

Jari Korpela
PALKITSEMISJÄRJESTELMÄN MÄÄRITTELY JA TOTEUTUS

Tietotekniikan koulutusohjelma
Ohjelmistotekniikan suuntautumisvaihtoehto
2011

PALKITSEMISJÄRJESTELMÄN MÄÄRITTELY JA TOTEUTUS

Korpela, Jari
Satakunnan ammattikorkeakoulu
Tietotekniikan koulutusohjelma
Maaliskuu 2011
Ohjaaja: Aarinen, Reino
Sivumäärä: 53
Liitteitä: 4

Asiasanat: Kyselytutkimus, Suunnittelu, PHP, SQL, Ohjelmisto

Opinnäytetyön aiheena oli suunnitella ja toteuttaa kannustinjärjestelmä Massidea.org-sivustolle. Työn tavoite oli kannustaa sivuston käyttäjäkuntaa pysymään aktivoituneena ja innokkaana luomaan uutta sisältöä sivustolle. Tavoitteeseen pyrittiin pääsemään tekemällä sivuston käytöstä hausempaa ja kilpailuhenkisempää, johon pyrittiin suunnitteleamalla ja luomalla pelinomainen palkitsemisjärjestelmä.

Työn alussa tutkittiin nykyistä kannustinjärjestelmää, jonka jälkeen tutkittiin uuden palkitsemisjärjestelmän vaatimuksia. Tutkimuksissa päädyttiin käyttämään pelimekaniikkoja hyödyntävää palkitsemisjärjestelmää. Suunnittelun jälkeen aloitettiin käyttäjäkyselylomakkeen laadinta, jolla oli tarkoitus selvittää käyttäjien mielipiteet tulevasta järjestelmästä. Lomakkeeseen ei työn aikana ehditty saamaan tarpeeksi vastauksia, jotta niitä olisi voitu lähteä analysoimaan.

Työn toisessa vaiheessa tutustuttiin itse sivustossa käytettäviin ohjelmiin ja tekniikoihin, koska niitä piti osata käyttää työssä. Näiden opetteluun meni hyvin paljon aikaa, ja osa tekniikoista vaihdettiin sivustolla toiseen kesken työn, jonka vuoksi työn konkreettista luomista ei päästy kunnolla tekemään. Tekniikoiden vaihdon jälkeen nämä uudet tekniikat täytyi opetella, jonka jälkeen pystyttiin luomaan tietokantasuunnitelma sekä ohjelmistosuunnitelma.

Työn viimeisessä vaiheessa oli tarkoitus toteuttaa laaditut suunnitelmat, mutta sivustoa ei saatu tähän mennessä siirrettyä riittävästi uudelle alustalle. Tästä johtuen suunnitelmat jäivät suurimmaksi osaksi vain teoreettiselle asteelle, eikä niitä ehditty panna kokonaan täytäntöön. Palkitsemisjärjestelmä saatiin kuitenkin suunniteltua halutunlaiseksi, vaikka konkreettista toteutusta ei päästy tekemään.

DEFINING AND IMPLEMENTING A ROYALTY PROGRAM

Korpela, Jari

Satakunnan ammattikorkeakoulu, Satakunta University of Applied Sciences

Degree Programme in Information Technology

March 2011

Supervisor: Aarinen, Reino

Number of pages: 53

Appendices: 4

Keywords: User Survey, Planning, PHP, SQL, Program

The purpose of this thesis was to define and implement a royalty program for the Massidea.org website. The objective of the work was to encourage the site's user base to stay and be active and enthusiastically create new content to the site. To reach the objective, it was decided to make using of the site more fun and more competitive, which was aimed at by designing and creating a game-like reward system.

At the beginning of the work the current royalty system was examined and after that, the requirements of the new reward system were studied and explored. In the studies it was decided to use a reward system that utilises game mechanics. After the system was planned, drawing up a user-questionnaire was started. It was intended to determine users' opinions on the planned system. The form did not receive enough answers during this work so the form results couldn't be analysed.

In the second stage, the programs and technologies used in the site had to be familiarised with, because they were known to be used in this work. A lot of time was used to learn all these and some technologies were changed to others in the middle of the work, which caused the rewarding system not being properly and concretely programmed. After the technologies were changed, the new technologies had to be learnt and after learning, plans for a database and software were made.

In the last stage, the plans that had been drawn up were supposed to be executed, but a sufficiently large part of the site had not yet been moved to the other platform. As a result, the plans were for the most part only in the theoretical stage, and they couldn't be fully implemented to the website. The reward system however was designed as it was desired to be, although the actual implementation was missing.

SISÄLLYS

1	JOHDANTO	8
1.1	Työn tausta	8
1.2	Työn tavoitteet ja rajoitukset.....	8
1.3	Työn rakenne.....	9
2	KÄYTETYT TEKNIIKAT JA OHJELMAT.....	10
2.1	Microsoft Word.....	10
2.2	PHP -kieli	10
2.3	ZEND Framework.....	10
2.4	CakePHP Framework.....	10
2.5	MVC arkkitehtuuri.....	11
2.6	JavaScript	11
2.7	jQuery JavaScript kirjasto	11
2.8	jQuery UI	12
2.9	CSS tyylitiedostot	12
2.10	Eclipse ohjelmointiympäristö	12
2.11	MySQL tietokanta.....	12
2.12	MySQL Workbench.....	13
2.13	UML mallinnus	13
2.14	astah* professional.....	13
2.15	Git versionhallinta.....	13
2.16	GitHub isännöintipalvelu	14
3	PALKITSEMISJÄRJESTELMÄN SUUNNITTELU	15
3.1	Nykyinen HoF järjestelmä	15
3.2	Uudessa järjestelmässä huomioon otavat asiat	15
3.3	Gamification.....	15
3.4	Pistejärjestelmien tutkiminen	16
3.4.1	Tasapainotettu pistejärjestelmä	16
3.4.2	Toimintopainotettu pistejärjestelmä	16
3.4.3	Progressiivinen pistejärjestelmä	17
3.5	Pistejärjestelmän valinta	17
3.6	Negatiivisten puolien neutraloiminen	18
3.7	Palkitseminen.....	19
3.7.1	Palkitsemisen tavat.....	19
3.7.2	Saavutustasot	19
3.7.3	Saavutusmerkit	19
3.7.4	Edistymispalkit	20

3.7.5	Virtuaalinen raha	20
3.7.6	Virtuaalilahjat	21
3.7.7	Sijoituslistat	21
3.8	Järjestelmän laajentaminen	22
4	KÄYTTÄJÄKYSELYLOMAKE	23
4.1	Lomakkeen tarkoitus	23
4.2	Kohderyhmä	23
4.3	Lomakkeen jäsentäminen	23
4.4	Alustavan version laadinta	24
4.4.1	Ensimmäinen osio	24
4.4.2	Toinen osio	24
4.4.3	Kolmas osio	25
4.4.4	Neljäs osio	26
4.4.5	Viides osio	27
4.5	Alustavan version testaus	27
4.5.1	Lukijatestaus	27
4.5.2	Alustava palaute	27
4.5.3	Yhteenvedo	27
4.6	Lopullisen version laadinta	28
4.6.1	Ensimmäinen osio	28
4.6.2	Toinen osio	28
4.6.3	Kolmas osio	28
4.6.4	Neljäs osio	30
4.6.5	Viides osio	31
4.7	Vastausten kerääminen	31
4.8	Tulosten analysointi	31
5	TIETOKANNAN SUUNNITTELU	33
5.1	Muutokset projektissa	33
5.2	Työvaiheet	33
5.3	Vaatimusten määrittely ja analyysi	34
5.4	Käsitteellinen mallintaminen	35
5.4.1	Mallintamisen vaiheet	35
5.4.2	Käyttäjä	35
5.4.3	Tapahtuma	36
5.4.4	Saavutus	36
5.4.5	Palkinto	36
5.5	Looginen suunnittelu	36
5.5.1	Huomioitavat asiat	36

5.5.2	Suunnitelma.....	37
5.5.3	Toimenpiteet.....	37
5.5.4	CakePHP nimeämiskäytännöt.....	38
5.5.5	EER diagrammin luonti.....	38
5.6	Tietokannan asennus ja testaus	39
6	OHJELMISTON SUUNNITTELU	40
6.1	Vaatimukset	40
6.2	Toiminnallisuus.....	40
6.3	Luokkasuunnittelu.....	41
6.4	Ulkoasun ja käyttöliittymän suunnittelu	42
6.4.1	Palkitsemisjärjestelmän osa-alueet.....	42
6.4.2	Tapahtumat.....	42
6.4.3	Saavutukset.....	43
6.4.4	Palkinnot.....	43
6.4.5	Virtuaalilahjat ja -raha.....	43
6.4.6	Sijoituslistat	44
7	PALKITSEMISJÄRJESTELMÄN TOTEUTTAMINEN	45
7.1	Lähtökohdat	45
7.2	Tietokannan toteuttaminen.....	45
7.2.1	MySQL kannan luonti.....	45
7.2.2	CakePHP Console	45
7.2.3	Mallien luonti	46
7.3	Ohjelmiston toteuttaminen.....	46
7.3.1	Sivuston tilanne	46
7.3.2	App Controller.....	46
7.3.3	Palkitsemisjärjestelmän ohjain	47
7.3.4	Ohjaimen toiminnot.....	47
7.3.5	Näkymät	48
8	YHTEENVETO	50
8.1	Työn onnistumisen arviointi	50
8.2	Tulevaisuuden pohdintaa	50
	LÄHTEET.....	52
	LIITTEET	

SYMBOLIT JA LYHENTEET

CakePHP	PHP ohjelmistokehys. Esitelty luvussa 2.4.
CSS	Tyyliohjeiden laji eritoten WWW dokumenteille. (Cascading Style Sheets). Esitelty luvussa 2.9.
DOM	Asiakirjan oliomallinne. DOM on selaimesta ja kielestä riippumaton yleissopimus sille miten olioita esitetään ja käytetään HTML, XHTML ja XML asiakirjoissa. (Document Object Model)
EER	Korkean tason käsitteellinen tietomalli, joka on laajennus alkupe- räiseen ER malliin. (Enhanced Entity-Relationship)
Flash	Sivustolla käytetty tekniikka tapahtumien ilmoittamiseen.
Flag	Sivustolla käytetty järjestelmä roskaviestien merkitsemiseen.
Gamification	Pelimekaniikkojen hyödyntämistä kuvaava termi.
HoF	Kunniataulu (Hall of Fame)
Kojelauta	Graafinen käyttöliittymä, joka muistuttaa auton kojelautaa (Dash- board)
MVC	Ohjelmistoarkkitehtuurityyli. Malli-Näkymä-Ohjain. (Model-View- Controller). Esitelty luvussa 2.5.
MySQL	C ja C++ kieleen pohjautuva RDBMS. Esitelty luvussa 2.11.
OOP	Oliokeskeinen ohjelmointi (Object Oriented Programming)
PHP	Ohjelmointikieli. (Hypertext Preprocessor). Esitelty luvussa 2.2.
RDBMS	Relaatiotietokantojen hallintajärjestelmä (Relational DataBase Ma- nagement system)
RSS	Päivittyvä digitaalinen tietosyöte (Really Simple Syndication)
SQL	Relaatiotietokantojen kyselykieli (Structured Query Language)
UML	Graafinen mallinnuskieli. Esitelty luvussa 2.13.
ZEND	PHP ohjelmistokehys. Esitelty luvussa 2.3.

1 JOHDANTO

1.1 Työn tausta

Massidea.org on uudenlainen sosiaalisen median toimintaperiaatetta hyödyntävä avoin innovaatioyhteisö, jossa käyttäjät voivat jakaa muiden käyttäjien kanssa ideoita, havaitsemiaan ongelmia ja tulevaisuuden näkymiä. Massidea.org pyrkii parantamaan käyttäjiensä luovuutta yhdistelmällä älykkäästi erilaisia sisältöjä, ja sisältöjä tuottaneet käyttäjät. Yhteisön keskeisenä tavoitteena on kannustaa käyttäjiä aktiiviseen kanssakäymiseen keskenään. Sivustolla on jo toiminnassa HoF (Hall of Fame) tyylinen kannustinjärjestelmä, joka halutaan uudistaa. Uudistuksessa halutaan luoda palkitsemisjärjestelmä, joka edesauttaisi yhteisön keskeisen tavoitteen toteutumista.

1.2 Työn tavoitteet ja rajoitukset

Tavoitteena on saada selville millainen palkitsemisjärjestelmä sopii sivustolle parhaiten, sekä luoda kyseinen järjestelmä. Järjestelmän myötä nykyiset sivuston käyttäjät on tarkoitus saada aktiivisesti tuottamaan uutta sisältöä sivustolle. Työssä täytyy ottaa huomioon myös nykyisten käyttäjien mielipiteet tulevasta järjestelmästä, ja tavoitteena on säilyttää täysi käyttäjätyytyväisyys uutta järjestelmää kohtaan.

Sivustossa käytetään ZEND ohjelmistokehystä, jonka ohjelmointikieli on PHP (Hypertext Preprocessor). Laaditun palkitsemisjärjestelmän tulee olla myös toteutettu sivustossa olevalla ohjelmistokehyksellä. Ohjelmistokehys vaatii, että sivusto on toteutettu MVC (Model-View-Controller) arkkitehtuuria noudattaen, jonka vuoksi myös uuden järjestelmän täytyy noudattaa MVC arkkitehtuuria. Sivustolla on käytössä MySQL RDBMS (Relational DataBase Management System) ja järjestelmän täytyy olla yhteensopiva nykyisen tietokantarakenteen kanssa. Helpon yhteensopi- vuus saadaan toteuttamalla palkitsemisjärjestelmä myös MySQL:llä.

1.3 Työn rakenne

Toisessa luvussa käyn ensin läpi työssä käytettäviä tekniikoita ja ohjelmia joihin minun on täytynyt tässä työssä tutustua. Kolmannessa luvussa tutkin palkitsemisjärjestelmän erilaisten toteutusten soveltuvuutta sivustoon ja mitä tämän tulisi pitää sisällään. Suunnittelen myös millainen uuden palkitsemisjärjestelmän tulee olla, jotta se saadaan liitettyä nykyiseen järjestelmään. Neljännessä luvussa suunnittelen käyttäjäkyselylomakkeen. Käyn siinä läpi mitä siinä halutaan kysyä, mitä se pitää sisällään ja miten tuloksia kerätään. Viidennessä luvussa suunnittelen palkitsemisjärjestelmän tietokannan ja käyn läpi suunnittelussa huomioonotettavat asiat. Kuudennessä luvussa suunnittelen palkitsemisjärjestelmän ohjelmallisen osan ja ulkoasut. Seitsemännessä luvussa lähdän toteuttamaan edellä tehtyjä suunnitelmia. Kahdeksannessa luvussa käyn läpi omia ajatuksiani tekemästäni työstä ja siitä miten työ on onnistunut, sekä pohdin hieman palkitsemisjärjestelmän tulevaisuutta.

2 KÄYTETYT TEKNIIKAT JA OHJELMAT

2.1 Microsoft Word

Microsoft Word on Microsoftin kehittämä tekstinkäsittelyohjelmisto. Tässä työssä käytettiin Wordin 2007 versiota käyttäjäkyselylomakkeen laadintaan, koska se oli itselleni tutuin ja hyväksi havaituin tekstinkäsittelyohjelmisto.

2.2 PHP -kieli

PHP on laajasti käytetty vapaaseen lähdekoodiin perustuva yleiseen käyttöön tarkoitettu skriptikieli, joka soveltuu eritoten web kehitysympäristöihin ja sitä voidaan käyttää HTML koodin seassa. Kaikki PHP koodi suoritetaan serverin puolella. Sillä voidaan generoida koodia, joka sitten näytetään sivuston lataajalle. (PHP 2011).

2.3 ZEND Framework

Zend ohjelmistokehys on avoimen lähdekoodin olio-ohjelmointiin perustuva verkko-sovellus PHP 5 alustalle. Zendä sanotaan usein komponenttikirjastoksi, koska sillä on monia irrallisia komponentteja, joita voi käyttää enemmän tai vähemmän itsenäisesti. Zendissä on myös kehittynyt MVC ohjelmistoarkkitehtuurirakenne, jonka avulla voidaan rakentaa tarvittava perusrakenne web sivustolle. (ZEND 2011). Zend ohjelmistokehys on hyvin suuri ohjelmistokehys ja sen versio 2 oli käytössä sivustolla ennen kuin tämä työ aloitettiin.

2.4 CakePHP -framework

CakePHP on nopeaan kehittämiseen tarkoitettu ohjelmistokehys PHP:lle. CakePHP käyttää ZENDin tavoin MVC ohjelmistoarkkitehtuurirakennetta. Nopean kehittymisen mahdollisuus johtuu CakePHP:ssa paljon käytetyistä PHP:n ”magic” toiminnoista. Monet web sivustoilla yleisesti käytetyt kokonaisuudet ja pienetkin asiat on tehty nopeammin ohjelmoitaviksi. (CakePHP 2011). CakePHP on paljon yksinkertaisempi

ja nopeampi kehittää kuin ZEND, mutta ei tarjoa yhtä kehittyneitä valmiita komponentteja. Työn puolesta välissä sivustolla päätettiin vaihtaa ZEND ohjelmistokehitys CakePHP 1.3 versioon, koska sivuston kehittämisestä haluttiin nopeampaa.

2.5 MVC arkkitehtuuri

MVC-arkkitehtuuri eli malli–näkyvä–ohjain on ohjelmistoarkkitehtuurityyli, jonka tarkoituksena on erottaa sivuston mallinnus, esittäminen ja toiminnot toisistaan. MVC-arkkitehtuurissa ohjelma jaetaan kolmeen osaan: malliin, näkymään ja ohjaimen. Malli huolehtii järjestelmän kaikenlaisesta tiedon käsittelystä. Näkyvä käsittelee miten tieto näytetään käyttöliittymässä. Ohjain eli kontrolleri vastaanottaa käyttäjältä tulevat käskyt, sekä muuttaa mallia ja näkymää vastauksena niihin. (MVC-Arkkitehtuuri 2010).

2.6 JavaScript

JavaScript on pääasiassa web-ympäristössä käytettävä komentosarjakieli, jonka tärkein sovellus on mahdollisuus lisätä Web-sivuille dynaamista toiminnallisuutta (JavaScript 2011). JavaScriptiä käytetään sivustolla ja työssä tehtävässä palkitsemisjärjestelmässä jQueryä ja jQuery UI:ta käyttäen.

2.7 jQuery JavaScript kirjasto

jQuery on selaimesta riippumaton, avoimen lähdekoodin, JavaScript kirjasto, joka on suunniteltu yksinkertaistamaan JavaScript skriptausta. Sen avulla voidaan helposti muun muassa valita DOM (Document Object Model) elementtejä, luoda animaatioita ja hallita tapahtumia. (jQuery. 2011). jQueryä käytetään sivustolla paljon. Sitä käytetään myös suunniteltavassa palkitsemisjärjestelmässä muun muassa animaatioiden ja ajaxin muodossa.

2.8 jQuery UI

jQuery UI on vapaan lähdekoodin käyttöliittymälaajennus jQuery kirjastoon (jQuery UI. 2011). Se sisältää valmiita efektejä ja hyödyllisiä liitännäisiä, jonka vuoksi sitä käytetään tuomaan sivustoon interaktiivisuutta. Sitä aiotaan käyttää myös uudessa palkitsemisjärjestelmässä samasta syystä.

2.9 CSS tyylitiedostot

CSS, eli porrastetut tyylit, määrittelevät miten WWW-dokumentit näytetään. CSS:ssä dokumentille voi määritellä useita tyyliloheja, jotka yhdistyvät säännöksi. (Cascading Style Sheets. 2011). Sivuston tyylittelyssä tullaan käyttämään CSS versiota 2, joka on yhteensopiva vanhemmillakin selaimilla.

2.10 Eclipse ohjelmointiympäristö

Eclipse on avoimeen lähdekoodiin perustuva ohjelmointiympäristö, joka tukee monia ohjelmointikieliä, muun muassa PHP. Siitä on olemassa erilaisia versioita eri käyttötarkoituksiin. Eclipse tukee myös CSS ja JavaScriptiä. (Eclipse. 2011). Eclipseä käytetään työssä sivuston ohjelmoinnissa, koska siinä tarvitaan kaikkia edellä mainittuja.

2.11 MySQL -tietokanta

MySQL on suosittu SQL-tietokannan hallintajärjestelmä, ja se on hyvin suosittu web-palveluiden tietokantana. Sen päälle rakennettava ohjelmalogiikka on tehty usein PHP ohjelmointikielellä, mutta sitä voi käyttää hyvin monella eri ohjelmointikielellä. (MySQL. 2011.). Kuten aiemmin mainittiinkin, MySQL:ää käytetään sivustolla, ja sitä kannattaa käyttää myös tehtävässä palkitsemisjärjestelmässä.

2.12 MySQL Workbench

MySQL Workbench on MySQL:n kehittämä alustariippumaton, visuaalinen tietokannasuunnittelutyökalu (MySQL Workbench. 2009). Tässä työssä ohjelmalla kyetään tekemään käsitteellinen mallintaminen ja looginen suunnittelu. Se tarjoaa myös valmiiksi generoidut tietokannan luomiseen tarvittavat koodit, jotka se saa generoitua tehdystä loogisesti mallista.

2.13 UML mallinnus

UML-mallinnus (Unified Modeling Language) on Object Management Groupin (OMG) vuonna 1997 standardoima graafinen mallinnuskieli, joka sisältää 13 erilaista kaaviota. Kaavioista kuudella kuvataan rakennetta, kolmella käyttäytymistä ja neljällä vuorovaikutusta. Se on alun perin kehitetty järjestelmä- ja ohjelmistokehitystä varten, ja se syntyi yhdistämällä kolme johtavaa oliomallinnustekniikkaa: 1) OMT, 2) Booch ja 3) OOSE. (UML-Mallinnus 2010). Palkitsemisjärjestelmän suunnittelussa ei varsinaisesti käytetä UML:ää, vaan vuokaaviota, joka on lähes identtinen UML:n aktiviteettikaavion kanssa, mutta yksinkertaisempi. Kaavion tarkoitus on esittää, miten ohjelma etenee kun se ajetaan.

2.14 astah* professional

astah* on UML kehitysympäristö, joka tarjoaa sen lisäksi myös monia muita kaavio-suunnittelutyökaluja esimerkiksi vuokaavioihin, ajatuskarttoihin ja ER kaavioihin. Sitä käytetään tässä työssä vuokaavion tekemiseen ohjelmiston suunnittelun yhteydessä.

2.15 Git versionhallinta

Git on ilmainen avoimen lähdekoodin versionhallintaohjelmisto, joka on suunniteltu toimimaan hajautetusti ja mahdollisimman tehokkaasti. Se kehitettiin alun perin Li-

nux-ytimen kehitystiimiä varten. (Git. 2011). Gitia käytetään sivustolla ohjelmiston päivittämiseen.

2.16 GitHub isännöintipalvelu

GitHub on internetissä toimiva isännöintipalvelu ohjelmistokehitysprojekteille, jotka käyttävät Git versionhallintaa. GitHub on kirjoitettu Ruby on Railsilla. Se tarjoaa sekä kaupallisia tilejä kaupallisiin projekteihin, että ilmaisia tilejä vapaan lähdekoodin projekteille. (GitHub. 2011). Githubia käytetään sivustossa Gitin ohella. Kaikki päivitykset on nähtävissä GitHubista.

3 PALKITSEMISJÄRJESTELMÄN SUUNNITTELU

3.1 Nykyinen HoF järjestelmä

Sivustolla nykyisin toimiva HoF järjestelmä laskee käyttäjille, ryhmille, kaupungeille ja maille pisteitä viiden eri kategorian mukaan: 1) sisältöjen lukumäärän 2) lukijoiden lukumäärän 3) kommenttien lukumäärän 4) luettujen sisältöjen määrän 5) saatu- jen arvostelujen summan (positiiviset ja negatiiviset äänet yhteenlaskettu) mukaan. Kaikki muut, paitsi käyttäjät, saavat pisteitä myös jäsenten lukumäärän mukaan. Kaikista näistä kategorioista saa yhden pisteen per toiminto.

3.2 Uudessa järjestelmässä huomioitavat asiat

Uutta palkitsemisjärjestelmää valitessa on hyvä ottaa huomioon, että tarkoitus on uudistaa sivustossa oleva HoF järjestelmä, eikä luoda sen perustoinnista jotain täysin poikkeavaa. Palkitsemisjärjestelmän tulee luoda käyttäjille tarve käyttää sivustoa monipuolisesti, jolloin palkitsemisen kriteereiden täytyy olla tasapainossa. Uudessa järjestelmässä täytyy ensin valita käytettävä pistejärjestelmä. Valittavassa pistejärjestelmässä olisi hyvä ottaa jollain muotoa huomioon jo saavutetut pisteet, etteivät sivuston käyttäjät närkästyisi HoF pistejärjestelmässä saavutettujen pisteiden menetyksestä. Järjestelmä tulee kehittää sellaiseksi, jotta sen hyväksikäyttämiseksi ei ole järkeviä syitä, eikä sen hyväksikäyttö ole palkitsevaa. Tämä on tärkeää, jotta sivuston väärinkäytöksistä vältyttäisiin mahdollisimman hyvin. Sivuston ennustetaan myös kasvavan ja laajenevan vielä kauan, jolloin järjestelmän tulee olla järkevästi laajennettavissa.

3.3 Gamification

Hyvä tapa saada käyttäjille tarve käyttämään sivustoa, on lisätä sivustolle Gamification tekniikkaa. Gamification tarkoittaa pelimekaniikkojen käyttämistä hyödyntäen ihmisten psykologista taipumusta pelaamiseen. Termi on alalla uusi, jonka vuoksi sille ei ole vielä suomenkielistä vastinetta, eikä koko ilmiölle ole tehty vielä mitään virallista objektiivista tutkimusta.

Gamification toimii kahdella tavalla: 1) Se tekee palvelun käyttämisestä kiinnostavampaa, sekä 2) kannustaa käyttäjiä halutunlaiseen käyttäytymiseen. Tämän avulla voidaan ihmiset saada tekemään jotakin, mitä he eivät normaalisti viitsisi tehdä. (Gamification 2011.)

Massidea.org -sivustoon sopivia gamification tekniikoita ovat sijoituslistat, saavutusmerkit ja saavutustasot, koska ne tukevat jo sivustolla olevaa HoF järjestelmää. Uusina elementteinä sivulle sopisivat edistymispalkit, virtuaalinen raha ja lahjajärjestelmä. Edistymispalkkeja voidaan käyttää saavutusten etenemisen näyttämiseksi. Virtuaalista rahaa käyttäjät saisivat saavutuksista ja niillä käyttäjä voi ostaa mahdollisuuden muokata enemmän esimerkiksi profiilisivuansa. Lahjajärjestelmän avulla käyttäjät voisivat antaa toisilleen virtuaalisia lahjoja, mikä edistäisi käyttäjien välistä kommunikaatiota ja yhteistoimintaa.

3.4 Pistejärjestelmien tutkiminen

3.4.1 Tasapainotettu pistejärjestelmä

Tämä on käytössä vanhassa HoF järjestelmässä. Järjestelmä antaa kaikille toiminnolle yhtä paljon pisteitä. Järjestelmä ei anna eri toiminnolle erilaisia painoarvoja, vaan antaa vain lähinnä tilastollista tietoa eri käyttäjien toiminnosta. Tämä pistejärjestelmä ei luo käyttäjille haluttua tarvetta tehdä sivustolla vaikeampia tai monipuolisempia toimintoja, jos helpoimmistakin toiminnosta saa yhtä paljon pisteitä.

3.4.2 Toimintopainotettu pistejärjestelmä

Painotettu pistejärjestelmä tarkoittaisi, että eri toiminnosta saisi erisuuruisen määrän pisteitä. Tämä pistemäärä riippuisi toiminnon vaativuudesta. Pisteiden painottamisella sivusto kykenee osoittamaan käyttäjilleen millaisia toimintoja sivuston haltijat arvostavat eniten ja millaisia käyttäjien toimintoja sinne halutaan saada. Tällä järjes-

telmällä kyetään myös osoittamaan paremmin sivuston keskeinen tavoite ja ohjaamaan käyttäjiä siihen.

Vaarana tässä järjestelmässä on, että pisteiden painotus ei toteudu halutulla tavalla. Tästä voi seurata, että käyttäjät eivät tee sivustolla muita toimintoja kuin niitä, joista saa parhaiten pisteitä. Vaarana on myös, että jos joillekin asioille ei anneta tarpeeksi pisteitä, niitä ei viitsitä tehdä, koska niiden tekeminen tuntuisi turhalta.

3.4.3 Progressiivinen pistejärjestelmä

Progressiivinen pistejärjestelmä tarkoittaisi, että toiminnoista saatujen pisteiden määrä riippuisi tehtyjen toimintojen määrästä. Tällä järjestelmällä sivusto kykenee osoittamaan toimintojen määrän arvokkuutta. Tämä myös kannustaisi aktiivisuuteen, koska aktiivisempi käyttäjä saisi samoista toiminnoista enemmän pisteitä kuin epäaktiiviset.

Mikäli pisteitä ei kuitenkaan ole painotettu ollenkaan, on tässä myös sama vaara kuin tasapainotetussa pistejärjestelmässä. Haittavaikutuksena voi myös olla, että jos pisteiden saanti alkaa liian alhaalta, tai jos progressio on liian jyrkkää, uusilla käyttäjillä ei riitä motivaatio pisteiden keräämiseen. Tarkoituksena on kuitenkin saada myös uusia käyttäjiä palveluun pysyvästi.

3.5 Pistejärjestelmän valinta

Mikään edellä tutkimistani pistejärjestelmävaihtoehdoista ei tuntunut täysin sopivalta sivuston käyttöön. Vanhalla tasapainotetulla pistejärjestelmällä on turha yrittää jatkaa, koska sillä ei saada mitenkään aikaan haluttuja asioita. Toimintapainotettu järjestelmä oli kolmesta tutkitusta paras. Progressiivisessa järjestelmässä oli hyvänä puolena mahdollinen aktiivisuuden parantumisen mahdollisuus.

Päädyin siihen tulokseen, että on paras valita toimintapainotettu pistejärjestelmä, jossa on progressiivisen pistejärjestelmän elementtejä mukana saavutustasojen muodossa. Tällä tavalla on mahdollista saada molempien järjestelmien hyvät puolet käyt-

töön. Huomioitavaksi seikoiksi pitää ottaa huomioon järjestelmien tuomat negatiiviset puolet, ja keksiä ratkaisuja niiden neutraloimiseksi.

3.6 Negatiivisten puolien neutraloiminen

Kuten aiemmin todettiin, järjestelmän tulee olla sellainen, ettei sen hyväksikäyttäminen ole järkevää eikä palkitsevaa. Tämä on välttämätöntä, jotta järjestelmään kohdistuvat mahdolliset hyökkäykset saadaan pidettyä kurissa, eikä niistä koidu kenellekään normaalikäyttäjälle mielipahaa ja täten karkota heitä pois sivustolta. Normaalikäyttäjien piste-eduntavoittelut voivat johtaa siihen, että he yrittävät saada pisteitä mahdollisimman vähällä vaivalla. Tämä voi johtaa häiriökäyttäytymiseen siten, että käyttäjät voivat innostua kirjoitteluun. Lyhyitä spam, eli roskaviestejä, jotta he saisivat esimerkiksi tietyn määrän kommentteja tehtyä, joista he sitten saisivat jonkin palkinnon. Koko järjestelmältä menee uskottavuus, jos siihen ei voida luottaa, ja sille ei ole asetettu selkeitä sääntöjä, jotka estävät tällaisen toiminnan.

Sivuston tulee pidättää oikeus siihen, että se saa tehdä käyttäjien pisteille mitä se haluaa. Mikäli häiriökäyttäytymistä tapahtuu, voidaan häirikkö poistaa palvelusta pisteistä ja saavutuksista välittämättä. Mikäli normaalikäyttäjä tekee häirikköviestin tai viestejä, täytyy sivuston valvojan puuttua siihen poistamalla kyseiset häiriöviestit.

Pelkkä poisto ei kuitenkaan voi olla riittävä, sillä se ei sisällä rangaistusta. Ilman rangaistusta käyttäjät voisivat kokeilla onneaan sillä, ettei heidän häiriöviestejään huomattaisi. Sivustolla on onneksi olemassa niin sanottu ”flagaus” järjestelmä, jolla kaikki käyttäjät voivat ilmoittaa häiriökäyttäytymisestä sivustolla. Sivuston valvojat näkevät nämä ilmoitetut asiat, ja voivat siten puuttua niihin. Paras keino rangaista normaalikäyttäjää häirikkökäytöstapauksessa on poistaa häneltä tietty määrä hänen ansaitsemiaan pisteitä, jotta hän ymmärtäisi häiriökäytöstapauksissa menettävänsä, eikä varsinkaan ansaitsevansa pisteitä.

3.7 Palkitseminen

3.7.1 Palkitsemisen tavat

Luvussa 3.3 käytiin lävitse erilaisia palkitsemisen tapoja: 1) saavutusmerkit, 2) saavutustasot, 3) edistymispalkit, 4) virtuaalinen raha, 5) virtuaalilahjat ja 6) sijoituslisät. Näillä palkitsemisen tavoilla pyritään saavuttamaan asetetut aktiivisuustavoitteet. Jokainen palkitsemisen tapa tulee kuitenkin suunnitella erikseen, jotta tavoitteissa onnistutaan mahdollisimman tehokkaasti. Palkinnoista ei saa tulla sellaisia, joita käyttäjä ei arvosta, eikä mitään sellaista joka on täysin ylitse muiden tehden muista palkinnoista sellaisia, joita ei viitsitä edes tavoitella.

3.7.2 Saavutustasot

Käyttäjä voi saada erilaisista sivuston toiminnoista saavutuksia. Näillä saavutuksilla voi olla erilaisia tasoja, ja korkeammalla tasolla vaaditaan enemmän työtä seuraavalle saavutustasolle pääsemiseen. Tasojen määrä vaihtelee toiminnosta riippuen. Tasoja ei saa olla liikaa, tai tasojen saavuttaminen ei enää tunnu käyttäjistä hyvältä. Tasolla eteneminen ei saa tapahtua liian tiheästi, tai ne menettävät merkitystään. Saavutustasolle pääsemisenkin tulee olla saavutus itsessään. Jos tasot saadaan mitoitettua oikein, on oletettavissa, että käyttäjissä saadaan herätettyä saavutuksen tunteita, jotka innostaisivat sivuston aktiivisempaan käyttämiseen.

3.7.3 Saavutusmerkit

Saavutusmerkkejä ansaitaan saavutustasoilla etenemisestä. Merkkien määrä kertoo käyttäjän kokemuksesta sivuston käyttäjänä. Laaja merkkivalikoima tarkoittaisi, että käyttäjä käyttää sivustoa hyvin monipuolisesti ja on monella osaa aktiivisesti mukana sivuston toiminnassa. Saavutusmerkkien tarkoitus on luoda käyttäjille ylpeyden tunteita, ja sen toivotaan aktivoivan kunnianhimoisia käyttäjiä. Merkkien tulee olla mahdollisimman uniikin näköisiä, eikä kaikkia saa saada liian helposti, jotta ne tuntuvat arvokkaimmilla saavutusta. Merkin saamisen pitää tuntua palkinnolta ja kiitok-

selta, joista käyttäjän toivotaan saavan arvostuksen tunnetta, joka taas innostaisi tätä käyttämään sivustoa jatkossakin.

3.7.4 Edistymispalkit

Käyttäjälle on tärkeää pystyä näyttämään, että hän edistyy sivustolla. Tätä varten on hyvä käyttää edistymispalkkeja, joilla voidaan näyttää saavutustasojen välisten toimintojen määrä ja käyttäjän eteneminen tällä välillä kohti seuraavaa tasoa. Tämän tarkoitus on herättää käyttäjässä halua saada edistymispalkki valmiiksi, ja siinä samalla käyttäjä käyttäisi sivustoa aktiivisesti. Kun käyttäjä saa edistymispalkin kokonaan valmiiksi, on sen tarkoitus herättää samanlaisia arvostuksen tunteita hänessä kuin saavutusmerkkienkin kohdalla. Käyttäjää voidaan palkita edistymispalkin täyttämisestä esimerkiksi seuraavassa kohdassa käsiteltävällä virtuaalirahalla.

3.7.5 Virtuaalinen raha

Virtuaalisella rahalla voidaan antaa käyttäjälle tunne, että hän olisi ikään kuin rikas, ja pystyy tekemään sivustolla vähän kaikenlaista. Raha on siitä hyvä väline, että sille voidaan kehittää monia erilaisia käyttötarkoituksia, ja se jättää tilaa tulevaisuudinkin laajennuksille.

Tässä suunnitelmassa virtuaalirahalla voidaan ostaa virtuaalilahjoja, jotka käsitellään seuraavassa luvussa. Sillä voidaan myös ostaa sivustolta auki erilaisia lukittuja toimintoja, kuten esimerkiksi oman profiilisivun tyylimuokkaus.

Virtuaalirahan saanti tulee rajoittaa tarkoin, tai se menettää kokonaan arvonsa. Sen täytyy olla yksilöllistä jokaiselle käyttäjälle, joka tarkoittaa, että raha itsessään ei voi vaihtaa omistajaa. Mikäli raha toimisi tässä tapauksessa vaihdon välineenä kuten oikea raha, voi se johtaa järjestelmän hyväksikäyttämiseen. Tästä syystä ainoa mahdollinen keino palkita käyttäjää virtuaalirahalla, on hänen edetessä saavutustasoissa. Tasoa on vain tietty määrä, joten rahaakin voi saada vain tiettyyn pisteeseen asti, tai kunnes järjestelmää tullaan laajentamaan.

3.7.6 Virtuaalilahjat

Virtuaalilahjoilla on tarkoitus parantaa käyttäjien välistä kommunikaatiota. Tässä palvelussa virtuaalilahjat tarkoittaisivat lähinnä arvostuksen antamista toiselle käyttäjälle. Sen sijaan, että lahjoja annettaisiin suoraan käyttäjille, ne annetaan käyttäjien tekemille sisällöille.

Nykyisessä sivuston mallissa käyttäjät voivat antaa vain plus äänen jollekin käyttäjän tekemälle sisällölle. Jos käyttäjä lukee jonkun sisällön, jonka kanssa hän on hyvin paljon samaa mieltä, hän haluaisi varmasti näyttää sen paremminkin kuin vain yhdellä plus äänellä. Virtuaalilahjat ovat ratkaisu tähän. Käyttäjä voi antaa monia erilaisia lahjoja sisällölle.

Virtuaalilahjoja voi ostaa virtuaalirahalla. Lahjat ovat kertaostoksia, joka tarkoittaa, että lahjoja on monia erilaisia, mutta jokainen erilainen lahja tarvitsee ostaa vain kerran. Kun lahja on kerran ostettu, voi sitä käyttää loputtomasti. Ehtona täytyy kuitenkin olla jonkinlainen aikarajoitus, joka riippuu lahjan arvosta, jotta niitä ei voida antaa ylenpalttisesti tai ne menettävät arvostusta. Sama sisältö voi saada samalta henkilöltä vain yhden samanlaisen lahjan.

3.7.7 Sijoituslistat

Kuten aiemmassakin HoF järjestelmässä, täytyy nykyisessä järjestelmässä olla myös jonkinlaiset sijoituslistat, jotta tietynlaisen kilpailuvietin omaavat käyttäjät voivat kisata keskenään paremmuudesta. Käyttäjät, jotka pitävät kisaamisesta, pysyvät myös aktiivisena, koska he eivät halua menettää sijoituksiaan.

Sijoituslistat täytyy kuitenkin tehdä kokonaan uusiksi, sillä vanhoja HoF järjestelmän listoja ei voida käyttää. Sijoituslistat on hyvä tehdä kokonaisuudessaan ansaitun virtuaalirahan mukaan, jolloin ne perustuvat sivuston monipuoliseen ja aktiiviseen käyttämiseen, eivätkä vain jonkin tietyn toimintatyyppin arvostukseen. Toinen vaihtoehto ja mahdollisuus on luoda lista yhteenlasketuista pisteistä, jotka on saatu kaikista toiminnoista.

3.8 Järjestelmän laajentaminen

Palkitsemisjärjestelmän tulevia osa-alueita täytyy olla helppo laajentaa. Tästä syystä sen täytyy olla hyvin mukautuva, eikä siihen saa asettaa liikaa ehdottomia ehtoja.

Pistejärjestelmän pisteytystä täytyy pystyä muuttamaan myöhemmin, mikäli sivustoon tulee uusia asioita, joista kannattaa antaa pisteitä. Samalla pitää olla mahdollisuus myös palkintojen lisäämiseen ja muokkaamiseen mahdollisimman vaivatta.

4 KÄYTTÄJÄKYSELYLOMAKE

4.1 Lomakkeen tarkoitus

Työn keskeisenä tavoitteena on säilyttää täysi käyttäjätyytyväisyys uutta palkitsemisjärjestelmää koskien. Käyttäjäkyselyllä pyritään keräämään käyttäjien mielipiteitä tulevasta palkitsemisjärjestelmästä. Kyselyllä saadaan käyttäjät tuntemaan, että he pääsevät vaikuttamaan asioihin ja olemaan osa sivustoa, ja että heidän mielipiteensä on sivustolle erittäin tärkeä. Lomake tehdään käyttäen Microsoft Word 2007 tekstinkäsittelyohjelmaa.

4.2 Kohderyhmä

Vaikkakin sivusto on kansainvälinen, on Suomi eniten edustettuna. Suurin osa sivuston käyttäjistä on ammattikorkeakoulussa opiskelevia opiskelijoita ja vanhempia ihmisiä. Näistä syistä lomake voidaan suunnitella suomenkielisenä ja sen lukijoilta voidaan odottaa tietyn asteen teknistä tietämystä ja ymmärtämistä.

4.3 Lomakkeen jäsentäminen

Lomakkeen ensimmäisessä osiossa esitellään sivusto, ja mistä sivustossa on kyse. Tällöin uudemmatkin sivuston käyttäjät voivat täyttää lomakkeen ja lomaketta voidaan jakaa myös sellaiselle, joka ei sivustoa ennen ole käyttänyt.

Lomakkeen toisessa osiossa kerrotaan miksi lomake on tehty ja mihin sivusto tarvitsee käyttäjien apua. Tässä kohtaa täytyy myös esitellä lomakkeen sisältöä käyttäjälle.

Lomakkeen kolmannessa osiossa aloitetaan itse kysely. Palkitsemisjärjestelmässä palkitsemisen kriteereiden tasapainotus on erittäin tärkeää, jotta palkitsemisjärjestelmä onnistuu. Onnistumisen takaamiseksi, lomakkeen tässä kohdassa käydään läpi käyttäjien mielipiteitä palkitsemisen kriteereistä ja toimintojen arvokkuudesta.

Lomakkeen neljännessä osiossa käydään läpi käyttäjien mielipiteitä saavutuksista. Kerrotaan saavutuksien jaottelumisesta ja yleistä tietoa niiden saavuttamisesta ja palkinnoista, sekä kysytään erilaisten saavutusten mielekkyyttä käyttäjänäkökulmasta.

Lomakkeen viidennessä ja viimeisessä osiossa kysytään käyttäjältä yleisesti hänen mielipiteitään koko järjestelmästä ja kirjoittamisen motivaatioista. Tarkoitus on hahmottaa millaisia luonteenpiirteitä vastaajalla on.

4.4 Alustavan version laadinta

4.4.1 Ensimmäinen osio

Sivuston esittelyssä käydään ensimmäiseksi läpi mikä Massidea.org -sivusto on, esitellään sivuston toimintaa yleisellä tasolla, ja mitkä ovat sen keskeiset tavoitteet. Toisessa kohdassa kerrotaan Massidea.org:n toimintaperiaatteet käyttäjätunnuksen luonnista sisällön tuottamiseen, eli kuinka Massidea.org toimii käytännössä. Kolmannessa kohdassa esitellään ketkä ovat Massidea.org:n takana, sekä ohjeistetaan käyttäjää tutustumaan halutessaan sivustoon ennen kyselyyn vastaamista.

4.4.2 Toinen osio

Lomakkeen esittelyssä käydään ensimmäiseksi läpi kyselyn taustat, eli kerrotaan, että sivustolle ollaan luomassa palkitsemisjärjestelmää, jonka avulla aktiivisia käyttäjiä voidaan palkita. Seuraavaksi kerrotaan miltä kohderyhmältä näitä vastauksia halutaan saada, minkä tyyppisistä toiminnoista ja kuinka paljon pisteitä tulisi kertyä, ja minkälaiset palkinnot kannustaisivat opiskelijoita aktiivisesti osallistumaan Massidea.org -sivuston toimintaan. Selitetään sivuston keskeiset toiminnot, joiden ympärillä sivusto toimii ja joihin kysely keskittyy. Lopuksi kerrotaan millä asteikolla pisteitä voidaan antaa ja ehdotetaan asteikkoa 0-1000, sekä kerrotaan muita pisteiden antamiseen liittyviä sääntöjä.

4.4.3 Kolmas osio

Varsinaiset pistekyselyt ovat kolmannessa osiossa, joka on jaettu sivuston toimintojen mukaan seuraavasti: 1) sisällön tuottamiseen liittyvät toiminnot, 2) sisällön ja muiden arvostelut, 3) ryhmän luonti ja ryhmän linkitys toiseen ryhmään 4) kampanjan luonti ja kampanjan linkitys toiseen kampanjaan, 5) linkittäminen, 6) kommentointi, 7) muut toiminnot, ja viimeisenä 8) negatiiviset toiminnot.

Sisällön tuottamisen toimintoihin liittyviä kysymyksiä on 3. Niissä 1) esitellään luotavan sisällön kolme mahdollista tyyppiä: haaste, idea ja tulevaisuuden visio, 2) kerrotaan sivuston haluavan sisällön vastaavan mikä, miksi, kuka, milloin ja missä kysymyksiin, ja 3) kerrotaan sivuston monikielisuudesta ja kielenkääntämisominaisuuksista. Kysymyksiä yhteydessä käyttäjältä kysytään, paljonko hän antaisi itselleen pisteitä kyseisistä toiminnoista.

Sisällön ja muiden arvosteluiden toimintoihin liittyviä kysymyksiä on 2. Niissä 1) kysytään, paljonko käyttäjä antaisi itselleen pisteitä erilaisten sisällönarvosteluskenaarioiden mukaan, 2) ja pitäisikö arvosteluiden olla samanarvoisia käyttäjään katsomatta.

Ryhmän luonti ja ryhmän linkitys toiseen ryhmään toimintoihin liittyviä kysymyksiä on 4. Niissä 1) esitellään kolme eri ryhmätyyppiä: yleinen, suljettu, ja yksityinen, 2) niiden ääretön luontimahdollisuus, 3) ryhmään liittyminen ja 4) ryhmien keskeinen linkitys. Kysymyksiä yhteydessä käyttäjältä kysytään, paljonko hän antaisi itselleen pisteitä kyseisistä toiminnoista.

Kampanjan luonti ja kampanjan linkitys toiseen kampanjaan toimintoihin liittyviä kysymyksiä on 3. Niissä 1) esitellään kampanjan luominen, 2) sisällön lisääminen kampanjaan ja kysytään, paljonko käyttäjä antaisi itselleen pisteitä näistä toiminnoista, ja lopuksi 3) tiedustellaan ovatko kaikki kampanjaan lisätyt sisällöt käyttäjän mielestä samanarvoisia.

Linkittäminen ja sen toimintoihin liittyviä kysymyksiä on 3. Ensimmäiseksi käydään läpi kaikki mahdolliset linkittämisen skenaariot ja kysytään käyttäjältä, paljonko hän

antaisi kustakin skenaariosta itselleen pisteitä. Toisena kysytään minkälaista linkittämisen ajatusmallia käyttäjä pitää hyvänä, ja ovatko linkitykset eriarvoisia keskenään riippuen siitä millainen ja mikä linkitetään millaiseen ja mihinkin. Kolmantena kysytään pitäisikö ryhmän kerätä erikseen ryhmälle pisteitä ja miten ne tulisi jakaa.

Kommentointi toimintoihin liittyviä kysymyksiä on 3. Ensimmäiseksi esitellään kaikki kommentoinnin skenaariot ja kysytään, paljonko käyttäjä antaisi itselleen pisteitä eri skenaariovaihtoehtojen kommentoinneista. Toiseksi ja kolmanneksi kysytään, onko sillä väliä kommentoiko toisen kommentointia, kommentoiko omaa sisältöä ja kommentoiko samaa asiaa useamman kerran.

Muihin toimintoihin liittyviä kysymyksiä on 5. Niissä 1) esitellään suosikkisisältöjen lisääminen, 2) omien ja suosikkisisältöjen seuraaminen, 3) yksityisviestien lähettäminen ja vastaanottaminen ja 4) RSS (Really Simple Syndication) -syötteiden lisääminen. Kysymyksien yhteydessä käyttäjältä kysytään, paljonko hän antaisi itselleen pisteitä näistä toiminnoista.

Negatiiviset toiminnot toimintoihin liittyviä kysymyksiä on 1. Siinä kysytään, paljonko käyttäjän itsensä pitäisi menettää pisteitä, mikäli hän lähettäisi asiatonta sisältöä palvelussa.

4.4.4 Neljäs osio

Käyttäjälle selitetään suunniteltua saavutusjärjestelmää saavutustasoinen, sekä kerrotaan saavutusten olevan jaettu kolmeen eri kategoriaan: 1) jatkuvat aktiiviset saavutukset, 2) aikaan perustuvat saavutukset ja 3) yksinkertaiset saavutukset. Käyttäjältä kysellään yleistä mielipidettä saavutusjärjestelmästä ja millaisia tasoja hän pitää mielekkäänä. Lopuksi kysytään yksityiskohtaisesti erilaisten saavutusten mielekkyyttä.

4.4.5 Viides osio

Viimeisessä osiossa kysytään käyttäjän yleisiä kirjoittamisen motivaatioita, sekä mitä mieltä hän on ylipäättään koko järjestelmästä. Lopuksi annetaan vielä mahdollisuus antaa vapaata palautetta. Lomakkeen pituudeksi tuli 20 sivua.

4.5 Alustavan version testaus

4.5.1 Lukijatestaus

Kun alustava versio saatiin laadittua, tulostettiin siitä kopioita. Näitä kopioita annettiin luettavaksi opiskelijoille, jotka antoivat siitä palautetta. Tällä haluttiin varmistaa ymmärtävätkö käyttäjät lomakkeen, ja onko siinä epäselvyyksiä, ennen kuin varsinainen lomake julkaistaan.

4.5.2 Alustava palaute

Palautetta tuli eritoten siitä, että lomake oli aivan liian pitkä, eikä sitä jaksaisi kukaan täyttää. Myös pisteiden antaminen oli usean mielestä vaikeaa, koska skaalaus oli niin suuri. Käyttäjä unohti helposti, montako pistettä oli antanut millekin, eikä pystynyt helposti päässään suhteuttamaan pisteitä. Eritoten olisi toivottu valmiita vastausvaihtoehtoja. Lomake oli käyttäjien mielestä kuitenkin ymmärrettävästi laadittu.

4.5.3 Yhteenveto

Palautteesta viisastuneena koko lomake suunniteltiin uudestaan, mutta rakenne pyrittiin pitämään suurin piirtein samanlaisena. Lomaketta täytyi lyhentää ainakin puoleen alustavasta versiosta ja pisteiden antamisjärjestelmää muuttaa.

4.6 Lopullisen version laadinta

4.6.1 Ensimmäinen osio

Alustavan version ensimmäisessä osiossa ei havaittu mitään puutteita. Käyttäjien mielestä se antoi hyvän kuvan siitä, miten sivusto toimii ja mistä sivustossa on kyse.

4.6.2 Toinen osio

Palautteen antajat pitivät hyvänä asiana, että heille selitettiin, mistä tässä lomakkeessa onkaan kyse. Osion alku toimi hyvin, mutta sivuston keskeisten toimintojen selittäminen, joiden ympärille kysely keskittyi, ei toiminut toivotulla tavalla. Nämä toiminnot kirjoitettiin uudestaan ja kyselyä pyrittiin keskittämään paremmin näihin toimintoihin. Pisteiden antamiselle toivottiin vaihtoehtoja, mutta niitä ei voida antaa olematta johdattelematta käyttäjää tietynlaisiin vastauksiin. Pisteiden skaalaus oli alustavassa versiossa 0-1000 ja käyttäjien oli vaikea hahmottaa niin suurella piste skaalalla toimintojen pisteiden suhteuttaminen toisiinsa. Tästä syystä pisteskaalaus päätettiin laskea 0-100 asteikolle pisteiden suhteuttamisen helpottamiseksi.

4.6.3 Kolmas osio

Kolmannen osion rakenne päätettiin rakentaa uudelleen, jotta kyselystä saatiin toimintokeskitetympi ja lyhyempi. Itse kyselyosa jaettiin osiin, jotka olivat: 1) sisällön tuottamiseen liittyvät toiminnot, 2) ryhmän luonti, 3) kampanjan luonti 4) linkittäminen, 5) kommentin antaminen, 6) lahjoituspisteet, 7) pisteiden menettäminen ja 8) muut toiminnot.

Sisällön tuottamiseen liittyvät toiminnot osio pysyi lähes samana alustavaan versioon nähden. Ainoana muutoksena päätettiin lisätä sivuston monikielisyydestä kertovaan osioon kysely, paljonko käyttäjä antaisi itselleen pisteitä kääntäessään oman tai toisen sisällön toiselle kielelle. Tällä haluttiin katsoa kuinka arvokkaana käyttäjät pitävät monikielistä sisältöä, ja kuinka arvokkaana prosessina he kääntämistä pitivät.

Ryhmän luonti toimintoihin liittyvät kysymykset siirrettiin toiseksi, ja alustavassa versiossa olevat arvostelut siirrettiin viimeiseen muut toiminnot osioon. Ryhmän luontiin liittyvät kysymykset pysyivät lähes samoina kuin alustavassa versiossa, mutta siitä poistettiin linkitykseen liittyvät kysymykset, joita tarkastellaan erikseen myöhemmin linkittäminen osiossa.

Kampanjan luonti toimintoihin liittyvät kysymykset pysyivät samoina lukuun ottamatta sisällön lisäämisen eriarvoisuutta koskevaa kysymystä, joka päätettiin poistaa, koska lomaketta piti saada lyhyemmäksi ja sen ei todettu antavan mitään sellaista tietoa, jota olisi voitu pitää ehdottoman tärkeänä.

Linkittämiseen liittyvien toimintojen kysymyksiä karsittiin kovalla kädellä, ja näihin liittyviä kysymyksiä pyrittiin keskittämään ja yksinkertaistamaan paremmin. Linkittämisen toiminnoissa käydään läpi vain neljä linkityksen skenaariota, jotka liittyvät sisältöön, ryhmään, kampanjaan ja profiilisivuun. Linkittämisen ajatusmallikysymystä hiottiin myös tarkemmaksi, ja kysyttiin molempiin ajatusmallivaihtoehtoihin käyttäjän pisteytystä. Muut linkitykseen liittyvät kysymykset poistettiin kokonaan, sillä niiden ei katsottu antavan tarpeeksi tärkeää tietoa.

Komentointiin liittyviä kysymyksiä päätettiin yksinkertaistaa rankasti. Käyttäjälle kerrottiin vain kolme pääasiallisinta kommentoinnin skenaariota, joita käyttäjää pyydettiin sitten pisteyttämään. Muut kommentointiin liittyvät kysymykset poistettiin, sillä niiden ei katsottu antavan tarpeeksi tärkeää tietoa.

Lahjoituspisteet osiota ei ollut alustavassa versiossa, mutta se päätettiin ottaa mukaan lopulliseen versioon, koska käyttäjälle haluttiin kertoa myös lahjamahdollisuuksista, joita ei alustavassa versiossa kerrottu ollenkaan. Käyttäjälle ei kuitenkaan kerrota koko lahjoitussysteemin toimintaa, ettei lomake mene liian monimutkaiseksi. Käyttäjälle kerrotaan vain periaatteellinen toimintatapa ja kysytään, paljonko toiminnoista saatavia pisteitä tarvitaan lahjapisteiden saamiseen.

Negatiiviset toiminnot osio korvattiin pisteiden menettäminen osiolla, jossa kerrottiin samat asiat kuin alustavassa versiossakin, mutta kysymysten määrää kavennettiin huomattavasti. Todettiin, ettei ollut mielekästä kysyä pisteiden menettämisestä liian

yksityiskohtaisesti ja että on riittävä kysyä niistä vain pääpiirteittäin. Alustavasta versiosta puuttui myös kokonaan selitys siitä, kuka näistä pisteiden menetyksistä päättää. Myös sivustolla toimivasta ”flagaus”, eli liputussysteemistä ei kerrottu mitään. Flag systeemillä tavalliset käyttäjät voivat ilmoittaa (liputtaa) epäasiallisista sisällöistä sivuston valvojille, jotka sitten voivat poistaa nämä sisällöt. Käyttäjiltä kysyttiin, paljonko ilmoittamisesta pitäisi antaa pisteitä.

Viimeiseksi osioksi jätettiin muut toiminnot, jossa kerrottiin lähes samoja asioita kuin alustavassa versiossa, mutta lisäten vielä muutamia kysymyksiä. RSS syötteet jätettiin pois, koska ne eivät olleet niin relevantteja kyselyn kannalta. Uusiksi kysymyksiksi lisättiin kysymykset siitä, paljonko käyttäjä antaisi itselleen pisteitä sivuston sisältöjen, ryhmien, kampanjoiden ja profiilisivujen katsomisesta ja paljonko käyttäjä antaisi itselleen pisteitä siitä, että joku muu katsoo näitä edellä mainittuja sivuja. Tähän liittyen kysyttiin myös arvostaako käyttäjä enemmän pisteitä, joita hän saa muiden tekemistä toiminnoista, vai pisteitä, joita hän saa omalla toiminnallaan. Osion loppuun siirrettiin myös alustavan version viimeisestä viidennessä osiossa laadittu vapaa palaute kohta.

4.6.4 Neljäs osio

Neljäs osio kirjoitetaan kokonaan uusiksi. Alustavassa versiossa kerrottiin saavutuksista, mutta kolmannen osion kysymykset ajavat lähes saman asian ja niistä saatua tietoa voidaan käyttää myös saavutuksissa. Sen sijaan palkitsemisjärjestelmästä ja itse motivaatioista ei ollut tarpeeksi kysymyksiä. Siksi neljännessä osiossa kysytäänkin vain motivaatiokysymyksiä, eli syitä miksi käyttäjät käyttävät tai käyttäisivät sivustoa. Näihin kysymyksiin käyttäjä vastaa käyttäen valmiiksi annettua 7 asteista Likertin asteikkoa (Likertin asteikko. 2006. William M.K. Trochim). Tästä on apua palkitsemisen tarkemmassa suunnittelussa, kun tiedetään minkälaiset palkitsemisen skenaariot motivoivat käyttäjiä eniten. Samalla saatiin lomaketta paljon lyhyemmäksi.

4.6.5 Viides osio

Viidennessä eli viimeisessä osiossa kysytään vähän perustietoja kyselyyn vastaajasta. Tätä tietoa voidaan käyttää tulosten vertailussa. Annetaan käyttäjälle myös mahdollisuus tulla kasvokkain haastatteluun, mikäli hän haluaa. Lopullisen lomakkeen pituudeksi tuli 10 sivua (Liitteeseen jaoteltuna 12:lle sivulle tyylien vuoksi) [LIITE 1].

4.7 Vastausten kerääminen

Tuloksia kerättiin pääsääntöisesti sähköisesti, koska lomake oli pitkä ja vastauksia haluttiin saada maanlaajuisesti. Pidin myös mahdollisena paperiversion käyttämistä, mikäli sille olisi jonkinlainen erityinen mahdollisuus ja tarve tullut.

Sähköisessä keräämisessä käytettiin SAMK:n (Satakunnan Ammattikorkeakoulu) käyttämää E-lomaketta. E-lomake on ohjelma, jolla voi tehdä verkkolomakkeita. Ohjelma luo lomakkeelle oman internet-osoitteen, jossa sen voi käydä täyttämässä. Ohjelmassa on toiminnot vastausten lajitteluun ja käsittelyyn. (E-lomake. Eduix Oy).

E-lomakkeella luotiin lopullisen version käyttäjäkyselylomake, ja se saatiin vastaanmaan sellaista kuin haluttiinkin. Sähköisen lomakkeen ansiosta tulokset olivat myös helposti suoraan saatavilla, ilman että niitä piti erikseen ottaa ylös.

4.8 Tulosten analysointi

Tuloksia analysoidaan sillä periaatteella, että yritetään saada suhteutettu pistekeskisarvo sivuston pääasiallisille toiminnoille. Näitä myös pyritään suhteuttamaan sivustolla toimivan HoF järjestelmän pisteisiin. Toiseksi halutaan tietää motivaatioiden jakautuminen, sillä siitä voidaan päätellä parhaat palkinnot ja palkitsemisjärjestelmät sivustolle. Näiden molempien tuloksia voidaan myös verrata eri ikäryhmittäin ja lääneittäin, josta voidaan huomata eri-ikäisten ihmisten suhtautuminen tämäntapaiseen järjestelmään ja se, otetaanko se erilaisella alueella paljonkin eri tavalla vastaan.

Itse tuloksia ei ehditty saamaan tarpeeksi, jotta niitä olisi voitu vielä käyttää konkreettisena tietona toimintojen arvon asettamisessa. Se ei kuitenkaan estä järjestelmän suunnittelua tai toteuttamista, sillä niitä asioita voidaan säätää myöhemmin halutunlaisiksi.

5 TIETOKANNAN SUUNNITTELU

5.1 Muutokset projektissa

Massidea.org -sivustossa päätettiin kirjoittaa koko sivusto uudelleen CakePHP -ohjelmistokehyksellä ZEND ohjelmistokehyksen sijaan, ja samalla tietokantaan päätettiin ottaa käyttöön ohjelmoinnista tuttu periyttämismallin mukainen järjestelmä tukemaan sivustolla esiintyvää polymorfismia. Vaihto tehtiin, koska CakePHP:ta pidettiin nopeampana alustana kehittää. Sivustossa tapahtuvien muutosten johdosta tietokanta täytyykin suunnitella nyt CakePHP -ohjelmistokehyksen ja uuden tietokannan kanssa yhteensopivaksi.

5.2 Työvaiheet

Tietokantasuunnittelu on monivaiheinen prosessi. Se on jaettu neljään eri vaiheeseen, joiden avulla tietokanta saadaan lopulta valmiiksi.

- 1) Vaatimusten määrittelyssä ja analyysissä selvitetään järjestelmän vaatimia ominaisuuksia erilaisin keinoin, kuten haastattelemalla, kirjallisella materiaalilla ja keskustelemalla tässä tapauksessa sivuston ylläpitäjien kanssa.
- 2) Käsitteellisessä mallintamisessa laaditaan käsitekaavio, jolla kuvataan tietokannan sisältö ja rakenne. Tämä rakennetaan vaatimusten määrittelyjen pohjalta.
- 3) Tietokannan loogisessa suunnitteluvaiheessa edellä tehdyn käsitekaavan pohjalta laaditaan sisällöstä ja rakenteesta relaatiokaavio. Tämän kaavion pohjalta voidaan suoraan rakentaa suunniteltu tietokanta.
- 4) Toteutusvaiheessa toteutetaan edellisessä osiossa laadittu kanta. Kantaa testataan pienellä käyttöliittymällä ja sillä testataan toteutuksen toimivuus. (Tietokannan suunnittelu. 2004. Antti Ekonoja, Tommi Lahtonen, Jukka Mäntylä).

Näiden edellä mainittujen prosessien lisäksi täytyy ottaa huomioon, että kyseessä ei ole tavallinen relaatiotietokanta, sillä osassa tietokantaa käytetään periyttämismallia. Syy tällaisen kannan käyttämiseen johtuu siitä, että MySQL -relaatiotietokanta ei tue polymorfismia. Suunnitelmaa ei siis voida tehdä aivan perinteisin menetelmin vaikka

suunnitelman osat ovatkin samanlaiset. Tietokantaa lähdettiin suunnittelemaan MySQL Workbench tietokannasuunnitteluohjelmalla.

5.3 Vaatimusten määrittely ja analyysi

Tietokantaa suunniteltaessa, täytyy ensin tutkia mitä tietoa on jo saatavilla, ja mikä tieto täytyy lisätä järjestelmään. Palkitsemisjärjestelmä tulee tarvitsemaan tiedot 1) käyttäjistä, 2) kaikista sivuston tapahtumista ja niistä annettavista pisteistä, 3) käyttäjien tekemistä tapahtumista, 4) saavutuksista, 5) käyttäjän saamista saavutuksista, 6) palkinnoista ja 7) käyttäjän ansaitsemista palkinnoista.

Näiden edellä mainittujen lisäksi tarvittaisiin tiedot virtuaalilahjoista. Eli mitä virtuaalilahjoja on kullakin käyttäjällä, ja mitä lahjoja on annettu kullekin sisällölle. Näitä ei kuitenkaan voida vielä työn tässä vaiheessa suunnitella tarkasti tai toteuttaa, koska ne liittyvät vahvasti uuteen suunnitteilla olevaan tietokantaan, ja on vielä mahdoton tietää, miten se tulee toteutumaan.

Käyttäjätiedot saadaan helposti sivustolla jo olevasta tietokannasta. Käyttäjään voidaan viitata vain käyttäjän omalla tunnistenumeroilla.

Kaikista sivuston tapahtumista tarvitsee kerätä tieto talteen. Tätä varten täytyy ensin käydä läpi kaikki sivuston mahdolliset tapahtumat. Käytännössä riittää, että käydään läpi vain ne tapahtumat, joista pisteitä tahdotaan antaa, mutta ottaen samalla huomioon myös järjestelmän laajenemismahdollisuus. Tämä tieto täytyy kyetä ottamaan talteen mistä tahansa tapahtumasta, jolloin sen tallennuslogiikka täytyy pitää yksinkertaisena. Tapahtumista saadaan myös pisteitä, jolloin tietokannassa täytyy olla tieto, paljonko pisteitä saa mistäkin tapahtumasta. Tallentamalla tiedot tietokantaan, ne ovat myös paljon helpommin muokattavissa, jonka avulla voidaan helposti säätää pisteitä vielä jälkikäteen, mikäli sille on tarvetta. Jotta tapahtumien tallentamisesta olisi jotain hyötyä, täytyy niillä olla tekijöitä. Tarvitaan lista näiden toimintojen tekijöistä, eli käyttäjistä. Listasta täytyy käydä ilmi mitä jokainen eri käyttäjä on sivustolla tehnyt.

Käyttäjä saa saavutuksen, kun hän on tehnyt tietyn määrän tiettyjä tapahtumia. Nämä saavutukset täytyy löytyä tietokannasta, josta täytyy selvittää myös se, mitä saavutuksia on kenelläkin ja mistä ja monestako tapahtumasta saavutuksen saa.

Tietystä pistemäärästä ja saavutustasoilla etenemisestä saa tiettyjä palkintoja, kuten esimerkiksi saavutusmerkin. Näistä palkinnoista ja niihin vaadittavista pistemääristä tai saavutustasovaatimuksista täytyy olla tieto tietokannassa. Siellä täytyy myös olla olemassa tieto, mitä eri palkintoja eri käyttäjillä on. Palkintoja on monia erilaisia, joten tallennuslogiikka täytyy pitää yksinkertaisena.

5.4 Käsitteellinen mallintaminen

5.4.1 Mallintamisen vaiheet

Käsittemallin luonnissa käytettiin MySQL Workbenchin Data Modeling työkalua, jolla voidaan luoda EER (Enhanced Entity-Relationship) malleja. Mallit koostuvat kolmesta osasta: 1) entiteetistä, 2) entiteetin ominaisuuksista ja 3) entiteetin relaatioista eli suhteista toisiin entiteetteihin.

Entiteetillä tarkoitetaan kokonaisuutta ja kohdetta, johon toiminnot perustuvat. Palkitsemisjärjestelmän entiteettejä ovat: käyttäjä, tapahtuma, saavutus, palkinto, sekä virtuaalilahja. Relaatioita voi olla kolmenlaisia: 1) yhden suhde yhteen, 2) yhden suhde moneen ja monen suhde yhteen, sekä 3) monen suhde moneen. Entiteetit täytyy käydä läpi yksitellen ja selvittää niiden ominaisuudet ja suhteet. Virtuaalilahjoja ei voida käydä kokonaisuudessaan läpi, joten ne on jätetty kokonaan pois epäselvyyksien välttämiseksi.

5.4.2 Käyttäjä

Käyttäjä entiteetillä on käyttäjän ominaisuudet, eli esimerkiksi käyttäjän nimi. Nämä tiedot löytyvät jo olemassa olevasta kannasta. Käyttäjällä voi olla monia tapahtumia, ja tapahtuman on voinut tehdä useampi käyttäjä (monen suhde moneen relaatio).

Käyttäjällä voi olla myös monia palkintoja ja palkintoja voi olla usealla käyttäjällä (monen suhde moneen relaatio). Käyttäjällä voi olla myös monia saavutuksia, ja saavutus voi olla monella käyttäjällä (monen suhde moneen relaatio).

5.4.3 Tapahtuma

Tapahtuman ominaisuuksia ovat tapahtuman tunnistenumero ja nimi sekä tapahtumasta saatavat pisteet. Kuten aiemmin todettiin, tapahtumalla on monen suhde moneen relaatio käyttäjään. Tapahtuma voi liittyä monen erilaisten saavutuksen saantiin, ja johonkin saavutukseen voidaan vaatia monta eri tapahtumaa (monen suhde moneen relaatio).

5.4.4 Saavutus

Saavutuksen ominaisuuksia ovat saavutuksen tunnistenumero ja nimi, sekä saavutukseen vaadittavien tapahtumien lukumäärä. Kuten aiemmin todettiin, saavutuksilla on monen suhde moneen relaatio tapahtumiin. Saavutuksista voi saada myös monia palkintoja (yhden suhde moneen relaatio).

5.4.5 Palkinto

Palkinnon ominaisuuksia ovat palkinnon tunnistenumero ja nimi, sekä palkintoon vaadittava saavutus ja palkinnosta saatava virtuaaliraha. Kuten aiemmin todettiin, palkinnoilla on monen suhde yhteeseen relaatio saavutuksiin.

5.5 Looginen suunnittelu

5.5.1 Huomioitavat asiat

Suunnitelmassa täytyy ottaa huomioon se, joka jo aikaisemminkin mainittiin, että kyseessä ei ole normaali relaatiotietokanta. Sivustolla toimivassa tietokannassa on

tarkoitus käyttää osaksi periyttämismallia. Se tarkoittaa sitä, että on olemassa yksi päätaulu, jossa on olemassa kaikki oliot, joilla on moneen suhde moneen relaatioita keskenään. Tästä taulusta periytetään jokainen olio omaksi taulukseen, jossa on sen tarkemmat ominaisuudet. Kaikki näiden olioiden väliset relaatiot suoritetaan yhdessä linkitys taulussa, joka on käytännössä päätaulun ja sen itsensä välinen monen suhde moneen linkitys. Linkitys ei ole vain relaatio, vaan se on vanhempi-lapsi suhde, jossa toinen olio on toisen olion alainen, eli tässä tapauksessa toisen taulun rivi on toisen taulun rivin alainen. Jotta palkitsemisjärjestelmän vaatiman tietokannan lisääminen onnistuisi, täytyy ottaa selvää tarvitseeko järjestelmä kytkeä tähän periyttämismallia tukevaan järjestelmään, vai saako sen kytkettyä ilman sitä ja onko se järkevää.

5.5.2 Suunnitelma

Palkitsemisjärjestelmä on osaksi erillinen järjestelmä muuhun sivustoon nähden, ja sen monen suhde moneen relaatiot ovat vain sen oman järjestelmän sisäisiä, eivätkä suunnitellun järjestelmän entiteetit liity muuhun järjestelmään kuin käyttäjä taulun kautta. Tästä syystä ainoaksi huomioitavaksi asiaksi jää se, että käyttäjätaulu ja muu järjestelmä osataan liittää oikein.

Jotta tämä onnistuisi, täytyy ensin järjestelmästä tehdä EER kaavio, jotta tulevan järjestelmän ja nykyisen tietokannan välinen yhteys voidaan hahmottaa ja selvittää. Sivuston tietokantaa ei ole työn tässä vaiheessa suunniteltu vielä tarpeeksi pitkälle, jotta aiemmat virtuaalilahjoihin liittyvät asiat saataisiin vielä liitettyä mukaan.

5.5.3 Toimenpiteet

Loogisessa suunnittelussa toteutetaan suunniteltu kanta varsinaisiksi relaatioiksi. Kaikki entiteetit muutetaan omiksi tauluikseen. Jokainen monen suhde moneen suhde muutetaan relaatioiksi, joka tarkoittaa sitä, että näiden välille luodaan uusi taulu. Tämän taulun ominaisuuksissa on molempien taulujen tunnistenumerot, joiden avulla tämä taulu linkittää toiset taulut yhteen. Näiden lisäksi tauluun tulee luontikenttä, josta tiedetään koska tämä yhteys on luotu. Tauluun asetetaan myös asetus, joka poistaa tai päivittää linkin, mikäli jommassakummassa taulussa tapahtuu muutos. Monen

suhde yhteen suhteissa taas lisätään siihen tauluun toisen taulun tunnistenumero, jossa on moneen suhde. (Tietokannan suunnittelu. 2004. Antti Ekonoja, Tommi Lahtonen, Jukka Mäntylä).

5.5.4 CakePHP nimeämiskäytännöt

Tekovaiheessa tuli ottaa huomioon, että taulujen ja ominaisuuksien nimet täytyi olla englanninkielisiä, jotta CakePHP osaa automaattisesti tehdä omia toimintojaan. Taulujen nimet täytyy olla monikossa ja pienellä kirjoitettuna. Kaksiosaiset taulun nimet täytyy olla erotettuna alaviivalla.

Monen suhde moneen relaatioiden välissä oleva taulu nimetään asettamalla taulun nimeksi molempien taulujen nimet alaviivalla erotettuna aakkosjärjestyksessä. Näissä tauluissa saa olla vain tarvittavat kaksi tunnistenumeroa, ja ne eivät saa olla pääavaimia, koska CakePHP ei tunnista yhdistelmäavaimia. Jos monen suhde moneen relaation väliin tulevaan tauluun tulee saada jotakin muuta tietoa, ei edellä mainittu nimeämissääntö päde, koska sitten kyseessä on yhden suhde moneen relaatio tähän välitauluun, ja silloin käytetään aikaisemmin mainittuja nimeämissääntöjä. Tämän lisäksi tauluun tulee nimetä oma erillinen pääavainkenttä.

Yhden suhde moneen relaatioissa monen suhde tauluun asetettava toisen taulun tunnistenumero tulee olla nimetty yhden suhde taulun yksikössä olevalla nimellä, jonka jälkeen tulee alaviiva id. (CakePHP Conventions. 2011)

5.5.5 EER diagrammin luonti

Ensimmäisenä luotiin events (tapahtuma) taulu, ja sille asetettiin edellä laaditut ominaisuudet kentiksi. Events taulun ja users (käyttäjä) taulun välille luotiin uusi taulu, koska niiden välillä on monen suhde moneen relaatio. Taulun nimeksi annettiin user_events (käyttäjän tapahtumat). Tähän tauluun luotiin tarvittavat tunnistenumeroa ja created kenttä, johon tulee päivämäärä milloin käyttäjä teki tapahtuman.

Toisena luotiin achievements (saavutukset) taulu, ja sille asetettiin edellä laaditut ominaisuudet kentiksi. Achievements taulun ja events taulun välille luotiin uusi taulu, koska niiden välillä on monen suhde moneen relaatio. Taulun nimeksi annettiin cakePHP nimeämissääntöjen mukaan achievements_events. Tässä taulussa on vain vaadittavat tunnistenumerokentät. Achievements taulun ja users taulun välillä on myös monen suhde moneen relaatio, jonka vuoksi myös niiden väliin luodaan uusi taulu. Taulun nimeksi annettiin user_achievements (käyttäjän saavutukset). Tähän tauluun luotiin tarvittavat tunnistenumerokentät, sekä created kenttä, johon tulee päivämäärä milloin käyttäjä sai saavutuksen.

Kolmantena luotiin rewards (palkinnot) taulu, ja sille asetettiin edellä laaditut ominaisuudet kentiksi. Rewards taulun ja achievements taulun välillä on monen suhde yhteen relaatio, jonka vuoksi rewards taulussa täytyy olla tunnistenumerokenttä saavutukselle. Rewards ja users taulun välillä on monen suhde moneen relaatio, jonka vuoksi niiden välille luotiin uusi taulu. Taululle annettiin nimi user_rewards (käyttäjän palkinnot), ja siihen laitettiin tarvittavat tunnistenumerokentät sekä created kenttä, jonka avulla saadaan tietää milloin käyttäjä on saanut kyseisen palkinnon.

5.6 Tietokannan asennus ja testaus

Kun suunnitelma saatiin valmiiksi, eli suunnitelma saatiin muutettua relaatioksi, ja EER diagrammi muodostettua (LIITE 2), saatiin generoitua SQL luontikomennot järjestelmän luomiseksi (LIITE 3). Luontikomennot syötettiin testitietokantaan, ja päästiin testaamaan sen toimivuutta. Muutaman testidatan jälkeen todettiin tietokanta toimivaksi.

6 OHJELMISTON SUUNNITTELU

6.1 Vaatimukset

Tärkeänä vaatimuksena itse ohjelmiston teossa oli käyttää OOP (Object Oriented Programming) ohjelmointitapaa, jonka lisäksi järjestelmä täytyy ohjelmoida muita hyviä ohjelmointitapoja noudattaen. Sivustolla on sovittu käytettäväksi CakePHP:n ohjelmointistandardit.

Järjestelmän täytyy olla myöhemmin helposti laajennettavissa, jonka vuoksi sen täytyy olla mahdollisimman dynaaminen, ja helposti muokattava. Järjestelmän liittäminen sivuille tulee olla mahdollisimman yksinkertaista ja vähän tilaa vievää, koska järjestelmä tulee toimimaan jokaisella sivuston alueella. Ohjelmiston avulla täytyy pystyä käyttämään koko palkitsemisjärjestelmää.

Sivustolle täytyy myös suunnitella ulkoasu tulevalle järjestelmälle. Ulkoasussa käytetään normaaleja sivuston tyylitelyjä. Tarkoitus olisi saada käyttöliittymästä mahdollisimman mukava, helppokäyttöinen ja interaktiivinen.

6.2 Toiminnallisuus

Jotta järjestelmä saadaan liitettyä mukaan mahdollisimman mutkattomasti koko sivustolle, täytyy se käynnistää CakePHP:n `app_controller.php` tiedostosta. Tämä tiedosto ladataan aina, kun sivustolla liikutaan, jonka vuoksi ohjelman käynnistys voidaan aloittaa täältä ilman, että sitä tarvitsisi moneen paikkaan kirjoittaa. Täältä käsin voidaan lähettää ohjelmalle paljon tietoa kerralla. On kuitenkin mahdoton saada kaikkea tietoa, koska ei voida tietää, läpäisevätkö tiedot kaikki tarkistukset. Tästä syystä tietyt asiat on pakko laittaa muualle koodin sekaan. Tämä täytyy kuitenkin hoitaa erittäin yksinkertaisella tavalla, jotta sivuston koodi pysyy siistinä. Kaikki toiminnallisuus täytyy olla koodissa, jossa tarkistetaan mitä kukin palkinto muun muassa oikeuttaa. Näistä seikoista johtuen on järkevintä luoda järjestelmästä hyvä luokkahierarkia.

Jotta järjestelmästä olisi mitään iloa, tarvitsee se hienon ulkoasun ja käyttöliittymän. Sivuston käyttäjän ei tarvitse tehdä mitään erikoista uuden järjestelmän myötä. Käyttäjän tulee vain käyttää sivua normaaliin tapaan. Kun käyttäjä saa saavutuksen tai palkinnon, ilmoitetaan hänelle siitä sivustolla toimivalla flash tekniikalla. Tällä ei tarkoiteta Adobe Flashia, vaan CakePHP flash tekniikkaa. Tekniikan avulla käyttäjälle voidaan antaa lyhykestoinen viesti sivuston ylälaitaan, kun hän saa saavutuksen ja palkinnon. Normaaleja toimintoja ei käyttäjälle raportoida tällä tavalla, mutta niitä voidaan katsoa palkitsemisjärjestelmän omalta sivulta.

Tarkoitus on kannustaa käyttäjiä käyttämään sivustoa ja luomaan sisältöä, joten järjestelmä täytyy tuoda näyttävästi esiin, jonka vuoksi se tarvitsee hienon käyttöliittymäsivun. Käyttäjän täytyy nähdä sivulta kaikki hänen palkintonsa ja saavutukset ja koska hän on ne saanut. Myös muut tapahtumat täytyy pystyä näkemään, sekä käyttäjän oma sijoitus muiden käyttäjien joukossa.

6.3 Luokkasuunnittelu

Järjestelmässä tulee olemaan periaatteessa kolme kokonaisuutta: tapahtumat, saavutukset, palkinnot ja virtuaalilahjat. Jokainen näistä on suhteellisen suuri kokonaisuus, jolloin jokaisesta kannattaa luoda oma luokkansa. Jokaisella näillä luokalla on jollain tavalla yhteys toisiinsa, joten kannattaa luoda myös abstrakti luokka, jonka nämä luokat periyttävät. Näitä jokaista kolmea luokkaa täytyy kyetä hallitsemaan yhdestä luokasta käsin, joten tarvitaan myös pääluokka, joka hoitaa näiden luokkien välisen käsittelyn. Toiminnallisuutta ja luokkasuunnitelman toimintaa kuvaamaan rakennettiin vuokaavio [LIITE 4], josta nähdään miten palkitsemisjärjestelmän tulee toimia kun sivu ladataan. Tarkempaa luokkasuunnittelua on vielä tässä vaiheessa mahdoton tehdä, koska sivuston ohjelmointi vasta aloitettiin uudestaan CakePHP:lla ja moni asia tulee olemaan eri tavalla kuin vanhassa versiossa. Jotta tarkempaa luokkasuunnittelua voitaisiin tehdä, pitäisi tietää miten sivu tullaan ohjelmoimaan uudestaan.

6.4 Ulkoasun ja käyttöliittymän suunnittelu

6.4.1 Palkitsemisjärjestelmän osa-alueet

Palkitsemisjärjestelmä tarvitsee omat alueensa sivustolle. Järkevintä on kuitenkin pitää koko järjestelmä samassa paikassa. Loogisin paikka tälle on käyttäjän oma profiilisivun alla oleva sivu. Tämän sivun on hyvä toimia niin sanottuna kojelautana, eli dashboardina (graafinen käyttöliittymä, joka muistuttaa auton kojelautaa), josta käyttäjä näkee heti suoraan kaikki uusimmat tapahtumat, saavutukset ja palkinnot, jotka hän on saanut. Sivulla voidaan näyttää myös saavutuksia, joiden saavuttaminen on lähellä, ja ansaitut virtuaalilahjat, sekä virtuaalirahan määrä. Oma henkilökohtainen sijoituslista on myös näkyvässä, josta käyttäjä voi nähdä oman sijoituksensa eri tapahtumissa. Toisinsanoen sivulla on kaikki oleellinen tieto kerralla näkyvässä pieninä kojelautaosina järjestettynä. jQuery ja jQuery UI laajennusta käyttäen voidaan antaa käyttäjälle mahdollisuus myös liikutella näitä osioita siihen järjestykseen kuin hän haluaa, jolloin jokainen voi tehdä siitä mieleisensä näköisen. Järjestämiseen käytetään jQuery UI laajennuksen Sortable Portlets tekniikkaa. Jokainen näistä kojelaudan osista on kuitenkin jaettu myös omiin sivuihinsa, josta käyttäjä voi nähdä enemmän ja tarkempaa tietoa.

6.4.2 Tapahtumat

Tapahtumat sivulla käyttäjä kykenee selaamaan kaikkia tekemiään tapahtumia, ja hän näkee myös milloin kyseiset tapahtumat ovat tapahtuneet. Käyttäjä voi myös selata tapahtumakalenteria tietyltä aikaväliltä nähdäkseen mitä tapahtumia sillä aikavälillä hän on tehnyt. Tämän lisäksi hän näkee sivulta, montako tietynlaista tapahtumaa hän on tehnyt yhteensä, ja kalenterista hän voi määrittää aikavälin, jolta hän tahtoo nämä tapahtumat nähdä.

6.4.3 Saavutukset

Saavutukset sivulla käyttäjä näkee omat saavutuksensa saavutusmerkkeineen uusimmasta vanhimpaan päivämäärät mukaan luettuna. Lisäksi sivulla näkyvät saavutusmerkit, joita käyttäjä ei ole vielä saanut, sekä merkit jotka käyttäjä saisi päästessään seuraavalle tasolle. Merkeistä näkee myös, moniko käyttäjä on kyseisen merkin saanut, jolloin tietää kuinka harvinainen kyseinen merkki on.

Saavutuksesta voi olla monta tasoa, ja aina tasolla edetessä, hän saa vanhan merkin tilalle uuden merkin. Lisäksi käyttäjä näkee edistymispalkin, paljonko hän vielä tarvitsee tapahtumia päästäkseen seuraavalle saavutustasolle. Kuten tapahtumissakin, käyttäjä voi määrittää aikavälin, jolta hän tahtoo saavutuksiaan katsoa.

6.4.4 Palkinnot

Palkinnot sivulta käyttäjä voi katsoa kaikki saavuttamansa palkinnot. Palkinnoista hän voi katsoa niiden tarkemmat tiedot, jotka kertovat mistä saavutuksista palkinto on saatu, ja mihin se palkinto oikeuttaa, sekä paljonko siitä sai palkinnoksi virtuaalirahaa. Listassa näkyvät myös palkinnot, jotka käyttäjä saavuttaa seuraavilla saavutus-tasoilla, sekä kaikki muut vielä saavuttamatta olevat palkinnot. Käyttäjä voi määrittää aikavälin, jolta hän tahtoo selata saamiaan palkintoja.

6.4.5 Virtuaalilahjat ja -raha

Virtuaalilahjasivulla käyttäjä voi ostaa virtuaalirahalla virtuaalilahjoja itselleen. Sivulla on listaus mahdollisista virtuaalilahjoista ja niiden tarkemmista ominaisuuksista, kuten esimerkiksi lahjan arvostuksen suuruus, sekä kauanko käyttäjän pitää vielä odottaa, että hän voi antaa lahjan uudelleen jonkun toisen käyttäjän sisällölle. Samalta sivulta hän näkee ostamansa virtuaalilahjat, sekä sisällöt joille hän on kutakin lahjaa antanut, sekä keltä ja mihin sisältöihin hän on virtuaalilahjoja saanut. Samalta sivulta käyttäjä näkee myös virtuaalirahansa määrän, jotta hän tietää mihin lahjoihin hänellä on varaa.

6.4.6 Sijoituslistat

Sijoituslistat sivulla käyttäjä voi selata mikä on hänen sijoituksensa erilaisissa tapahtumissa. Hän näkee myös kokonaispisteensä, jotka hän on saanut kaikista tapahtumista ja kokonaissijoituksensa, joka perustuu näihin pisteisiin. Näiden lisäksi ovat käytössä vielä lahjasijoituslistat, jotka perustuvat sekä saatuihin, että annettuihin lahjoihin, sekä virtuaalirahalistat, jotka perustuvat kerätyn virtuaalirahan määrään. Käyttäjä voi myös selata sijoituslistoja nähdäkseen ketkä ovat korkeammalla ja alemmalla sijoituksella häneen itseensä nähden. Sivustolla jo oleva HoF lista korvataan kokonaissijoituslistoilla, joka muuten toimii samalla tapaa kuin edellinenkin.

7 PALKITSEMISJÄRJESTELMÄN TOTEUTTAMINEN

7.1 Lähtökohdat

Palkitsemisjärjestelmä aiotaan toteuttaa edellä laadittujen suunnitelmien mukaan. Ensin integroidaan tietokanta nykyiseen järjestelmään, ja testataan sen käyttöä, että se toimii myös siellä niin kuin pitää. Kun tietokanta on saatu integroitua, aloitetaan ohjelmiston ohjelmoiminen.

7.2 Tietokannan toteuttaminen

7.2.1 MySQL -tietokannan luonti

Tietokannan MySQL puolen toteuttaminen on tässä tapauksessa yksinkertainen prosessi. Kirjautumme tietokantaan ja kopioimme aikaisemmin generoidun tietokannan luontikomennot MySQL:n komentoriville. Operaatio onnistui ilman virheitä.

7.2.2 CakePHP Console

Tietokannan toteuttamisen lisäksi täytyy luoda myös MVC arkkitehtuurin mallit, eli mallit, joiden avulla tietokantaa tullaan käyttämään. CakePHP tarjoaa itse hyvin automatisoidun työkalun, jonka avulla mallit voidaan luoda pienellä vaivalla. CakePHP Console on komentorivipohjainen ohjelma, jonka avulla voidaan automaattisesti luoda pohjia eri CakePHP:n tiedostoille. Kirjoittamalla ”cake bake” päästään CakePHP:n sanoin ”leipomaan” koodia. Komennolla pääsee valikkoon, jonka kautta voi luoda automaattisesti valmiiksi tietokannan konfiguraatitiedoston, mallit, näkymät, ohjaimet, projekti pohjat, fixturet ja testi tapaukset. (CakePHP Console. 2011). Tässä työssä konsolia käytetään vain mallien luonnissa.

7.2.3 Mallien luonti

CakePHP console pystyy tunnistamaan tietokannan tiedoista millainen tietokanta on suurin piirtein kyseessä. Ohjelmalla luodaan malli jokaisesta taulusta, joka ei ole monen suhde moneen relaation välitaulu. Ohjelma kysyy mallia luodessa ensin mahdollisia kelpuuttamissääntöjä kentille, esimerkiksi maksimipituutta ja mitä ne saavat sisältää. Näiden lisäksi se kysyy taulun väliset relaatiot muihin tauluihin. Ohjelma osaa itse ehdottaa mahdollisia relaatioita, jotka se on saanut selville tietokannan tiedoista. Nämä säännöt pitää kuitenkin tarkistaa, että ne ovat oikein. Jos relaatiota ei ole, sen voi itse määrittää. Kun kaikki nämä on luotu, ohjelma generoi mallitiedoston CakePHP:n models kansioon, joka nimetään tietokannan taulun nimen mukaan yksikössä. Kaikki mallitiedostot täytyy myös tarkistaa, että ne on varmasti luotu oikein, ja lisätä puuttuvat kelpoisuussäännöt mikäli sellaisia on, sekä määrittää yhteydet tarkemmin, jos sille on tarvetta. Kun kaikista tauluista on luotu mallitiedostot, voidaan niitä käyttää CakePHP:n ohjaimissa halutulla tavalla.

7.3 Ohjelmiston toteuttaminen

7.3.1 Sivuston tilanne

Sivuston omassa projektissa sivustoa ei ole vielä ohjelmoitu uudelleen tarpeeksi pitkälle CakePHP -alustalla, jotta ohjelmistoa voitaisiin alkaa ohjelmoimaan. Voidaan kuitenkin ajatuksellisesti suunnitella miten ohjelma rakentuisi CakePHP:n sisälle.

7.3.2 App Controller

App Controller, eli ohjelmiston ohjain on kaikkien ohjainten pääohjain, kuten aiemmin ohjelmiston suunnitelmassa jo mainittiinkin. Tässä ohjaimessa voidaan käyttää niin sanottuja callback funktioita, joita kutsutaan ohjelmiston suorituksen eri kohdissa. Näitä callback funktioita ovat: beforeFilter, beforeRender ja afterFilter.

beforeFilter kutsutaan ennen minkään ohjaimen toimintojen suorittamista. beforeRender kutsutaan kun ohjaimen toiminto on suoritettu, mutta näkymää ei vielä tulostettu. afterFilter kutsutaan kun ohjaimen toiminto on suoritettu ja näkymä tulostettu. (CakePHP Controller callbacks. 2011). Nämä funktiot ovat erittäin hyödyllisiä palkitsemisjärjestelmän logiikan kannalta.

Palkitsemisjärjestelmän pääluokan alustus ja muut alustustoimenpiteet voidaan suorittaa beforeFilter funktiossa. Tällöin ne on valmiiksi asetettu, kun seuraavaksi ajettava toimintologiikka suoritetaan.

Kun toimintologiikkaa suoritetaan, voidaan asettaa sen toimintoja pääluokan kautta muistiin. Nämä muistissa olevat toiminnot ajetaan beforeRender funktion sisällä, kun toiminnot on suoritettu. Tämän jälkeen sieltä asetetaan flash viesti seuraavaksi tulostettavalle sivulle. Flash viesti pitää sisällään mahdolliset saavutetut saavutukset ja saadut palkinnot.

Tarvitavat lopputoimenpiteet voidaan suorittaa afterFilter funktiossa, kun sivu on jo ladattu. Jos näkymässä on esimerkiksi erilaisia näkymävaihtoehtoja, voidaan afterFilterin avulla suorittaa haluttuja toimintoja tähän tiettyyn näkymään liittyen.

7.3.3 Palkitsemisjärjestelmän ohjain

Palkitsemisjärjestelmä ei tarvitse omaa ohjaintaan, koska kaikki siihen liittyvät asiat liittyvät täysin suoraan käyttäjään. Tästä syystä palkitsemisjärjestelmän ohjaimena kannattaa toimia ohjain, joka käsittelee käyttäjiä. Käyttäjäohjaimen kautta malleilla on myös paras pääsy kaikkiin palkitsemisjärjestelmän osa-alueisiin, koska käyttäjät ovat sen keskus.

7.3.4 Ohjaimen toiminnot

Ohjaimen voidaan asettaa actioneita eli toimintoja. Kukin toiminto viittaa jollain tapaa ohjaimen. Kullakin toiminnolla on oma view, eli näkymä. Toiminnolla voi

olla myös useita näkymiä riippuen sen vastaanottamasta tiedosta. Toimintojen tarkoitus on kuitenkin käsittää vain yksi toiminto.

Tässä työssä tarvitaan toiminnot, jotka hakevat ohjelmiston suunnittelussa suunniteltujen ulkoasujen sisällöt, jotta ne voidaan näyttää näkymissä. Lisäksi tarvitaan toiminnot, joilla voi ostaa virtuaalilahjoja itselleen, ja joilla voi antaa toisen käyttäjän sisällöille ostamiaan virtuaalilahjoja. Työssä suunniteltu tietokanta ei ole kuitenkaan tässä tapauksessa riittävä, jotta virtuaalilahjoja voitaisiin antaa sisällöille tai ostaa itselle. Tämä johtuu siitä, että ei vieläkään tiedetä miten sisältötaulu tulee muodostumaan, ja kuinka siihen voidaan tehdä liitettäväksi taulu, jossa nämä virtuaalilahjat näkyisivät ja koska ne on annettu.

Aiemmin mainitun kojelaudan näyttämiseksi tarvitaan oma toiminto, joka voidaan nimetä esimerkiksi viewDashboard. Tämän toiminnon täytyy hakea käytännössä kaikki palkitsemisjärjestelmään liittyvä tieto käyttäjään liittyen. Tieto haetaan käyttäjän aiemmin tehtyjä malleja. Kojelaudan tapauksessa näitä ei kuitenkaan haeta heti toiminnon suorituksessa, vaan vasta kun näkymä on ladattu. Tämä johtuu siitä, että haettavaa tietoa on erittäin paljon haettavaksi joka kerta, kun sivu ladataan. Näkymässä voidaan vielä jälkeinpäin pyytää tiettyjä tietoja eri toiminnoilta. Nämä tiedot välitetään lopulta kojelaudan näkymän eri osille, jotka yhdessä tulostavat näkymän. Etuna tässä on se, että ne voidaan asettaa välimuistiin halutun määrätiksi ajaksi, joka nopeuttaa sivun latautumista ja parantaa palvelimen suorituskykyä huomattavasti.

Tarvitaan myös tapahtumat viewEvents, viewAchievements, viewRewards, viewGifts ja viewRecords, jotka tulostavat kojelaudalla esiintyvien osien tiedot suunnittelussa mainituilla erillisillä sivuilla. Kukin näistä hakee mallien avulla tarvitsemansa tiedot ja välittää ne näkymilleen. Tämä tieto tallennetaan välimuistiin, josta se poistetaan, mikäli sivulle tulee jotain uutta näytettävää.

7.3.5 Näkymät

Näkymät tehdään kaikista tehdyistä toiminnoista lukuun ottamatta virtuaalilahjojen osto- ja lahjoitustoimintoja. Näkymät pitävät sisällään XHTML merkkäusta. Ohjain

ja toiminto voivat lähettää näkymälle tulostettavaa tietoa. Tämä tieto voidaan tulostaa haluttuun paikkaan näkymässä esimerkiksi XHTML merkkauksen sisään. Lopulta tämä näkymä asetetaan yleisesti määritellyn layoutin, eli ulkoasun sisälle ja tämä kokonaisuus näytetään sivun lataajalle.

Ohjaimen toiminnoissa mainittuja näkymän osia kutsutaan elementeiksi. Elementit ovat osia, joita voidaan tulostaa näkymän eri paikkoihin. Elementtejä voidaan tallentaa cacheen, eli välimuistiin, jolloin ne latautuvat paljon nopeammin.

Tyylit on jaettu moneen eri tyyli tiedostoon, ja tässä vaiheessa ei vielä tiedetä mistä tyyli tiedostosta mikin tyyli löytyy. Sivustolla ei ole määritelty tyyli tiedostoja kokonaan valmiiksi, jonka vuoksi näkymiä on hyvin hankala suunnitella työn tässä vaiheessa. Palkitsemisjärjestelmässä on kuitenkin käytettävä sivuston omia tyyli telyjä, vaikkakin se vaatii myös paljon omia tyyli telyitä. Sivusta ei voida tietää millainen siitä tulisi, ellei käytä juuri niitä tyyli telyjä, kuin sivustossa pitääkin käyttää. Vaikka sivuston suunnittelisikin jo valmiiksi omilla tyyli telyillä, voi niitä joutua muuttamaan hyvin paljon riippuen muusta sivuston tyyli telystä. Tästä syystä on parasta jättää näiden tarkempi laadinta myöhemmälle, eli kunnes sivusto on tarpeeksi pitkälle siirretty.

8 YHTEENVETO

8.1 Työn onnistumisen arviointi

Työ oli erittäin opettavainen ja laaja. Työn aikana käytettiin hyvin paljon uusia ohjelmia ja tekniikoita, sekä saatiin ymmärrystä siitä, miten järjestelmiä tutkitaan ja suunnitellaan. Työtä ei päästy tekemään niin pitkälle, että voitaisiin sanoa onnistuttiinko työssä pääsemään alussa asetettuihin tavoitteisiin asti, mutta työ onnistui kuitenkin hyvin siihen asti, mitä sitä päästiin tekemään.

Työssä huomioitiin kaikki muut työlle asetetut rajat ja vaatimukset ja niitä noudatettiin. Suunniteltuun käyttäjäkyselylomakkeeseen ei ehditty työn aikana saamaan kuitenkaan tarpeeksi vastauksia, jotta sen tuloksia olisi päästy analysoimaan ja hyödyntämään. Osittainen hidas vastausten saaminen voi johtua siitä, että lomake jäi edelleen aika pitkäksi, ja se on käyttäjälle aikaa vievä täyttää.

Työn puolella välissä sivuston järjestelmän vaihtuminen teki lisätyötä, koska kokonaan uusi ohjelmistokehys täytyi opetella. Järkeväksi vaihtoehdoksi jäi tehdä vain alustavat suunnitelmat tehtävästä palkitsemisjärjestelmästä, koska uuteen ohjelmistokehykseen siirtyminen vei sivustolla niin paljon aikaa. Työ jäi tästä syystä harmittavan paljon vajavaiseksi toteuttamisen kannalta.

Työ itsessään vaati paljon enemmän aikaa, kuin alussa kuviteltiin, johtuen osaksi opeteltavista asioista ja tekniikoista. Vaikka työssä ei päästykään konkreettisesti toteuttamaan suunniteltua järjestelmää, voi sen toteuttaa kuka tahansa tässä työssä laadittujen suunnitelmien pohjalta.

8.2 Tulevaisuuden pohdintaa

Järjestelmä tullaan luultavasti myöhemmin toteuttamaan sivustolla. Työ on joka tapauksessa tehty sillä tavalla, että sitä voi hyödyntää myös muissa samankaltaisissa hankkeissa. Kun palkitsemisjärjestelmä saadaan joskus toteutettua sivustolle, kannattaa työhön ottaa mukaan myös graafikko luomaan hienoja kuvakkeita muun muassa

ansaituille saavutusmerkeille. Näin sivustosta ja järjestelmästä saataisiin paljon näyttävämpi. Järjestelmää pystytään laajentamaan halutessa niin paljon kuin halutaan, niin kuin työn alussa vaatimuksissa oli määritelty. Tämä antaa vapaat ovet kelle tahansa laajentaa järjestelmä mielensä mukaiseksi.

Palkitsemisjärjestelmä itsessään on mielenkiintoinen konsepti käyttäjien aktivoimiseen ja toiminnan ohjaamiseen, ja sitä tullaan todennäköisesti tutkimaan tulevaisuudessa vielä enemmän. Vielä löytyy varmasti hyviä ideoita hyödynnettäväksi.

LÄHTEET

CakePHP. 2011. Cake Software Foundation, Inc. Viitattu 22.2.2011.
<http://cakephp.org/>

CakePHP Console. 2011. Cake Software Foundation, Inc. Viitattu 28.2.2011.
<http://book.cakephp.org/view/1522/Code-Generation-with-Bake>

CakePHP Controller callbacks. 2011. Cake Software Foundation, Inc. Viitattu 28.2.2011. <http://book.cakephp.org/view/984/Callbacks>

CakePHP Conventions. 2011. Cake Software Foundation, Inc. Viitattu 28.2.2011.
<http://book.cakephp.org/view/903/Model-and-Database-Conventions>

Cascading Style Sheets. 2011. W3Schools. Viitattu 28.02.2011.
http://www.w3schools.com/css/css_intro.asp

E-lomake. Eduix Oy. Viitattu 2011.12.2.
<https://e-lomake.fi/web/pintaasyyvemmalta/kuvaus/>

Eclipse. 2011. The Eclipse Foundation. Viitattu 28.02.2011. <http://www.eclipse.org>

Gamification. 2011. Wikipedia. Viitattu 2011.1.2.
<http://en.wikipedia.org/w/index.php?title=Gamification&oldid=410037113>

Git. 2011. Scott Chacon. Viitattu 28.02.2011. <http://git-scm.com/>.

GitHub. 2011. GitHub Inc. Viitattu 28.2.2011. <https://github.com/>

JavaScript. 2011. W3Schools. Viitattu 28.02.2011.
http://www.w3schools.com/js/js_intro.asp

jQuery. 2011. The jQuery Project. Viitattu 22.2.2011. <http://jquery.com/>

jQuery UI. 2011. jQuery UI Team. Viitattu 22.2.2011. <http://jqueryui.com/>

Likertin asteikko. 2006. William M.K. Trochim. Viitattu 22.2.2011.
<http://www.socialresearchmethods.net/kb/scallik.php>

MVC-arkkitehtuuri. 2011. Microsoft. Viitattu 28.02.2011.
<http://msdn.microsoft.com/en-us/library/ff649643.aspx>.

MySQL. 2010. Oracle Corporation and/or its affiliate. Viitattu 28.02.2011.
<http://www.mysql.com/about/>.

MySQL Workbench. 2009. MySQL Workbench Team. Viitattu 13.2.2011.
http://wb.mysql.com/?page_id=6

PHP. 2011. The PHP Group. Viitattu 22.2.2011.
<http://www.php.net/manual/en/intro-what-is.php>

Tietokannan suunnittelu. 2004. Antti Ekonoja, Tommi Lahtonen, Jukka Mäntylä.
Viitattu 12.2.2011. <http://appro.mit.jyu.fi/doc/tiedonhallinta/suunnittelu/>

UML-mallinnus. 2010. Wikipedia. Viitattu 22.02.2011.
<http://fi.wikipedia.org/w/index.php?title=UML-mallinnus&oldid=9021163>.

ZEND. 2011. Zend Technologies Ltd. Viitattu 22.2.2011.
<http://framework.zend.com/manual/en/learning.quickstart.intro.html>

Auta kehittämään Massidea.org innovaatio yhteisön kannustinjärjestelmää

Mikä on Massidea.org?

Massidea.org on uudenlainen sosiaalisen median toimintaperiaatetta hyödyntävä avoin innovaatioyhteisö, jossa käyttäjät voivat jakaa muiden käyttäjien kanssa ideoitaan, havaitsemiaan ongelmia ja tulevaisuuden näkymiä. Massidea.org pyrkii parantamaan käyttäjiensä luovuutta yhdistelmällä älykkäästi erilaisia sisältöjä, ja sisältöjä tuottaneet käyttäjät. Yhteisön keskeisenä tavoitteena on kannustaa käyttäjiä aktiiviseen kanssakäymiseen keskenään.

Kuinka Massidea.org toimii?

Massidea.org palvelun toimintaperiaate on seuraava:

- 1) Luot käyttäjätunnuksen
- 2) Ryhdyt tuottamaan idea, ongelma ja tulevaisuus kuvauksia ja jaat ne muiden käyttäjien kanssa
- 3) Jokainen tuottamasi uusi sisältöä kasvattaa henkilökohtaista sisältövarastoasi
- 4) Henkilökohtainen sisältövarastosi toimii rajapintana muihin käyttäjiin ja heidän sisältöihinsä. Mitä suurempi sisältö varastosi on, sitä paremmat vuorovaikutus mahdollisuudet sinulla on
- 5) Uudet jopa odottamattomat yhteydet toisiin käyttäjiin ja heidän sisältöihinsä ruokkivat luovuuttasi ja synnyttävät virikkeellisen ympäristön.

Kuka on Massidea.org palvelun takana?

Massidea.org verkkopalvelua kehittämiseen osallistuu 10 ammattikorkeakoulua mukaan lukien sinun oma ammattikorkeakorkeakoulusi. Palvelun kehittämistä rahoittaa EU:n sosiaalirahasto.

Mikäli haluat tutustua Massidea.org verkkopalveluun ennen kyselyyn vastaamista, siirry osoitteeseen www.massidea.org. Muutoin jatka kysymyslomakkeeseen vastaamista seuraavalla sivulla.

Mihin tarvitsemme apusi?

Massidea.org verkkopalvelun seuraavassa kehitysvaiheessa on tarkoitus luoda pelinomainen kannustinjärjestelmä, jonka avulla aktiivisia käyttäjiä voidaan nostaa esiin ja palkita.

Kannustinjärjestelmässä käyttäjät kerryttävät pisteitä Massidea.org verkkopalvelussa tekemistä erilaisista toimista. Kerättyjen pisteiden avulla käyttäjät voivat kohota ”kunniagalleriassa” ylöspäin ja saavuttaa jopa palkintoja.

Kannustinjärjestelmän kehitystyön tueksi haluamme selvittää ammattikorkeakouluopiskelijoiden näkemyksiä 1) minkä tyyppisistä toiminnoista pisteitä tulisi kertyä ja kuinka paljon, sekä 2) minkälaiset palkinnot kannustaisivat opiskelijoita aktiivisesti osallistumaan Massidea.org palvelun toimintaan.

Jotta saisit mahdollisimman hyvän kokonaiskuvan miten Massidea.org toimii, on seuraavaan listaan koottu tärkeimmät toiminnot, jotka muodostavat palvelun ytimen:

1. Sisällön tuottaminen.
2. Ryhmän luonti
3. Kampanjan luonti
4. Linkittäminen
5. Kommentointi
6. Lahja- ja tukipisteet
7. Muut tapahtumat.

Käyttäjä voi luoda sisällön, jonka hän voi halutessaan linkittää toiseen sisältöön (esim. linkittää oman ongelma kuvauksen jonkun muun käyttäjän ongelma kuvaukseen tai ideaan). Yksittäisen sisällön voi kuka tahansa käyttäjä halutessaan myös arvostella peukalo ylös/alas toiminnolla. Lisäksi käyttäjät voivat antaa kirjoittaa sisältösivun loppuun omat kommenttinsa.

Käyttäjä voi perustaa ryhmän, johon hän haluaa kerätä käyttäjiä, jotka ovat samanhenkisiä johonkin aiheeseen liittyen. Ryhmä voi aloittaa kampanjan jostakin tietystä asiasta johon sitten ryhmään kuuluvat voivat liittää sisältöjään, ja näin kota aiheen ympärille paljon ajatuksia.

Käyttäjä voi kommentoida näitä ryhmiä ja kampanjoita, ja niitä on mahdollisuus myös linkittää keskenään: ryhmän ryhmään ja kampanjan kampanjaan. Kommentoida voi myös toista käyttäjää tämän profiilissa.

Muita tapahtumia ovat mm. yksityisviestin lähettäminen toiselle käyttäjälle ja sisältöjen lisääminen suosikkeihin.

Pisteiden antaminen

Voit antaa pisteitä väliltä 0-100. Myös 0 pistettä on mahdollinen, jos et koe toiminnon olevan sellainen, josta pisteitä pitäisi antaa. Voit antaa samoista toiminnoista saman määrän pisteitä, mikäli koet niiden olevan samanarvoisia.

Seuraavilla sivuilla käymme läpi edellä mainittuja tapahtumia, joita pyydämme Teitä arvos-telemaan suhteuttamalla niitä toisiinsa.

Sisällön tuottamiseen liittyvät toiminnot

Käyttäjät voivat tuottaa Massidea.org palveluun kolmen tyyppisiä sisältöjä:

- 1) **Tämän päivän haasteita:** Haaste voi olla henkilökohtainen, liike-elämään tai yleisemmin yhteiskuntaan liittyvä ongelma, tarve, tilanne, tapahtuma tai havainto, joka käyttäjä mielestä on vielä ratkaisematta häntä tyydyttävällä tavalla.
- 2) **Tulevaisuuden visio:** Tulevaisuuden visiot pyrkivät ennustamaan minkälainen maailmaa on lähitulevaisuudessa. Lähitulevaisuudella tarkoitetaan yleensä vähintään 10 vuoden päähän sijoittuvaa ajanjaksoa. Tulevaisuuden visiot voivat käsitellä skenaarioita, teknologian kehittymistä, tieteellisiä löydöksiä, poliittisia päätöksiä trendejä tai anti-trendejä, jotka syntyvät trendien vastailmiöiksi. Näiden avulla pyritään ennustamaan, minkälainen maailma todennäköisesti tulee olemaan seuraavan kymmenen vuoden kuluttua. Toisaalta visiot voivat käsitellä heikkoja signaaleja jotka ovat merkkejä asioista ja tapahtumista, jotka toteutuessaan voivat merkittävästi vaikuttaa meidän kaikkien elämään, mutta vielä tällä hetkellä niiden toteutuminen vaikuttaa epävarmalta. Tulevaisuuden visioihin liittyy myös keskeisesti vision mukanaan tuoma mahdollisuus, mutta myös tähän toteutuneeseen muutokseen liittyvä uhka.
- 3) **Idea kuvaus:** Idea kuvaukset voivat olla ratkaisuja tämän päivän haasteisiin tai ne voivat ehdottaa ratkaisuja kuinka tulevaisuuden visioissa tunnistettuja mahdollisuuksia voidaan hyödyntää ja uhka kuviin löytää ratkaisuja. Idea kuvaus voi olla pieni inkrementaalinen eli vaiheittainen parannus olemassa olevaan tuotteeseen, palveluun, prosessiin tai organisoititapaan. Toisaalta se voi olla radikaali, vallankumouksellinen uudenlainen ratkaisu ja ajattelutapa joka muuttaa nykyisen valitsevan käsityksemme siitä kuinka asioiden tulisi olla.

Asteikolla 0-100 kuinka paljon pisteitä sinun tulisi saada seuraavista toiminnoista?

- a) Kirjoitat **tämän päivän haasteen** muiden käyttäjien saataville: _____ pistettä
- b) Kirjoitat **tulevaisuuden vision** muiden käyttäjien saataville: _____ pistettä
- c) Kirjoitat **idea kuvauksen** muiden käyttäjien saataville: _____ pistettä

Jotta muut käyttäjät ja erityisesti entuudestaan asiaa tuntemattomat käyttäjät voisivat mahdollisimman helposti ja nopeasti ymmärtää kirjoittamasi ajatukset, olemme kehittäneet Massidea.org verkkopalveluun yhtenäisen sisältöformaatin jotka käyttäjien tulisi omissa kirjoituksissaan noudattaa.

Tähän liittyen jokaisen sisällön riippumatta siitä onko se haaste, visio tai idea tulisi vastata seuraaviin kysymyksiin:

- 1) **Mikä** on haaste/idea/visio (ts. kuvaus mistä oikein on kyse)
- 2) **Miksi** haaste/idea/visio on tärkeä ja merkityksellinen (ts. tärkeyden perustelu)
- 3) **Kuka** on haasteen/idean/vision kohderyhmä (ts. kenen tulisi olla asiasta kiinnostunut)
- 4) **Milloin** haaste/idea/visio on ajankohtainen (ts. asian ajallinen ulottuvuus)
- 5) **Missä** haaste/idea/visio on ajankohtainen (esim. maantieteellinen, paikka, tai tilanne).

Asteikolla 0-100 kuinka paljon pisteitä sinun tulisi saada seuraavista toiminnoista?

- a) Annat sisällöissäsi vastauksen **Mikä**-kysymykseen: _____ pistettä
- b) Annat sisällöissäsi vastauksen **Miksi**-kysymykseen: _____ pistettä
- c) Annat sisällöissäsi vastauksen **Kuka**-kysymykseen: _____ pistettä
- d) Annat sisällöissäsi vastauksen **Milloin**-kysymykseen: _____ pistettä
- e) Annat sisällöissäsi vastauksen **Missä**-kysymykseen: _____ pistettä

Käyttäjä voi kirjoittaa sisällön Massidea.org verkkopalveluun haluamallaan kielellä. Sisältöä luodessa englanninkieli on kuitenkin suosituksena, jotta sisältö tavoittaisi mahdollisimman monta käyttäjää.

Asteikolla 0-100 kuinka paljon pisteitä sinun tulisi saada sisällön (ongelma, idea tai tulevaisuuden visio) kirjoittamisesta...

- a) Englannin kielellä _____ pistettä
- b) Jollain muulla kielellä (esim. Suomi) _____ pistettä

Massidea.org sivustolla on mahdollista kääntää automaattisesti teksti toiselle kielelle hyödyntämällä palveluun integroitua Google-kääntäjä ominaisuutta. Automaattinen kääntäjä ei kuitenkaan kaikissa tapauksissa ole yhtä laadukas kuin ihmisen kääntämä teksti, Tämän

johdosta Massidea.org haluaakin tarjota käyttäjälle myös mahdollisuuden julkaista sisältö useammalla kielellä.

Asteikolla 0-100 kuinka paljon pisteitä sinun tulisi saada kun

- c) Käännät **oman** tekstisi toiselle kielelle _____ pistettä
d) Käännät **jonkun muun käyttäjän** tekstin toisella kielelle _____ pistettä

Ryhmän luonti

Ryhmän tarkoitus on koota yhteen erilaisia käyttäjiä, joilla on samoja mielenkiinnon kohteita tai muita yhdistäviä tekijöitä kuten opiskelu, yhdistystoiminta, harrastus, työ tai asuinpaikka. Yksittäiseen ryhmään voi liittyä rajaton määrä jäseniä ja yksittäinen käyttäjä voi luoda haluamansa määrään ryhmiä.

Massidea.org verkkopalvelussa ryhmiä on kolmenlaisia:

- **Avoim ryhmä:** Avoimeen ryhmään voi liittyä kuka tahansa, milloin tahansa. Ryhmiin liittyvät sisällöt näkyvät kaikille käyttäjälle, myös niille jotka eivät kuulu ryhmään.
- **Suljettu ryhmä:** Suljettuun ryhmään lähettämällä liittymispyynnön ryhmän omistajalle joka voi halutessaan hyväksyä tai hylätä käyttäjän liittymispyynnön. Vastavasti kuin avoimessa ryhmässä, ryhmiin liittyvät sisällöt näkyvät kaikille käyttäjälle, myös niille jotka eivät kuulu ryhmään.
- **Yksityinen ryhmä:** Yksityiseen ryhmään liitytään lähettämällä liittymispyynnön ryhmän omistajalle joka voi hyväksyä tai hylkää käyttäjän liittymispyynnön. Toisin kuin suljetussa ryhmässä, yksityisen ryhmän sisältö näkyy **vain** ryhmän jäsenille.

Asteikolla 0-100 kuinka paljon pisteitä sinun tulisi saada ryhmän luonnista?

- a) **Avoimen** ryhmän luonnista _____ pistettä
b) **Suljetun** ryhmän luonnista _____ pistettä
c) **Yksityisen** ryhmän luonnista _____ pistettä

Asteikolla 0-100 kuinka paljon pisteitä sinun tulisi saada seuraavista toiminnoista?

- a) Liityt jonkun toisen käyttäjän perustamaan ryhmään _____ pistettä
b) Joku toinen käyttäjä liittyi perustamaasi ryhmään _____ pistettä

Kampanjan luonti

Kampanja on ryhmän omistajan organisoitu ja koordinoitu pyrkimys edistää sisällön tuotantoa määrittelemäänsä teema alueeseen. Kampanjan avulla ryhmän omistaja kannustaa jäseniään tuottamaan sisältöä juuri sellaisiin teemoihin jotka ovat ryhmälle tärkeitä. Kampanjat voivat olla ikuisia tai aikaan sidottuja eli ne kestävät vain tietyn ajan.

Käyttäjät osallistuvat kampanjaan linkittämällä julkaisemansa sisällön kyseiseen kampanjaan. Sisällön linkittäminen kampanjaan edellyttää ryhmän jäsenyyttä. Käyttäjä voi halutesaan linkittää saman sisällön useaan kampanjaan.

Asteikolla 0-100 kuinka paljon pisteitä sinun tulisi saada seuraavista toiminnoista?

a) Luot kampanjan _____ pistettä

Asteikolla 0-100 kuinka paljon pisteitä sinun tulisi saada seuraavista toiminnoista?

a) Linkität sisältösi jonkun toisen käyttäjän kampanjaan _____ pistettä

b) Joku toinen käyttäjä linkittää sisältönsä sinun kampanjaasi _____ pistettä

Linkittäminen

Linkittämällä erilaisia sisältöjä keskenään, voidaan edistää Massidea.org verkkopalvelun käyttäjien luovuutta. Linkityksen pyrkimyksenä onkin edesauttaa ennalta arvaamattomien sisältökokonaisuuksien muodostumista, jotka stimuloivat uusia ajatuksia käyttäjien päähän.

Massidea.org verkkopalvelussa sisältöjä voidaan linkittää useilla eri tavoilla:

1. Sisällön linkittäminen sisältöön (esim. idean linkittäminen ongelmaan)
2. Sisällön linkittäminen eli lisääminen kampanjaan
3. Ryhmän linkittäminen ryhmään
4. Kampanjan linkittäminen kampanjaan

Sisällön linkitys toimii kahdensuuntaisesti. Eli linkittäessäsi esimerkiksi haasteen ideaan, tulet samalla linkittäneeksi ks. idean haasteeseen. Käytännössä tämä tarkoittaa että linkki kahden linkitetyn sivun välillä näkyy kummallakin sivulla, riippumatta siitä kumpaa sivua katselet. Mikään ei estä käyttäjiä linkittämästä kahta eri ideaa yhteen haasteeseen, jolloin ideat ikään kuin kilpailevat ratkaisun paremmuudesta keskenään, tai voivat toisaalta olla yhdessä se ratkaiseva tekijä kyseisen haasteen ratkaisussa. Myös kaksi samaa tyyppiä olevaa sisältöä voidaan linkittää keskenään (esim. idean linkittäminen ideaan). Tällöin linkityksen perusteena voi olla vaikkapa sisältöjen samanlaisuus tai eri näkökulma samasta asiasta.

Myös yksittäinen ryhmän voidaan linkittää useampaan ryhmään kuin myös yksittäinen kampanja voidaan linkittää useampaan kampanjaan.

Asteikolla 0-100 kuinka paljon pisteitä sinun tulisi saada seuraavista linkityksistä?

a) Linkität **oman ryhmäsi** jonkun **toisen ryhmään**: _____ pistettä

b) Linkität **oman kampanjasi** jonkun **toisen kampanjaan**: _____ pistettä

c) Linkität **oman sisältösi** jonkun **toisen sisältöön**: _____ pistettä

d) Linkität **oman profiilisivusi** jonkun **toisen profiilisivuun**: _____ pistettä

Kumpaa alla esitetyistä linkitysmalleista pidät parempana? (laita X valitsemasi vaihtoehdon eteen)

___ Luomasi linkki näkyy toisen käyttäjän ryhmä-, kampanja-, sisältö- tai profiilisivulla **välittömästi**.

___ Luomasi linkki näkyy toisen käyttäjän ryhmä-, kampanja-, sisältö- tai profiilisivulla **vasta tämän hyväksytyä linkkisi**.

Vastaa kumpaankin seuraavaan kysymykseen riippumatta siitä kummin valitset edellisessä kysymyksessä.

Jos luomasi linkki näkyisi toisen käyttäjän ryhmä-, kampanja-, sisältö- tai profiilisivulla **välittömästi, asteikolla 0-100** kuinka paljon pisteitä...

- a) sinun tulisi **saada** kun teet linkin _____ pistettä
- b) sinun tulisi **menettää** jos toinen käyttäjä poistaa sinun tekemäsi linkin sen nähtyään _____ pistettä

Jos luomasi linkki näkyisi toisen käyttäjän ryhmä-, kampanja-, sisältö- tai profiilisivulla **vasta tämän hyväksytyä linkkisi, asteikolla 0-100** kuinka paljon pisteitä...

- a) sinun tulisi **saada** kun teet linkin _____ pistettä
- b) sinun tulisi **saada** jos toinen käyttäjä hyväksyy sinun tekemäsi linkin _____ pistettä

Kommentin antaminen

Massidea.orgissa voit kommentoida muiden käyttäjien tuotoksia kunkin ryhmä-, kampanja-, sisältö- tai käyttäjän profiilisivulla sivun alalaidassa olevalla keskustelutoiminnolla. Kaikki sivulla olevat kommentit näkyvät kaikille käyttäjille jonka johdosta kommentit muodostavat eräänlaisen keskustelupuu.

Asteikolla 0-100 kuinka paljon pisteitä sinun tulisi saada seuraavista kommentoinneista?

- a) Kirjoitat kommentin **oman** ryhmäsi, kampanjasi, sisältösi tai profiilisivusi keskusteluosioon _____ pistettä
- b) Kirjoitat kommentin **jonkun toisen** käyttäjän ryhmä, kampanja, sisältö tai profiilisivun keskusteluosioon _____ pistettä
- c) **Joku toinen** käyttäjä kirjoittaa kommentin **sinun** ryhmän, kampanjan, sisällön tai profiilisivun keskusteluosioon _____ pistettä

Lahjoituspisteet

Osallistumalla aktiivisesti Massidea.org verkkopalvelun toimintaan, käyttäjä kartuttaa omaa pistetiliään. Pistetilin karttuessa käyttäjälle kertyy samalla lahjoituspisteitä, joita hän voi edelleen jakaa haluamilleen käyttäjille.

Esimerkiksi ryhmän omistajalle kertyy lahjapisteitä, kun muut käyttäjät linkittävät omia sisältöjään hänen ylläpitämiinsä kampanjoihin. Ryhmän omistaja voi puolestaan jakaa näin kertyneitä pisteitä haluamilleen käyttäjille. Lahjoituspisteillä käyttäjä voi osoittaa toiselle käyttäjälle tukeaan ja motivoida häntä yhteistyöhön.

Asteikolla 0-100 kuinka paljon pisteitä sinun tulisi kerätä, jotta saisit itsellesi yhden lahjoituspisteen jonka voit edelleen antaa jollekin toiselle käyttäjälle. _____ pistettä

Pisteiden menettäminen

Koska Massidea.org on kaikille avoin verkkopalvelu, voi sinne eksyä myös häiriköitä. Nämä häiriköt voidaan poistaa helposti ylläpidon toimesta, mikäli heistä ilmoitetaan. Tähän liittyen jokaisella Massidea.org sivulla on ns. flag-toiminto, jota klikkaamalla käyttäjät voivat ilmoittaa asiattomista sisällöstä ja siihen liittyvästä käyttäjätunnuksesta.

Asteikolla 0-100 kuinka paljon pisteitä sinun tulisi saada jos?

a) Ilmoitat epäasiallisesta sisällöstä flag-toiminnon avulla _____ pistettä

Asteikolla 0-100 kuinka paljon pisteitä sinun tulisi menettää jos?

a) Administraattori poistaa sinun luomasi ryhmän: _____ pistettä

b) Administraattori poistaa sinun luomasi kampanjan: _____ pistettä

c) Administraattori poistaa sinun luomasi sisällön: _____ pistettä

d) Administraattori poistaa sinun luomasi kommentin: _____ pistettä

Muut toiminnot

Edellä kuvattujen toimintojen lisäksi Massidea.org pitää sisällään joukon muitakin toimintoja.

Asteikolla 0-100 kuinka paljon pisteitä sinun tulisi saada seuraavista toiminnoista?

- a) Lähetät yksityisviestin toiselle käyttäjälle _____ pistettä
b) Luet toisen käyttäjän lähettämän yksityisviestin _____ pistettä
- a) Lisäät sisällön suosikkilistaasi _____ pistettä
b) Sisältösi lisätään jonkun toisen suosikkilistaan _____ pistettä
- a) Katsot jonkun toisen käyttäjän **ryhmä**-sivun _____ pistettä
b) Katsot jonkun toisen käyttäjän **kampanja**-sivun _____ pistettä
c) Katsot jonkun toisen käyttäjän **sisältö**-sivun _____ pistettä
d) Katsot jonkun toisen käyttäjän **profiilisivun** _____ pistettä
- e) Joku toinen käyttäjä katsoo sinun **ryhmä**-sivun _____ pistettä
f) Joku toinen käyttäjä katsoo sinun **kampanja**-sivun _____ pistettä
g) Joku toinen käyttäjä katsoo sinun **sisältö**-sivun _____ pistettä
h) Joku toinen käyttäjä katsoo sinun **profiilisivun** _____ pistettä
- a) Arvostelet jonkun toisen käyttäjän sisällön _____ pistettä
b) Joku toinen käyttäjä arvostelee sinun sisältösi positiivisella äänellä _____ pistettä
c) Joku toinen käyttäjä arvostelee sinun sisältösi negatiivisella äänellä _____ pistettä

Kumpia pisteitä arvostat enemmän? (laita X valitsemasi vaihtoehdon eteen)

_____ Pisteitä, joita voin itse kerryttää omalla toiminnallani
(esim: teen kommentin, lisään sisällön, luon ryhmän tai kampanjan, jne.)

_____ Pisteitä, jotka voin saada vain toisen käyttäjän toiminnan kautta
(esim: joku muu kommentoi sisältöäsi, lukee sisältösi, liittyy ryhmääsi, liittää sisällön kampanjaasi)

Mikäli haluat antaa kehittäjille avoimia kommentteja ja/tai kehitysehdotuksia voit kirjoittaa ne tähän

15. kerätkäkseni uusia näkökulmia	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
16. edelleenjalostaakseni omia ideoitani	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
17. aktivoidakseni oman luovuuteni	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
18. kehittääkseni itseäni	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
19. koska haluan tehdä yhteistyötä muiden kanssa	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Kerro lopuksi hieman itsestäsi

a. Ikä: _____ a. Opiskeluvuosi: 1. 2. 3. 4. 5. vuosi

b. Sukupuoli: Mies Nainen

c. Onko sinulla kotona käytössäsi nopea Internet-yhteys (laajakaista)? Kyllä Ei

d. Pidätkö itseäsi

1. erittäin tottuneena verkkoviestintäympäristöjen ja niiden sovellusten käyttäjänä

2. kohtalaisen tottuneena verkkoviestintäympäristöjen peruskäyttäjänä

3. aloittelijana verkkoviestintäympäristöjen käytössä

e. Minkälaiseksi verkkoviestintäympäristöjen käyttäjäksi kuvailisit itseäsi

1. Luen ja tarkkailen, mutten osallistu aktiivisesti (esim. tuota sisältöä, kommentoi, äänestä, arvostelee)

2. Osallistun ajoittain, mutta pääasiassa keskityn lukemiseen ja tarkkailuun

3. Osallistun aktiivisesti ja usein

f. Minä vuonna aloit käyttämään verkkoviestintäympäristöjä? _____

h. Oppilaitos ja yksikkö: _____ i. Koulutusohjelma: _____

g. Haluatko osallistua kasvokkain toteutettavaan haastatteluun, jonka tarkoituksena on arvioida opiskelijoiden sosiaalisen verkostojen käyttöä ja asenteita syvällisemmin?

Kyllä En

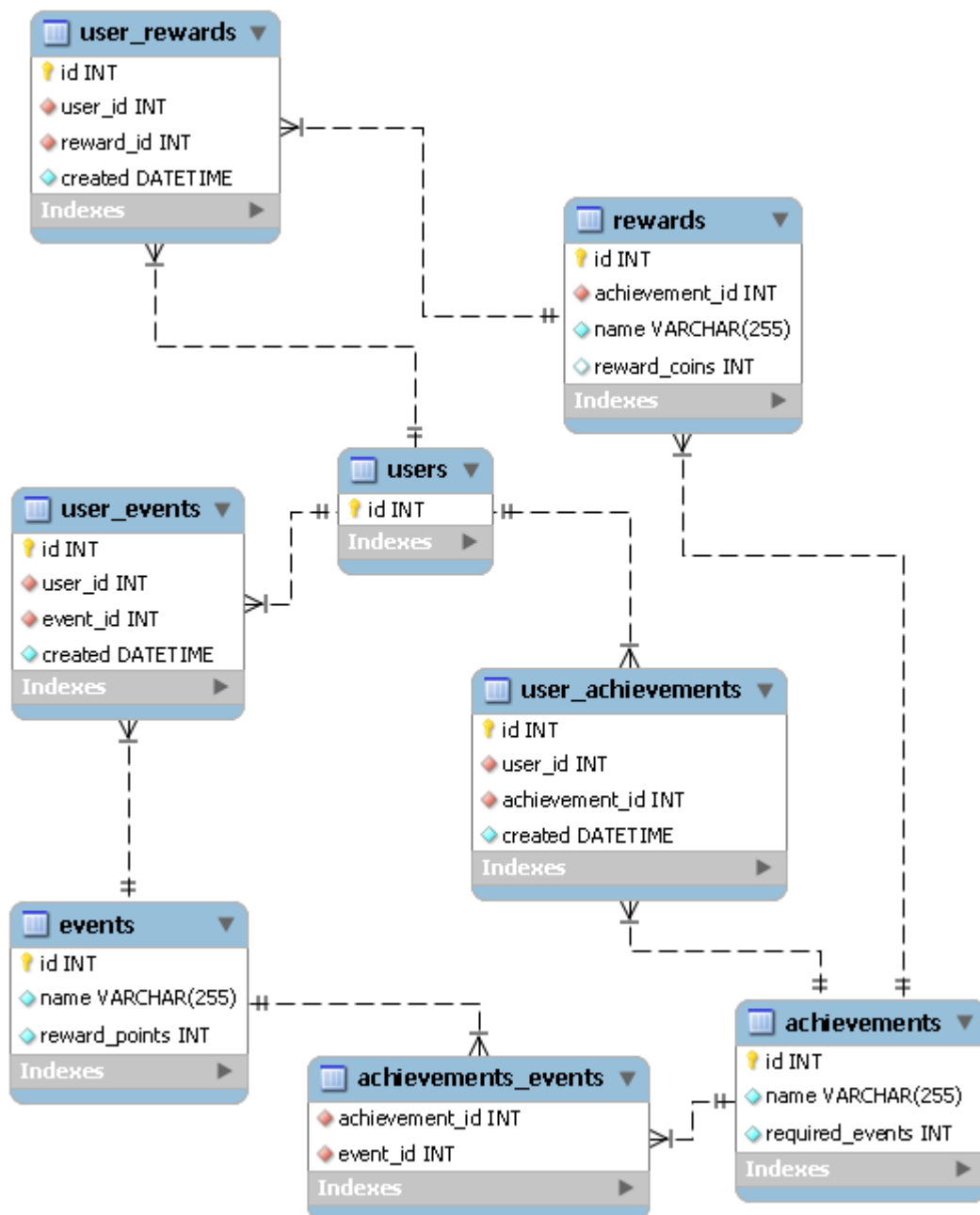
Jos vastasit kyllä, kirjoita yhteystietosi alle, jotta voimme ottaa sinuun myöhemmin yhteyttä

Nimi: _____ Puhelinnumero: _____

E-mail _____

Kiitos osallistumisestasi!

EER diagrammi



Tietokannan luomislauseet:

```
SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS,
UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS,
FOREIGN_KEY_CHECKS=0;
SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='TRADITIONAL';
```

```
USE `massidea` ;
```

```
-----
-- Table `massidea`.`achievements`
-----
```

```
CREATE TABLE IF NOT EXISTS `massidea`.`achievements` (
  `id` INT NOT NULL ,
  `name` VARCHAR(255) NOT NULL ,
  `required_events` INT NOT NULL ,
  PRIMARY KEY (`id` )
ENGINE = InnoDB;
```

```
-----
-- Table `massidea`.`rewards`
-----
```

```
CREATE TABLE IF NOT EXISTS `massidea`.`rewards` (
  `id` INT NOT NULL ,
  `achievement_id` INT NOT NULL ,
  `name` VARCHAR(255) NOT NULL ,
  `reward_coins` INT NULL DEFAULT 0 ,
  PRIMARY KEY (`id` ) ,
  INDEX `fk_rewards_achievement` (`achievement_id` ASC),
  CONSTRAINT `fk_rewards_achievement`
```

```
FOREIGN KEY (`achievement_id` )
REFERENCES `massidea`.`achievements` (`id` )
ON DELETE NO ACTION
ON UPDATE CASCADE)
ENGINE = InnoDB;
```

```
-----
-- Table `massidea`.`events`
-----
```

```
CREATE TABLE IF NOT EXISTS `massidea`.`events` (
  `id` INT NOT NULL ,
  `name` VARCHAR(255) NOT NULL ,
  `reward_points` INT NOT NULL ,
  PRIMARY KEY (`id` )
ENGINE = InnoDB;
```

```
-----
-- Table `massidea`.`user_events`
-----
```

```
CREATE TABLE IF NOT EXISTS `massidea`.`user_events` (
  `id` INT NOT NULL ,
  `user_id` INT NOT NULL ,
  `event_id` INT NOT NULL ,
  `created` DATETIME NOT NULL ,
  INDEX `fk_user_events_event` (`event_id` ASC) ,
  PRIMARY KEY (`id` ) ,
  CONSTRAINT `fk_user_events_user`
  FOREIGN KEY (`user_id` )
  REFERENCES `massidea`.`users` (`id` )
  ON DELETE CASCADE
  ON UPDATE CASCADE,
  CONSTRAINT `fk_user_events_event`
```

```
FOREIGN KEY (`event_id` )
REFERENCES `massidea`.`events` (`id` )
ON DELETE NO ACTION
ON UPDATE CASCADE)
ENGINE = InnoDB;
```

```
-----
-- Table `massidea`.`user_rewards`
-----
```

```
CREATE TABLE IF NOT EXISTS `massidea`.`user_rewards` (
  `id` INT NOT NULL ,
  `user_id` INT NOT NULL ,
  `reward_id` INT NOT NULL ,
  `created` DATETIME NOT NULL ,
  INDEX `fk_user_rewards_reward` (`reward_id` ASC) ,
  PRIMARY KEY (`id`) ,
  CONSTRAINT `fk_user_rewards_user`
  FOREIGN KEY (`user_id` )
  REFERENCES `massidea`.`users` (`id` )
  ON DELETE CASCADE
  ON UPDATE CASCADE,
  CONSTRAINT `fk_user_rewards_reward`
  FOREIGN KEY (`reward_id` )
  REFERENCES `massidea`.`rewards` (`id` )
  ON DELETE NO ACTION
  ON UPDATE CASCADE)
ENGINE = InnoDB;
```

```
-----
-- Table `massidea`.`user_achievements`
-----
```

```
CREATE TABLE IF NOT EXISTS `massidea`.`user_achievements` (
```

```

`id` INT NOT NULL ,
`user_id` INT NOT NULL ,
`achievement_id` INT NOT NULL ,
`created` DATETIME NOT NULL ,
INDEX `fk_user_achievements_achievement` (`achievement_id` ASC),
PRIMARY KEY (`id`),
CONSTRAINT `fk_user_achievements_user`
  FOREIGN KEY (`user_id` )
  REFERENCES `massidea`.`users` (`id` )
  ON DELETE NO ACTION
  ON UPDATE NO ACTION,
CONSTRAINT `fk_user_achievements_achievement`
  FOREIGN KEY (`achievement_id` )
  REFERENCES `massidea`.`achievements` (`id` )
  ON DELETE NO ACTION
  ON UPDATE NO ACTION)
ENGINE = InnoDB;

```

```

-----
-- Table `massidea`.`achievements_events`
-----

```

```

CREATE TABLE IF NOT EXISTS `massidea`.`achievements_events` (
  `achievement_id` INT NOT NULL ,
  `event_id` INT NOT NULL ,
  INDEX `fk_achievements_events_event` (`event_id` ASC),
  CONSTRAINT `fk_achievements_events_achievement`
    FOREIGN KEY (`achievement_id` )
    REFERENCES `massidea`.`achievements` (`id` )
    ON DELETE CASCADE
    ON UPDATE CASCADE,
  CONSTRAINT `fk_achievements_events_event`
    FOREIGN KEY (`event_id` )
    REFERENCES `massidea`.`events` (`id` )

```

```
ON DELETE NO ACTION  
ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

```
SET SQL_MODE=@OLD_SQL_MODE;  
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;  
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
```

LIITE 4

