

Elina Kuivinen

# Verkkopelin päivitysprojektin suunnittelu ja toteutus

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Mediatekniikan koulutusohjelma

Insinöörityö

5.9.2016

Tekijä Otsikko	Elina Kuivinen Verkkopelin päivitysprojektin suunnittelu ja toteutus
Sivumäärä Aika	32 sivua + 1 liite 5.9.2016
Tutkinto	Insinööri (AMK)
Koulutusohjelma	Mediatekniikka
Suuntautumisvaihtoehto	Digitaalinen media
Ohjaaja	Yliopettaja Kari Salo
<p>Insinööriyön tarkoituksena oli suunnitella ja toteuttaa Flash-ohjelmalla rakennetun Arkkitehtuurimme vuosikymmenet -verkkopelin päivittämissuunnitelma. Alkuperäisen pelin toteutustavan vuoksi pelin toimimattomuus mobiililaitteissa oli aiheuttanut sen, ettei se enää tavoittanut kohderyhmäänsä tarpeeksi laajasti. Lisäksi pelin yleinen ulkoasu ei sopinut yhteen tilaajaorganisaation uudistetun visuaalisen ilmeen kanssa.</p> <p>Ennen uudistamisprojektin aloittamista pelin kohderyhmälle tehtiin kysely, jonka perusteella arvioitiin pelin etenemislogiikan, pisteiden laskun ja muiden pelillisten elementtien muutostarve. Saadun palautteen pohjalta rakennettiin uusi alustariippumaton pelisovellus pääosin HTML- ja JavaScript-kielten avulla. Tulostusta varten luotiin yksinkertainen taulu MySQL-tietokantaan ja pisteet tallennettiin tietokantaan yhdistelemällä PHP- ja Ajax-koodaustekniikoita.</p> <p>Uuden pelisovelluksen rakentamisessa käytettiin responsiivisia verkkokehitystekniikoita, jotta pelin pelaaminen olisi mahdollisimman sujuvaa usealla eri päätelaitteella. CSS-merkin- täkielen avulla sovelluksen ulkoasu suunniteltiin mukautumaan käytettävän päätelaitteen ruutukoon mukaisesti. Pelisovelluksen graafinen ilme ja visuaaliset elementit suunniteltiin ja toteutettiin yhteistyössä tilaajaorganisaation graafikon kanssa.</p> <p>Pelisovelluksesta on tarkoitus tehdä myös ruotsinkielinen versio siten, että valmis suomenkielinen sovellus kloonataan ja suomenkieliset tekstit korvataan niiden ruotsinkielisillä versioilla. Työn viimeistelyvaiheessa valmis kaksikielinen sovellus siirretään tilaajaorganisaation palvelimelle ja linkki valmiiseen sovellukseen lisätään tilaajaorganisaation verkkosivuille.</p>	
Avainsanat	Flash, mobile first, opetuspelejä, pelisuunnittelu

Author Title Number of Pages Date	Elina Kuivinen Planning and implementing an upgrade project for an online learning game 32 pages + 1 appendix 5 September 2016
Degree	Bachelor of Engineering
Degree Programme	Media Technology
Specialisation option	Digital Media
Instructors	Kari Salo, Principal Lecturer
<p>The purpose of this thesis was to plan and implement an upgrade project for a Flash-based learning game called Arkkitehtuurimme vuosikymmenet. Due to the execution of the original game it did not work on mobile devices and thus did not reach the target audience widely enough. In addition, the overall layout of the game was not following the client organization's renewed visual guidelines.</p> <p>Before starting the practical part of the project a survey was performed among the target group. The purpose of this survey was to evaluate the need for change for several gaming elements. Based on the feedback received, a new cross-platform game application was constructed by using mainly HTML- and JavaScript-coding languages. A simple table for the result list was created in MySQL-database and the function for saving the score into the database was created by combining PHP- and AJAX-coding techniques.</p> <p>Responsive web design techniques were used in order to build the renewed application so that the gaming experience would not depend on the device used to play the game. The layout of the application was designed to adapt to the screen size of the user's device by using the CSS-markup language. The visual elements of the game were designed and carried out in co-operation with the client organization's graphic designer.</p> <p>It is intended to create a Swedish-language version of the game later by cloning the Finnish-language application and replacing the Finnish texts with their Swedish versions. When the bilingual application is ready, it will be transferred to the client organization's server and a link to the finished application will be added to the organization's website.</p>	
Keywords	Flash, game design, learning game, mobile first

## Sisällys

1	Johdanto	1
2	Flash	1
2.1	Yleistä	1
2.2	Ongelmat	3
2.3	Tulevaisuus	4
3	Mobiilioptimoitu verkkosovellus	6
3.1	Responsiivisuus	6
3.2	Mobile first	7
4	Pelisuunnittelu	8
4.1	Konseptisuunnittelu	8
4.2	Käyttöliittymäsuunnittelu	9
4.3	Pelin rakenne	9
4.4	Fyysiset ominaisuudet: rajoitukset	10
4.5	Pelillistäminen opetuksessa	11
5	Arkkitehtuurimme vuosikymmenet -verkkopeli	12
5.1	Alkutilanne	12
5.2	Tärkeimmät kehityskohteet: suunnittelu ja toteutus	13
5.2.1	Visuaalinen ilme	13
5.2.2	Pelilogiikka	16
5.3	Uudistamisprojektin eteneminen	17
5.3.1	Käyttäjätutkimus	17
5.3.2	Käytetyt tekniikat	19
5.4	Valmis peli	25
5.4.1	Toiminta	25
5.4.2	Jatkokehitys	27
6	Yhteenveto	27
	Lähteet	30
	Liitteet	
	Liite 1. Arkkitehtuurimme vuosikymmenet -pelin käyttäjätutkimus	

## 1 Johdanto

Insinööriyön tavoitteena on suunnitella ja toteuttaa Flash-ohjelmalla rakennetun Arkkitehtuurimme vuosikymmenet -verkkopelin päivittämisprojekti. Flash-ohjelmistoympäristössä luodut sovellukset eivät toimi ilman ulkopuolisen liitännäisen lataamista, ja varsinkin mobiililaitteilla Arkkitehtuurimme vuosikymmenet -pelin pelaaminen on käytännössä mahdotonta. Pelin toimimattomuus mobiililaitteissa on aiheuttanut sen, ettei se tavoita kohderyhmäänsä, nuoria, tarpeeksi tehokkaasti, joten se on päivitettävä nykyaikaan siten, että se toimii saumattomasti mahdollisimman monella eri päätelaitteella.

Alkuperäinen peli on osa tilaajaorganisaation, Suomen arkkitehtuurimuseon, kiinteää Arkkitehtuurimme vuosikymmenet -näyttelykokonaisuutta. Peli tuotettiin 2000-luvun alkupuolella osaksi kouluopetuksen oheisaineistoa. Kyseessä on lineaarisesti etenevä yksinkertainen kysymys-vastaustyyppinen peli, joka sisältää runsaasti tekstiä ja jonkin verran animaatioita. Pelin sisältö muotoutui aikoinaan kehitysvaiheessa sellaiseksi, että se kohdentui suunniteltua vanhemmalle ikäryhmälle, ja siksi lapsekkaat animaatiot ovat ristiriidassa pelin sisällön kanssa.

Pelin tarkoituksena on tiedon lisääminen ja toimiminen opetusmateriaalin tukena. Insinööriyössä keskitytäänkin pohtimaan pelillisyyden merkitystä ja sen vaikutuksia opetuksessa ja sitä, mitä eri vaiheita pelisuunnittelu sisältää. Insinööriyön tavoitteena on rakentaa kohderyhmältä saadun palautteen perusteella ja modernien verkkokehitystyökalujen avulla mobiilioptimoitu selainpohjainen peli, joka noudattaa tilaajaorganisaation uudistettua visuaalista ilmettä. Lisäksi alkuperäinen peli on koettu raskaaksi, joten myös pelilogiikan ja pelin etenemisen uudelleenarviointi on paikallaan.

## 2 Flash

### 2.1 Yleistä

Flashilla voidaan tarkoittaa monta eri asiaa. Sillä voidaan viitata SWF-tiedostoon (Small Web Format), joka on Adobe Systems -yhtiön Flash-ohjelmistolla tuotetun sovelluksen,

kuten esimerkiksi pelin tai mainosbannerin, verkkosivustolle upotettava lopullinen tiedostomuoto. Flash-ohjelmistolla voidaan luoda myös yksinkertaisia videotiedostoja, eli FLV-tiedostoja (Flash Video). [1.]

Flashista puhuttaessa eniten sekaannusta aiheuttaa Flash Player, liitännäinen, joka on asennettava selaimeen, mikäli haluaa tarkastella Flash-sisältöä. Insinööriydessäni viitataan jatkossa Flashilla kehitysympäristöön, jossa Flash-sovellusten luomisprosessi tapahtuu.

Flash syntyi, kun Macromedia-niminen yhtiö osti vuonna 1996 silloin FutureSplash Animator -nimellä tunnetun monialustaisen animaatiotyökalun ja muutti sen nimen Flashiksi. Flashin suosio kasvoi vähitellen, ja viimeistään Adobe Systems -nimisen yhtiön ostettua Macromedian vuonna 2005 Flash oli vakiinnuttanut asemansa yhtenä suosituimpana multimedia-alustana. [2.] Flashin avulla pystyi tuottamaan animaatioita, videoita ja interaktiivisia elementtejä sisältäviä sovelluksia verkkosivustoille liitettäväksi. Koska Flash-animaatioihin pystyi sisällyttämään tekstiä ja vektorigrafiikkaa, ne eivät vieneet kovinkaan paljon tilaa, ja näin ollen verkkosivustojen Flash-sisältö latautui suhteellisen nopeasti. [3.] Flash oli erityisen suosittu alusta etenkin animaatioita sisältävien mainosbannereiden ja pelisovellusten luomiselle, sillä Flash-sovellukset kykenevät reagoimaan muun muassa käyttäjän hiiren ja näppäimistön avulla syöttämiin komentoihin.

Samana vuonna kuin Adobe Systems osti Flashin, sai alkunsa startup-yritys nimeltä YouTube, joka on nyt maailman suosituin online-videopalvelu. YouTuben suosion myötä Flashista tuli olennainen multimediatoimija verkossa, sillä YouTube valitsi Flash-soittimen oletussoittimekseen videontoistoon ja siten myös Flash-videon suosio kasvoi. Ongelmat alkoivat, kun YouTube päätti vuonna 2007 ryhtyä tarjoamaan pääsyä videoihinsa sellaisessa formaatissa, joka oli optimoitu mobiililaitteille, mikä lakkautti Flashin tarpeen. [4.] Aikaisemmin kaikki YouTuben videomateriaali oli .flv-muodossa, mutta videoita alettiin kääntää H.264-tiedostomuotoon, jotta ne saataisiin tarjolle Applen Apple TV-laitteisiin [5].

## 2.2 Ongelmat

Adoben Flash-tuoteperheeseen kuuluvat ohjelmat ovat tarvinneet historiansa aikana lukuisia tietoturvapäivityksiä, sillä ne ovat altistuneet useasti erilaisille hakkereiden toteutetuille tietoturvahyökkäyksille. Lokakuussa 2015, vain päivä Flash Playerin kuukausittaisen turvallisuuspäivitysten jälkeen, Adobe ilmoitti suuresta turvallisuuteen liittyvästä haavoittuvaisuudesta, joka vaikutti kaikkiin Flash Playerin versioihin Windows-, Mac- ja Linux-tietokoneissa. Käyttäen hyväkseen tätä haavoittuvaisuutta hyökkääjän, eli hakkerin, oli pahimmassa tapauksessa mahdollista saada kokonaan hallintaansa hyökkäyksen kohteena ollut tietokone. Tilanne eteni lopulta niin pahaksi, että jopa Adoben virallinen kanta oli, että parhaiten käyttäjä pystyi suojelemaan itseään tietoturvahyökkäyksiltä poistamalla Flash Playerin asennuksen tietokoneeltaan. [6.]

Flash-liitännäisen asentaminen tai sen ottaminen käyttöön voi hidastaa tietokoneen suorituskykyä merkittävästi. PCWorld testasi vuonna 2015 viidellä eri selaimella, onko liitännäisen poistamisella käytöstä merkittäviä vaikutuksia tietokoneen suorittimeen ja muistin kulutukseen. Tulokset olivat lähes shokeeraavia: esimerkiksi Opera 31 -selaimella Flash-liitännäisen poistaminen käytöstä kulutti suorittimen sykleistä lähes 75 prosenttiyksikköä vähemmän. Myös muilla selaimilla ero näkyi tuloksissa selkeästi. [7.] Flash-liitännäisen poistaminen käytöstä selaimessa on nykyään niin helppoa, että näiden tulosten perusteella on ymmärrettävää, miksi joku niin tekisi. Tämä ei tietenkään ole hyvä uutinen Flashin kannalta.

Suurimmaksi vedenjakajaksi Flashin kohdalla muodostui kuitenkin Applen haluttomuus liittää Flash-tukea mobiililaitteisiinsa, eli iPhoneen, iPadiin ja iPodiin. Applen perustaja Steve Jobs ei uskonut Flashilla olevan tulevaisuutta ja julkaisi huhtikuussa 2010 avoimen kirjeen ”Thoughts on Flash”. Jobs arvosteli muun muassa sitä, kuinka Adoben Flash-tuotteet ovat sataprosenttisesti patentoituja, jolloin Adobe kontrolloi yksin esimerkiksi niiden hinnoittelua ja jatkokehitystä. Adoben tuotteet ovat toki laajasti saatavilla, mutta eivät avoimessa käytössä. Jobs uskoi avoimiin standardeihin, kuten HTML- ja CSS-kuvauskieliin sekä JavaScript-skriptikieleen, jotka eivät tarvitse kolmannen osapuolen liitännäisen, kuten Flashin, asentamista toimiakseen saumattomasti useassa eri päätelaitteessa ja selaimessa. [8.]

Steve Jobsin mielestä Flashin kulta-aika oli ohi, sillä se kehitettiin PC-aikakauden aikana, jolloin hiiren käyttäminen oli kosketusnäytön käyttämistä yleisempää. Viimeistään 2010-

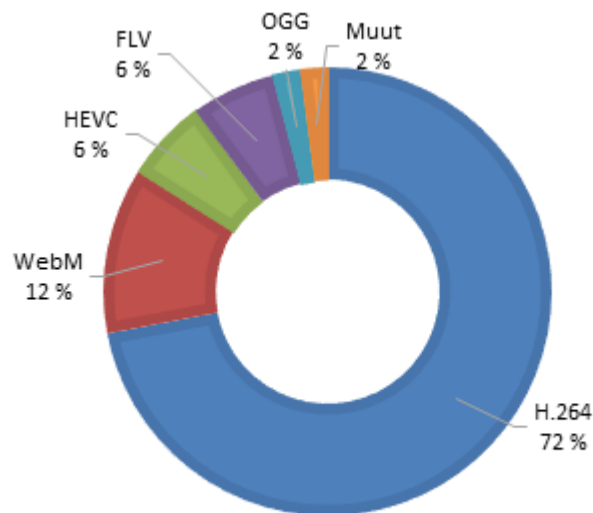
luku merkitsi mobiiliaikakaudelle siirtymistä. Tällöin pääosaan nousivat pienitehoiset laitteet, kosketusnäytöt ja avoimet web-standardit. Näillä osa-alueilla Flash ei enää riitä, vaan tulevaisuus on HTML5-standardissa. [8.]

### 2.3 Tulevaisuus

#### Flash-video (FLV)

Encoding.com-sivuston uusimman, vuosittain julkaistavan maailmanlaajuisen mediaformaattiraportin mukaan vain kuudessa prosentissa mobiili- ja verkkovideoista käytettiin Flash-videonpakkausstandardia vuonna 2015. Vastaava luku oli edeltävänä vuonna 21 prosenttia, joten muutos on huomattava. Sivusto ennustaakin Flash-videoiden katoavan raportistaan kokonaan seuraavien 24 kuukauden aikana. [9.]

Kuten kuva 1 osoittaa, tämän hetken suosituin videonpakkausstandardi on H.264. Encoding.com-sivuston mukaan standardin käyttöaste jopa kasvoi hieman vuoteen 2014 verrattuna. H.264:n ohella myös WebM-standardi jatkaa kasvuaan Flash-videoiden osuuden pienentyessä.



Kuva 1. Käytetyimmät videonpakkausstandardit vuonna 2015 [muokattu lähteestä 9].



Suurimmat videopalveluntarjoajat ovat joko korvanneet tai korvaamassa Flash-pohjaiset videosoitimensa jollain toisella tekniikalla. Esimerkiksi Yleisradio Oy eli Yle ilmoitti helmikuussa 2016 pyrkivänsä korvaamaan Areena-videopalvelunsa selainversiossa käyttämänsä Flash-soittimen HTML5-soittimella vuoden 2016 aikana [10].

Internetin suosituin videontoistopalvelu YouTube siirtyi käyttämään HTML5-soitinta oletussoittimenaan jo tammikuussa 2015. Syynä olivat esimerkiksi MSE:n (MediaSource Extensions) luomat mahdollisuudet mukautuvalle tiedonsiirtonopeudelle suoratoistossa, reaaliaikaisen lähetyksen toistaminen pelikonsoleissa, mediasoitimissa ja selaimissa sekä VP9-koodekin käyttö, joka mahdollistaa paremman kuvanlaadun pienemmällä kais-tanleveyden käytöllä. Niin ikään suosittu internetin VOD-palvelut (Video On Demand), kuten Netflix ja Viaplay, käyttävät videoiden toistoon HTML5-soitinta. [11.]

### **Flash-sovellukset (SWF)**

Flashin alasajoa ja katoamista verkosta ovat hidastaneet varsinkin mainosbannerit, sillä vielä kesäkuussa 2015 noin 84 prosenttia internetin bannereista oli toteutettu Flashilla [12]. Google kuitenkin ilmoitti samassa kuussa ottavansa syyskuusta 2015 lähtien Chrome-selaimessaan automaattiseen käyttöön asetuksen, joka pysäyttää oletusarvoisesti tietynlaisten liitännäisten toiminnan parantaen näin sivustojen latausnopeutta ja toimintakykyä sekä päätelaitteiden akun kestoa. Liitännäisillä tarkoitettiin pääasiassa animoituja Flash-mainoksia, ja Google kehottikin hakukonemarkkinointityökalunsa AdWordsin käyttäjiä joko vaihtamaan tai konvertoimaan olemassa olevat Flash-mainok-sensa Googlen avulla HTML5-mainoksiksi [13].

Jopa Flashin omistava yhtiö, Adobe, päätyi vuoden 2016 alussa korvaamaan Creative Cloud -tuoteperheeseen kuuluvan Adobe Flash -kehitysympäristön uudella Adobe Animate -ohjelmistolla. Adoben blogissa vuoden 2015 loppupuolella julkaistussa uutisessa uudistuksen syiksi luettiin muun muassa HTML5:n syntyminen ja nykyaikaisia web-standardeja hyödyntävien animaatioiden kysynnän kasvu. Korostaakseen ohjelmistonsa asemaa verkon johtavana animaatiotyökaluna, yhtiö päätti uudelleenbrändätä tuot-teensa vaihtamalla sen nimen Flashista Animateksi. [14.] Esimerkiksi HTML5 Canvas -animaatioiden luominen on ollut aikaisemminkin mahdollista Flashilla, mutta nimenmuu-tos nostaa esille ajatuksen yhtiön tarpeesta suunnata katseensa tulevaisuuteen ja päästä lopullisesti eroon Flashin synnyttämistä negatiivisista mielikuvista.

Animaten tuomat muutokset eivät kuitenkaan poista sitä tosiasiaa, että Flash-kehitysympäristön ominaisuudet säilyvät jatkossakin. Verkossa on edelleen paljon Flashilla luotua sisältöä, esimerkiksi videoita ja pelejä, joten Adobella ei ole aikomusta lopettaa Flash Playerin kehitys- ja päivitystyötä [15]. Nähtäväksi jää, milloin kaikesta Flash-sisällöstä lopulta luovutaan.

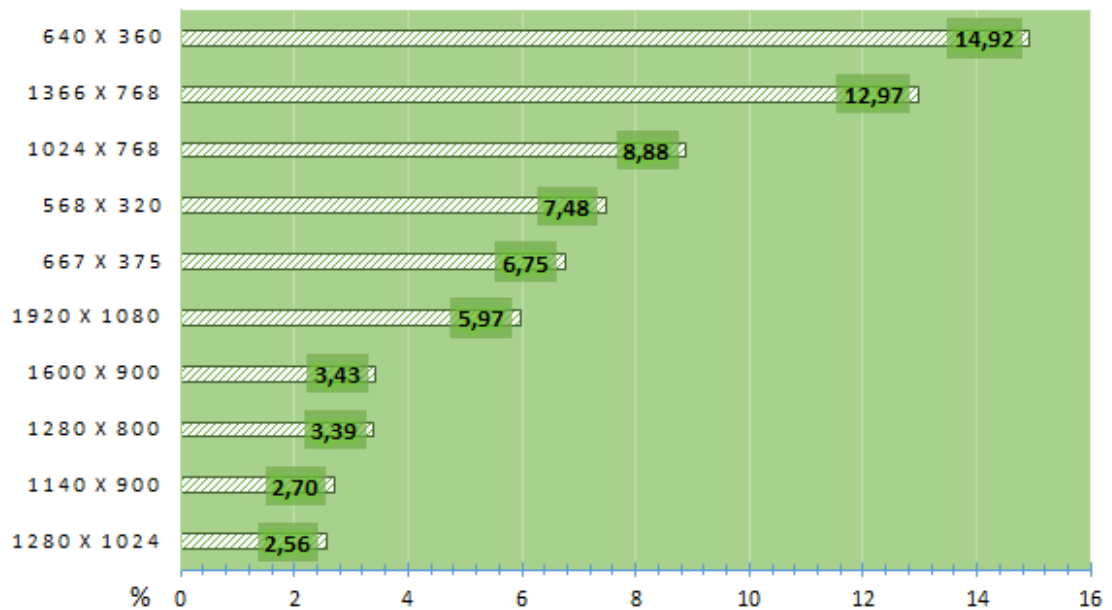
### **3 Mobiilioptimoitu verkkosovellus**

#### **3.1 Responsiivisuus**

Responsiivinen suunnittelu (responsive design) tarkoittaa sitä, kuinka selain reagoi (engl. responds) ympäristöönsä. Lähtökohtaisesti responsiivisen suunnittelun päätavoite on suunnitella verkkosivusto käyttöliittymältään sellaiseksi, että se kykenee mukautumaan, eli skaalautumaan, käytettävän päätelaitteen asettamien ominaisuuksien mukaisesti. Näin verkkosivuston selaaminen on miellyttävää myös mobiililaitteella, eikä käyttäjä joudu esimerkiksi turvautumaan vaakasuuntaiseen vierityspalkkiin nähdäkseen kaiken verkkosivun sisältämän sisällön.

Responsiivisessa verkkosuunnittelussa ei voida määritellä vain yhtä kaiken kattavaa koodikokonaisuutta, joka toteuttaisi halutun lopputuloksen, vaan se on joukko strategioita, joita yhdistelemällä saadaan aikaiseksi sovellus, joka toimii saumattomasti esimerkiksi päätelaitteen ruudun koosta riippumatta. [16, s. 5.] Tällainen toteutus voi olla hyvin haastavaa, sillä markkinoilla olevien laitteiden kirjo on suuri ja näyttöjen koko vaihtelee mobiililaitteiden pienistä ruuduista koneiden leveisiin monitoreihin.

Kuvassa 2 on esitetty maaliskuun 2016 aikana kymmenen eniten käytettyä näytön resoluutiota ja niiden prosentuaalinen osuus kaikista käytössä olevista laitteista. Tiedot perustuvat W3Counter-sivuston keräämään dataan. Kuvasta huomaa selkeästi, että sekä mobiililaitteiden että pöytäkoneiden näytön resoluutiot vaihtelevat huomattavasti. Suosituin laite internetin selaamiseen on puhelin, mutta myös kannettavat ja pöytätietokoneet sijoittuvat listan kärkisijoille. Tästä voidaan päätellä, että käytettävien laitteiden monimuotoisuus kannustaa yhä enemmän responsiiviseen verkkosuunnitteluun.



Kuva 2. Kymmenen käytetyintä näytön resoluutiota maaliskuussa 2016 [muokattu lähteestä 17].

Tilanteen helpottamiseksi on kehitetty laaja valikoima valmiita toimintamalleja ja -tapoja, joiden avulla sivut saadaan skaalautumaan eri resoluutioiden välillä erilaisiksi. On olemassa myös valmiita JavaScript-koodikirjastoja, jotka voi linkittää verkkosovelluksen yhteyteen ja jotka keventävät verkkokehittäjien työtä.

### 3.2 Mobile first

Ennen mobile first -ajattelutapaa yleinen käytäntö verkkosivustojen kehitystyössä oli rakentaa pääasiallisen verkkosivuston lisäksi erillinen mobiililaitteille suunnattu versio, jonne mobiililaitteiden käyttäjät ohjattiin suoraan. Tämä tarkoitti kuitenkin käytännössä, että verkkokehittäjien oli rakennettava alusta asti kaksi erillistä versiota, mikä lisäsi heidän työmääräänsä. Mobile first -sovellusten idea perustuu sille ajatuspohjalle, että ensisijainen laite, jolla käyttäjä tarkastelee sovellusta, on mobiililaitte. Näin ollen ei ole tarkoituksenmukaista rakentaa sovelluksesta ensin täysikokoista versiota, joka toimii suuremmilla päätelaitteilla ja sen jälkeen karsia siitä toissijaisia elementtejä, jotta se toimisi paremmin mobiililaitteilla, vaan päinvastoin: mobiililaitteiden käyttäjät ovat ajattelutavan mukaan ensisijainen kohderyhmä sovelluskehityksessä, ja mobiilioptimoitun verkkosivuston on tarkoitus mukautua ketterästi suuremmille päätelaitteille.

Mobile first -ajattelutapa syntyi vähitellen, kun mobiililaitteiden käyttäjäkunta kasvoi. Kasvun taustalla oli kaksi suurta tekijää: samalla kun laitteiden ominaisuudet ja toiminnollisuudet paranivat, niiden hinnat laskivat. Näin myös ne ihmiset, joilla ei aiemmin ollut varaa ostaa pöytä- tai kannettavaa tietokonetta, pääsivät verkkoon edullisempien mobiililaitteiden ja tiedonsiirtosopimusten avulla. Lisäksi nopeampien verkkoyhteyksien myötä myös mobiililaitteissa siirrettävän datan määrä kasvoi. Esimerkiksi vuonna 2010 matkapuhelinverkkojen nopeudet kaksinkertaistuivat ja tämä korreloi suoraan myös mobiililaitteiden käytön kasvuun. [18, s. 11.]

Ajattelutavan koetaan johtavan verkkosivuston tai sovelluksen parempaan kokonaiskäyttökokemukseen ja avaavan uusia mahdollisuuksia kasvulle. Ajattelutapa pakottaa myös priorisoimaan eli keskittymään sovelluksen kannalta oleellisiin asioihin, sillä mahdollisuuksien lisäksi myös mobiilin verkon rajoitteet on otettava huomioon. Toisaalta mobile first -ajattelutavan myötä on mahdollista luoda innovatiivisia kokemuksia rakentamalla sovellus uusien, mobiililaitteille jo valmiiksi luontaisten ominaisuuksien varaan. [18, s. 1.]

## **4 Pelisuunnittelu**

### **4.1 Konseptisuunnittelu**

Kaikista yksinkertaisimmankin pelin konseptisuunnitteluun kuuluu monta eri vaihetta. Suunnitteluvaiheen tärkeyttä ei tule aliarvioida: hyvä suunnitelma säästää aikaa ja vaivaa ja mahdollistaa sen, että projekti ylipäätään valmistuu. Huolellinen suunnittelu takaa myös, että suunnittelijalla on aikaa kirkastaa se ajatus, joka hänen on tarkoitus välittää yleisölle. [19, s. 39.]

Jokaisen pelikonseptin takana on idea. Perusidean lisäksi pelissä on oltava ”koukku” eli jokin ominaisuus, joka herättää pelaajan mielenkiinnon ja saa tämän jatkamaan pelamista. Ennen kuin peliä voidaan kuitenkaan alkaa luoda, on ideaa jalostettava. Suunnitteluvaihe on hyvä dokumentoida, jolloin idean jalostaminen konkreettisemmaksi on helpompaa. Idean, eli pelin ytimen, ympärille luodaan asteittain muitakin elementtejä, kuten esimerkiksi pelin tyyli, pelisäännöt, peliympäristö ja pelihahmot. [19, s. 41–43.] Käytännössä konseptisuunnitteluvaiheessa määritellään, mitä pelissä tapahtuu ja millä keinoin peliä aletaan markkinoida kohdeyleisölle.

## 4.2 Käyttöliittymäsuunnittelu

Kun pelin konsepti on muotoutunut, on aika kiinnittää huomiota pelin käyttöliittymään ja sen suunnitteluun. Käyttöliittymä tarkoittaa käytettävän ohjelmiston tai tuotteen sitä osaa, jonka kautta käyttäjä käyttää tuotetta. Näin ollen se on elementti, joka vaikuttaa eniten pelikokemukseen, sillä se määrittelee vuorovaikutussuhteen pelaajan ja pelin välillä [20, s. 147].

Käyttöliittymäsuunnittelu on yksi vaikeimmista osa-alueista onnistuneen pelin suunnitteluprosessissa, sillä käytettävien laitteiden ruutukoot vaihtelevat eikä videopeliteollisuudessa ole yleistä standardia käyttöliittymäsuunnittelulle [20, s. 147–148]. Hyvän käyttöliittymän perustavoitteena tulisi olla se, että pelaaja kokee kontrolloivansa pelikokemustaan. Siksi suunnitteluvaiheessa tulisi kiinnittää huomiota seuraaviin kysymyksiin:

1. Toimiiko rajapinta halutulla tavalla, kun pelaaja käyttää sitä?
2. Onko käyttöliittymä helppo omaksua?
3. Tunteeko pelaaja voivansa vaikuttaa vahvasti pelin lopputulokseen?

Onnistunut käyttäjäkokemus motivoi pelaajaa jatkamaan, joten edellä oleviin kysymyksiin on tärkeää saada myöntävä vastaus ennen pelin viimeistelyä. Mikäli vastaus on kieltevä, on mahdollisiin ongelmiin pureuduttava syvemmin: Miksi jokin asia ei toimi halutulla tavalla? Mikä tekee omaksumisesta vaikean, ja millä toimilla tämän voi korjata? Mikäli pelaaja ei tunne kontrolloivansa pelin etenemistä, millä toimilla hänet voi saada tuntemaan toisin? [21, s. 222–223.]

## 4.3 Pelin rakenne

Alustasta ja vaikeustasosta riippumatta jokaisella pelillä on selkeä perusrakenne, joka koostuu vähintään kolmesta eri osasta: aloituksesta, pelisilmukasta ja lopetuksesta [20, s. 123]. Kuvassa 3 on esitetty hyvin yksinkertainen rakennekaavio.



Kuva 3. Yksinkertainen kuvaus pelirakenteesta.

Aloituksessa asetetaan arvot niille elementeille, joita ilman peli ei voi edetä. Käytännössä tämä tarkoittaa, että asetetaan arvot pelin muuttujille. Esimerkiksi pelihahmon lähtöpiste ja parametrit, vihollisten määrä ja lähtöpiste, kerättävät esineet ja niistä saatavat bonukset sekä vaikeustason asetukset on määriteltävä ennen pelin aloittamista. Listasta voi muodostua hyvinkin pitkä. [20, s. 123.]

Pelisilmukka on pelin sydän, ja tässä kohtaa tapahtuvat pelin etenemistä määrittävät rutiinit, jotka jatkuvat niin kauan, kuin pelaaja jatkaa pelaamista. Pelisilmukassa saadaan käyttäjäsyötettä, jonka perusteella voidaan laskea seuraamukset pelaajan toimille. Seuraamukset piirtyvät edelleen pelaajan näytölle ja vaikuttavat tämän tuleviin valintoihin. [20, s. 123–124.]

Kun pelaaja päättää lopettaa pelaamisen, suoritetaan lopetus eli järjestelmän muistista puhdistetaan kaikki jäljelle jääneet pelirutiinit ja suoritetaan tarpeelliset sammuttamistoimet. Tämä osio on erityisen tärkeä varsinkin mobiililaitteille suunnatuissa peleissä, sillä laitteilla voi suorittaa yhtä aikaa monia eri toimia eikä ole käyttäjän edun mukaista, että peli jää pyörimään taustalle kuluttamaan laitteen järjestelmän resursseja ja heikentämään sen suorituskykyä. [20, s. 124.]

#### 4.4 Fyysiset ominaisuudet: rajoitukset

Pelisuunnittelussa on otettava huomioon pelin sisällön ja toiminallisuuden lisäksi myös fyysisen päätelaitteen, eli sen, jolla peliä pelataan, ominaisuudet ja rajoitukset. Mobiilipeliä suunniteltaessa ensimmäinen fyysinen ominaisuus, joka tulee vastaan, on tietysti päätelaitteen ruutukoko. Suurimmassa osassa puhelimista näyttö on suhteellisen pieni ja se aiheuttaa seuraavia ongelmia: Miten ruudulla pystytään näyttämään tarpeeksi informaatiota niin, ettei pelin pelattavuus kärsi, ja miten tästä informaatiosta tehdään tar-

koituksenmukaista ja tunnistettavaa? Miten sama peli näyttää hyvältä ja sen toiminnallisuus säilyy ruutukoon kasvaessa tai pienentyessä? Miten ruutukoko vaikuttaa pelimekaniikkaan? [20, s. 277.]

Myös aikaa ominaisuutena on hyvä tarkastella, ja sen asettamat rajoitukset on otettava huomioon. Varsinkin mobiililaitteita käytetään usein silloin, kun käyttäjällä on hetki aikaa sillä välin, kun hän odottaa jotain muuta tapahtuvaksi. Kun tällaisena aikana aloitetaan pelin pelaaminen, tulisi pelaajalle tarjota tyydyttävä pelikokemus alle kolmessa minuutissa. Tämä tarkoittaa, että pelissä saavutetaan jonkinlainen tavoite, kuten palkinto tai tallennuspiste, kyseisen aikarajan puitteissa. [20, s. 279.] Tästä syystä pelin kokonaispituudellekin on omat rajoituksensa, eikä mobiilipelin läpipelaaminen yleensä vaadikaan kymmeniä tunteja peliaikaa.

#### 4.5 Pelillistäminen opetuksessa

Pelaamista ei enää nykypäivänä pidetä pelkästään lasten ja nuorten harrastuksena, vaan se on merkittävä osa vallitsevaa kulttuuria ja yhteiskuntaa. Viihdekäytön lisäksi pelejä ja pelillisyyttä käytetään entistä enemmän hyväksi myös esimerkiksi opetuksessa [22, s. 10]. Opetuksen pelillistämisen on huomattu tukevan oppimista varsinkin sellaisten henkilöiden kohdalla, joilla on vaikeuksia pysyä mukana perinteisessä kouluopetuksessa. Opetuspelit pelityyppinä pyrkivät yhdistämään oppimisen ja viihteen mielekkäällä tavalla, jolloin parhaassa tapauksessa oppimisesta tulee sekä hauskaa että vuorovaikutteista. Tämä on monen mielestä perinteisiä menetelmiä mukavampi tapa oppia. [22, s. 66.]

Opetuspeliä suunniteltaessa on huomioitava tasapainoa tuovat seikat: peli ei saa olla liian haastava tai liian helppo ja sen on oltava riittävän mutta ei liian viihdyttävä. Pelin on motivoitava pelaajaa jatkamaan pelaamista ja opetussisällön on oltava selkeää ja jäseneltyä [22, s. 64]. Tämän takia on tärkeää määritellä etukäteen ne seikat ja aihepiirit, joista halutaan pelin avulla välittää tietoa.

Itse pelisovelluksen roolia opetuksessa on myös syytä pohtia. Tarjoaako peli itsessään kaiken oppimisen kannalta tärkeän materiaalin vai onko se osa suurempaa kokonaisuutta? Onko pelaajan tarkoitus oppia pelin edetessä vai vaatiiko pelin pelaaminen jo lähtökohtaisesti jotain tietoa? Voiko opetukseen käytettävän materiaalin ja opetuspelin

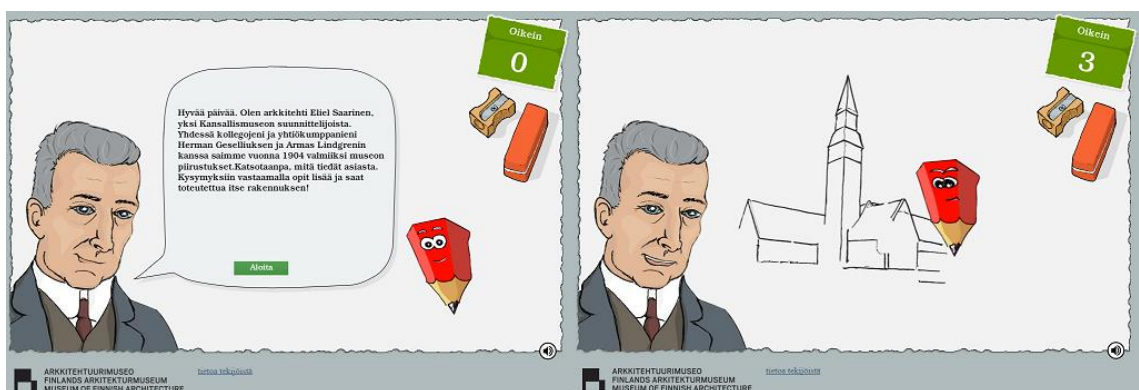
integroida saumattomasti yhteen vai täytyykö materiaali olla täysin hallussa ennen pelin aloittamista? Miten pelin sisältö vaikuttaa pelilogiikan rakentumiseen?

## 5 Arkkitehtuurimme vuosikymmenet -verkkopeli

### 5.1 Alkutilanne

Alkuperäinen Arkkitehtuurimme vuosikymmenet on yksinkertainen kysymys-vastaus-tyyppinen peli, joka koostuu kahdesta viisi tasoa sisältävästä osiosta. Jokainen taso keskittyy tiettyyn vuosikymmeneen 1900- ja 1990-luvun väliltä, ja pelattuaan läpi osion kaikki tasot pelaaja voi tallentaa pisteensä ja lähettää ne tuloslistalle. Tuloslistalla näytetään sata parasta tulosta viimeisen kuuden kuukauden ajalta.

Jokaisella tasolla joku tunnettu suomalainen arkkitehti esittää sarjan kysymyksiä pelaajalle. Kysymykset liittyvät tasolla käsiteltävän aikakauden arkkitehtuuriin ja yleisesti sen ajan tyyliin ja designiin. Tasolla esiintyvä arkkitehti toimii ikään kuin pelaajan oppaana pelin edetessä, ja tasolla käsiteltävän aikakauden esimerkkirakennus on aina tämän arkkitehdin suunnittelema. Kun pelaaja vastaa oikein arkkitehdin kysymykseen, kynä piirtää osan esimerkkirakennuksesta. Pelaajan vastattua oikein kymmeneen kysymykseen esimerkkirakennus valmistuu ja taso on pelattu läpi. Pelitason aloitusnäkyminen ja tasolla eteneminen on havainnollistettu kuvassa 4.



Kuva 4. Pelin ensimmäisen osion ensimmäisen tason aloitusteksti (vasemmalla) ja esimerkkirakennuksen valmistuminen (oikealla).



Sekä ennen kunkin pelitason pelaamista että sen jälkeen pelaajan on mahdollista tutustua tarkemmin kyseisen vuosikymmenen arkkitehtuuriin ja tunnettuihin arkkitehteihin klikkaamalla auki peliin linkitetyt lisätietolaatikot. Pelitasojen kysymykset pohjautuvat näiden lisätietolaatikoiden teksteihin, joten pelaajan on halutessaan mahdollista valmistautua tason kysymyksiin jo etukäteen.

## 5.2 Tärkeimmät kehityskohteet: suunnittelu ja toteutus

Insinööriyön tilaajaorganisaation edustajien kanssa käydyissä palaverissa käytiin läpi, mitä ominaisuuksia alkuperäisestä pelistä tulisi säilyttää ja mitä muuttaa. Koska kyseessä on vahvasti tekstipohjainen peli, jonka kysymys- ja lisätietolaatikoiden tekstit muotoutuivat nykyiseen muotoonsa pitkäjänteisen ja tarkan suunnittelu- ja tiedonhaku-prosessin jälkeen, ei niitä voinut juurikaan muokata. Sovelluksesta on tarkoitus tuottaa lisäksi ruotsinkielinen versio, joten myös tekijänoikeusyistä pelin olemassa olevat tekstit ja niiden ruotsinkieliset käännösversiot oli säilytettävä alkuperäisessä muodossaan.

Koska peli sisälsi paljon tekstiä, oli keksittävä muita keinoja sen keventämiseksi ja pelattavuuden lisäämiseksi. Tärkeimmiksi kehityskohteiksi muotoutuivatkin pelin visuaalisen ilmeen uudistaminen ja pelilogiikan päivittäminen. Oli myös mietittävä uusien pelillisten elementtien, kuten esimerkiksi aikarajoitteisuuden tai pelaajan palkitsemisen, lisääminen peliin ja arvioitava niiden tuoma lisäarvo tai sen puute.

### 5.2.1 Visuaalinen ilme

Sen lisäksi, että Arkkitehtuurimme vuosikymmenet oli toteutettu tekniikalla, joka ei enää vastaa nykypäivän mobiiliyhteiskunnan tarpeita, se oli visuaaliselta ilmeeltään lapsekkaan pelkistetty eikä istunut arkkitehtuurimuseon graafiseen ilmeeseen. Osoiden päävalikon elementit olivat palikkamaisia, ja varsinkaan pelin värimaailma ei ollut enää linjassa arkkitehtuurimuseon omien verkkosivujen kanssa.

Tilaajaorganisaatio oli huolissaan pelin piirrosmaisesta etenemisgraafiikasta. Suunnittelu-vaiheessa pelin kohderyhmäksi valittiin alakouluikäiset lapset, mutta alkuperäisen projektin edetessä pelin kysymykset muotoutuivat sellaisiksi, että ne koettiin liian haasta-

viksi alkuperäiselle kohderyhmälle. Julkaisuvaiheessa kohderyhmäksi uudelleenarvioitiin 11–15-vuotiaat lapset ja nuoret, joten etenemisgrafiikan lapsekkuus oli ristiriidassa kohderyhmän iän kanssa.

## Fontti

Insinöörityötä tehdessä käytössä oli arkkitehtuurimuseon visuaalinen ohjeisto, jossa linjataan organisaation omissa materiaaleissa käytettävä värimaailma ja muut graafiset elementit, kuten esimerkiksi fontit. Museon painetuissa materiaaleissa on käytössä Cyrus Highsmithin suunnittelema Scout-niminen fontti ja sen eri versiot [23]. Kyseessä on kaupallinen fontti, joten mahdollisten lisenssiongelmien välttämiseksi päätettiin pelisovellukseen etsiä suoraan kyseisen fontin korvaajaksi avoimen lähdekoodin fontti.

Lopullisessa sovelluksessa päädyttiin käyttämään Christian Robertsonin suunnittelemaa Roboto Condensed -nimistä Google Fonts tarjoamaa fonttia, sillä useita eri fontteja vertailtaessa se muistutti eniten Scoutia [24]. Lisäksi fontin käytölle saatiin myös tilaajaorganisaation graafikon lupa. Google Fonts on Googlen ylläpitämä fonttiarkisto, jonka sisältämät fontit ovat verkkokehittäjien avoimessa ja ilmaisessa käytössä. Näin mahdollistetaan monipuolisempien fonttien käyttö verkkosovelluksissa ja sen myötä laadukkaampi design. [25.] Sekä Scout- että Roboto Condensed -fontit on esitelty kuvassa 5.

### Scout

#### Scout Thin

ABCDEFGHIJKLMNOPQRSTUVWXYZÄÖ  
abcdefghijklmnopqrstuvwxyzääö  
0123456789

#### Scout Regular

ABCDEFGHIJKLMNOPQRSTUVWXYZÄÖ  
abcdefghijklmnopqrstuvwxyzääö  
0123456789

#### Scout Light

ABCDEFGHIJKLMNOPQRSTUVWXYZÄÖ  
abcdefghijklmnopqrstuvwxyzääö  
0123456789

#### Scout Bold

ABCDEFGHIJKLMNOPQRSTUVWXYZÄÖ  
abcdefghijklmnopqrstuvwxyzääö  
0123456789

## Roboto Condensed

ABCDEFGHIJKLMNOPQRSTUVWXYZÄÖ  
abcdefghijklmnopqrstuvwxyzääö  
0123456789

Kuva 5. Scout- ja Roboto Condensed -fontit [muokattu lähteistä 23 ja 24].

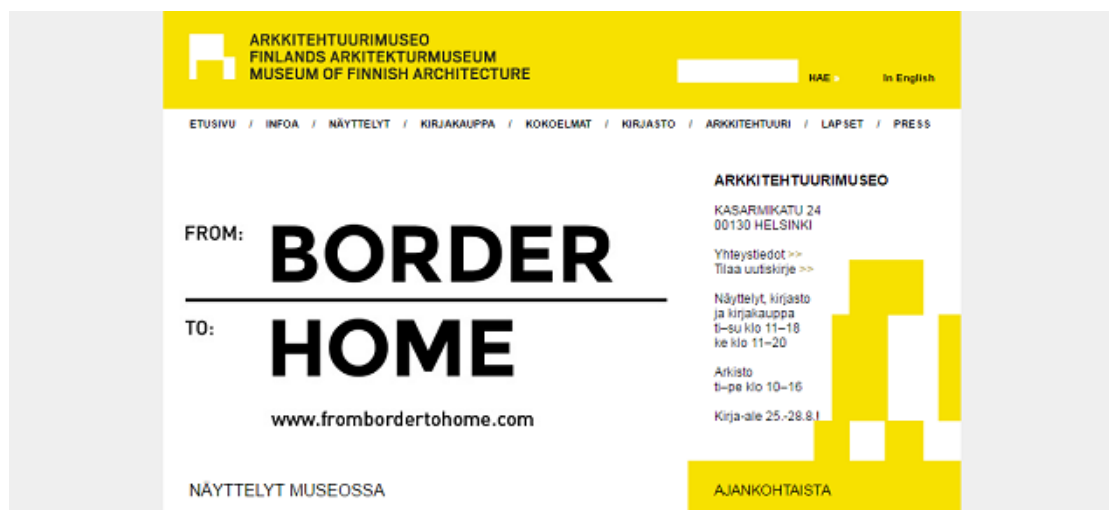
## Värimaailma

Arkkitehtuurimuseon visuaalisessa ohjeistossa museon pääväriksi on määritelty keltainen. Harmaata voi käyttää otsikko- ja ingressitasolla, leipätekstin väri on musta. Kuvassa 6 on esitetty museon painetussa materiaalissaan käyttämät värit.



Kuva 6. Arkkitehtuurimuseon värit [23].

Kuvassa 7 on kuvankaappaus arkkitehtuurimuseon verkkosivuista havainnollistamassa visuaalisen ohjeiston määrittämän värimaailman mukaista ulkoasua. Kuten kuvasta ilmenee, ei alkuperäisen pelisovelluksen harmaa värimaailma soinnu kovinkaan hyvin yhteen museon nykyisen linjan kanssa.



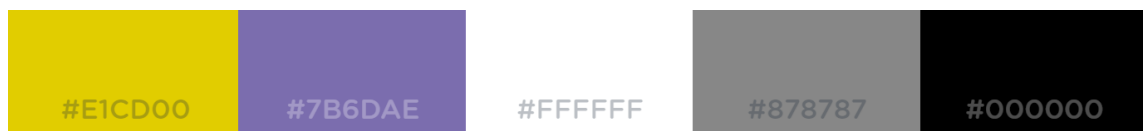
Kuva 7. Arkkitehtuurimuseon verkkosivut [26].

Koska arkkitehtuurimuseolla on oma graafikko, joka on vastuussa kaikista museon ulkoasuun liittyvistä asioista, sovittiin, että pelin rakenteen valmistuessa hän auttaa pelin graafisen ilmeen suunnittelussa, jotta lopputulos olisi museon linjan mukainen. Graafikko Salla Bedard toimitti luonnoksen pelin toivotusta ulkoasusta (kuva 8), ja pelin lopullista versiota rakennettaessa tätä luonnosta pyrittiin noudattamaan mahdollisimman tarkasti.



Kuva 8. Luonnos Arkkitehtuurimme vuosikymmenet -pelin uudistetusta aloitusnäkyästä [27].

Päävärien lisäksi museon graafisessa aineistoissa on käytetty myös violettiä väriä, ja sitä päätettiin käyttää uudistetussa pelisovelluksessa kontrastin luomiseksi. Kuvassa 9 on esitelty pelisovelluksessa käytetty väripaletti.



Kuva 9. Arkkitehtuurimme vuosikymmenet -pelin uudistuksessa käytetty väripaletti.

### 5.2.2 Pelilogiikka

Koska kohderyhmä oli rajattu, oli tarpeen pohtia, mikä vetoaa kohderyhmän ikäisiin henkilöihin. Herättääkö hitaasti etenevä kysymys-vastaustyyppinen peli pelaajan mielenkiinnon siten, että hän jaksaa pelata pelin loppuun asti, vai olisiko jonkin toisen tyylilajin eli genren peli vetoavampi? Tarvitaanko pistelaskurin lisäksi myös muita motivoivia tekijöitä, kuten esimerkiksi vaikeutuvat tasot, aikarajoitteisuus ja sosiaalinen ulottuvuus?

Suunnitteluvaiheen kantavana ajatuksena oli, että lopullinen tuote täyttää kohderyhmän tarpeet. Sen lisäksi, että pelin on oltava kiinnostava, sen täytyy olla myös miellyttävä ja helposti pelattavissa. Projektin suunnitteluvaiheessa todettiin, että paras tapa löytää pelille sopivin etenemistyyli on tehdä käyttäjätutkimus kohderyhmälle. Samalla voidaan karhottaa, millaisia lisäominaisuuksia peliin tulee mahdollisesti lisätä. Pelin tyyli ja vaihtoehdot siinä etenemiselle rajattiin ja määriteltiin pääpiirteittäin yhteistyössä arkkitehtuurimuseon henkilökunnan edustajien kanssa. Henkilökunnan kanssa pidetyssä palaverissa innovoitiin muutamia pelaajan palkitsemistapoja ja se, miten pelissä eri tasoilla tehdyt valinnat vaikuttavat pelin lopputulokseen. Ideoinnin lopputuotteena syntyi kolme erilaista taso-esimerkkiä, jotka testiryhmään kuuluvien käyttäjien oli pelattava läpi.

### 5.3 Uudistamisprojektin eteneminen

#### 5.3.1 Käyttäjätutkimus

Pelin etenemistyylin määrittävä käyttäjätutkimus toteutettiin toukokuun 2016 ensimmäisellä puoliskolla, jolloin kohderyhmän ikäluokkaan kuuluvia koululaisia kutsuttiin pelaamaan testitasot läpi. Kukin testitaso koostui satunnaisesti valitun vuosikymmenen pelikysymyksistä ja erilaisista elementeistä, jotka vaikuttivat esimerkiksi pelin etenemiseen ja pisteytykseen. Pelattuaan tasot läpi testiryhmän jäsenet vastasivat anonyymisti testitasoja koskeviin kysymyksiin, ja vastausten perusteella tarjolla olevista vaihtoehdoista valittiin testiryhmää eniten miellyttävät etenemismallit lopulliseen sovellukseen.

#### **Esimerkkitasot**

Esimerkkitasolla 1 piti vastata jokaiseen tason kysymykseen. Oikeasta vastauksesta sai yhden pisteen eikä väärästä vastauksesta rangaistu, joten tasoa ei voinut hävitä. Esimerkkitasolla oli yhteensä 17 kysymystä, ja taso päättyi, kun kaikkiin oli vastattu. Tason tarkoituksena oli testata, kuinka moneen kysymykseen pelaaja jaksaa vastata yhdellä tasolla.

Esimerkkitason 2 läpäisi vastaamalla oikein viiteen eri kysymykseen. Pelaaja menetti yhden elämän väärästä vastauksesta. Peli päättyi, kun elämiä ei ollut enää jäljellä. Tason tarkoituksena oli testata, miten pelaaja reagoi väärästä vastauksesta annettavaan rangaistukseen.

Esimerkkitasolla 3 testattiin pelaajan paineensietokykyä. Tason läpäisi vastaamalla oikein viiteen eri kysymykseen. Väärästä vastauksesta menetti pisteen, kuitenkin siten, että peli ei päätynyt pelaajan pistemäärän pudotessa noltaan eikä pelaaja voinut saada negatiivisia pisteitä. Pelaajalla oli kaksi minuuttia aikaa pelata taso läpi, ja peli päättyi joko tason läpäisyyn tai ajan loppumiseen.

Kun testiryhmän jäsenet olivat pelanneet esimerkkitasot läpi, heitä pyydettiin etenemään kyselylomakkeeseen (liite 1). Testaajilta toivottiin vastauksia etenkin seuraaviin kysymyksiin:

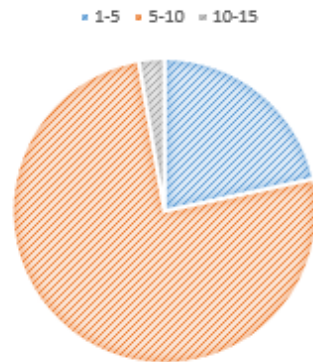
- Mikä on sopiva pituus yksittäiselle pelitasolle?
- Miten aikarajoitteisuus vaikuttaa peli-intoon?
- Miten pelaaja haluaa seurata omaa etenemistään pelin edetessä?
- Millainen pisteytysmalli kannustaa jatkamaan pelaamista?
- Tuleeko pelaajaa rangaista väärästä vastauksesta?
- Haluaako pelaaja jakaa tuloksensa sosiaaliseen mediaan vai riittääkö peliin integroitu tuloslista tulosten vertailuun?

## **Tulokset**

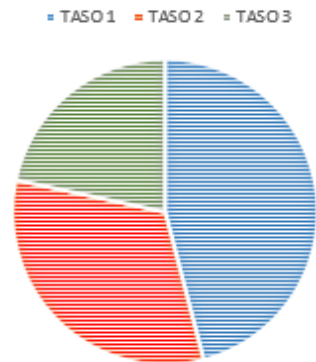
Arkkitehtuurimuseon edustaja välitti esimerkkitasot testiryhmälle testattavaksi. Vastajiksi valikoitui yläkoululaisia, lukiolaisia ja muutama opettaja. Testaukseen osallistui yhteensä 37 henkilöä. Nuorin kyselyyn vastannut henkilö oli 13-vuotias ja vanhin 69-vuotias. 68 prosenttia kyselyyn vastanneista oli iältään 16–17-vuotiaita.

Kyselyn vastauksista ilmeni, että aikarajoitteisuus ja väärästä vastauksesta rankaiseminen eivät olleet hyviä lisäelementtejä peliin. Suurin osa vastanneista kertoi avoimessa palautteessa pelaavansa opetuspelejä nimenomaan oppimistarkoituksessa, jolloin sovelluksen viihdearvo on toissijainen. Pelisovellukseen tuovat lisäarvoa selkeä etenemistyli ja reilu pisteytys. Kuvassa 10 on esitetty grafiikan avulla tärkeimmät käyttäjätutkimuksen tuloksista.

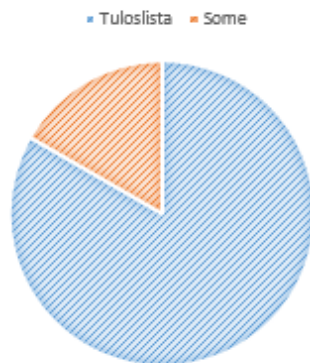
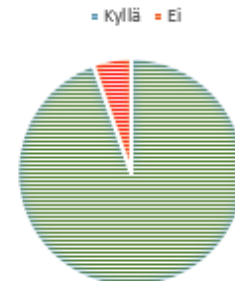
## SOPIVA MÄÄRÄ KYSYMYKSIÄ PER TASO



## PARAS TASO



## TULOSTEN VERTAILU

JÄLJELLÄ OLEVIENTEN KYSYMYSTEN  
MÄÄRÄN NÄKYMINEN PELIN  
EDETESSÄ

Kuva 10. Käyttäjätutkimuksen tuloksia.

Alkuperäisessä pelisovelluksessa pelaaja läpäisi tason vasta vastattuaan oikein kymme-  
neen kysymykseen ja pisteiden tallennus onnistui vasta, kun osion kaikki tasot oli pelattu  
läpi. Kyselyn tulosten perusteella päätettiin muuttaa pelin etenemislogiikkaa siten, että  
pelaaja vastaa jokaisella tasolla tasan kymmeneen kysymykseen ja saa yhden pisteen  
jokaisesta oikeasta vastauksesta. Väärästä vastauksesta ei rangaista. Kun pelitasoja on  
kymmenen, on pelistä saatava kokonaispistemäärä 100 pistettä. Myös tulosten tallen-  
nuksen mekaniikkaa päätettiin muuttaa lisäämällä pisteiden tallennusmahdollisuus jokai-  
sen tason loppuun. Näin oman tuloksen saa tuloslistalle, vaikka pelaamiseen varattu  
aika loppuisikin, ennen kuin kaikki tasot on ehditty pelata läpi.

### 5.3.2 Käytetyt tekniikat

Insinööriöprojektin lähtökohtana oli rakentaa olemassa oleva sovellus uudelleen siten,  
että se toimisi ongelmitta mahdollisimman monella eri päätelaitteella ja kaikissa käyte-

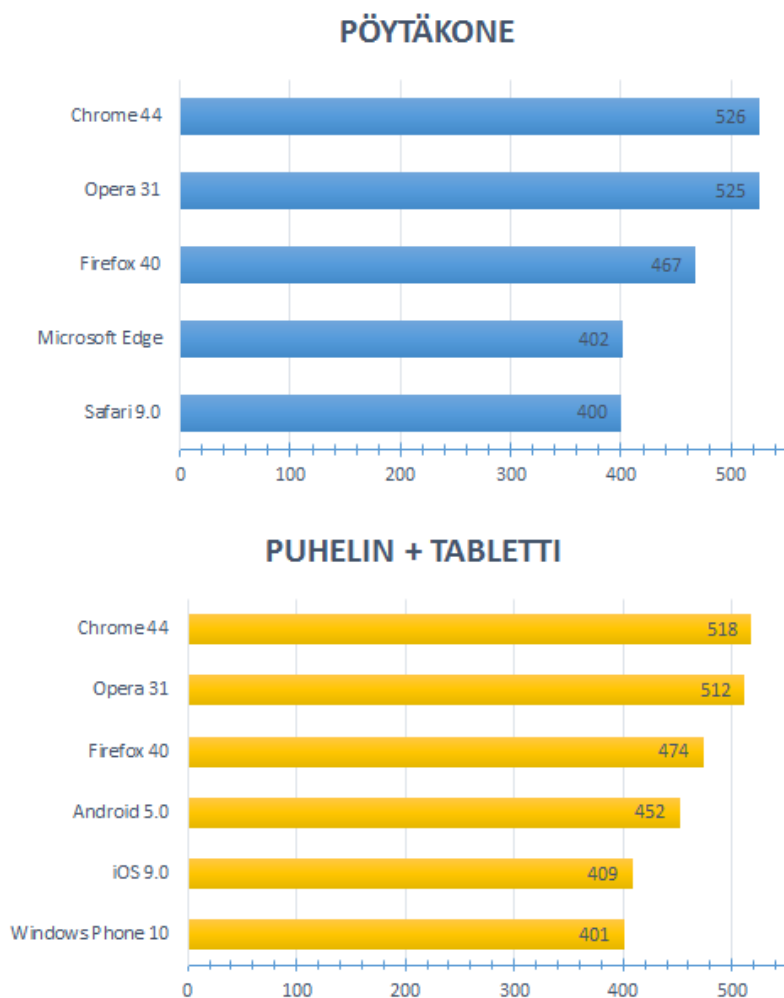
tyimmissä selaimissa. Koska projekti haluttiin pitää työmäärältään kohtuullisena ja lopullinen sovellus mahdollisimman kevyenä, ei ennen kehitysvaiheen alkamista lähdetty opiskelemaan mitään uutta standardia tai ohjelmointikieltä, vaan lopullinen sovellus toteutettiin tuttujen tekniikoiden pohjalta.

## HTML5

Sovelluksen runko päätettiin rakentaa käyttämällä HTML-kuvauskielen (Hypertext Markup Language) uusinta versiota, HTML5:tä. HTML5-standardi on tätä nykyä tuettu kaikissa suurimmissa selaimissa niin pöytä- kuin mobiililaitteissa, ja se mahdollistaa hyvinkin mutkikkaiden verkkosovellusten kehittämisen. Tästä syystä HTML5:tä pidetäänkin ihanteellisena valintana monialustaiselle pelinkehitykselle. Yksi HTML5:n parhaista ominaisuuksista on kuitenkin se, että se on suunniteltu käytettäväksi matalan suorituskyvyn laitteissa, kuten älypuhelimissa ja tableteissa. [20, s. 119–120.]

Kuvassa 11 on esitetty grafiikan avulla pöytä- ja mobiililaitteiden suosituimpien selaimien HTML5-valmiudet html5test.com-sivuston keräämiä tietoja mukaillen. Sivuston testin pisteytys perustuu siihen, mitkä HTML5:n ominaisuudet kyseessä olevassa selaimessa toimivat, sekä arvioon siitä, miten oleellisia nämä ominaisuudet ovat web-kehittäjien näkökulmasta ja kuinka vaikeaa niiden toteuttaminen on. Mikäli selain läpäisee kaikki testit, se saa maksimipistemäärän 555. [28.] Pelisovellusta rakennettaessa oleellisinta oli onnistunut kommunikointi käyttäjän ja palvelimen välillä sekä responsiiviset elementit, kuten kuvat ja tekstikentät. Testitulokset olivat hyviä jokaisella selaimella, joten HTML5:n käyttäminen pelisovelluksen rakentamisessa oli luonnollinen valinta. Kuvan 11 tiedot perustuvat vuoden 2015 lopun tilastoihin, joten tätä insinööriötä tehdessä uusia selainversioita on jo ilmestynyt ja testitulokset saattavat niillä olla jo huomattavasti parempia.





Kuva 11. Pöytä- ja mobiililaitteiden suosituimpien selainten HTML5-valmiudet pisteytettynä [muokattu lähteestä 28].

## CSS

Sovelluksen ulkoasuun liittyvät seikat hoidettiin CSS-tyyliohjeiston (Cascading Style Sheets) uusimman version CSS3:n avulla. Tyyliohjeistoon määriteltiin sovelluksen elementtien värit, leveydet ja pituudet sekä esimerkiksi fonttien koot. Tärkeimmät tyylimäärittelyt ovat kuitenkin niitä, jotka noudattavat responsiivisen suunnittelun periaatteita ja edesauttavat sovelluksen mukautumisen ketterästi usealle eri päätelaitteelle.

CSS-tyyliohjeistoon lisättiin pysäytyspisteet (engl. breakpoints) mediakyselyjen (engl. media query) avulla. Tällä tekniikalla määritellään samaan tyyliohjeistoon useita erillisiä tyylisääntöjä, jotka otetaan käyttöön, kun tietynlaiset ehdot toteutuvat. Tässä tapauksessa ohjeiston koodissa tarkistetaan käytettävän päätelaitteen ruutukoko, jotta sovellus

osaa ottaa käyttöön kyseiselle ruutukoolle optimoidun tyylisäännön, mikä parantaa selailukokemusta. Arkkitehtuurimme vuosikymmenet -sovelluksen tyyliohjeistoon määriteltiin kaksi erillistä tyyliä, eli karkeasti sanottuna oma tyyli isomille ja pienille laitteille. Tyyli vaihtuu päätelaitteen ruudun leveyden pienentyessä 375 pikseliin. Tämä pysäytyspiste valittiin kuvan 3 taulukon tulosten perusteella, ja 667 x 375 oli suurin puhelimen näytön resoluutio kärkikymmenikössä.

Sovelluksessa esiintyvien kuvien enimmäisleveydeksi määriteltiin sata prosenttia. Tällä tekniikalla kuva skaalautuu automaattisesti pienemmäksi, mikäli päätelaitteen ruudun leveys on pienempi kuin kuvan leveys, eikä kuvasta tarvitse tehdä erillistä versiota mobiililaitteella selaamista varten.

## **JavaScript**

Sovelluksen toiminallisuus toteutettiin JavaScript-skriptikielellä, josta käytän jatkossa lyhennettä JS. Skriptikieli on ohjelmointikieli, jota käytetään sellaisten ohjelmien luomisessa, jotka toimivat toisessa sovelluksessa. Niiden avulla verkkosivustolle on mahdollista lisätä esimerkiksi interaktiivisuutta. Skriptin voi sisällyttää HTML-sivustolle joko suoraan tai linkittämällä se erillisen tiedoston kautta. [29, s. 2.]

Skriptikieliä on paljon erilaisia, ja niiden pääasiallinen tehtävä on automatisoida ennalta määritettyjä toimia. Monet ohjelmistoalustat, kuten Flash, tarjoavat sisäisiä skriptikieliä, joilla voidaan suorittaa käyttäjän toimia. Esimerkiksi 3D-mallinnusohjelmat, kuten 3D Studio Max, Maya ja Blender, sisältävät kukin oman sisäisen skriptikielensä, jolla voi ohjelmoida niin sanottuja makroja, jotka automatisoivat ohjelmistoympäristössä saatavilla olevia toimintoja. Skriptaus on tällaisissa tapauksissa tapa, jolla säästetään aikaa ja vaivaa, kun käyttäjän on suoritettava toistuvia toimintoja. [20, s. 123.]

Ohjelmien sisäisten skriptikielien lisäksi on myös yleisiä skriptikieliä, kuten JS, jolla on tuki useassa suosituksessa pelimoottorissa, kuten Unity, UDK ja Corona SDK. Näissä pelimoottoreissa skriptikieltä käytetään esimerkiksi pelilogiikan ohjelmoimiseen, peliobjektien käytösmallien ja pelin käyttöliittymän luomiseen sekä minkä tahansa sellaisen aspektin luomiseen, joka liittyy jotenkin pelin pelattavuuden kehitykseen kyseessä olevan pelimoottorin sisällä. [20, s. 123.]

Arkkitehtuurimme vuosikymmenet -pelin uudistettu versio rakentuu neljästä erityyppisestä JS-tiedostokokonaisuudesta: pelifunktiot, kysymystaulukot, lisätietolaatikoiden funktiot ja JS-kirjastot. JS-kirjastoja lukuun ottamatta erilliset JS-tiedostot olisi ollut mahdollista yhdistää yhdeksi kokonaisuudeksi, mutta ne päätettiin jakaa osiin selkeyden kannalta. Tiedostoihin tehtävät mahdolliset korjaustyöt on myös helpompi tehdä, kun kokonaisuudet on paloiteltu.

### **JavaScript-kirjastot**

JS-kirjasto on valmiiksi kirjoitettu JS-koodikokonaisuus, jonka ohjelmoijat voivat liittää helposti oman sivurakenteensa yhteyteen ja käyttää sen sisältämiä toiminallisuuksia hyväkseen omassa projektissaan [30]. JS-kirjastot nopeuttavat JS-pohjaisten sovellusten kehitystyötä, kun kaikkea toiminallisuutta ei tarvitse rakentaa alusta asti itse, ja niiden tarkoitus onkin tehdä JS:n käyttämisestä helpompaa, sillä ne sisältävät monia yleisiä funktioita, joiden toteuttaminen vaatii useita rivejä JS-koodeja. Nämä pidemmät koodikokonaisuudet on muunnettu metodeiksi, joita kirjastoa käyttävä kehittäjä voi kutsua omassa sovelluksessaan parhaassa tapauksessa vain yhden koodirivin avulla. [31.]

Insinööriyössä käytettiin kahta avoimen lähdekoodin JS-koodikirjastoa: jQuery ja Bootbox.js. JQueryyn päädyttiin, koska se on maailman suosituin JS-kirjasto, jolla on monialustainen tuki, eli se toimii moitteetta useassa eri selaimessa. Se on myös mobiiliystävällinen ja helpottaa Ajaxin käyttämistä. Lisäksi kirjaston käyttämisestä oli jo olemassa jonkin verran kokemusta, eli kaikkea ei tarvinnut opetella alusta asti.

Bootbox.js pohjautuu suoraan Bootstrap-koodikirjastoon, joka on erityisen suosittu kehitysalusta mobile first -projekteille, ja sen tarkoitus on luoda yksinkertaisia ohjelmallisia valintaikkunoita. Lopulliseen peliversioon tuli useita suuria tekstikokonaisuuksia, jotka piti saada avautumaan uuteen näkymään ilman pelin keskeytymistä, sekä muutama varoitusikkuna. Muutaman JS-kirjaston vertailun jälkeen Bootbox.js tuntui luontevimmalta vaihtoehdolta edellä mainittujen ominaisuuksien rakentamiseen, sillä perimmäisenä tarkoituksena oli varmistaa lopullisen sovelluksen mobiiliystävällisyys ja -käytettävyys.

Kuvassa 12 on esitetty esimerkki pelin päälle avautuvasta lisätietolaatikosta ja sen koodirakenteesta. Bootbox.js-koodikirjaston avulla lisätietolaatikon ilmestymiseen tarvittavan funktion rakentaminen onnistuu muutamalla rivillä.



Kuva 12. Esimerkit pelisovelluksen lisätietolaatikosta ja lisätietolaatikon koodin rakenteesta.

## PHP ja Ajax

Pelin tulostusta varten oli luotava yksinkertainen MySQL-tietokanta, johon pelaajan nimerkki ja pistemäärä tallentuvat. Palvelinpuolen skriptikieli PHP:n (Hypertext Preprocessor) avulla oli mahdollista yhdistää tietokantaan ja hakea sinne tallennettu tieto näkyviin tulostilan muodossa. Kuvassa 13 on esitettyä edellä mainittu toimenpide koodin muodossa.

```

33 <?php
34 $servername = "PALVELIN";
35 $username = "TUNNUS";
36 $password = "SALASANA";
37 $dbname = "TIETOKANTA";
38 $conn = new mysqli($servername, $username, $password, $dbname);
39 if ($conn->connect_error) {
40     die("Connection failed: " . $conn->connect_error);
41 }
42 $sql = "SELECT name, score FROM scores ORDER BY score DESC LIMIT 10";
43 $result = $conn->query($sql);
44 $rank = 0;
45 $temp_score = 0;
46 echo "<div id='tulokset'><table id='scoreboard'>";
47 echo "<h1>Arkkitehtuurimme vuosikymmenet</h1>";
48 echo "<tr><th>Sijoitus</th><th>Nimi</th><th>Pisteet</th></tr>";
49 if ($result->num_rows > 0) {
50     while($row = $result->fetch_assoc()) {
51         if ($temp_score!=$row['score']) {
52             $rank++;
53         }
54         echo "<tr><td># " . $rank. "</td><td>" . $row["name"]. "</td><td>" . $row["score"]. "</td></tr>";
55     }
56 } else {
57     echo "Ei tallennettuja huipputuloksia";
58 }
59 echo "</table></div>";
60 $conn->close();
61 ?>

```

Kuva 13. Tulostilan luominen pistemäärän perusteella laskevaan järjestykseen PHP-koodin avulla.

PHP:ta käytettiin lisäksi tulosten tallentamiseen sovelluksesta tietokantaan. Kaikki pelaajan tulokset tallentuvat pelisovelluksessa kuitenkin ensin JS-tiedostoon määriteltyihin muuttujiin, joten muuttujien sisältämät arvot piti viedä PHP-tiedostoon Ajaxin avulla. Ajax (Asynchronous JavaScript And XML) on sekoitus eri verkkokehitystekniikoita, jotka mahdollistavat kommunikoinnin palvelimen kanssa ja verkkosivuston eri osioiden päivittämisen ilman, että koko sivusto täytyy ladata uudelleen [32]. Ajaxin avulla oli mahdollista pyytää pelaajaa syöttämään sovellukseen nimensä tai nimimerkkinsä, yhdistää yksilöity pelaaja tämän pistemäärään ja välittää saatu informaatio selaimen puolella PHP:lle ja tätä kautta välittää tieto edelleen tietokantaan.

## 5.4 Valmis peli

### 5.4.1 Toiminta

#### **Alkunäkymä**

Pelisovelluksen etusivulla pelaaja voi valita, aloittaako pelaamisen heti vai tarkasteleeko ensin tuloslistaa. Etusivulla on myös linkki arkkitehtuurimuseon omille verkkosivuille, mistä löytyy lisäinfoa pelistä. Kun peli alkaa, on pelaajan valittava, mitä tasoa hän haluaa pelata. Valittavissa on kymmenen eri tasoa, jotka voi pelata haluamassaan järjestyksessä. Mikäli pelaaja valitsee tason, jonka hän on jo pelannut läpi, näytölle ilmestyy varoitusteksti, jossa kerrotaan tason pisteiden nollaantuvan, mikäli taso aloitetaan uudelleen.

#### **Ennen tason aloittamista**

Pelitason näkymä koostuu kahdesta päällekkäisestä peliruudusta. Ennen kuin varsinainen pelitaso käynnistetään, ylempään peliruutuun ilmestyy pelattavaa aikakautta lyhyesti kuvaava teksti, jonka alla olevista näppäimistä pelaaja voi valita, aloittaako tason vai palaako tasojen valintanäkymään. Alempaan peliruutuun ilmestyy kuva aikakauden tyyliä ilmentävästä rakennuksesta. Kuvan alta löytyvät lisätietoa aikakaudesta -linkit. Linkkejä painamalla pelaaja saa näkyviin lisätietoa, josta on apua pelitason kysymyksiin vastatessa.

## Tasolla eteneminen

Kun pelaaja aloittaa tason, pelifunktio aloittaa tiedon hakemisen tason kysymystaulukosta. Yksittäinen kysymystaulukko koostuu 16–17 erilaisesta kysymyskokonaisuudesta ja algoritmista, jolla sekoitetaan kysymyskokonaisuudet ja niiden sisältämät vastausvaihtoehdot siten, että ne ilmestyvät pelissä satunnaisessa järjestyksessä. Kuvassa 14 on hahmoteltu kysymyskokonaisuuden rakenne.



Kuva 14. Kysymyskokonaisuuden rakenne.

Kysymyskokonaisuus sisältää kysymystekstin, kolme vastausvaihtoehtoa, oikean vastauksen tunnisteen ja selitetekstin. Peli etenee siten, että tason alussa peli arpoo kysymystaulukosta yhden kokonaisuuden, jonka kysymysteksti ja vastausvaihtoehdot ilmestyvät ylempään pelikenttään. Pelaaja valitsee yhden vaihtoehdon, minkä jälkeen hän painaa tarkistusnäppäintä. Pelifunktio tarkistaa kysymystaulukosta, onko kokonaisuuden vastaus oikein vai väärin. Lopputuloksesta riippuen pelaajan pistemäärä päivitetään pelifunktion sisältämään muuttuun, vastausvaihtoehdon näppäimen taustaväri muuttuu joko vihreäksi tai punaiseksi ja alempaan pelikenttään ilmestyy "oikein"- tai "väärin"-tekstillä alkava seliteteksti, jossa kerrotaan oikea vastaus.

Pelaamista jatkettaessa seliteteksti pyyhkiytyy pois, kysymyksen järjestysnumero kasvaa ja edellinen kysymys korvataan seuraavalla. Pelifunktio valitsee kysymystaulukosta sellaisen kokonaisuuden, joka ei ole vielä ilmestynyt pelissä. Näin vältetään tilanne,

jossa pelaaja joutuu vastaamaan samaan kysymykseen useaan otteeseen. Nämä toimenpiteet toistetaan, kunnes pelaaja on vastannut kymmeneen kysymykseen. Tämän jälkeen pelifunktiossa siirrytään lopetukseen.

### **Tason lopetus ja pisteiden tallennus**

Kun taso päättyy, pelaajalle kerrotaan ylemmässä pelikentässä hänen tällä tasolla ansaitsemansa pistemäärä ja kokonaispistemäärä. Alempaan pelikenttään ilmestyy kuva aikakauden esimerkkirakennuksesta. Pelaaja voi seuraavaksi joko jatkaa pelaamista tai tallentaa tuloksensa tuloslistalle. Tuloksen tallentamisen yhteydessä pelaajalta kysytään hänen nimeään tai nimimerkkiään. Pisteitä ei ole mahdollista tallentaa tuloslistalle ilman nimen syöttämistä tekstikenttään. Kun tulos on tallennettu, pelaaja ohjataan pelisovelluksen etusivulle.

#### **5.4.2 Jatkokehitys**

Lopullisen sovelluksen ensimmäinen versio valmistui elokuussa 2016, ja se lähetettiin heti valmistuttuaan tilaajaorganisaatiolle testaukseen. Organisaation edustajilla ei ollut lainkaan kokemusta verkkokehityksestä, joten he kykenivät arvioimaan vain pelin toiminnallisuutta ja ulkoisia seikkoja. Työstä ei saatu yksityiskohtaista palautetta, mutta tilaajaorganisaation edustajat olivat yleisellä tasolla tyytyväisiä valmiiseen sovellukseen eikä heiltä tullut negatiivista palautetta.

Sovelluksesta on tarkoitus tehdä myös ruotsinkielinen versio siten, että valmis suomenkielinen sovellus kloonataan ja kloonattuun versioon vaihdetaan ruotsinkieliset tekstit. Ruotsinkielisen version on tarkoitus valmistua syyskuussa 2016, minkä jälkeen valmis sovellus siirretään arkkitehtuurimuseon palvelimelle. Kun siirto on suoritettu, tilaajaorganisaatio huolehtii itse pelin mahdollisesta uudelleenjulkaisusta.

## **6 Yhteenveto**

Flash-ohjelmistoympäristössä tuotetut sovellukset eivät vastaa enää nykypäivän mobiiliyhteiskunnan tarpeisiin, joten verkkosovellusten kehitystyössä pyritään yhä enemmän käyttämään responsiivisia ja mobiiliystävällisiä kehitystekniikoita. Tämä mahdollistaa sen, että tuotetut sovellukset toimivat saumattomasti päätelaitteesta riippumatta. Flashin

suosio hiipuu vähitellen, sillä modernien web-standardien avulla voidaan rakentaa kevyitä ja monialustaisia verkkosovelluksia ilman, että toiminallisuuksia joudutaan karsimaan. Arkkitehtuurimme vuosikymmenet -pelisovelluksen päivittäminen nykyaikaan oli täten looginen ratkaisu.

Ennen varsinaisen uudistamisprojektin aloitusta tutkittiin pelisuunnittelun eri vaiheita sekä pelillisyyden merkitystä opetuksessa. Konsepti- ja käyttöliittymäsuunnittelun tärkeyttä ja keinoja analysoitiin uudistamisprojektin etenemisen sujuvoittamiseksi. Lisäksi pelisovelluksen käytettävyyttä rajoittavia tekijöitä tutkittiin ja saatua tietoa käytettiin hyväksi pelin rakennusvaiheessa.

Insinööriöprojektin alkaessa oli hieman epäselvää, kuinka suurta kasvojen kohotusta Arkkitehtuurimme vuosikymmenet -pelisovellus kaipasi. Alussa alkuperäistä pelisovellusta yritettiin jäljitellä siten, että mahdollisimman monta sen sisältämää elementtiä olisi päivitetty nykypäivään. Projektin edetessä selvisi kuitenkin, että pelin suunnittelu- ja toteutusvaiheessa sai käyttää omaa luovuuttaan hyvin vapaasti ja kaikkia alkuperäisen sovelluksen elementtejä ei ollut tarkoituskaan siirtää uuteen sovellukseen.

Tekstien pysyminen tekijänoikeusyistä ennallaan vaikutti suunnitteluvaiheessa lopullisen pelilogiikan muotoutumiseen. Koska tekstiä ja siten luettavaa oli paljon, piti pelikonaisuuden pysyä yksinkertaisena ja siten helposti pelattavana. Lopulta paras ratkaisu oli testauttaa kohderyhmällä muutamia eri pelin etenemisvaihtoehtoja ja rakentaa lopullinen sovellus saadun palautteen perusteella. Arkkitehtuurimuseon edustajat osallistui-  
vat kiitettävästi erilaisten vaihtoehtojen ideoimiseen ja organisoivat testisovelluksen ja-  
kamisen testiryhmälle.

Itse sovelluksen kehittäminen ja siihen liittyvät vaiheet sujuivat pääasiassa kivuttomasti. Koska insinööriötä ei ollut mahdollista tehdä kokopäiväisenä palkkatyönä, oli suurin haaste löytää tarpeeksi aikaa ja motivaatiota työskentelyyn. Alussa aikaa vei loogisesti toimivan pelisilmukan rakentaminen siten, että kysymykset ilmestyvät satunnaisessa jär-  
jestyksessä ja pisteet pysyvät tallennettuina funktion sisältämiin muuttujiin pelitasolta toi-  
selle siirryttäessä. Kun kaksi pelitasoa oli saatu toimimaan keskenään, pystyi niiden si-  
sältämää koodia käyttämään pohjana muita tasoja suunniteltaessa. Pelin tiedostojen si-  
sältämän koodimäärän kasvaessa oli oltava erityisen huolellinen, sillä pienikin virhe pe-  
lisilmukoiden koodissa saattoi aiheuttaa sovelluksen täydellisen toimimattomuuden.



Sovelluksen valmistumiselle ei ollut asetettu tarkkaa määräaikaa, joten sen rakentaminen eteni maltillisesti. Tarkoitus oli kuitenkin julkaista valmis sovellus alkusyksystä 2016, joten sovelluksen suomenkielisen version valmistuttua elokuussa 2016 voitaneen todeta, että tavoite saavutetaan. Sovellukseen on lisättävä vielä muutama elementti, kuten esimerkiksi ruotsinkielinen versio ja kielen valinta, ja se on siirrettävä tilaajaorganisaation palvelimelle, mutta nämä toimenpiteet on mahdollista toteuttaa nopeasti lyhyellä aikavälillä.

## Lähteet

- 1 Ozer, Jan & Richter, Stefan. Everything You Don't Know About Flash. Verkkodokumentti. HTMLGoodies. <<http://www.htmlgoodies.com/beyond/reference/article.php/3696596/Everything-You-Dont-Know-About-Flash.htm>>. Luettu 26.4.2016.
- 2 Aune, Sean P. 2010. A Brief History Of Flash. Verkkodokumentti. TechnoBuffalo. <<http://www.technobuffalo.com/2010/04/05/a-brief-history-of-flash>>. Luettu 26.4.2016.
- 3 Flash. Verkkodokumentti. TechTerms. <<http://techterms.com/definition/flash>>. Luettu 26.4.2016.
- 4 Warren, Christina. 2012. The Life, Death and Rebirth of Adobe Flash. Verkkodokumentti. Mashable. <<http://mashable.com/2012/11/19/history-of-flash/#kZ0b657i25qQ>>. Luettu 27.4.2016.
- 5 Mäki, Matias. 2007. Youtube streamataan Apple TV:isiin h.264-muodossa. Verkkodokumentti. Digitoday. <<http://www.digitoday.fi/viihde/2007/06/06/youtube-streamataan-apple-tvisiin-h264-muodossa/200713947/66>>. Luettu 1.9.2016.
- 6 Epstein, Zach. 2015. Adobe confirms major Flash vulnerability, and the only way to protect yourself is to uninstall Flash. Verkkodokumentti. BGR. <<http://bgr.com/2015/10/15/adobe-flash-player-security-vulnerability-warning/>>. Luettu 27.4.2016.
- 7 Hachman, Mark. 2015. Tested: How Flash destroys your browser's performance. Verkkodokumentti. PCWorld. <<http://www.pcworld.com/article/2960741/browsers/tested-how-flash-destroys-your-browsers-performance.html>>. Luettu 27.4.2016.
- 8 Jobs, Steve. 2010. Thoughts on Flash. Verkkodokumentti. Apple. <<http://www.apple.com/hotnews/thoughts-on-flash/>>. Luettu 18.4.2016.
- 9 Global Media Format Report. 2016. Verkkodokumentti. Encoding.com. <<https://www.encoding.com/files/2016-Global-Media-Formats-Report.pdf>>. Luettu 26.4.2016.
- 10 Koivisto, Matti. 2016. Yle Areenan Flash-toisto siirtyy historiaan. Verkkodokumentti. Yle. <[http://yle.fi/uutiset/yle\\_areenan\\_flash-toisto\\_siirtyy\\_historiaan/8669105](http://yle.fi/uutiset/yle_areenan_flash-toisto_siirtyy_historiaan/8669105)>. Luettu 26.4.2016.
- 11 Leider, Richard. 2015. YouTube now defaults to HTML5 <video>. Verkkodokumentti. YouTube. <[http://youtube-eng.blogspot.fi/2015/01/youtube-now-defaults-to-html5\\_27.html](http://youtube-eng.blogspot.fi/2015/01/youtube-now-defaults-to-html5_27.html)>. Luettu 26.4.2016.

- 12 Evans, David. 2015. Flash Will Soon Be Obsolete: It's Time for Agencies to Adapt. Verkkodokumentti. AdvertisingAge. <<http://adage.com/article/digital-next/flash-obsolete-time-agencies-adapt/298946/>>. Luettu 26.4.2016.
- 13 Bringing Better Performance to Rich Media on Chrome. 2015. Verkkodokumentti. Google. <<https://plus.google.com/+GoogleAds/posts/UQSBzWiMpEG?e>>. Luettu 26.4.2016.
- 14 Lee, Rich. 2015. Welcome Adobe Animate CC, a new era for Flash Professional. Verkkodokumentti. Adobe Animate Team Blog. <<http://blogs.adobe.com/animate/welcome-adobe-animate-cc-a-new-era-for-flash-professional>>. Luettu 26.4.2016.
- 15 Barrett, Brian. 2015. Adobe Flash is Dead in Name Only. Verkkodokumentti. Wired. <<http://www.wired.com/2015/12/adobe-flash-is-dead-in-name-only>>. Luettu 26.4.2016.
- 16 Gonzalez, Jason. 2013. Mobile First Design with HTML5 and CSS3. Birmingham: Packt Publishing.
- 17 Browser & Platform Market Share. 2016. Verkkodokumentti. W3Counter. <<https://www.w3counter.com/globalstats.php>>. Luettu 29.4.2016.
- 18 Wroblewski, Luke. 2011. Mobile First. New York: A Book Apart.
- 19 Lecky-Thompson, Guy W. 2008. Video Game Design Revealed. Course Technology.
- 20 Nolte, David & Scolastici, Claudio. 2013. Mobile Game Design Essentials. Birmingham: Pack Publishing.
- 21 Schell, Jesse. 2008. The Art of Game Design: A Book of Lenses. Burlington: Morgan Kaufmann Publishers.
- 22 Harviainen, Tuomas; Meriläinen, Mikko & Tossavainen, Tommi (toim.). 2013. Pelikasvattajan käsikirja. Helsinki: Mediakasvatus- ja kuvaohjelmakeskus.
- 23 Arkkitehtuurimuseon graafinen ohjeisto. 2011. Arkkitehtuurimuseon sisäinen ohjeisto. Dog Design.
- 24 Roboto Condensed. Verkkodokumentti. Google Fonts. <<https://fonts.google.com/specimen/Roboto+Condensed>>. Luettu 18.8.2016.
- 25 Making the web more beautiful, fast and open through great typography. Verkkodokumentti. Google Fonts. <<https://fonts.google.com/about>>. Luettu 18.8.2016.

- 26 Etusivu. 2012. Verkkodokumentti. Arkkitehtuurimuseo. <<http://mfa.fi/>>. Luettu 18.8.2016.
- 27 Bedard, Salla. 2016. Graafinen suunnittelija, Suomen arkkitehtuurimuseo, Helsinki. Sähköpostikeskustelu 15.8.2016.
- 28 Leenheer, Niels. About the test. Verkkodokumentti. HTML5test.com. <<https://html5test.com/about.html>>. Luettu 27.4.2016.
- 29 Negrino, Tom & Smith, Dori. 2007. Javascript - Tehokas hallinta. Helsinki: Readme.fi.
- 30 Rauch, Joseph. 2015. A Quick Guide to Learning JavaScript Libraries. Verkkodokumentti. SkilledUp. <<http://www.skilledup.com/articles/quick-guide-learning-javascript-libraries>>. Luettu 29.4.2016.
- 31 jQuery Introduction. Verkkodokumentti. W3Schools.com. <[http://www.w3schools.com/jquery/jquery\\_intro.asp](http://www.w3schools.com/jquery/jquery_intro.asp)>. Luettu 29.4.2016.
- 32 jQuery AJAX Methods. Verkkodokumentti. W3Schools.com. <[http://www.w3schools.com/jquery/jquery\\_ref\\_ajax.asp](http://www.w3schools.com/jquery/jquery_ref_ajax.asp)>. Luettu 18.8.2016.

## Liite 1. Arkkitehtuurimme vuosikymmenet -pelin käyttäjätutkimus

# Arkkitehtuurimme vuosikymmenet - peli

Seuraavaksi pääset vastaamaan muutamaa kysymykseen koskien äsken pelaamiasi pelitasoja. Vastaathan jokaiseen kysymykseen! Vastaaminen vie noin 5-10 minuuttia.

\*Pakollinen

### 1. Ikäsi? \*

Oma vastauksesi

---

### 2. Pelasit äsken läpi kolme eri tasoa. Jos pelissä on 10 tasoa, mikä olisi sopiva määrä kysymyksiä per taso? \*

- 1-5
- 5-10
- 10-15
- yli 15

### 3. Jos pelissä on 10 eri tasoa, miten toivoisit pelin etenevän? \*

- Kaikki tasot on pakko pelata läpi aikajärjestyksessä, minkä jälkeen tulokset tallentuvat
- Kaikki tasot on pakko pelata läpi, mutta järjestyksellä ei ole väliä. Tämän jälkeen tulokset tallentuvat.
- Kaikkia tasoja ei ole pakko pelata. Haluan pelata läpi yksittäisiä tasoja ja tulokset tallentuvat tasokohtaisesti.
- Haluan pelata yksittäisiä tasoja, tulosten ei tarvitse tallentua.

4. Haluatko pelin edetessä nähdä montako kysymystä on vielä jäljellä? \*

Kyllä

En

5. Äskeisistä tasoista pidin eniten... \*

1910-luku (SININEN)

1920-luku (PUNAINEN)

1930-luku (VIHREÄ)

5. Jos ennen tasoa olisi tarjolla infoteksti tason aikakaudesta, olisiko siitä apua? \*

Kyllä

Ei

6. Mitkä alla olevista ominaisuuksista lisäävät pelin mielekkyyttä? Voit valita useamman vaihtoehdon. \*

Aikarajoitteisuus

Vähenevät "elämät"

Väärästä vastauksesta ei sakoteta

Vaikka pisteet putoaisivat nolnaan, peli ei kuitenkaan pääty

Saan palkinnon läpäistyäni tason

7. Mitkä alla olevista ominaisuuksista eivät kannusta jatkamaan pelaamista? Voit valita useamman vaihtoehdon. \*

- Aikarajoitteisuus
- Vähenevät "elämät"
- Väärästä vastauksesta ei sakoteta
- Vaikka pisteet putoaisivat nolnaan, peli ei kuitenkaan pääty
- En saa palkintoa läpäistyäni tason

8. Kumpi alla olevista vaihtoehdoista olisi mielestäsi paras pisteytystapa Arkkitehtuurimme vuosikymmenet -pelissä? \*

- Vastaan kaikkiin kysymyksiin. Tällöin mahdolliselle tuloslistalle tallentuu kokonaispistemääräni.
- Vastaan kysymyksiin vain niin kauan kunnes tietty pistemäärä täyttyy. Tällöin mahdolliselle tuloslistalle tallentuu vastaamiseen käyttämäni aika.

9. Kumpi alla olevista vaihtoehdoista olisi mielestäsi paras tapa tulosten vertailuun? \*

- Pisteeni tallentuvat pelistä löytyvälle tuloslistalle
- Voin jakaa saamani pistemäärän sosiaaliseen mediaan ja haastaa ystäväni mukaan peliin

10. Mikä tekee opetuspelistä mielenkiintoisen?

Oma vastauksesi

---