäilevät toisiinsa. Tähän tarvitaan mekanismi, joka hallitsee törmäyksiä.
- Ääniefektit: Ääniefektit lisäävät tunnelmaa peliin.
- Teksti-palautte: On tärkeää näyttää pelaajalle, mitä pelissä tapahtuu tai kuinka peli etenee. Tämä yleensä näkyy pelissä, kun esitetään käyttöliittymässä pistetilanne ja jäljellä olevien elämien määrä.
- Uudelleen peluu -painike: Jossain vaiheessa peliä peli päättyy. Oli se sitten pelin läpi pääsyn tai kuoleman jälkeen. Tällöin tarvitaan mekanismi, joka käynnistää pelin uudelleen alusta.

Kun nämä asiat ovat hallinnassa, voidaan aloittaa varsinainen pelin lähdekoodin rakentaminen. Kun on kyse HTML5-pelistä, henkilökohtaisesti koen seuraavanlaisen prosessin toimivan:

1. Haetaan pelimoottorikirjasto ja asetetaan se käyttöön.
2. Luodaan Spritet.
  - a. Luodaan pelaaja-hahmo toimintoinen.
  - b. Luodaan vihollishahmot/esteet toimintoinen.
3. Valmistellaan kohtausta (scene), johon asetetaan pelitaso ja otetaan se käyttöön.
4. Lisätään tasoon kaikki hahmot (pelaaja ja viholliset).
5. Lisätään näiden välille törmäystarkistus ja siihen liittyvät toiminnot
6. Lisätään peliin käyttöliittymä.
7. Lisätään tasoon päämäärä.
8. Luodaan lisää tasoja, jos tarpeellista.
9. Lopuksi ladataan ja käynnistetään kohtausta

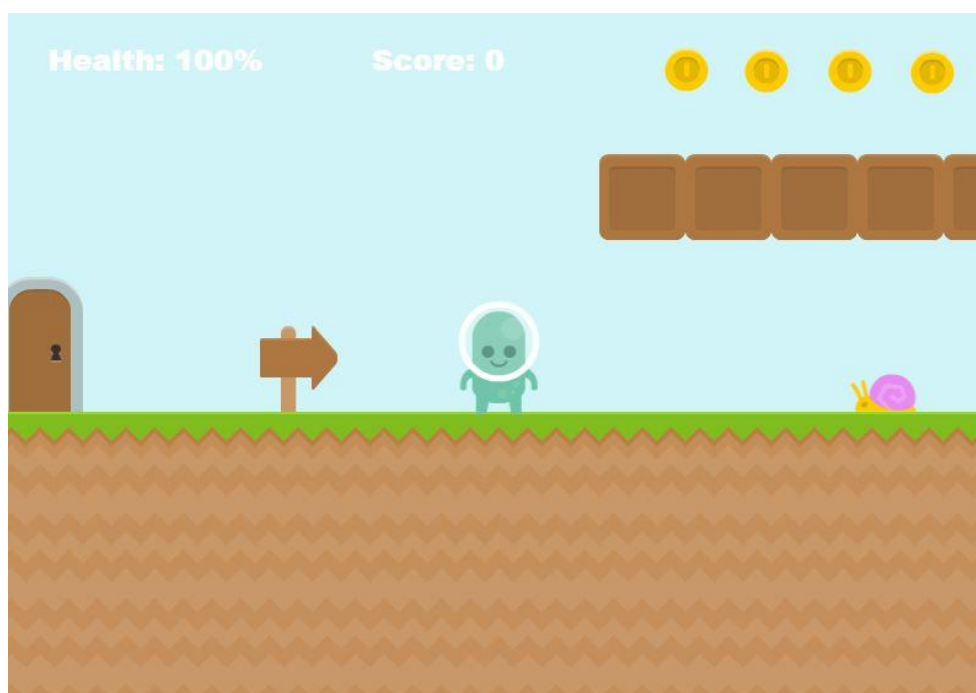
Jokaisella pelikehittäjällä on omat prosessinsa pelin luomiseen, mutta tämä on hyvä lähtökohta, mikäli on vasta aloittelemassa pelikehittämistä HTML5:n parissa. Kun alkaa kehittämään pelikehittäjänä, löytää itselle sopivimmat prosessit pelien luomiseen ja rakentamiseen.

## 6 HTML5-PELI – THE JUMPER

### 6.1 Tausta ja idea

Osana tätä opinnäytetyötä päätin tehdä testimielessä yksinkertaisen HTML5-pelin. Pelityypin valinta tuotti hieman vaikeuksia. Alun perin suunnittelin kehittäväni Zeldatyyllisen seikkailu-/roolipelin, mutta pidemmän pohdinnan jälkeen päädyin kuitenkin yksinkertaiseen tasoloikkapeliin (ks. Kuvio 11).

Pelissä on tavoitteena päästä pelin loppuun kuolematta ja kerätä matkalla mahdollisimman paljon pisteitä. Pelissä kerätään pisteitä keräämällä kolikoita sekä tuhoamalla vihollisia. Pelin sankarihahmo kestää 4 iskua viholliselta ennen kuolemista ja rotkoihin, veteen tai laavaan putoaminen tarkoittaa suoraa kuolemaa. Jos pelihahmo kärsii paljon vahinkoa vihollisilta, on mahdollista kerätä sydämiä, jotka palauttavat hahmolle yhden osuman verran lisää terveyttä (Health). Pelissä on toistaiseksi vain 3 tasoa ja muutenkin pelissä on vielä paljon kehittämisen varaa. Pelin vaikeustaso on vielä toistaiseksi todella alhainen.



KUVIO 11. Kuvankaappaus The Jumper -pelistä.

## 6.2 Testaaminen

Testaus suoritettiin useilla eri laitteilla: puhelimilla ja erityisesti tableteilla. Testilaitteita oli saatavilla rajoitetusti, mutta kuitenkin hyvin eri käyttöjärjestelmillä varustettuna. Näin saatiin hyvä kuva siitä kuinka peli toimii eri laitteistoilla. Alla olevasta taulukosta 1 löytyy testatut laitteet ja niiden käyttöjärjestelmät.

TAULUKKO 1. Testilaitteet.

Laitteen malli	Käyttöjärjestelmä
Apple iPad Air	iOS 8.1.3
Nokia Lumia 635	Windows Phone 8.1
Samsung Galaxy S5 Mini	Android 4.4.2
Samsung Galaxy Tab3 Lite	Android 4.2.2

Testilaitteita oli niin rajoitetusti, että testiin ei välttämättä kannata liikaa luottaa. Mahdollisia laitteita on olemassa tuhansia ja pelin toimivuutta ei voida taata joka ikiselle. Jos peli julkaistaisiin, testi tehtäisiin julkisesti, jolloin testattavien laitteiden määrä kasvaisi moninkertaiseksi ja saataisiin laajempi kuva pelin toimivuudesta.

Peli toimi pienien hienosäätöjen jälkeen lähes moitteettomasti kaikilla alustoilla, paitsi Windows Phone 8.1 -käyttöjärjestelmällä. Kyseisellä alustalla peli ei toiminut ollenkaan. Hieman pidemmän testaustuokion jälkeen tulin siihen tulokseen, että Quintus-pelimoottori ei jostain syystä toimi alustalla.

Yleisenä ongelmana oli, että pienemmillä laitteilla, kuten Samsung Galaxy S5 Mini ja Samsung Galaxy Tab3 Lite, peli ei skaalautunut täysin oikein. Tämä hieman hankaloitti pelaamista, mutta ei kuitenkaan tehnyt pelistä pelaamiskelvotonta. Quintus-pelimoottorissa on siis vielä hieman parantamisen varaa, jotta se toimisi kaikilla alustoilla moitteettomasti.

## 7 POHDINTA

HTML5:n soveltuvuudesta pelikehittämiseen voidaan olla montaa eri mieltä. Omieni kokemuksieni mukaan se on kuitenkin täysin toimiva tapa toteuttaa selainpelejä. Mikä parasta, pelit toimivat laitteella kuin laitteella, kunhan vain laitteen selain tukee HTML5:n ominaisuuksia. Internet on täynnä hienoja ja toimivia HTML5-pelejä, joita kokeilemalla voi ymmärtää HTML5:n mahdollisuudet.

HTML5 ja CCS3 ovat laajoja ja ne tulevat vielä kasvamaan ja kehittymään. Tällä hetkellä ne kuitenkin sisältävät jo monia uusia ominaisuuksia, joita voidaan hyödyntää pelikehityksessä ja jotka voivat mahdollisesti korvata nykyisiä standardeja kuten Flash.

Ongelmia ilmeni lähinnä Quintus-pelimoottorin toimivuuden kanssa testivaiheessa. Muun muassa peli ei tahtonut toimia ollenkaan iOS 8.1.3 -käyttöjärjestelmällä. Ongelmaksi selvisi, että pelimoottorin ominaisuuksissa oli pieni virhe, jonka onnistuin löytämään usean tunnin etsiskelyn jälkeen. HTML5-pelimoottoreita on niin monia, että olisin varmasti voinut löytää paremmankin toimivuudeltaan, mutta koska virhe löytyi niin myöhäisessä vaiheessa, pelimoottorin vaihtaminen olisi pitkittänyt projektin kestoja hurjasti.

Opinnäytetyön aikataulu pääsi venymään aivan liian paljon. Tähän olisi tietenkin auttanut, jos opinnäytetyölle olisi löytynyt toimeksiantaja. Jatkossa täytyy parantaa omaa panostusta aikataulujen kiinni pitämiseen seuraavien projektien parissa. Kokonaisuudessaan olen opinnäytetyön lopputulokseen kuitenkin tyytyväinen.

## LÄHTEET

Makzan. 2011. HTML5 Games Development by Example. Beginner's Guide. Birmingham: Packt Publishing Ltd.

Harris, A. 2013. HTML5 Game Development for Dummies. Hoboken, New Jersey: John Wiley & Sons Inc.

Seidelin, J. 2012. HTML5 Games Creating Fun with HTML5, CSS3 and WebGL. Chichester, West Sussex: John Wiley & Sons Inc.

W3Schools. 2015. CSS3 Animations. Luettu 24.2.2015.

[http://www.w3schools.com/css/css3\\_animations.asp](http://www.w3schools.com/css/css3_animations.asp)

Google. 2015. Google Chrome ominaisuudet. Luettu 10.2.2015.

<https://www.google.com/chrome/browser/features.html>

Html5test. 2015. How well does your browser support html5? Kuvakaappaus HTML5 tukevien selaimien pisteytyksestä. Luettu 10.2.2015.

<https://html5test.com/results/desktop.html>

Adobe Photoshop CC. 2015. Adobe Systems.

GIMP 2. 2015. The GIMP Development Team.

Adobe Illustrator CC. 2015. Adobe Systems.

Inkscape. 2015. The Inkscape Team.

Audacity. 2015. Kuvakaappaus Audacityn Windows käyttöliittymästä.

<http://audacity.sourceforge.net/about/images/audacity-windows.png>

ZygoteBody. 2015. Kuvakaappaus ZygoteBody 3D-anatomia työkalusta.

<https://zygotebody.com/> (vaatii kirjautumisen)

Quintus. 2015. Quintus HTML5-pelimoottori. Luettu 1.3.2015.

<http://www.html5quintus.com/guide/intro.md>