

Julius Hakkarainen

INFOPORTAALI

Liiketalouden koulutusohjelma
Tietoverkkopalveluiden suuntautumisvaihtoehto
2011

INFOPORTAALI

Hakkarainen, Julius
Satakunnan ammattikorkeakoulu
Tietoverkkopalveluiden sv
Lokakuu 2011
Ohjaaja: Kuisma, Pekka
Sivumäärä: 99
Liitteitä:

Asiasanat: verkkopalvelut, ohjelmistotuotanto, portaalit, järjestelmät

Tämän opinnäytetyön tarkoituksena oli laatia internetissä toimiva, PHP-kielellä tehty infoportaali itse, osittain tai kokonaan valmiilla sovelluksella ja laatia portaaliin sen käyttöohje. Infoportaalin avulla voidaan esitellä jotakin yritystä ja/tai palvelua tekstin, kuvien ja videoiden avulla. Portaalin toimintaa kuvaamaan siihen on asennettu esimerkkipalvelu.

Työn vaiheisiin kuului sivujen toteutukseen tarvittavien asioiden määrittely, suunnittelu, sivujen toteutus, testaus ja pienissä määrin käyttöönotto. Vain asioiden esittely olisi saanut olla monipuolisempaa. Sivuja voisi siis periaatteessa käyttää, mutta esittelymahdollisuuksien monipuolistaminen olisi asiakkaiden määrän kannalta parempi.

Työn teoriaosassa käydään läpi ohjelmistotuotantoon liittyviä asioita ja erilaisia vaihejakomalleja. Tämän jälkeen käydään läpi ilmaisia valmiita sovellusvaihtoehtoja ja mietitään, käytetäänkö valmista sovellusta vai tehdäänkö ohjelma kokonaan vai osittain itse. Valitaan portaalin pohja ja perustellaan valinta.

Suunnitteluvaiheen jälkeen toteutetaan tehty suunnitelma, testataan sovellusta, korjataan mahdolliset viat ja lopuksi ohjelma otetaan käyttöön internetin ilmaiselle testipalvelimelle. Työn lopputuloksena syntyy siis PHP-pohjaiset sivut, jolla käyttäjät voivat esitellä haluamiaan asioita. Lisäksi sivuilla on omat käyttöohjeet vierailijoille, käyttäjille ja ylläpitäjille.

Työ antaa vastauksen muun muassa seuraaviin kysymyksiin: Miten internetissä oleva portaali, jolla voidaan esitellä jonkin palvelun toimintaa, kannattaa laatia? Kannattaako portaali suunnitella ja ohjelmoida itse vai käyttää jotakin valmista sovellusta? Mitä asioita portaalin suunnittelussa ja toteutuksessa pitää ottaa huomioon: tietoturva, yhteensopivuus ainakin yleisimpien selaimien kanssa ja miten projektin pitäisi edetä käytännössä?

Viimeisenä asiana ovat sivujen esittely tekstien ja kuvien avulla ja infoportaalin tulevaisuuden pohtiminen. Esittely ei ole perusteellinen, vaan siinä kerrotaan ja esitellään olennaiset asiat infoportaalista.

INFORMATION PORTAL

Hakkarainen, Julius
Satakunta University of Applied Sciences
Information Network Services

October 2011
Supervisor: Kuisma, Pekka
Number of pages: 99
Appendices:

Key words: web services, software engineering, portals, systems

The purpose of this thesis was to create an information portal in PHP language. The portal might be made entirely by oneself or using partially or wholly a premade application. Another task was to write a user manual of the information portal. The information portal could be used for demonstrating a firm or/and a service with the help of texts, pictures and videos. It also includes a demo to describe the function of the portal.

The work phases included a requirement specification, the design and implementation of sites, testing and, to some extent, initialization. There could have been more functions for the task of demonstrating. In principle, the sites could be used but extending the range of functions related to demonstrating could increase the number of customers.

The theoretical part of this thesis first handles with issues related to software engineering and different kinds of phasing models. After this, certain free application alternatives are tested to help in deciding whether to use a readily available application or a partially or wholly premade one. Lastly, the template of the portal is chosen and the reason for the choice is given.

After the design phase, the design is implemented, the application is tested, possible errors are corrected and, finally, the application is initialized on a free host. As a result, we have sites based on PHP language where users are able to demonstrate what they wish. Additionally, the sites include user manuals for visitors, users and web masters.

The thesis answers the following questions: How should a portal demonstrating a service on the Internet be created? Is it worth designing and programming the portal by oneself or should one use a premade application? What things should be taken into account when designing and implementing the portal; such as data security, compatibility at least with the most common browsers, and how the project should advance in reality.

At the end of the thesis, sites are presented with the help of texts and pictures, and the future of the information portal is discussed. The presentation is not thorough, but it explains and describes the essential parts of the information portal.

SISÄLLYS

1	SYMBOLI- JA TERMILUETTELO.....	6
2	JOHDANTO.....	8
3	OPINNÄYTETYÖSUUNNITELMA.....	9
3.1	Opinnäytetyöongelma.....	9
3.2	Ongelmaan liittyvät osaongelmat.....	10
3.3	Teoreettinen viitekehys.....	10
3.4	Käytettävät menetelmät, aineiston kerääminen, tulosten luotettavuus ja työn arvioitu aikataulu.....	11
4	OHJELMISTOTUOTANTO.....	11
4.1	Yleisiä ongelmia ohjelmistotyössä.....	12
4.1.1	Monimutkaisuus, näkymättömyys, muunnettavuus ja ainutkertaisuus.....	12
4.1.2	Skaalautumattomuus ja epäjatkuvuus.....	13
4.2	Vaihejakomallit.....	14
4.2.1	Vesiputousmalli.....	14
4.2.2	Prototyypimalli.....	15
4.2.3	Evo-malli.....	16
4.2.4	Spiraalimalli.....	17
4.2.5	Mallien vaiheet.....	18
5	TOTEUTUSYMPÄRISTÖ.....	22
5.1	XAMPP.....	22
5.2	Internet-selaimet.....	23
5.3	Palvelimen kokoonpano.....	24
6	PHP-NUKE 8.0.....	24
6.1	Asennus.....	24
6.2	PHP-Nuken käyttö ja sen lisenssi.....	25
6.2.1	Sivujen ulkonäkö ja tietokanta.....	26
6.2.2	Ylläpidon käyttöliittymä.....	28
6.2.3	Nuken tiedostojärjestelmä.....	31
6.2.4	Nuken yleisiä ohjelmointiohjeita ylläpidettävyyden vuoksi.....	34
6.2.5	Esimerkki projektin tehtävien jaosta.....	40
7	INFOPORTAALI.....	40
7.1	Vaatimukset.....	42
7.2	Toiminnallinen ja tekninen määrittely.....	42
7.2.1	ER-malli.....	47

8	TOTEUTUS	50
8.1	Taulujen MySQL-koodit	51
8.2	Ota yhteyttä.....	55
8.3	Käyttäjän salasanan vaihtaminen	56
8.4	Portaalin esittelyn lukeminen.....	57
8.5	Sisäänkirjautuminen.....	58
8.6	Uloskirjautuminen.....	59
8.7	Salasanan vaihtaminen	60
8.8	Teeman vaihto	60
8.9	Käyttöohjeet	61
8.10	Sisältö	63
8.11	Sivut.....	65
8.12	Esitykset	67
8.13	Äänestäminen	68
8.14	Esityksen käynnistäminen	69
8.15	Uutiset.....	70
8.16	Vierailijoiden ja käyttäjien toimintojen käyttäminen	71
8.17	Äänestysten hallinta.....	72
8.18	Uutisten hallinta.....	73
8.19	Yhteydenotot	75
8.20	Ylläpitäjien hallinta	76
8.21	Sivujen ominaisuuksien muuttaminen.....	77
8.22	Käyttöohjeet	81
8.23	Käyttäjien hallinta.....	83
8.24	Käyttäjien sisältöjen hallinta	84
8.25	Moduulien hallinta.....	85
8.26	Osioiden hallinta.....	86
8.27	Kiellettyjen IP-osoitteiden hallinta.....	88
8.28	Tietokannan kopiointi	89
8.29	Tietokannan optimointi.....	91
9	TESTAUS.....	92
9.1	Moduulitestaus	92
9.2	Integrointitestaus.....	93
9.3	Järjestelmättestaus.....	93
10	LOPPUTULOKSET.....	94
10.1	Lyhyt esittely valmiista sivuista	95
11	TULEVAISUUDEN NÄKYMÄT.....	97
	LÄHTEET.....	98

1 SYMBOLI- JA TERMILUETTELO

Portaali: Internetissä oleva verkkopalvelu, joka luo yhteyksiä eri tahojen välille (Wikipedia, Portaali (internet)).

Olio-ohjelmointi: Modulaarisuuteen perustuva ohjelmointitapa (Haikala & Märijärvi 2004, 309).

Moduuli: Kutsutaan ohjelmoinnissa yleisesti myös olioksi. Ohjelmiston ”rakennuspalikka” tai toimintayksikkö. Valmis ohjelmisto koostuu monista moduuleista.

Vaihejakomalli: Tapa, jolla ohjelmiston kehitystyö jaetaan eri vaiheisiin (Haikala & Märijärvi 2004, 36).

Hierarkia: Ohjelmiston toimiminen niin, että sen moduulit käyttävät toisiaan hyväkseen (Haikala & Märijärvi 2004, 308).

Modulaarinen: Moduuleista koostuva ja sen mukaisesti toimiva ohjelmisto (Wikipedia, Moduuli).

Kotelointi: Moduulien tekemistä sellaisiksi, että niiden sisältämiä tietoja voi käsitellä vain itsensä kautta (Haikala & Märijärvi 2004, 308).

Tiedon piilotus: Katso termi kotelointi. Moduulin tiedon piilotus on onnistunut, kun ohjelmiston yksittäisen moduulin käyttäminen vaatii mahdollisimman vähän tietoa moduulin toteutuksesta/rakenteesta (Haikala & Märijärvi 2004, 308).

Abstrakti tietotyyppi: Asia, jota moduuli kuvastaa. Tätä asiaa ei voi konkretisoida. Tällaisia ovat mm. liukuluvut ja laiteajurit (Haikala & Märijärvi 2004, 309).

Sisällönhallintajärjestelmä: Englanniksi content management system, CMS. Tietojärjestelmä, joka helpottaa jonkun asian, kuten verkkopalvelun, hallitsemista. Esimerkiksi PHP-Nuke helpottaa www-sivujen tekemistä tarjoamalla yksinkertaistettuja tapoja lisätä sivulle esimerkiksi uusi uutisartikkeli ja hallitsemalla tätä artikkelia muokkaamalla tai poistamalla sen (Wikipedia, Sisällönhallintajärjestelmä).

Intranet: Suppeampi versio internetistä. Lähiverkon synonyymi. Internet koostuu tietokoneista tai laitteista, kuten intranet. Intranet on kuitenkin ryhmä laitteita, jotka voivat kommunikoida vain keskenään, ei muiden ryhmien laitteiden kanssa.

Webmaster/ylläpitäjä: Www-sivuja hallinnoiva taho (Wikipedia, Webmaster).

Käyttöliittymä: Ohjelmiston osa, jolla käyttäjä käyttää ohjelmistoa. Www-sivuilla esimerkiksi napit ovat osa käyttöliittymää. (Wikipedia, Käyttöliittymä.)

Tietokanta: Tietotekniikkaan, erityisesti ohjelmointituotantoon liittyvä termi. Se on varasto tiedoille, joita ohjelmistot käyttävät. Ohjelmistot voivat esimerkiksi lisätä ja poistaa tietokannasta tietoja tarpeen mukaan. (Wikipedia, Tietokanta.)

Muuttuja: Ohjelmointikielen termi. Se on kuin varasto, johon laitetaan tietynlaista tavaraa. Yhteen muuttujaan voi esimerkiksi tallentaa vain tekstin, toiseen kirjaimen ja toiseen kokonaisluvun.

HTML: Lyhenne sanoista Hypertext Markup Language tai hypertekstin merkintäkieli. Se on ohjelmointikieli, jota käytetään www-sivun rakentamiseen. (Wikipedia, HTML.)

RSS-syöte: Tarkoittaa samaa kuin RDF. RSS-syötteet ovat palvelimella olevia tiedostoja, joita käytetään tiedon saamisen helpottamiseksi. Ideana on, että RSS-syötettä käyttävä selain osaa kertoa käyttäjälleen, mitä kyseisen syötteen omistamalla sivulla tapahtuu. Siksi syötteitä ladataan omalle koneelle esimerkiksi uutisia sisältäviltä sivuilta.

2 JOHDANTO

Tämä opinnäytetyö on viimeisiä tehtäviä, joita ammattikorkeakoulusta saa tehtäväksi. Sen tarkoitus on mitata opiskelijan osaamista. Tämä työ on tehty kokonaisuudessaan kotona, ei koulussa, koska siellä näin suurta työtä ei moni jaksaisi tehdä. Työn idea muotoutui alkuperäisestä aiheesta, joka oli tehdä info-tv -järjestelmä jollekin yritykselle. Sellaista yritystä ei ollut kuitenkaan näköpiirissä, mutta suurin syy oli halu tehdä enemmän omaan tahtiin, koska arvasin työn tekemisen kestävän alkuperäistä suunnitelmaa kauemmin, koska aihe liittyy ohjelmointiin. Yrityksellekin olisi ollut mukavampi, jos työn olisi saanut edes suhteellisen ajoissa valmiiksi.

Aihe muodostui lopulliseen muotoonsa vasta aivan työn aloituksen alussa. Työn aiheeksi/ongelmaksi tuli tehdä www-sivut, jossa olevia palveluita joku yritys voisi myydä asiakkailleen. Asiakkaat voisivat käyttää www-sivuja esitelläkseen joitain omia tuotteitaan/palveluitaan joillekin, esimerkiksi niiden omille asiakkailleen. Www-sivujen palveluiden käytöstä joutuisi kuitenkin maksamaan kuukausimaksua. Ongelman ratkaiseminen ei ollut ihan yksinkertaista, vaikka harrastuspohjaista ja koulusta saatua ohjelmointikokemusta olikin jo ennen työn aloitusta. Ohjelmointitekniikka ei ehkä sinänsä muuttunut paljonkaan työn vuoksi. Ongelmia tuli siitä, että tarkoitus oli tehdä www-sivut ja sitä varten piti opetella joukko uusia ohjelmointikieliä. Pääasiassa olen käyttänyt vain C#-kieltä, jota ei käytetä Internet-sivujen teossa. Toinen suurempi asia oli se, että sivut oli tarkoitus tehdä mahdollisimman oikeaoppisesti. Siksi työhön kuuluu määrittely-, suunnittelu-, toteutus- ja testausvaihe.

Lopputuloksesta tuli onneksi tarpeeksi hyvä, että sillä voidaan esitellä mitä asioita tahansa. Jos aikaa olisi ollut vielä enemmän, sivuja olisi voinut hyvinkin parantaa. Parannusideoita on tälläkin hetkellä mielessä. Erityisesti asioiden esittelyn olisi saanut monipuolisemmaksi. Jos jotain lähtisin parantamaan nyt, juuri tuo olisi ensimmäinen asia. Tämän raportin lopussa on lyhyt esitelmä valmiista työstä, kuvien kera.

3 OPINNÄYTETYÖSUUNNITELMA

3.1 Opinnäytetyöongelma

Opinnäytetyöni tarkoituksena on laatia internetissä toimiva, PHP-kielellä tehty infoportaali itse, osittain tai kokonaan valmiilla sovelluksella ja laatia portaaliin sen käyttöohje. Infoportaalien avulla voidaan esitellä jotakin yritystä ja/tai palvelua tekstin, kuvien ja videoiden avulla. Portaalien toimintaa kuvaamaan siihen on asennettu esimerkkipalvelu. Valitsin tämän työn koulun ehdottamista valmiista aiheista, koska tämä oli niistä mielenkiintoisin. Minulla oli itse keksitty aihe valittuna jo ennen tätä, mutta sen idea kaatui asian tutkimisen vaikeuteen. Tämän työn alkuperäisen aiheen mukaan työllä olisi ollut toimeksiantaja, mutta tässä työssä sitä ei ole. Tarkoitus on tehdä tämä työ kuitenkin siltä pohjalta, että tätä portaalia voidaan tarvittaessa hyödyntää oikeastikin.

Työn teoriaosassa käydään läpi ohjelmistotuotantoon liittyviä asioita ja erilaisia vaihejakomalleja. Tämän jälkeen käydään läpi ilmaisia valmiita sovellusvaihtoehtoja ja mietitään, käytetäänkö valmista sovellusta vai tehdäänkö ohjelma kokonaan vai osittain itse. Valitaan portaalien pohja ja perustellaan valinta. Jos valinta kohdistuu kokonaan tai osittain omaan sovellukseen, mietitään portaalien suunnittelussa/toteutuksessa tarvittavia asioita.

Työn empiriaosan sisältö vaihtelee sen mukaan, päädytäänkö käyttämään valmista sovellusta. Jos ei käytetä, suunnitellaan portaalien rakenne, sisältö ja ulkoasu. Jos käytetään, niin yksityiskohtaista suunnittelua ei tarvita. Suunnitteluvaiheen jälkeen toteutetaan tehty suunnitelma, testataan sovellusta, korjataan mahdolliset viat ja lopuksi otetaan käyttöön internetin ilmaiselle testipalvelimelle.

Työn lopputuloksena syntyy siis PHP-pohjaiset sivut, jolla käyttäjät voivat esitellä haluamiaan asioita. Lisäksi sivuilla on omat käyttöohjeet vierailijoille, käyttäjille ja ylläpitäjille. Vierailijoiden ohjeet kertovat yrittävät samalla houkutellessa käyttäjäksi.

3.2 Ongelmaan liittyvät osaongelmat

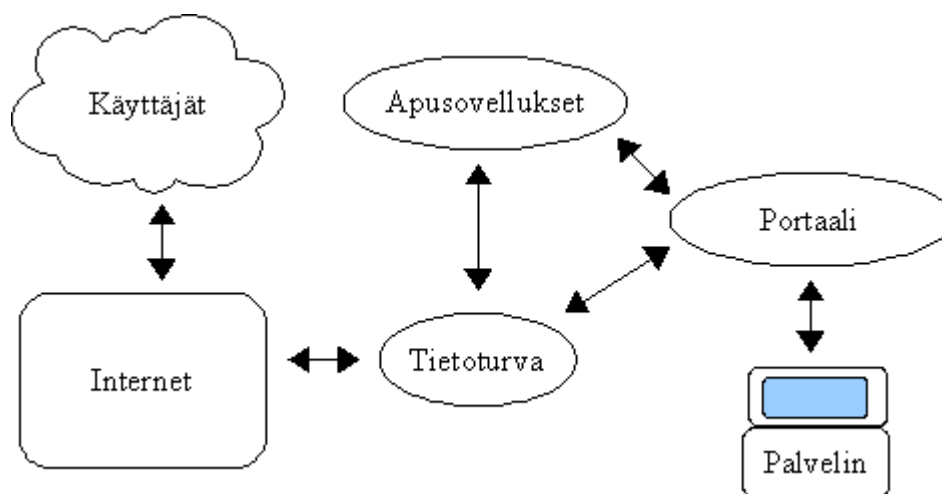
Työ antaa vastauksen muun muassa seuraaviin kysymyksiin:

- Miten internetissä oleva portaali, jolla voidaan esitellä jonkin palvelun toimintaa, kannattaa laatia? Kannattaako portaali suunnitella ja ohjelmoida itse vai käyttää jotakin valmista sovellusta?
- Mitä asioita portaalin suunnittelussa ja toteutuksessa pitää ottaa huomioon? (esim. tietoturva, yhteensopivuus ainakin yleisimpien selaimien kanssa)
- Miten projektin pitäisi edetä käytännössä?

3.3 Teoreettinen viitekehys

Kuviossa 1 on selvitetty aiheeseen liittyvät asiat. Tässä työssä keskitytään pääasiassa portaalin suunnitteluun ja toteutukseen tiettyjä apusovelluksia käyttäen.

Palvelin on tietokone, jota tarvitaan vain portaalin toteutukseen. Siksi siihen on asennettu muutamia sovelluksia auttamaan portaalin kehittämistä, apusovelluksia. Tietoturva puoleen hoitaa tässä työssä palomuuriohjelmisto, joka on siis sovelluksen lailla asennettu palvelimelle. Se toimii vaarallisen tietoliikenteen suodattimena internetin ja palvelimen välillä takaamaan turvallisen portaalin kehitystyön.



Kuvio 1. Viitekehys

3.4 Käytettävät menetelmät, aineiston kerääminen, tulosten luotettavuus ja työn arvioitu aikataulu

Työn ohjelmointiosuus toteutetaan ilmaisella PHP-ohjelmistolla, ohjelmistoprojektimallin mukaisesti. Työn teoriaosassa käytetään apuna kirjallisuutta ja Internet-lähteitä. Työn onnistuminen voidaan todeta käyttämällä opinnäytetyössä laadittua sovellusta.

Työssä käytetään lähdekirjassa olevaa sovellettua UML-ohjelmointimenetelmää sen verran, kun tuntuu järkevältä. Sitä on siis osittain sovellettu tässäkin tapauksessa. Menetelmän päävaiheet ovat kuitenkin samat: vaatimukset, määritelmä, suunnittelu, toteutus, testaus.

Työ alkaa tammikuussa 2010 ja loppuu syksyllä 2011.

4 OHJELMISTOTUOTANTO

Ohjelmistotuotanto on tämän ja muidenkin sovellusten tärkein tai ainakin yksi tärkeimpiä asioita. Helpon sen ymmärtää siitä, miten se on virallisesti määritelty: ”The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software”. Se siis tarkoittaa hallittua sovelluksen kehitystä, mikä voidaan toteuttaa käytännössä erilaisten mallien avulla. (Haikala & Märijärvi 2004, 16.)

Pienien sovellusten kohdalla tällaisten valmiiden mallien jättämättä käyttäminen ei ole kuitenkaan yhtä vakavaa, jos ohjelman pystyy toteuttamaan hallitusti. Mitä isompi ohjelma on, sitä vaikeampaa sen hallinta on. Sama pätee moneen asiaan - suunnitteleminen tekemisiään ei onnistu yhtä todennäköisesti kuin suunnittelemalla. Vastaan tulee luultavasti erilaisia ongelmia, joita ei niin yhtäkkiä ole pystynyt huomaamaan ja ongelman vuoksi jotkin muut ohjelmakoodin osat eivät toimikaan keskenään ja jo kirjoitettua koodia täytyy korjailla tai poistaa kokonaan. Asiaan vaikuttaa myös huo-

no ohjelmointitekniikka, jolla voi pilata hyvänkin suunnitelman toteuttamalla sen huonosti. Seuraavissa kappaleissa puhutaan ohjelmistotuotannossa käytettävistä projektinhallintamalleista, niin sanotuista vaihejakomalleista ja kehitysohjon liittyvistä ongelmista.

4.1 Yleisiä ongelmia ohjelmistotyössä

4.1.1 Monimutkaisuus, näkymättömyys, muunnettavuus ja ainutkertaisuus

Ohjelmien pitää ratkaista usein monimutkaisia ongelmia, joten ohjelmienkin pitää olla monipuolisia ratkaistakseen mahdollisimman monet ongelmat mahdollisimman yksinkertaisesti. Ohjelmistosuunnittelulla yritetään tehdä ohjelmoinnista yksinkertaista. Tässä korostuu juuri ohjelmointitekniikka. Monimutkaisuuden vähentämiseksi on keksitty olio-ohjelmointi, jonka ansiosta ohjelmat saadaan helpommin jaettua moduuleihin. Mitä isompaa järjestelmää ohjelmoidaan, sitä enemmän toimintoja siinä on ja sitä monimutkaisempi siitä tulee. Tällöin olio-ohjelmointi on erityisen suositeltavaa. (Haikala & Märijärvi 2004, 28.)

Tämä ongelma ei oikeastaan haittaa harrasteliijaohjelmoijaa, vaan se on enemmänkin yritysten ongelma. Yritykset kun yrittävät pärjätä taloudellisesti. Siihen tarvitaan asiakkaita ja asiakkaat haluavat tietää heidän kustantamansa ohjelmistoprojektin etenemisestä, jotta he voisivat suunnitella omaa tulevaisuuttaan esim. sen mukaan, milloin projekti valmistuu. Tällä alalla asiakkaat maksavat teetetyistä projekteista vasta sitten, kun projektin kustannukset on selvillä riittävän tarkasti. Kustannuksien arviointi ei ole kuitenkaan yleensä helppoa, koska ohjelmointiprojektit eivät yleensä etene taasta tahtia, vaan siihen voi ihan hyvin päteä sanonta ”puoliksi suunniteltu on puoliksi tehty”. Subjektiiivisia arvioita projektin etenemisestä yritetään tarkentaa jakamalla projekti selviin vaiheisiin ja huolehtimalla laadunvarmistuksesta. (Haikala & Märijärvi 2004, 29.)

Ohjelmistoja on helppo muokata ja niin pitää ollakin. Toisaalta, muuttaminen pitää tehdä huolellisesti. On hyvä huomata, että huolimattomuudesta voi seurata vakavia-kin ongelmia projektin onnistumisen suhteen. Muutokset johtuvat yleensä siitä, että ohjelman toimintoihin halutaankin muutoksia, jolloin tehdyt ohjelmistosuunnitelmatkaan eivät välttämättä toimi kokonaisuudessaan. Samoin voi tapahtua myöhemmin, kun ohjelmisto on jo otettu käyttöön. Helpoin ratkaisu tähän on tehdä ohjelmasta hyvin ylläpidettävä. (Haikala & Märijärvi 2004, 29.)

Ainutkertaisuus voi olla ongelma, mutta ei kovin vakava sellainen. Ohjelmat ovat yleensä ainutlaatuisia, eikä toista ainakaan ihan samanlaista löydy. Siksi ei ole olemassa juuri tiettyä tapaa tehdä ohjelmia. Jos olisi, se säästäisi ohjelmistotuotannon kustannuksia. Valmiita moduuleita kannattaa käyttää apuna, jos sopivia löytyy ja jos ne on melko helppo sovittaa tekeillä olevaan sovellukseen. (Haikala & Märijärvi 2004, 30.)

4.1.2 Skaalautumattomuus ja epäjatkuvuus

Välillä ohjelmointiprojekteissa käy huonosti, mitä isommasta projektista on kyse. Kun tulee ongelma, yleensä on painetta ratkaista se mahdollisimman helposti ja halvalla. Pienissä ohjelmistoprojekteissa se onnistuu helpommin, koska järjestelmä on sen verran yksinkertainen, että projektityöntekijät ymmärtävät sen toiminnan. Silloin se on helpompi korjata. Suuri projekti taas on niin monimutkainen, että projektityöntekijät eivät voi hetkessä ymmärtää asioita kokonaisuutena ja siten korjata virheet helposti. Sellainen lisää kustannuksia, kun menee enemmän aikaa vian selvittämiseen. Uusien työntekijöiden palkkaaminenkaan ei auta, koska he ymmärtävät järjestelmästä vielä vähemmän. Kokeilun mukaan se on vain hidastanut projektia. (Haikala & Märijärvi 2004, 30.)

Epäjatkuvuus voi olla vakava ongelma sovelluksissa, mutta sitä on vaikea välttääkään. Joskus tulee kirjoitettua koodia, joka aiheuttaaakin vakavia ongelmia joissain tilanteissa. Niitä ongelmia on kuitenkin vaikea havaita, koska ne voivat tapahtua niin poikkeuksellisissa tilanteissa, että testauksessakaan ei ole huomattu sitä. Tällainen virhe voi kaataa koko järjestelmän. Sehän ei ole kovin vakavaa, jos on kyse vaikka

tekstinkäsittelysovelluksesta. Vakavampi ongelma se on sitten, jos esimerkiksi puhe-
linjärjestelmän välitysohjelma kaatuu. Niinkin on aikoinaan käynyt. (Haikala & Mä-
rijärvi 2004, 30.)

4.2 Vaihejakomallit

Tässä työssä portaalin kehittäminen etenee vaihejakomallin mukaan, kuten kaupalli-
sissa ohjelmistoprojekteissa tehdään. Yleisesti käytettyjä vaihejakomalleja on muu-
tamia: vesiputous-, prototyyppi, spiraali- ja evo-malli. Kaikki mallit perustuvat vesi-
putousmalliin ja mallien vaiheesta seuraavaan vaiheeseen siirrytään samalla kaavalla.
Vaiheesta luotua dokumenttia käytetään apuna seuraavassa vaiheessa. Eri malleja on
kehitelty siksi, että eri mallit sopivat eri tilanteisiin ja projekteihin. Toiset mallit ovat
joustavampia, toiset taas huomioivat paremmin ohjelmistotuotannon riskit. (Haikala
& Märijärvi 2004, 37.)

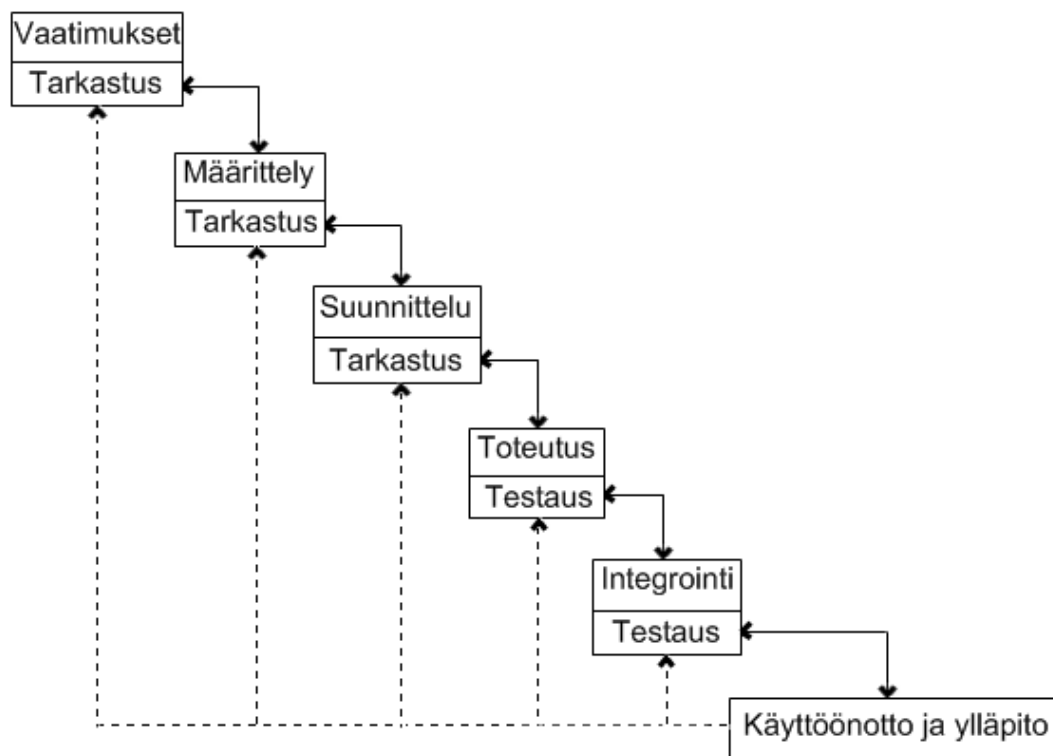
Evo-malli (katso kuvio 2) sopii tähän työhön hyvin, koska tämä projekti on pieni ver-
rattuna moniin kaupallisiin ohjelmistoihin ja mitä pienempi projekti on, sitä helpom-
pi sen hallinta on. Työtä yksinkertaistaa sekin, että teen tätä itse, joten projektinhal-
linnassa ei tarvitse kiinnittää huomiota muihin ihmisiin.

4.2.1 Vesiputousmalli

Tässä mallissa (kuvio 2) on kaikki sovelluskehityksen vaiheet, joita kaikkien ohjel-
mistoprojektien tulisi noudattaa omiin tarpeisiin sovellettuna, minkä vuoksi se onkin
yleisimmin käytetty malli. Kaikissa vesiputousmallia soveltaneissa malleissa on ole-
massa vähintään määrittely-, suunnittelu-, toteutusvaiheet. Kuviossa 2 vaiheet on eri-
telty yksityiskohtaisemmin kuin yleensä, sillä esimerkiksi vaatimukset-vaihe liittyy
määrittelyvaiheeseen. (Haikala & Märijärvi 2004, 37-38.)

Katkoviivat tarkoittavat, että tarvittaessa aiempaan vaiheeseen voidaan palata. Yleen-
sä se ei ole kuitenkaan hyvä merkki projektin onnistumisen kannalta, koska työtä
joudutaan silloin uusimaan. Takaisin joudutaan yleensä palaamaan jonkin verran,

mutta sitä ei saisi olla suhteettoman paljon. Kun vaiheesta siirrytään tai palataan johonkin vaiheeseen, tehdään vaiheen tarkastus tai testaus. Testaukseen sovelletaan V-mallia, joka selitetään tarkemmin kohdassa 4.4.5.

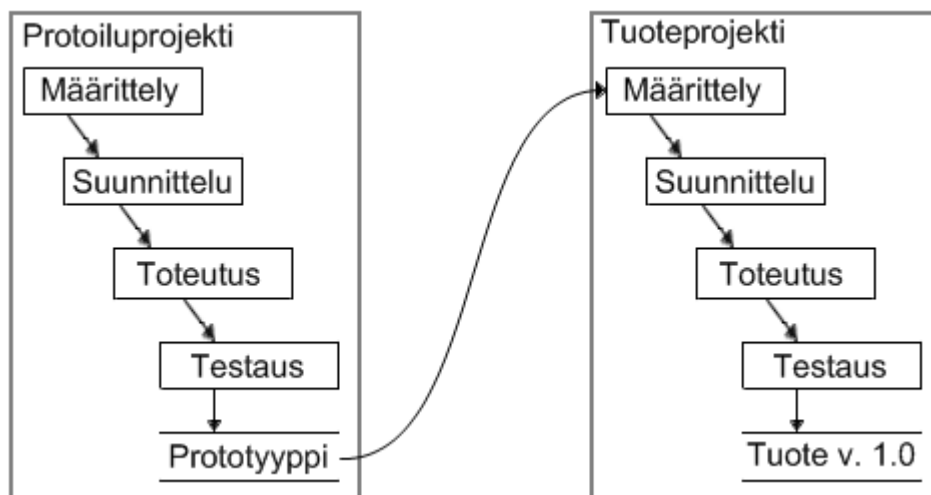


Kuvio 2. Perinteinen vesiputousmalli

4.2.2 Prototyypimalli

Prototyypimallia (kuvio 3) käytetään, kun halutaan kokeilla vaikka uutta ohjelman ominaisuutta tai selvittää yrityksen asiakkaan vaatimuksia ohjelman suhteen. Sillä testataan, että ohjelman perusidea ja runko on ymmärretty oikein, vaatimusten mukainen ja että kehitystyötä voidaan tehdä saman kaavan mukaan. Todellinen ohjelma ja proto-ohjelma ovat erillisiä projekteja, mutta todellisen ohjelman kehittäminen on se pitempikestoinen urakka. Projektit viedään läpi vesiputousmallia noudattaen. (Haikala & Märijärvi 2004, 42.)

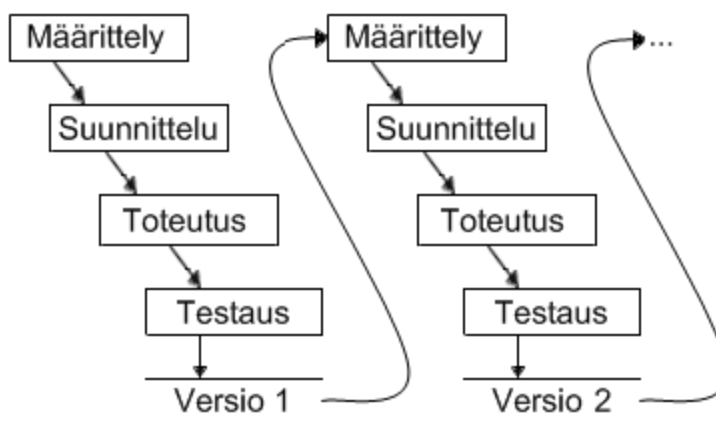
Kuviossa 3 projekti vietään läpi vesiputousmallin mukaisesti. Ensin tehdään proto-projekti ja sen jälkeen varsinainen tuote, tuoteprojekti. Tämä tuote on sen ensimmäinen versio, jota kehitellä tarvittaessa edelleen.



Kuvio 3. Prototyypimalli

4.2.3 Evo-malli

Evo-mallin (kuvio 4) kaltaisia malleja sanotaan inkrementaaliksi malleiksi, koska siinä ohjelma toteutetaan sarjalla peräkkäisiä vesiputousmalleja. Jokainen vesiputousmalli tuottaa ohjelmasta uuden version, joka on aina kehittyneempi kuin aikaisempi versio. Nykyisen version toimivuudesta ja uusista ideoista syntyy seuraavan version määrittelyvaihe, josta edetään taas tavalliseen tapaan. Ensimmäinen versio on normaalisti suurin kehitysaskel, koska ohjelma toimii kohtuullisesti jo silloin. (Haikala & Märijärvi 2004, 44-45.)

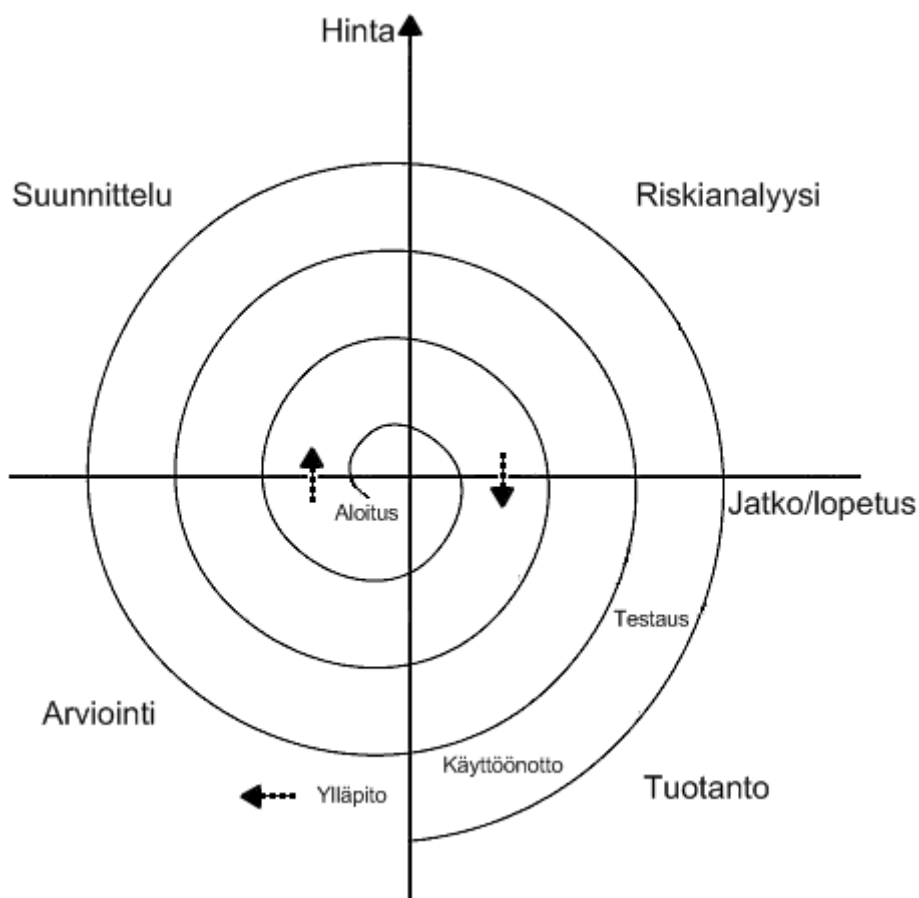


Kuvio 4. Evo-malli / inkrementaalinen malli

4.2.4 Spiraalimalli

Tämä on malleista se, joka huomioi taloudelliset asiat parhaiten. Sen toteuttaminen vaatii kuitenkin aikaa, koska jatkuva projektin kannattavuuden analysointi ja siihen perustuva suunnittelu vie aikaa ja rahaakin enemmän. Säästöä tulee siitä, kun ei tarvitse tehdä ylimääräistä työtä, esimerkiksi turhia suunnitelmia, joita ei toteutetakaan. Mallin etuihin kuuluu sekin, että iteraatiot voi viedä läpi vaikka vesiputous- tai prototyyppimallin mukaan. Spiraalimalli vain lisää siihen riskianalysoinnin. (Oulun Kauppaoppilaitos, Kehittämistyön vaiheet ja elinkaarimallit.)

Kuvion X spiraalimallissa edetään järjestyksessä vaiheesta vaiheeseen. Ensin arvioidaan projektin tulevaisuutta ja sen kannattavuutta. Sitten suunnitteluvaiheeseen, riskianalyysiin ja tuotantovaiheeseen. Projekti voidaan keskeyttää, jos sen riskit kasvavat liian suuriksi tai asiakas haluaa keskeyttää sen arviointivaiheessa, esimerkiksi arvioitua kalliimman projektin vuoksi. Mitä kauemmin projekti kestää, sen kalliimmaksi se tulee. Tämä kierre toistuu, kunnes projekti on valmis ja otettu käyttöön.



Kuvio 5. Spiraalimalli

4.2.5 Mallien vaiheet

Seuraavassa on selitetty vaihejakomallien (katso kuvat 1-5) yleisimmät vaiheet yksinkertaisesti. Ihan kaikkea ei ole kerrottu, vain portaalin kannalta tärkeimmät asiat.

Projektin vaiheiden suorittamiseen käytetään erilaisia menetelmiä, yleisimmin seuraavia: SA, OMT++ ja UML. Menetelmät tarjoavat yleensä hyväksi havaittuja tapoja viedä projekti läpi, mutta usein niitä tarvitsee soveltaa omaan projektiin sopivaksi. Menetelmät esittävät tapoja, joilla ohjelmistoprojekti kuvataan, mikä helpottaa sen suunnittelemista ja toteutusta. SA on menetelmistä vanhin ja se on väistymässä ns. oliokeskeisten menetelmien tieltä. Ohjelmointikielten luokkien ja olioiden käyttö on niin suosittua, ettei SA:ta tarvita samalla tavalla enää. Muun muassa OMT++ ja

UML ovat nykyään yleisiä oliokeskeisiä menetelmiä. Lisäksi tietokantojen mallintamiseen on suosittu ER-menetelmä (Haikala & Märijärvi 2004, 179-180.)

Määrittely-, suunnittelu- ja toteutusvaiheiden menetelmiin on kehitetty myös apuohjelmia, jotka helpottavat menetelmien toteutusta. Määrittely- ja suunnitteluvaiheiden apuvälineitä sanotaan edustavälineiksi ja toteutusvaiheen apuvälineitä taustavälineiksi. Edustavälineitä ovat esimerkiksi kuvankäsittely- ja tekstinkäsittelyohjelmat. Yleisimpiä taustavälineitä ovat kääntäjät, joilla kirjoitettu koodi muunnetaan tietokoneen ymmärtämäksi kieleksi. Helpointa ohjelmistojen kehittäjille on hankkia IPSE, Integrated Project Support Environment, johon on koottu yhdeksi ohjelmistoksi melkein kaikki edusta- ja taustavälineet. (Haikala & Märijärvi 2004, 83-84.)

Jos projektilla on asiakas, määrittelyvaiheessa selvitetään asiakkaan tarpeet ohjelmiston suhteen ja keksitään keinot niiden toteuttamiseksi. Tästä syntyy toiminnalliseksi määrittelyksi kutsuttu dokumentti, jota käytetään suunnitteluvaiheessa. Jos asiakasta ei ole, ohjelman tavoitteet selvitetään joka tapauksessa ja määritellään niiden saavuttamiseksi tarvittavat asiat. Määrittelyvaihe voidaan jakaa kahteen vaiheeseen: asiakas- ja ohjelmistovaatimusten selvittäminen. (Haikala & Märijärvi 2004, 78-79.)

Suunnitteluvaiheen tehtävänä on muuttaa määrittelyvaiheen dokumentin asiat sellaisiksi, että ne voidaan seuraavassa vaiheessa toteuttaa mahdollisimman helposti. Suunnitteluvaihe voidaan jaetaan kahteen tehtävään: arkkitehtuuri- ja moduulisuunnitteluun. Näistä tärkein on arkkitehtuurisuunnittelu, koska siitä syntyy dokumentti seuraavaan vaiheeseen. Tätä dokumenttia kutsutaan tekniseksi määrittelyksi. (Haikala & Märijärvi 2004, 81-83.)

Määrittelyvaiheen dokumentista löytyvät ohjelman ominaisuudet jaetaan arkkitehtuurisuunnitelmassa moduuleihin ja pyritään suunnittelemaan moduulien välinen toiminta niin, että ne eivät riipu toisistaan. Ohjelmistosta tulee selkeämpi, ymmärrettävämpi, tehokkaampi, luotettavampi, ylläpidettävämpi ja siirrettävämpi, jos jakaminen moduuleihin on onnistunut. (Haikala & Märijärvi 2004, 82.)

Moduulisuunnittelussa suunnitellaan moduulit niin, että ne täyttävät oman tehtävänsä. Tämän helpottamiseksi moduulisuunnittelussa kannattaa keskittyä erityisesti moduulien hierarkkisuuteen, modulaarisuuteen, rajapintojen tiivistämiseen, kotelointiin, tiedon piilottamiseen ja abstrakteihin tietotyyppeihin. (Haikala & Märijärvi 2004, 82-83.)

Suunnitteluvaiheen voi siis jakaa kahtia tai sitten ei, koska moduulisuunnittelun vaihe voidaan tehdä vasta ohjelmiston toteutusvaiheessa samalla, kun moduulit koodataan ja testataan (Haikala & Märijärvi 2004, 82-83).

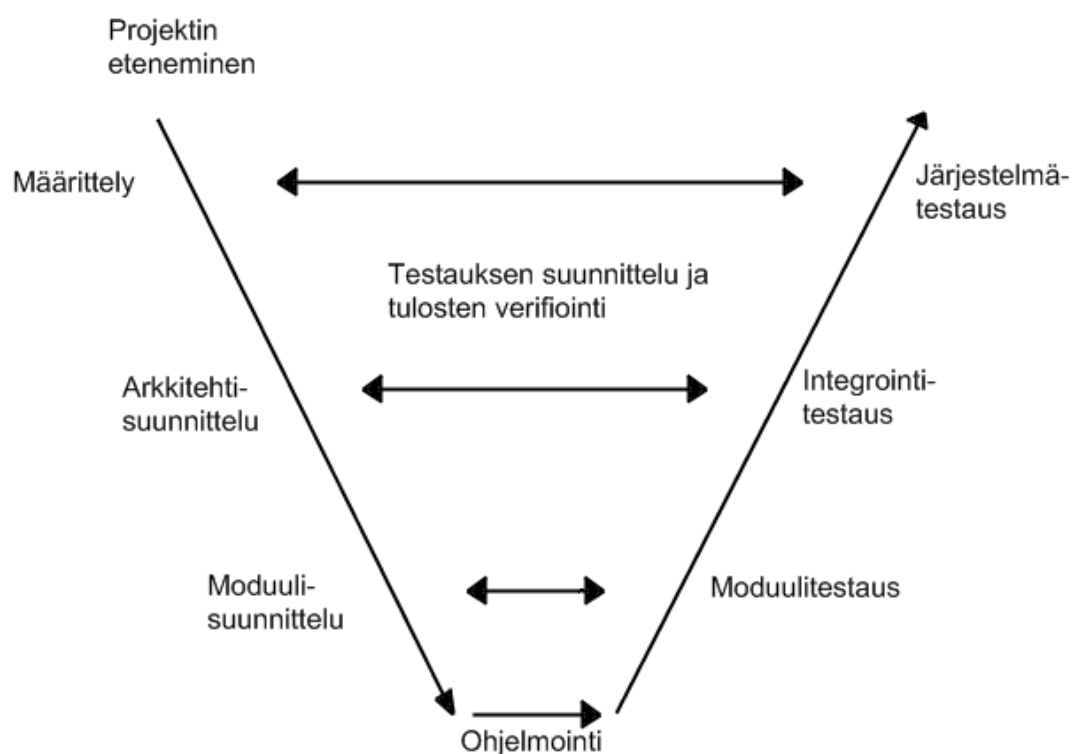
Toteutusvaihe on idealtaan yksiselitteinen. Se vie ohjelmistoprojektin vaatimasta ajasta paljon vaille puolet, jos kaikki menee tavallisesti. Toteutusvaihe käsittää kaiken koodin kirjoittamisen. Vaiheesta siirrytään testaukseen, kun on saatu aikaiseksi ensimmäinen virheetön ja toimiva versio ohjelmistosta (Haikala & Märijärvi 2004, 82-83).

Testausvaihe on merkittävä aikaa vievä vaihe, vaikka ei yhtä merkittävä kuin ohjelmiston ylläpitovaihe. Testauksen on arvioitu vievän jopa 80% ohjelmistoprojektin ajasta, sillä lähes kaikissa ohjelmistoissa on monia korjattavia virheitä. (Haikala & Märijärvi 2004, 40.)

Testaus etenee ns. v-mallin mukaan, joka saa nimensä v-kirjaimen muotoisesta kuvasta, kuten kuviossa 6. Tämäkään malli ei ole ehdoton, vaan sitä voi soveltaa projektiin sopivammaksi (Wikipedia, V-malli).

Kuviossa 6 on vasemmalla puolella määrittely, joka tarkoittaa vaihejakomallin määrittelyvaihetta. Arkkitehtuuri- ja moduulisuunnittelu yhdessä kuvaavat suunnittelu- vaihetta, ohjelmointi toteutusvaihetta ja moduuli-, integrointi ja järjestelmätestaus testausvaihetta. Niiden vieressä oleva viiva kulkee vasemmalta alaviistoon ja sieltä yläviistoon, mikä kuvaa projektin vaiheita ja ajan kulkua. Aikaviivan muodostaman V-kirjaimen välissä olevat nuolet kuvaavat, miten testaus tapahtuu projektin aikana. Jokaisella V:n vasemmalla puolella olevassa vaiheessa voidaan suunnitella vaiheen testausta ainakin alustavasti.

Oikealla puolella V:tä on varsinainen testausvaihe. Testaus tapahtuu samalla kaavalla kaikissa vaiheissa. Moduulitestauksessa testataan yksittäisiä moduuleita ja verrataan, miten moduulisuunnitelma on toteutunut kyseisen moduulin kohdalla. Integroititestauksessa moduulien yhteistoimintaa verrataan suunnitelmaan ja järjestelmätestauksessa verrataan järjestelmää/ohjelmistoa määrittelyvaiheesta syntyneeseen dokumenttiin. Testauksessa edetään siis pienimmistä yksityiskohdista koko ohjelmiston testaukseen. (Haikala & Märijärvi 2004, 40.)



Kuvio 6. V-malli

Ylläpito on viimeinen vaihe projekteissa. Sitä ennen on toki käyttöönotto vaihe, mutta siitä ei ole niin erityistä sanottavaa tässä tilanteessa. Ylläpito kestää niin kauan kuin ohjelmisto on käytössä. Testaus tulee tässä vaiheessa arvoon arvaamattomaan, koska projektin tekevän yrityksen tehtävä on hoitaa asiakkaansa ylläpito-ongelmat. Siihen kuuluu muun muassa ohjelmiston muokkaaminen, niin sanottu adaptiivinen ylläpito. Ohjelmistovirheisiin liittyviä ongelmia kutsutaan korjaavaksi ylläpidoksi ja täydentävä ylläpito tarkoittaa ohjelmiston parantamista, kuten uusien versioiden kehittämistä.

tä. Ohjelmistoprojekteissa toteutetaan vain täydentävää ylläpitoa. (Haikala & Märijärvi 2004, 41.)

5 TOTEUTUSYMPÄRISTÖ

Tässä kappaleessa kerron tärkeimmistä infoportaalin toteutukseen liittyvistä asioista, kuten joistakin käytettävistä sovelluksista. Tarkoitus on siis kertoa yleisesti tuntemattomammista asioista, eikä selitä perinpohjaisesti esimerkiksi dokumentoinnissa käytettävistä tekstinkäsittely- ja kuvankäsittelyohjelmista. Kaikkein tärkeimpänä on infoportaalin pohjana käytettävä sisällönhallintajärjestelmä PHP-Nuke, josta kerrotaankin siksi eniten.

Lähtökohtana on, että kaikki on ilmaista. Siksi kaikki tässä työssä käytettävät ohjelmistotkin ovat GPL- tai GNU GPL-lisenssin alaisia. GPL-lisenssi pitää olla lain mukaan sisällytettyinä julkisissa ilmaisohjelmistoissa, joka PHP-Nukekin siis on. Lisenssi määrittelee ohjelmiston käyttöön liittyviä lakeja. (Wikipedia, GNU General Public License.)

5.1 XAMPP

Tässä projektissa on käytössä ApacheFriends'ien ohjelmisto XAMPP 1.7.1. Versio ei ole uusin, mutta sillä ei ole portaalin toimimisen kannalta vaikutusta. XAMPP on koelma eri ohjelmistoja, joista sen nimikin tulee.

Sanan XAMPP kirjain X ei tule ohjelmiston nimestä, vaan se tarkoittaa yhteensopivuutta yleisimpien käyttöjärjestelmien kanssa. Saatavilla on lisäksi esimerkiksi vain Windowsille sopiva versio. A-kirjain tarkoittaa, että mukana on Apache HTTP Server -ohjelmisto. M-kirjain on MySQL, P-kirjain PHP ja toinen P-kirjain Perl. XAMPP:ssa on siis yhdessä ”paketissa” kaikki tarvittava ainakin tavallisten www-sivujen tekemiseen. PHP-editoria ei ole mukana. (Wikipedia, XAMPP.)

XAMPP käyttää tiettyä hakemistoa www-palvelimenaan, joten www-sivujen (alku)sivu avataan selaimella tuosta hakemistosta. Internet-osoitteeksi kirjoitetaan avatavan tiedoston nimi ja päätte ja sen hakemistopolku, kuten normaalistikin tehdään, jos halutaan viitata vaikka tiettyyn tiedostoon.

Koska kyseessä on www-palvelin, sen tiedostoihin viitataan kirjoittamalla osoitteen alkuun localhost tai vaihtoehtoisesti www-palvelimen IP-osoite. Esimerkiksi <https://localhost/xampp/op/html/index.php>. XAMPP pitää hakemistoa ”htdocs” www-palvelimena, joten infoportaalinkin kuuluu sijaita tuossa hakemistossa.

Tämä hakemisto on kuin Internet-osoite, sillä erolla, että portaalin sivut eivät ole internetissä, vaan omalla kotikoneella tai intranetissä. XAMPP:lla voi parantaa tietoturvaa ja tehdä omat www-sivut OpenSSL:llä salatun yhteyden kautta, jos vaikka intranetissä jotkut voisivat päästä katselemaan sivujasi. Tällöin Internet-osoitteen alku ”http” onkin ”https”.

5.2 Internet-selaimet

Ino-portaalin tekeminen onnistuneesti vaatii paljon testausta, joten pitää ajatella kuin käyttäjä sivuja käyttäessään. Tarkoitus on testata www-sivujen toimivuutta yleisimmin käytetyillä selaimilla. Kaikkia mahdollisia selaimia ei ole järkevää testata jo ajankin säästön vuoksi.

Testattavia ja yleisimmin käytettyjä selaimia on w3schools:n mukaan kolme: Internet Explorer 8, Firefox 3.6 ja Google Chrome 6 (W3schools, Browser Statistics). Olen asentanut koneeseen näiden selaimien uusimmat versiot, jotta infoportaali olisi mahdollisimman yhteensopiva ja nykyaikainen.

5.3 Palvelimen kokoonpano

Palvelimena toimii tässä tapauksessa oma pöytäkoneeni. Palvelimen ominaisuudet tarjoaa XAMPP-ohjelmisto. Muita koneita tai laitteita ei ole käytössä tässä projektissa, Internet-yhteyden vaatiman laitteen lisäksi. Kokoonpanolla ei ole niinkään merkitystä. Sen verran kannattanee kertoa, että käyttöjärjestelmänä on Windows XP Home Edition, jossa on uusin Service Pack 3 -päivitys asennettuna. Koneeseen on asennettu kolme Internet-selainta: Internet Explorer 8, Firefox 3.6.8 ja Google Chrome 6. Koneen suorituskyky ei rajoita tämän työn tekemistä.

6 PHP-NUKE 8.0

Tämän ilmaisprojektin valinta tämän projektin pohjaksi oli tarpeeksi hyvä, vaikka ei välttämättä paras. Mietin muutamaa eri vaihtoehtoa sisällönhallintajärjestelmäksi ja tällä tuntui helpoimmalta aloittaa infoportaalin teko, koska käyttöönotto oli ainakin tässä tapauksessa niin helppoa. XAMPP:ssa on kaikki tarvittavat ohjelmistot PHP-Nukea ajatellen. XAMPP oli jo valmiiksi asennettu koneeseen, joten minun ei tarvinnut kuin laittaa PHP-Nuken tiedostot XAMPP:n hakemistoon. Nukesta kerron vain mielenkiintoisimmista ja tärkeimmistä asioista, enkä kerro kaikista järjestelmän yksityiskohdista. Nekin selviävät sitten, jos alkaa käyttää PHP-Nukea.

6.1 Asennus

PHP-Nuken voi asentaa automaattisesti Internet-selaimen kautta tai manuaalisesti muuttamalla tiedostojen sisältöjä. Kummallakin tavalla tarkoitus on saada järjestelmän vaatimat asetukset vastaamaan palvelimen asetuksia. Esimerkiksi tiedostoa `config.php` muutetaan mm. siksi, että MySQL toimisi järjestelmän kanssa niin kuin pitäisi. Automaattinen asennus tapahtuu siis selaimen kautta. Nuken alkusivun tiedosto on alun perin `index.php`, joka löytyy siis `www`-palvelimelta. Se avataan selaimella. Jos

PHP-Nukea ei ole asennettu, sivulla näkyy kuvion 7 mukainen linkki, jota painamalla asennus aloitetaan.



**There seems that PHP-Nuke isn't installed yet.
(The values in config.php file are the default ones)**

You can proceed with the [web installation](#) now.

Kuvio 7. PHP-Nuke valmiina asennukseen

Tämä asennusjärjestelmä auttaa säätämään Nuken asetukset ja on paljon selkeämpi kuin manuaalinen asennus. Kun automaattinen asennus on suoritettu oikein, sivujen pääsivulla kerrotaan onnistuneesta asennuksesta, minkä jälkeen voikin jo aloittaa PHP-Nuken käytön.

6.2 PHP-Nuken käyttö ja sen lisenssi

PHP-Nuken käyttöä miettiessä kannattaa ottaa huomioon, että ohjelmointitaito monipuolistaa sivujen tekoa melkoisesti. Juuri PHP-kieltä ei ole pakko osata, koska senkin oppii suhteellisen nopeasti, jos tietää jo ohjelmoinnin periaatteita, tekniikoita, yms. Yksinkertaiset sivut on mahdollista tehdä ilman erikoisempia ohjelmointitaitoa, mutta monimutkaisemmat sivut on jo vaikeampaa. Internetistä löytää paljon apua, sillä sieltä on haettavissa muiden tekemiä moduuleita ja keskustelufoorumeilla voi kuka vain kysyä apua. Nuken paketin mukana tulee tärkeimmät käyttöohjeet Nukella ohjelmointiakin koskien, jos jotain jää epäselväksi tässä kerrottujen asioiden lisäksi. Seuraavassa kerrotaan asiat niin kuin ne ovat silloin, kun PHP-Nuke on juuri asennettu.

PHP-Nukeen on tehty automaattinen järjestelmä, jolla helpotetaan ja nopeutetaan www-sivujen tekemistä. Kaikkea sivuilla näkyvää pystyy kuitenkin muuttamaan, kun sivuja suunnittelee ja/tai tekee, koska PHP-Nuke on vain kokoelma tiedostoja. Siksi-kin on syytä huomata, että PHP-Nuken ilmainen käyttö estää poistamasta tekijänoikeuksista kertovaa asiaa sivuilta ja muutakin vastaavaa, mutta nämä säännöt koskevat vain www-sivujen levitystä. Kotikoneella pysyvillä tiedostoilla tällä ei siis ole väliä. PHP-Nuken kehittäjän tarjoama maksullinen lisenssi antaa luvan poistaa kaiken PHP-Nukeen liittyvän, jolloin sivuja voi käyttää hyödyntää esimerkiksi kaupalliseen toimintaan. Kun tällainen lisenssi ostetaan, sitä voidaan hyödyntää vain tietyllä verkkotunnuksella, esimerkiksi koti.fi. Jos esimerkiksi oma verkkotunnus muuttuisi, siitä pitää keskustella lisenssin myyjän kanssa.

6.2.1 Sivujen ulkonäkö ja tietokanta

Nukella tehtyjen sivujen rakenne on perinteinen, mutta sitä voi muuttaa paljonkin, jos osaa. Järjestelmä varaa sivuille neljä aluetta, jotka sisältävät osioita. Nämä osiot sisältävät moduuleita. Osiot ovat vasemmalla, oikealla, keskellä ylhäällä ja keskellä alhaalla. Moduulin sisältö näytetään keskellä sivua olevassa osiossa. Yläotsikko näytetään sivulla ylimpänä ja alaotsikko alimpana. Näillä, kuten muillakin alueilla näkyvä sisältö haetaan php-tiedostoista. Perinteisesti useimmat sivujen selaamiseen käytettävät osiot on sijoitettu vasemmalla reunalla oleviin osiin. Ne voidaan kuitenkin sijoittaa myös keskelle ylös tai keskelle alas moduulin ylä- ja/tai alapuolelle, oikealle reunalle tai vasemmalle reunalle (katso kuvio 8). Kuviossa punainen osoittaa osioiden sijainnit, vihreä moduulien, violetti yläotsikon ja sininen alaotsikon. Keskialue on merkattu punaisella ja vihreällä, joten siihen tulee moduuleja ja/tai osioita.



Kuvio 8. Nuken alkuperäinen sivujen rakenne

Osioiden sijoittamisessa kannattaa huomata, että oikealla puolella olevat osiot eivät näy aina. Tällä Nuken versiolla näkyminen määritellään moduulitiedoston alussa variablailla ”INDEX_FILE”, jolle annetaan esimerkiksi arvo true (erisuuri kuin nolla). Tämä kertoo järjestelmälle, että aina kun tämä moduuli on näkyvässä, oikean puolen osiot näkyvät. Vaikka alkusivuksi määrittäminen onkin epäloogista, se toimii hyvin. Eräs yksinkertainen vaihtoehto on sekin, että poistaa tiedostosta ”theme.php” (.../html/themes/) if-lauseen, joka tarkistaa, onko vakio ”INDEX_FILE” määritelty. Jos theme-tiedostoa muuttaa, sama muutos pitää tehdä jokaisen eri teeman theme-tiedostoon.

Keskelle alas ja keskelle ylös sijoitetut osiot näkyvät (alun perin) vain etusivulla, samalla tavalla kuin ”INDEX_FILE” määrittelee oikean puolen ja ”HOME_FILE” alaja yläosioiden näkyvyyden. Vasemman puolen osiot toimivat eri tavalla, sillä mikään vakio ei määrää sen näkyvyyttä. Koodi on kirjoitettu tiedostoon ”theme.php”. Vasemman puolen saa poistettua poistamalla funktiosta ”themeheader” rivin, jossa lu-

kee ”blocks(”left”)”. Jos haluaa muuttaa enemmän sivun rakennetta, pitää muokata teeman HTML-tiedostoja, kuten ”header.html”. Esimerkiksi ”blocks.html” määrittelee oikean ja vasemman puolen. Osioiden piilottamiselle voi olla muitakin käyttötarkoituksia, mutta yksi hyvä syy on tilan antaminen moduulille, jotta se näkyy sivulla kokonaisuudessaan.

Jos esimerkiksi pidetään kirjaa kaikista sivujen käyttäjistä, ohjelmoija luo MySQL-tietokantaan uuden taulun nimeltä ”käyttäjät”. Aina kun rekisteröidytään sivujen käyttäjäksi, tuohon tauluun lisätään rekisteröijän antamat tiedot. Ohjelmoija voi sopia ylläpitäjän kanssa, että hänen on hyvä tietää rekisteröityneet käyttäjät. Siksi ohjelmoijan pitää kirjoittaa sen mahdollistava koodi tiedostoihin, minkä jälkeen ylläpitäjä käyttää näin ohjelmoituja toimintoja hallinnoidakseen rekisteröityneiden käyttäjien tietoja. Kyse on siis selkeän käyttöliittymän tekemisestä ylläpitäjälle. Tietokanta on sivujen ydinasia, koska siellä on tarkoitus olla kaiken tiedon, joita sivulla käytetään. Tieto on ryhmitelty tietokantaan logiikan mukaisesti eri nimisiin ”tauluihin”. Tietoja muokataan, lisätään ja poistetaan tietokantakielellä, tässä työssä MySQL-kielellä. Nukea voi käyttää joillakin muillakin tietokantakielillä.

6.2.2 Ylläpidon käyttöliittymä

Kun ylläpitäjä haluaa poistaa esimerkiksi häiriköivän käyttäjän, hän voi käyttöliittymän kautta yksinkertaisesti poistaa kyseisen käyttäjän tietokannan taulusta ”käyttäjät”. Se tarkoittaa, että sivut eivät enää tunnista kyseistä käyttäjää, joten hän ei ole enää rekisteröitynyt käyttäjä. Käyttöliittymän on siis tarkoitus tehdä tietokannan käytöstä yksinkertaista muillekin kuin ohjelmointia osaaville. Ylläpitäjän käyttöliittymä on jaettu kahteen osaan: yleinen moduulien hallinta (kuvio 9) ja sivujen hallinta (kuvio 10).

Moduulien hallintaan on koottu moduulien käyttöliittymät. Ensin valitaan moduuli, johon halutaan vaikuttaa. Käyttöliittymä avautuu sivun alaosaan. Sillä voi esimerkiksi poistaa tietyn käyttäjän tai vaikka lisätä sivuille uutisen.

Modules Administration



News



Pages



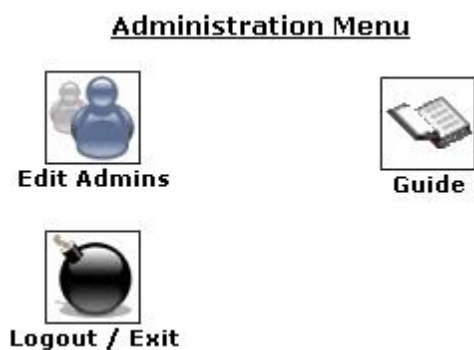
Users



Logout / Exit

Kuvio 9. Ylläpidon toimintoja moduulien hallintaan

Sivujen yleinen hallinta tarkoittaa toimintoja, jotka eivät pääasiassa liity sivujen sisällön hallintaan. Tällainen toiminto on esimerkiksi tiettyjen koneiden pääsemisen estäminen sivuille, mikä perustuu IP-osoitteisiin. Hyödyllisimmät toiminnot ovat osioiden näkyvyyden hallinta ("blocks") ja uusien tekeminen. Toinen on moduulien näkyvyyden hallinta ("blocks"). Näkyvyys tarkoittaa tässä montaa asiaa. Siihen vaikuttaa, onko osio otettu käyttöön, ketkä näkevät osion, mihin sivun alueelle osio on sijoitettu ja mikä osion tarkempi sijainti on verrattuna muihin saman alueen osioihin. Moduulin tapauksessa sille voidaan määrittää vain se, onko moduuli otettu käyttöön, ketkä sen näkevät ja onko se asetettu näkymään alkusivulla.



Kuvio 10. Ylläpidon toimintoja sivujen hallintaan

Osioiden hallitsemiseen kuuluu myös niiden lisääminen (kuvio 11). Lisättävälle osi-
olle voi antaa sisältöä valitsemalla moduuli listasta. Toinen tapa on vähän hitaampi,
mutta ei välttämättä vaadi ohjelmointitaitoa, vaikka siitä onkin hyötyä. Sisällön voi
hakea tietyistä Internet-osoitteesta tai sitten sisällöksi voi kirjoittaa HTML-koodia
ja/tai pelkkää tekstiä. Internet-osoite annetaan, kun osion sisällöksi halutaan RSS- ja
RDF-syötteitä. Tyypillisimpiä syötteitä ovat esimerkiksi päivän uutiset ja säätiedo-
tukset. Näistä kahdesta vaihtoehdosta voi valita vain jommankumman, vaikka mo-
lempien vaatimat lomakkeen kentät voidaankin täyttää. Huomion arvoista on, että
Nuke ei tee näin luodusta osiosta tiedostoa kansioon ”blocks”. Lopuksi asetetaan
muut osion asetukset niin kuin halutaan.

Ainakin sivujen ohjelmoijalle, osittain ylläpitäjällekin, on tärkeää tietää, mitä Nuken
tiedostot ja kansiot tarkoittavat ja mihin esimerkiksi moduulitiedostot pitäisi laittaa.
Seuraavassa siitä selitetään tarkemmin vain tärkeimmät kohdat.

Add a New Block

Title:

RSS/RDF file URL: Custom [Setup]
(Select Custom and write the URL or just select a Site from the list to grab news headlines)

Filename: (Select a custom Block to be included. All other fields will be ignored)

Content:

B *I* U ~~ABC~~ | |

If you fill the URL the content you write will not be displayed!

Position:

Language:

Activate? Yes No

Expiration: days

After Expiration:

Refresh Time: (Only for Headlines)

Who can View This?

Kuvio 11. Osion lisääminen

6.2.3 Nuken tiedostojärjestelmä

Tiedostojärjestelmä on rakennettu onneksi tarpeeksi loogiseksi, kuvion 13 mukaisesti. Kuviossa on kuvattu vain Nuken kansiot, ei niissä olevia tiedostoja. Niitä ympäröivät ”laatikot” tarkoittavat, että ne ovat samassa hakemistossa. Joissakin hakemistoissa, esimerkiksi moduulin ”download” hakemistossa, on myös kansio ”images”, jota ei näy kuviossa. Se ei ole kuitenkaan vakavaa, sillä mm. kansion ”images”, tarkoitus on joka paikassa sama ja sitä tietoa voi siten hyödyntää omassa ohjelmoinnissa. ”Images” kansio sisältää kyseisessä osiossa tai moduulissa käytettäviä kuvia. Kansion nimen ei tarvitse olla juuri ”images” toimiakseen, mutta kuuluu hyviin ohjelmointitapoihin käyttää nimeä ”images”. PHP-Nuke käyttää samoja nimiä samoissa tarkoituksissa, jotta kaikki ymmärtävät niiden tarkoituksen.

Alimpana on PHP-Nuken kansio ”html”, joka sisältää kaikki Nuken tiedostot. Tosin, ”sql” kuuluu myös PHP-Nukeen, mutta kansio ilmestyy oikealle paikalleen vasta asennuksen jälkeen. Asennukseen tarvitaan siis vain kansio ”html”.

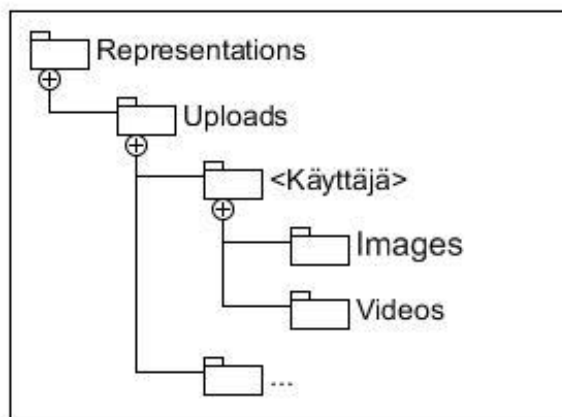
Kansiossa ”sql” on PHP-Nuken tietokanta, sql-loppuinen tiedosto. Siinä on PHP-Nuken tarvitsemat taulut taulua, joita on melkein sata kappaletta. Kaikki sivujen käyttämät taulut sisällytetään tähän samaan tietokantaan.

Tiedosto ”robots.txt” sisältää tietoa, joita internetin hakukoneet käyttävät löytääkseen sivut. Löytäminen riippuu tosin hakusanasta, joten sivut eivät löydy joka hakutuloksella. (PHP-Nuke, Directory structure).

Kansiossa ”admin” on neljä kansiota. Näissä säilytetään tiedostot, joita ylläpitäjä käyttää sivujen yleiseen hallintaan. Kansioon ”modules” luodaan uusi php-tiedosto, jos ylläpitäjän käyttöliittymään halutaan lisätä toiminto. Uuteen tiedostoon ohjelmoidaan käyttöliittymä, jolla uusi toiminto toteutetaan. Toiminnot voivat olla esimerkiksi olemassa olevien tietojen muokkaus ja uusien tietojen lisäys. Kansiossa ”links” on php-tiedosto, jonka tarkoitus on saada käyttöliittymän toiminto näkymään sivuilla. Kansion nimenkin mukaisesti, oikein ohjelmoitu tiedosto tekee sivuille linkin moduuliin, jonka sisältö ilmestyy ruudulle linkkiä painettaessa. Kansiossa ”case” olevan php-tiedoston tehtävä on ohjata järjestelmää niin, että se osaa tehdä sivujen käyttäjän haluaman toiminnon. Esimerkiksi sivuilla oleva toiminto ”poista tieto”, saa kutsumaan toimintoa (funktiota) ”poista tieto”, eikä esimerkiksi ”lisää tieto”. Viimeisenä on kansio ”language”, jolla on yksinkertainen tehtävä. Kansiossa oleva php-tiedosto sisältää tekstit, joita moduulin yhteydessä käytetään. Jokaiselle käytettävälle kielelle tehdään yksi tiedosto, esimerkiksi englannille ja suomelle. (PHP-Nuke, Directory structure; PHP-Nuke, Module creation.)

Kansiossa ”blocks” säilytetään osiot, eli niiden php-tiedostot. Kansiossa ”modules” on moduulin kansio, joka sisältää kyseisen moduulin käyttöön liittyviä asioita. Kuviossa 13 ”news” on esimerkkimoduuli. Jos moduulissa käytettäisiin kuvia, tehtäisiin uusi kansio nimeltään ”images”, johon ne kuvat laitettaisiin. Jos ylläpitäjälle halutaan antaa mahdollisuus hallita moduulia, lisätään kansio ”admin”. Tämä kansio lisää

toiminnon ylläpitäjälle, jolla hallitaan moduulia. ”Admin” kansioon riittää laittaa vain yksittäiset tiedostot, joita on yleensä kolme: ”index.php”, ”case.php” ja ”links.php”. Näiden lisäksi eri kielten tekstejä varten käytetään kansiota ”language”. Moduulin ”representations” kansiossa on erikoisuutena kansio ”uploads”, jossa sijaitsee kaikkien käyttäjien kuva- ja videotiedostot (kuvio 12).



Kuvio 12. Käyttäjien kuvien ja videoiden sijainti

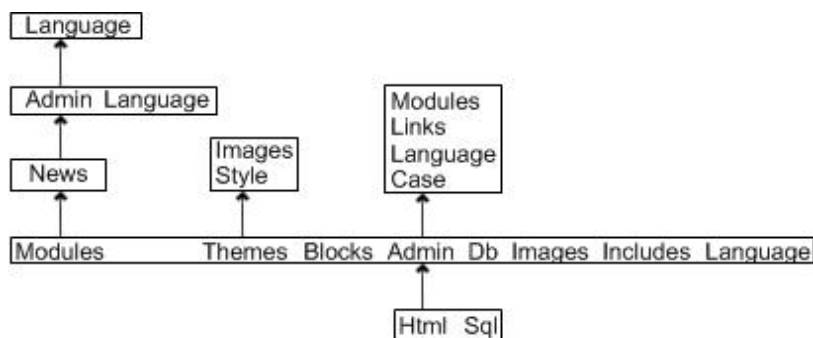
Kansiossa ”themes” on kaikki teemat, joita sivuilla voi käyttää. Jokainen teema on omassa kansiossaan, jossa on myös ainakin kansiot ”images” ja ”style”. Lisäksi on tiedosto ”tables.php”, jossa määritellään sivuilla näkyvien ympäröivien reunojen ulkonäköön vaikuttavia asioita, tarkemmin sanottuna funktioita. Näihin kuuluvat koodissakin usein käytetyt ”OpenTable” ja ”CloseTable”. Toinen tiedosto on ”theme.php”, joka vaikuttaa osaltaan mm. juuri sivujen rakenteeseen. Siinä määritellään esimerkiksi se, näkyykö sivujen oikean puolinen alue milloinkaan. Kansiossa ”images” löytyy kuvat, joita teema tarvitsee. Näitä on esimerkiksi osioita ympäröivät reunit. Kansio ”style” sisältää HTML-kielen tyylitiedostot, joita muuttamalla voidaan vaikuttaa sivujen ulkonäköön.

Kansion ”db” sisältö on tietokannan käyttöä varten. Kansion php-tiedostojen on tarkoitus mahdollistaa tietokannan käyttö kaikille käyttäjille, yhtenäistämällä käytettävän tietokantakielen komentoja.

Kansiossa ”images” on kuvia, joita sivut käyttävät. Moduulien käyttämät kuvat eivät kuulu tähän kansioon ensisijaisesti, vaan esimerkiksi sivujen yläotsikkojen kuvat ja osioiden käyttämät kuvat löytyvät tästä kansioista.

Kansiossa ”includes” on tiedostoja, jotka ovat Nuken toimimisen kannalta tärkeitä. Niitä ei tarvitse muuttaa, jos ei välttämättä halua. Jos niitä muuttaa, kannattaa perehtyä Nuken toimintaan tarkemmin. (PHP-Nuke, Directory structure).

Kansiossa ”language” on sivuilla käytettävien kielten sanat. Kielien sanat on määritelty kielen php-tiedostossa. Tämän kansion idea on samanlainen kuin esimerkiksi ”images” kansion.



Kuvio 13. PHP-Nuken tiedostorakenne

6.2.4 Nuken yleisiä ohjelmointiohjeita ylläpidettävyyden vuoksi

Kun tiedostorakenne ymmärretään ja halutaan tehdä jotain omaa sivuille, pitää tietää joitakin ohjelmointiinkin liittyviä ohjeita. Nämä tiedot löytyvät PHP-Nuken omista ohjetiedoista tai vaikka internetistä. Ohjeet kannattaa lukea, ettei tule turhia ongelmia. Ohjeiden noudattamisessa on sellainen suuri etu, että PHP-Nuken tekijä on noudattanut samoja ohjeita. Silloin voi olla varma, että PHP-Nuken koodi on yhteensopiva oman koodisi kanssa. Yksi ominaisuuksista on nimittäin se, että sen kehittämisestäkin voivat auttaa tavalliset käyttäjät, esimerkiksi lisäämällä ominaisuuksia moduulien avulla. Tämä järjestelmä ei toimi, jos kaikki käyttävät noudattavat omia

ohjeitaan. Tässä on kyse yhteensopivuudesta. Kaikki voivat asentaa PHP-Nuken tietokoneellensa, jolloin se on kaikilla samanlainen. Jos yksi henkilö muuttaa PHP-Nukeaan erilaiseksi ja lisää moduuleita miettimättä yhteensopivuusohjeita, järjestelmä ei ole enää samanlainen kuin muilla. Mikään siinä Nukessa toimivista moduuleista ei välttämättä enää toimi muiden järjestelmissä ilman koodin muuttamista, joten tällaisia moduuleita ei kannata jakaa muillekaan. Siksikin kannattaa lukea ja noudattaa ohjeita. Osioita koskevassa ohjeessa on kaksi osaa: toinen ohjelmoijalle ja toinen sivujen ylläpitäjälle. Tässä ne on liitetty yhteen, koska molemmissa on osittain samaa asiaa.

PHP-Nuke vaatii, että osion tiedoston nimi on oikeanlainen. Nimi alkaa sanalla ”block-”. Sen jälkeen tulee tiedoston nimi, jonka mahdolliset välilyönnit korvataan merkillä ”_”. Koska kyseessä on php-tiedosto, nimen lopussa pitää olla tiedoston tyyppin tunnuksena ”.php”.

Kun osio on lisätty sivuille ja otettu käyttöön, Nuke tekee, kuten tiedostoon on kirjoitettu. Kun sivut avataan esim. selaimella, vain muuttujaan ”content” tallennettu tieto näkyy. Jotta selain ymmärtää muuttujan tiedon, sen pitää olla HTML-kieltä. Jos muuttujassa ”content” ei ole mitään tietoa tai osion tiedosto poistetaan, vaikka se on käytössä, sivuilla näkyy sen osion kohdalla virheilmoitus. Koska vain muuttujassa oleva tieto on näkyvää, osio voi sisältää paljonkin koodia, josta sivuilla vieraileva ei tiedä mitään.

Tietoturvankin vuoksi PHP-Nuken ohjeena on, että tiedostoa ei voida avata suoraan selaimella. Avaamisen pitäisi tapahtua vain sivujen kautta, esimerkiksi linkkiä painamalla. Näin estetään se, että sivuille lähetettävää tietoa voitaisiin muokata selaimen osoiterivin kautta.

Osion tiedoston muuttujan ”content” sisältö ei saisi olla liian suuri, jotta sisältö mahdusi näkymään sivuilla. Liian suuri osio saa sivut näyttämään rumilta ja epäkäytännöllisiltä, koska silloin sisältö ei mahdu ruudulle ja sivuja pitää siirtää selaimen avulla, jotta näkee kaiken.

Jotta yhteensopivuus Internet-selaimien kanssa olisi paras mahdollinen, huomaa ainakin nämä asiat: HTML-kielellä kirjoitetuissa Internet-osoitteissa merkki "&" pitäisi korvata sanalla "&" ja HTML-kielen komentoja "FONT" ja "LI" ei pitäisi käyttää. Tyylitiedostoissa niitä tosin voi käyttää. Merkintää "UL" ei käytetä, koska se saa sivut näyttämään rumilta. Internetistä löytyy tavallisia merkkejä vastaavat HTML-kielen merkinnät, esimerkiksi osoitteessa <http://www.ascii.cl/htmlcodes.htm>.

Moduleillekin on omat ohjeet, joissa on samaa kuin osioita koskevissa ohjeissa. Moduulien kansiot pitää nimetä niin, että välilyönnit korvataan merkillä "_". Yleensä tiedostojen nimien kirjaimien koilla ei ole väliä, mutta PHP-Nuke voi ymmärtää samannimisetkin tiedostot eri tiedostoiksi.

Linkit moduuleihin muodostetaan seuraavasti. Jos esimerkiksi halutaan linkki moduuliin "news", siihen on kaksi tapaa riippuen siitä, mitä halutaan tehdä. Jos sivujen alkusivuksi on määritelty "index.php", eikä tarkoituksena ole kutsua funktiota, linkin osoitteeksi kirjoitettaisiin "https://localhost/xampp/html/modules.php?name=News". Jos alkusivuksi onkin asetettu esimerkiksi "startpage.php" ja moduulin halutaan kutsuvan funktiota "function", kirjoitetaan esimerkiksi näin: "https://localhost/xampp/html/modules.php?mop=modload&name=News&file=startpage&switchname=function&id=2". Osoitteessa on monta osaa/sanaa, jotka tekevät eri asioita. Sanan "name=" perään kirjoitetaan ladattavan moduulin nimi. Merkinnän "file=" jälkeen tulee alkusivun nimi ilman tiedoston päätettä, tässä esimerkissä "startpage". Merkinnän "switchname=" jälkeen tulee käytettävän funktion nimi. Funktion pitää silloin olla määriteltynä moduulin tiedostossa "startpage.php". Merkintä "id=2" tarkoittaa, että funktiolle välitetään tieto, joka on tässä esimerkissä numero kaksi. Funktiolle voidaan lähettää samalla tavalla muutakin tietoa. Tässä tapauksessa funktion ensimmäinen muuttuja saa id:n arvon. Kaikki osoitteessa olevien muuttujien nimet pitää olla samassa järjestyksessä funktiossa, jotta tiedot välittyisivät kuten ohjelmoija haluaisi. Siis, jos id:n jälkeen olisi määritelty muuttuja "testi", funktion toinen muuttuja saa "testin" arvon.

HTML-kielen lomaketoimintoa käytetään usein, erityisesti käyttöliittymien ohjelmoinnissa. Nukessa sitä käytetään niin, että lomakkeen tiedot lähetetään moduulitie-

doston ”index.php” funktiolle, joka hoitaa jatkokäsittelyn. Funktio pitää siis ensin määrittellä moduulitiedostoon. HTML-kielenä voitaisiin kirjoittaa esimerkiksi `<form action="modules.php?name=News&file=index&func=function" method="post"></form>`.

Moduulien ohjelmoinnissa on sama tietoturvaan liittyvä asia kuin osioidenkin ohjelmoinnissa. Käyttäjä ei saa päästä käsiksi sivujen tiedostoihin käyttämällä selaimen osoiteriviä. Siksi moduulitiedoston alkuun lisätään koodia, joka estää tämän. Käyttäjä voi käyttää moduuleita vain linkkien kautta.

Käytä Nuken kielijärjestelmää kirjoittamalla koodi moduulitiedoston alkuun. Esimerkit toimivista koodeista löytää Nuken ohjeista tai katsomalla mallia muista moduuleista.

Kun kirjoittaa koodia ja käyttää kielen käskyä ”switch”, sen kanssa ei pitäisi käyttää muuttujaa ”mop”, koska sitä käytetään eri tarkoitukseen Nukessa. Nukessa switch:n kanssa käytetään yleensä muuttujaa ”op”.

Nyt kun tunnetaan ohjelmointia koskevat ohjeet ja moduuli on valmiina, se pitää vain laittaa sivuille. Sitä ennen kannattaa suunnitella, miten tekee sivut. Moduulithan ovat erillisiä osia, joiden on tarkoitus toimia mahdollisimman itsenäisesti ja toteuttaa omaa tehtävänsä. Tätä periaatetta käytetään ihan kaupallisissakin ohjelmistoissa.

Nukessa sivut kootaan osioista ja moduuleista. Esimerkiksi osioita voidaan käyttää käyttöliittymän rakentamisessa ja moduuli voi toimia sitten käyttöliittymästä tulleiden tietojen perusteella. Moduulin lisääminen sivuille voidaan tehdä kahdella tavalla: lisäämällä sivulle linkki moduuliin ja/tai ottamalla käyttöön osio ”modules”.

Osiota ”modules” (kuvio 14) on helpompi ja nopeampi käyttää kuin ohjelmoimalla omat linkit jokaiselle moduulille, mutta vain tämän osion käyttö rajoittaa sivujen suunnittelua pahasti, koska tällöin moduulien linkit ovat samassa osiossa.

Moduulin asetuksia voidaan muokata Nuken kautta niin, että vain tietty käyttäjärhymä voi käyttää moduulia. Tietyn ryhmän käyttäjät, esimerkiksi tietyt ylläpitäjät, pys-

tyvät käyttämään kaikkia moduuleita, vaikka ne olisivatkin kielletty tai jopa otettu pois käytöstä. Tämä helpottaa ylläpitäjän työtä, koska moduuliin tehtyjä muutoksia on helppo kokeilla, kun kukaan muu ei voi käyttää tätä moduulia. Jos muokkaa käytössä olevaa moduulia, kannattaa varoa liian radikaaleja muutoksia ja tehdä ne joskus myöhemmin. Yksi tapa olisi varmuuskopioida sivut toiseen paikkaan ja kokeilla moduulin ja sivujen muutoksia niin, että sivuilla ei voi olla muita käyttäjiä. Tavalliset sivujen käyttäjät eivät pysty käyttämään kuin käytössä olevia moduuleita, ellei käyttäjästä tehdä ylläpitäjää.

Ylläpitäjäksi tekeminen tapahtuu Nuken kautta antamalla käyttäjälle lupa hallita joi-tain tiettyjä moduuleja. Niin sanottu ”super user” on toiseksi korkein arvo, joka voidaan antaa. Se antaa samat ylläpitotoiminnot kuin itse ylläpitäjällä, paitsi ”super user” ei voi vaikuttaa ylläpitäjiin. Näin paljon oikeuksia ei kannata antaa kenelle tahansa, vaan henkilön on paras olla hyvä tuttu. Korkein arvo on ”god admin”, jonka Nuke olettaa olevan sen asentanut henkilö. Asennuksen yhteydessä tehdään nimittäin käyttäjätunnuksetkin. Näin käy ainakin silloin, jos PHP-Nuke on asennettu käyttöliittymän kautta.

”Modules” osio toimii niin, että se jakaa moduulikansiota löytyvät moduulit kolmeen osaan niiden ominaisuuksien mukaan ja tekee linkin jokaiseen moduuliin. Osi-ossa ensimmäisenä ovat käytössä olevien moduulien linkit. Toisessa osassa ovat linkit käytössä oleviin moduuleihin, joiden linkki on piilotettu. Viimeisenä ovat linkit moduuleihin, jotka on otettu pois käytöstä.



Kuvio 14. Linkkeiä moduuleihin ”modules” osion sisällä

”Modules” osion linkkeihin voidaan vaikuttaa käyttämällä ylläpitäjän toimintoa ”modules”, jota käytetään muutenkin moduulien hallintaan. Silloin sivujen alaosaan ilmestyy lista, jossa on kaikki moduulikansiosta löydetyt moduulit samaan tapaan kuin kuviossa 15. Tätä kautta voidaan esimerkiksi ottaa moduuli pois käytöstä tai vaikka antaa jollekin käyttäjäryhmälle mahdollisuus käyttää moduulia.

Modules and Addons

See your Modules/Addons current status and change it by Activating or Deactivating them.
New Modules copied on the */modules/* directory will be automatically added with *inactive* status to the database when you reload this page.
If you want to remove a module just delete it from the */modules/* directory, the system will automatically update your database to show the changes.

== WARNING ==
Bold module's title represents the module you have in the Homepage.
 You can't Deactivate or Restrict this module while it's the default one!
 If you delete the module's directory you'll see an error in the Homepage.
 Also, this module has been replaced with *Home* link in the modules block.

[•] means a module which name and link will not be visible in Modules Block

Title	Custom Title	Status	Visible to	Group	Functions
[•] Advertising	Advertising		All Visitors	None	
AvantGo	AvantGo		All Visitors	None	
Content	Content		All Visitors	None	
Downloads	Downloads		All Visitors	None	
Encyclopedia	Encyclopedia		All Visitors	None	
FAQ	FAQ		All Visitors	None	

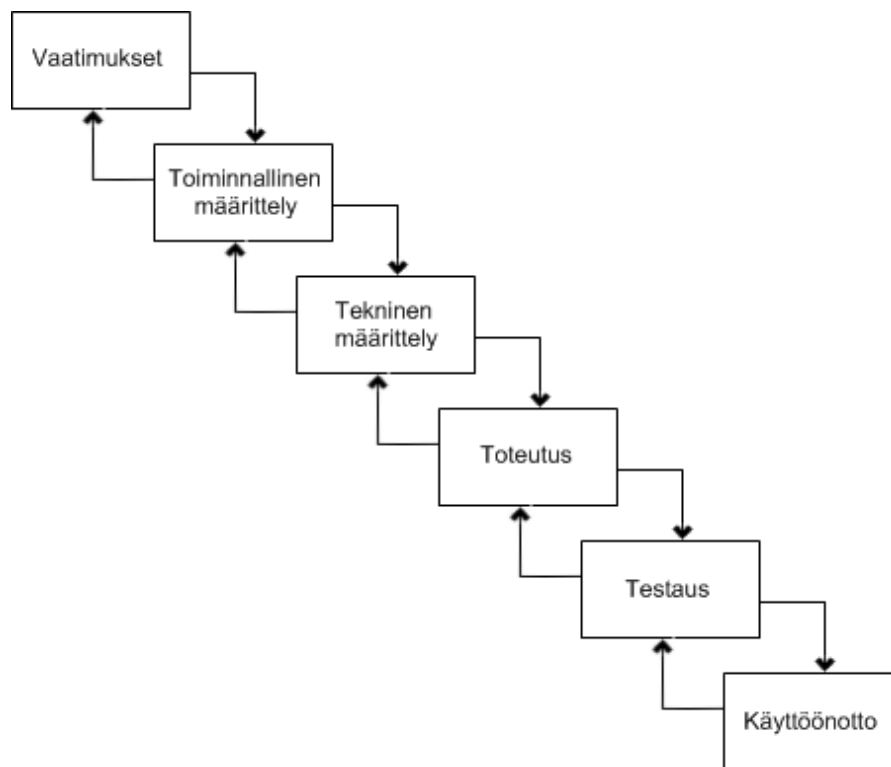
Kuvio 15. Moduulien hallintaa

6.2.5 Esimerkki projektin tehtävien jaosta

Todellisuudessa tehtävät on yleensä jaettu niihin erikoistuneille henkilöille. Ohjelmointi on ohjelmoijan tehtävä, joten www-sivujen ylläpitäjän ei tarvitse osata ohjelmointia. Otetaan vaikka tilanne, jossa kaksi henkilöä on päättänyt tehdä www-sivut. Toinen toimii ohjelmoijana ja toinen sivujen ylläpitäjänä. Tällainen tilanne voisi siis olla todellisuudessakin. Tässä tapauksessa ohjelmoija kirjoittaisi koodit tiedostoihin ja ylläpitäjän tehtäväksi jäisi muut asiat, kuten vastaaminen käyttäjien lähettämiin viesteihin. Yhteistyötä tietenkin tarvitaan, mutta ylläpitäjän ei välttämättä tarvitse tietää ohjelmoinnista mitään. Tämä on mahdollista, koska PHP-Nuke tarjoaa mahdollisuuden ohjata sivujen toimintaa pääosin sivujen kautta, koodiin koskematta. Ensin pitää vain päättää, miten ja mitä asioita ylläpitäjä voi muuttaa.

7 INFOPORTAALI

Tässä portaalissa käytetään kuviossa 16 olevaa sovellettua vesiputousmallia, jota käytetään iteratiivisesti. Käyttöönottovaihetta ei tosin tarvita tässä tapauksessa, mutta sekin pitää ottaa huomioon siltä varalta, että joskus tarvitaan.



Kuvio 16. Infoportaalin vaihejakomalli

Infoportaalin suunnittelussa noudatetaan osittain kirjan esimerkkiä, johon liittyy mm. UML, ER ja vuokaaviot (Kolehmainen 2006, 344). Kirjan esimerkistä kannattaa poiketa käytännöllisyyden ja Nukenkin vuoksi.

Perinteinen UML:kään ei sovi, koska siinä mallinnetaan luokkia, eikä infoportaalissa käytetä luokkia. Portaalin suunnittelussa pitää huomioida PHP-Nuken vaikutus myös muulla tavoin. Osa portaalista on jo valmiina, joten portaali syntyy Nuken valmiista ominaisuuksista ja omista lisätyistä ideoista ja ominaisuuksista.

Seuraavassa on ensin mietitty järjestelmän vaatimukset. Niiden jälkeen on vuorossa toiminnallinen ja tekninen määrittely, jotka on yhdistetty yhden otsikon alle vaiheiden lyhyden ja samankaltaisuuden vuoksi. Loput kolme vaihetta ovat erillisten otsikoiden alla.

7.1 Vaatimukset

Tällä infoportaalilla pitää voida esitellä yritystä tai palvelua PHP-pohjaisilla www-sivuilla. Vaikka tätä työtä ei tehdäkään asiakkaalle, osaa opinnäytetyöongelmasta voidaan pitää asiakasvaatimuksina, jotka yritetään saada toteutettua.

Opinnäytetyöongelmahan oli tämä: ”Tutkimusongelmana on laatia internetissä toimiva, PHP-kielillä tehty infoportaali itse, osittain tai kokonaan valmiilla sovelluksella ja laatia portaaliin sen käyttöohje. Infoportaalin avulla voidaan esitellä jotakin yritystä ja/tai palvelua tekstin, kuvien ja videoiden avulla. Portaalin toimintaa kuvaamaan siihen on asennettu esimerkkipalvelu.”

Ongelmasta saadaan esille ainakin seuraavia (asiakas)vaatimuksia.

- Portaalilla esitellään palvelua ja/tai yritystä
- Portaalin pitää perustua PHP- ja MySQL-kieliin.
- Portaalin käyttämiseen pitää olla ohjeet.
- Esittely tapahtuu teksteillä, kuvilla ja videoilla.
- Portaalissa on valmiina käytännön esimerkkinä esittely jostain palvelusta
- Portaalin pitää olla yhteensopiva yleisimpien selaimien kanssa ja muutenkin hyvien tapojen mukaisesti tehty.
- Portaali pitää voida siirtää mahdollisimman helposti www-palvelimelle.

7.2 Toiminnallinen ja tekninen määrittely

Portaaliin tulevia toimintoja ja asiakasvaatimuksia tarkennetaan käyttötapauskaavioilla ja sanallisesti. Niillä voidaan hakea samalla suuntaa portaalin käyttöliittymälle.

Vierailijan toiminnot

- o Sisäänkirjautuminen
- o Portaalin esittelyn lukeminen
- o Ota yhteyttä

- o Käyttäjän salasanan vaihtaminen

Käyttäjän toiminnot

- o Uloskirjautuminen
- o Salasanan vaihtaminen
- o Teeman vaihto
- o Ota yhteyttä
- o Uutiset
- o Käyttöohjeet
- o Sisältö
- o Sivut
- o Esittelyt
- o Äänestämisen
- o Esityksen käynnistäminen

Ylläpitäjän toiminnot

- o Uloskirjautuminen
- o Käyttöohjeet
- o Käyttäjien hallinta
- o Käyttäjien sisältöjen hallinta
- o Moduulien hallinta
- o Osioden hallinta
- o Kiellettyjen IP-osoitteiden hallinta
- o Tietokannan kopiointi
- o Tietokannan optimointi
- o Sivujen ominaisuuksien muuttaminen
- o Ylläpitäjien hallinta
- o Yhteydenotot
- o Uutiset
- o Äänestysten hallinta
- o Vierailijoiden ja käyttäjien toimintojen käyttäminen

Vierailijan toiminnot selitettynä:

- Sisäänkirjautuminen
Vierailija antaa käyttäjätunnuksen ja salasanan. Jos annetut tiedot vastaavat tietokannan tietoja, järjestelmään kirjaudutaan kyseiseksi käyttäjäksi ja portaalin uutiset tulevat esiin. Jos käyttäjätunnus ja/tai salasana ovat väärin, niitä kysytään uudestaan. Ylläpitäjäksi kirjauduttaessa pätevät samat asiat.
- Portaalin esittelyn lukeminen
Sivulla kerrotaan sanallisesti yleistietoa portaalista, kuten miten tunnuksen voi tilata ja miten portaalista voi kokeilla sitä varten tehdyllä tunnuksella.
- Ota yhteyttä
Vierailija voi kysyä infoportaalista tai tilata tunnuksen ylläpitäjältä antamalla kysytyt tiedot ja lähettämällä ne nappia painamalla.
- Käyttäjän salasanan vaihtaminen
Käyttäjän salasanan voi vaihtaa antamalla tunnusta vastaavat tiedot niin kuin ne ovat.

Käyttäjän toiminnot selitettynä:

- Uloskirjautuminen
Käyttäjistä tulee vierailija ja portaalin alkusivu avautuu.
- Salasan vaihtaminen
Käyttäjä voi vaihtaa käyttäjätunnuksensa salasanan kirjoittamalla uuden salasansa kahteen kertaan.
- Teeman vaihtaminen
Käyttäjä voi vaihtaa sivujen ulkonäköä ja joissain tapauksissa mm. linkkien sijainteja ennalta määriteltyjen vaihtoehtojen mukaan.
- Ota yhteyttä
Käyttäjä voi kysyä ylläpitäjältä asioista ja antaa palautetta infoportaalista täyttämällä lomakkeessa kysytyt tiedot ja lähettämällä ne ylläpitäjälle nappia painamalla.
- Uutiset
Käyttäjä näkee infoportaaliiin liittyvät uutiset sisään kirjautuessaan tai käyttäjävalikossa/-menussa olevan linkin kautta.
- Käyttöohje

Infoportaalin ohjeet ja niiden sisällysluettelo selittävät, miten portaalia käytetään. Ohjeet sisältävät esimerkin.

- **Sisältö**
Käyttäjä voi lisätä sisältöä, joita voidaan käyttää omissa esityksissä. Sivulla näkyy lisäksi jo lisätyt tekstit, kuvat ja videot, joita voi muokata.
- **Sivut**
Käyttäjä voi lisätä sivuja, joita käytetään omissa esityksissä. Sivuille voidaan lisätä itse tehtyä sisältöä, johon kuuluu tekstit, kuvat ja videot. Sivulla näkyy lisäksi jo tehdyt sivut, joita voi muokata.
- **Esitykset**
Käyttäjän tärkein apuväline, jota käytetään esityksien tekemiseen ja niiden muokkaamiseen.
- **Äänestäminen**
Käyttäjä voi antaa oman äänensä mielipidekyselyihin. Sama käyttäjä voi antaa yhden äänen jokaiseen kyselyyn.
- **Esityksen käynnistäminen**
Käyttäjä voi käynnistää haluamansa esityksen valitsemalla sen listasta, jossa on kyseisen käyttäjän esitykset aakkosjärjestyksessä.

Ylläpitäjän toiminnot selitettynä:

- **Uloskirjautuminen**
Ylläpitäjä asetetaan vierailijaksi ja alkusivu avautuu.
- **Käyttöohjeet**
Ylläpidon ohjeet. Siinä selitetään lyhyesti toiminnot ja niiden käyttö.
- **Käyttäjien hallinta**
Tällä ylläpitäjä voi hallita infoportaalin kaikkia käyttäjiä.
- **Käyttäjien sisällön hallinta**
Ylläpitäjä voi katsoa ja hallita käyttäjän esityksiä, sisältöjä ja sivuja.
- **Moduulien hallinta**
Tällä ylläpitäjä voi hallita infoportaalin kaikkia moduuleja.
- **Osioiden hallinta**
Ylläpitäjä voi hallita infoportaalin kaikkia osioita.
- **IP-osoitteiden hallinta**

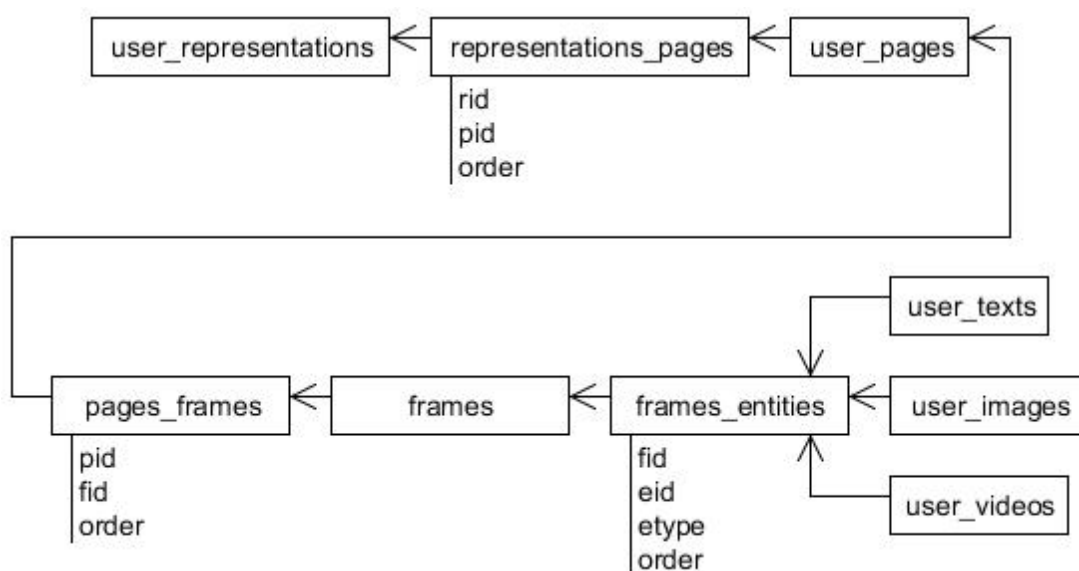
Näyttää listan IP-osoitteista, joilta on estetty pääsy sivuille. Listan yläpuolella on lomake IP-osoitteiden kieltämiseksi.

- Tietokannan kopiointi
Toiminto kopioi sivujen käyttämän tietokannan valinnaiseen paikkaan, kuten kovalevylle tai muistitikulle.
- Tietokannan optimointi
Toiminto pienentää tietokannan kokoa poistamalla niistä jotain turhaa tietoa.
- Sivujen ominaisuuksien muuttaminen
Mahdollistaa sivujen yleisten asetusten muuttamisen, kuten sivujen nimen, Internet-osoitteen.
- Ylläpitäjien hallinta
Näyttää listan sivujen ylläpitäjistä, joista jokaista voi hallita saman listan kautta. Listan vieressä on toiminto, jolla tehdään käyttäjästä ylläpitäjä.
- Yhteydenotot
Sisältää viestit, jotka vierailijat ja käyttäjät ovat lähettäneet ylläpitäjälle. Samalla sivulla näkyy viestit, jotka on haluttu säästää. Säästettyjä viestejä ei voi muokata, muilla tavoin viestejä voi hallita.
- Uutiset
Sivulla ovat listattuna nykyiset uutiset, joita voidaan hallita listan kautta tai antamalla uutisen tunnus. Samalla sivulla on mahdollista kirjoittaa uusia uutisia ja julkaista ne heti tai vasta tiettyyn aikaan. Niitä voi katsella etukäteen ennen julkaisua.
- Äänestysten hallinta
Äänestyksiä poistetaan, muutetaan ja luodaan tällä toiminnolla. Äänestyslouksia ei voi muuttaa.
- Pikalinkit
Nämä linkit ovat kahdessa eri osiossa. Toinen niistä helpottaa ylläpitäjiä muistuttamalla heitä yhteydenotoista. Toisessa osiossa on linkki ylläpitäjän sivulle, jolla sivuja hallitaan.
- Vierailijoiden ja käyttäjien toimintojen käyttäminen
Ylläpitäjä näkee käyttäjien ja vierailijoiden toimintovalikot niin kuin ne ovat. Niitä käytetään samalla tavalla, mutta ylläpitäjät jakavat kaiken sisällön, kuten esitykset ja niiden sisällöt.

7.2.1 ER-malli

ER-mallissa kuvataan portaalissa käytettävien tietokantojen taulut ja niiden väliset yhteydet, jotka auttavat ymmärtämään taulujen välistä kommunikointia. Tämä on esitetty kuviossa 17. Infoportaali koostuu yhdestä tietokannasta, jossa on kaksi osaa, koska ne liittyvät eri asioihin.

Tämä ensimmäinen ja suurin osa liittyy käyttäjien (ja ylläpitäjien) esityksiin ja niihin liittyviin asioihin, kuten sivuihin. Taulut, joihin on merkitty niiden muuttujia, esimerkiksi ”rid”, tekevät käyttäjille mahdolliseksi käyttää esimerkiksi monia esityksiä. Tähän tarvittaisiin periaatteessa vain kaksi muuttujaa, mutta esimerkiksi ”order” tarvitaan, jotta esityksissä käytettävien sivujen sisältö pysyy sen tekijän haluamassa järjestyksessä. Taulujen nuolet osoittavat tauluun, johon tämä kyseinen taulu sisältyy logiikan mukaan. Näin tästä saadaan ketju, josta saadaan haettua tietoa mistä kohdasta ketjua tahansa. Ketju on hyvä sana kuvaamaan tätä, koska ketjun lenkkien yli ei voi ”hypätä”, muuten tietoja ei saada niin kuin pitäisi. Jos esimerkiksi halutaan saada selville tietyn käyttäjän tietyn esityksen käyttämä sivu, käytetään kuvan kolmea ensimmäistä taulua: user_representations, representations_pages, user_pages.



Kuvio 17. Tietokannan taulujen väliset yhteydet

Seuraavat kuvat selvittävät tarkemmin, millaisia kuvion 17 taulut ovat.

Ominaisuus	Tietotyyppi	Pitää olla arvo?	Arvoalue	Huomautukset
id	bigint	ei	0-20	avain, auto_increment
rid	bigint	ei	0-20	"representation" indeksi
pid	bigint	ei	0-20	"page" indeksi
order	int	ei	0-3	

Kuvio 18. Representations_pages

Ominaisuus	Tietotyyppi	Pitää olla arvo?	Arvoalue	Huomautukset
id	bigint	ei	0-20	avain, auto_increment
owner_email	varchar	ei	0-64	
title	varchar	ei	0-64	
structure	varchar	ei	0-32	

Kuvio 19. User_pages

Ominaisuus	Tietotyyppi	Pitää olla arvo?	Arvoalue	Huomautukset
id	bigint	ei	0-20	avain, auto_increment
pid	bigint	ei	0-20	"page" indeksi
fid	bigint	ei	0-20	"frame" indeksi
order	int	ei	0-2	

Kuvio 20. Pages_frames

Ominaisuus	Tietotyyppi	Pitää olla arvo?	Arvoalue	Huomautukset
id	bigint	ei	0-20	avain, auto_increment
frame	int	ei	0-2	

Kuvio 21. Frames

Ominaisuus	Tietotyyppi	Pitää olla arvo?	Arvoalue	Huomautukset
id	bigint	ei	0-20	avain, auto_increment
fid	bigint	ei	0-20	"frame" indeksi
eid	bigint	ei	0-20	teksti/kuva/video indeksi
etype	int	ei	0-3	
order	int	ei	0-3	

Kuvio 22. Frames_entities

Ominaisuus	Tietotyyppi	Pitää olla arvo?	Arvoalue	Huomautukset
id	bigint	ei	0-20	avain, auto_increment
owner_email	varchar	ei	0-64	
text	text	ei		

Kuvio 23. User_texts

Ominaisuus	Tietotyyppi	Pitää olla arvo?	Arvoalue	Huomautukset
id	bigint	ei	0-20	avain, auto_increment
owner_email	varchar	ei	0-64	
source	varchar	ei	0-255	
file	varchar	ei	0-64	

Kuvio 24. User_images

Ominaisuus	Tietotyyppi	Pitää olla arvo?	Arvoalue	Huomautukset
id	bigint	ei	0-20	avain, auto_increment
owner_email	varchar	ei	0-64	
source	varchar	ei	0-255	
file	varchar	ei	0-64	

Kuvio 25. User_videos

Toinen tietokannan osa liittyy sivujen toimintoon ”ota yhteyttä” ja on hyvin yksinkertainen. Yhteydenotot tallennetaan tauluun ”contacts” (kuvio 28). Yhteydenotossy pitää hakea erillisestä taulusta ”contact_categories” (kuvio 27), koska infoportaalii on kaksikielinen. Jos kieliä lisätään, kyseiseen tauluunkin pitää lisätä muuttujia. Seuraavat kaksi kuvaa selvittävät taulujen sisällön.

Ominaisuus	Tietotyyppi	Pitää olla arvo?	Arvoalue	Huomautukset
id	bigint	ei	0-20	avain, auto_increment
f_category	varchar	ei	0-30	suomi
e_category	varchar	ei	0-30	englanti
viewer	int	ei	0-1	vierailijoille vai käyttäjille?

Kuvio 26. Contact_categories

Ominaisuus	Tietotyyppi	Pitää olla arvo?	Arvoalue	Huomautukset
id	bigint	ei	0-20	avain, auto_increment
send_date	varchar	ei	0-64	
category	varchar	ei	0-30	
subject	text	ei		
message	text	ei		
sender_email	varchar	ei	0-64	
answered	tinyint	ei	0-1	

Kuvio 27. Contacts

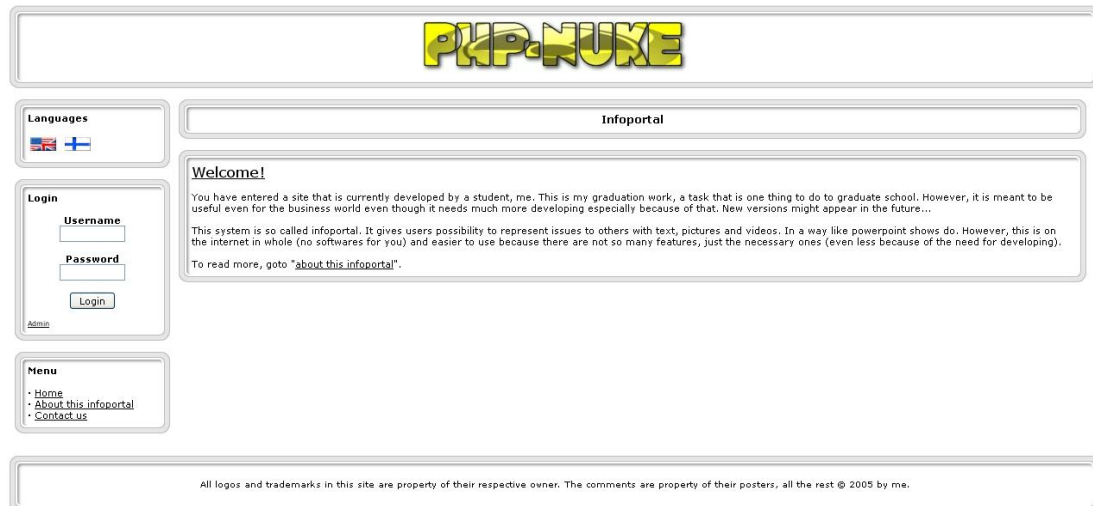
8 TOTEUTUS

Toteutus etenee yleensä tutkimalla Nuken valmista koodia ja muuttamalla sitä ja tietokannan tauluja tarvittaessa. Tämä on taitavalle ohjelmoijalle suhteellisen helppoa, koska moduulit on koteloitu ja Nuken käyttö on sopivan helppo oppia. Jos PHP-Nuke ei tarjoa toimintoa edes osittain, moduuli ja siihen liittyvä täytyy tehdä itse ja/tai etsiä internetistä toiminnon pohjaksi muiden tekemää koodia.

Sivujen tekeminen etenee suunnitellun mallin mukaan (kuvio 16). Ensin toteutetaan vierailijan toiminnot, minkä jälkeen on loogisestikin helpompi siirtyä käyttäjän ja ylläpidon toimintojen toteuttamiseen. Toteutusvaiheeseen kuuluu dokumentointia, tietokannan ja koodin muokkaamista.

Kuviossa 29 on infoportaalin etusivu, joka avautuu kaikille vierailijoille. Sivun keskiossa kerrotaan yleistietoa infoportaalista, jotta palveluun tutustujat tietävät siitä enemmän ilman linkkien painamista. Käyttäjätunnuksen omaavat kirjautuvat login-

ikkunan ja ylläpito samassa ikkunassa olevan admin-napin kautta. Infoportaali on kaksikielinen. Oletuskielenä on suomi. Alimpana vasemmalla ovat toiminnot, jotka sallitaan vierailijoille. Toiminnot vaihtuvat sivulla kävijän profiilin mukaan, eli onko profiilin käyttäjä vai vierailija. Ylläpito näkee molempien profiilien toiminnot erillisinä ikkunoina.



Kuvio 28. Infoportaalin alkusivu

Seuraavassa kerrotaan MySQL-taulujen luontikoodit. Niiden jälkeen kohdasta 8.2 alkaen toiminnoista näytetään kuvio ja kerrotaan, miten ne toimivat käytännössä. Taulujen väliset yhteydet (ER-malli) ja osa muuttujista ovat pysyneet samoina. Taulut ja ER-malli näkyvät sivuilla 45-48.

8.1 Taulujen MySQL-koodit

Joidenkin taulujen muuttujia piti vaihtaa erilaisiksi suunnitelluista, mutta ideat toimivat muuten melko hyvin. Järjestelmän kaikkia tauluja ei kerrota, koska ne sisältyvät PHP-Nukeen. Tässä lopulliset taulut ja kuinka ne tehdään MySQL-koodilla.

Taulu ”contact_categories”:

```
CREATE TABLE contact_categories (
  id bigint(20) NOT NULL auto_increment,
  f_category varchar(30) NOT NULL,
```

```

    e_category varchar(30) NOT NULL,
    viewer int(1) NOT NULL,
    PRIMARY KEY (id)
);

```

Taulu "contacts":

```

CREATE TABLE contacts (
    id bigint(20) NOT NULL auto_increment,
    send_date varchar(64) NOT NULL,
    category varchar(30) NOT NULL,
    subject text NOT NULL,
    message text NOT NULL,
    sender_email varchar(64) NOT NULL,
    answered tinyint(1) NOT NULL,
    PRIMARY KEY (id)
);

```

Taulu "user_representations":

```

CREATE TABLE user_representations (
    id bigint(20) NOT NULL auto_increment,
    owner_email varchar(64) NOT NULL,
    title varchar(64) NOT NULL,
    PRIMARY KEY (id)
);

```

Taulu "representations_pages":

```

CREATE TABLE representations_pages (
    id bigint(20) NOT NULL auto_increment,
    rid bigint(20) NOT NULL,
    pid bigint(20) NOT NULL,
    order int(3) NOT NULL,
    PRIMARY KEY (id),
    KEY rid (rid, pid)
);

```

);

Taulu "user_pages":

```
CREATE TABLE user_pages (  
    id bigint(20) NOT NULL auto_increment,  
    owner_email varchar(64) NOT NULL,  
    title varchar(64) NOT NULL,  
    structure varchar(32) NOT NULL,  
    PRIMARY KEY (id)  
);
```

Taulu "pages_frames":

```
CREATE TABLE pages_frames (  
    id bigint(20) NOT NULL auto_increment,  
    pid bigint(20) NOT NULL,  
    fid bigint(20) NOT NULL,  
    order int(3) NOT NULL,  
    PRIMARY KEY (id)  
);
```

Taulu "frames":

```
CREATE TABLE frames (  
    id bigint(20) NOT NULL auto_increment,  
    frame int(2) NOT NULL,  
    PRIMARY KEY (id)  
);
```

Taulu "frames_entities":

```
CREATE TABLE frames_entities (  
    id bigint(20) NOT NULL auto_increment,  
    fid bigint(20) NOT NULL,  
    eid bigint(20) NOT NULL,  
    etype int(3) NOT NULL,
```

```
order int(3) NOT NULL,  
PRIMARY KEY (id)  
);
```

Taulu "user_texts":

```
CREATE TABLE user_texts (  
  id bigint(20) NOT NULL auto_increment,  
  owner_email varchar(64) NOT NULL,  
  text text NOT NULL,  
  PRIMARY KEY (id)  
);
```

Taulu "user_images":

```
CREATE TABLE user_images (  
  id bigint(20) NOT NULL auto_increment,  
  owner_email varchar(64) NOT NULL,  
  source varchar(255) NOT NULL,  
  file varchar(64) NOT NULL,  
  PRIMARY KEY (id)  
);
```

Taulu "user_videos":

```
CREATE TABLE user_videos (  
  id bigint(20) NOT NULL auto_increment,  
  owner_email varchar(64) NOT NULL,  
  source varchar(255) NOT NULL,  
  file varchar(64) NOT NULL,  
  PRIMARY KEY (id)  
);
```

8.2 Ota yhteyttä












Vierailijan ja käyttäjän toiminto, jota käytetään kommunikointiin palveluntarjoajan kanssa ilman, että tarvitaan sähköpostia. Ylläpito tosin tarvitsee sähköpostia vastatesaan käyttäjän lähettämiin viesteihin. Kuviossa 30 näkyy käyttöliittymä viestin lähettämiseen.

Käyttäjä valitsee lähimmin viestiänsä kuvaavan aihealueen laatikosta ”message category”. Kohtaan ”subject” kirjoitetaan viestin aihe tarkemmin, minkä jälkeen kirjoitetaan itse viesti. Kirjoittamisessa on mahdollista käyttää HTML-kieltä tai muita ”message” osion alareunalla näkyviä toimintoja. Kun viesti on valmis, annetaan sähköpostiosoite, johon vastaus halutaan lähetettävän. Lopuksi viesti lähetään ylläpidolle painamalla nappia ”send”.

Message category
Question about infoportal ▼

Subject

Message

B ***I*** **U** ABC |     |   |   |   

Email (will be answered by us)

Kuvio 29. Käyttäjän käyttöliittymä

8.3 Käyttäjän salasanan vaihtaminen

Vierailijan toiminto, jolla vaihdetaan käyttäjätunnuksen salasana (kuvio 31). Jotta kuka tahansa ei voisi vaihtaa salasanaa, kaikki tunnuksen tiedot pitää tietää ja antaa ne ja uusi salasana (kahteen kertaan). Jos tunnuksen tiedot ovat oikeat, salasana vaihtuu uuteen.

Lost your Password?

No problem. Enter requested information of your account to prove you immediately after login with it.

Username:	<input type="text"/>
Email:	<input type="text"/>
Real name:	<input type="text"/>
Phone (without country number):	<input type="text"/>
Company:	<input type="text"/>
Address:	<input type="text"/>
Zip code:	<input type="text"/>
City:	<input type="text"/>
New password:	<input type="text"/>
Retype password:	<input type="text"/>

Kuvio 30. Unohtunut salasana vaihdetaan tällä lomakkeella

8.4 Portaalin esittelyn lukeminen

Vierailijan toiminto. Sivulla kerrotaan yleistä tietoa infoportaalista ja yritetään houkutella lisää käyttäjiä. Tämä näkyy kuviossa 32.

Welcome!

You have entered a site that is currently developed by a student, me. This is my graduation project, which is useful even for the business world even though it needs much more developing especially

This system is so called infoportal. It gives users possibility to represent issues to others via the internet in whole (no software for you) and easier to use because there are not so many things to develop (developing).

Trying this out

Probably, you want to try this out before deciding whether to use this or not. For that reason you will understand if it is a proper system for your needs. Everything can not be done as

The username for demo user is "demo" and the password is "de12mo34".

Ordering

If you liked it, ordering can be done only by filling a simple form at the menu on the left. Complete the rest of the form.

Write to "message" part the following information about you (account info):

- Username
- Real name
- Email
- Phone
- Company
- Home address
- Zip code
- City
- Country

The e-mail is needed so that you will be answered. We will check whether the given information is correct. If you do not want to use it, tell it in your message. We will make an account for you after we have received your validation by e-mail and when you have paid.

Monthly cost is 2€ and it should be paid until the 3rd day of the month. If we do not receive payment happens until the 8th day, your account will be closed and all the information related to it will be deleted.

Kuvio 31. Infoportaalin esittelysivu

8.5 Sisäänkirjautuminen

Vierailijan toiminto, jolla järjestelmään kirjaututaan sisään. Käyttäjäksi kirjautuminen tapahtuu antamalla tunnuksen käyttäjänimi ja salasana ja lopuksi painetaan "Login" -nappia (kuvio 33).



The image shows a login form with the following elements:

- Login** (Section Header)
- Username** (Label) followed by a text input field.
- Password** (Label) followed by a text input field.
- Login** (Submit Button)
- [Admin](#) (Link)

Kuvio 32. Käyttäjän sisäänkirjautuminen

Ylläpitäjäksi kirjaututaan sisään painamalla samassa osiossa olevaa pientä ”Admin” -nappia, jolloin avautuvalle sivulle annetaan käyttäjänimi ja salasana ja painetaan ”Login” -nappia (kuvio 34).



The image shows an administrator login form with the following elements:

- Admin ID** (Label) followed by a text input field.
- Password** (Label) followed by a text input field.
- Login** (Submit Button)

Kuvio 33. Ylläpitäjän sisäänkirjautuminen

8.6 Uloskirjautuminen

Ylläpitäjän ja käyttäjän toiminto, jolla kirjaututaan järjestelmästä ulos. Se tapahtuu painamalla ”Logout” -nappia (kuvio 35). Uloskirjautumaan pystyy joiltakin muiltakin infoportaalin sivuilta.



The image shows a user interface element with the following text:

- Login** (Section Header)
- User:
- [[Logout](#)]

Kuvio 34. Uloskirjautuminen

8.7 Salasanan vaihtaminen

Käyttäjän toiminto, jolla vaihdetaan käyttäjätunnuksen salasana. Vaihtaminen tapahtuu antamalla uusi salasana kahteen kertaan (kuvio 36).

Username:	Testaaja
Real Name:	Testi Henkilö
Email:	testi.henkilö@gmail.com
Phone (without country number):	000-3101331
Company:	Tmi. Testaajat
Address:	Testaajantie 1
Zip code:	12345
City:	Testi
Country:	Testimaa
Password:	<input type="text"/> <input type="text"/> (type a new password twice to change it)

Kuvio 35. Käyttäjän salasanan vaihtamiseen tarkoitettu lomake

8.8 Teeman vaihto

Käyttäjän toiminto, jolla muutetaan sivujen ulkonäköä ja mahdollisesti linkkien sijaintia (kuvio 37).

Select One Theme

▼

This option will change the look for the whole site.
The changes will be valid only to you.
Each user can view the site with different theme.

Kuvio 36. Käytettävän teeman valinta

8.9 Käyttöohjeet

Käyttäjän toiminto. Selittää infoportaalin käyttämiseen liittyviä asioita, kuten mitä kaikkea järjestelmällä voi tehdä ja miten ne tehdään. Asioita ei selitetä täysin tarkasti, koska oletettavasti käyttäjä osaa käyttää tietokonetta sen verran.

Ohjeet on jaettu sisällysluetteloon ja käyttöohjeisiin. Sisällysluettelon kautta pääsee lukemaan juuri haluamansa otsikkoon liittyviä ohjeita klikkaamalla sisällysluettelon otsikon nimeä.

Contents

Introduction

Languages box

Login box

Menu box

Survey

Example of using the infoportal

Introduction

This is an introduction about infoportal. This guide is made to help you to use all the features of the infoportal. It is not necessary because the infoportal is planned to be user-friendly (especially, for the elderly).

The features are as follows:

- Show and make representations to audience or anyone else.
- Include texts, videos and pictures into your representations.
- Give your opinion for surveys and see the current results of it.
- Be in contact with the staff of the infoportal. You will be answered to your e-mail.
- See the news of the infoportal.

We hope that the infoportal will be useful for you!

User interface

Languages box

Choose a language (english or finnish) for this site.

Login box

Used to login with your account info as well as logout. After login you start at your home page.

If you ever lose your account password, press "login" button. A new site will be opened where you must be able to give the requested account information. If the given info is valid, you will be able to login with this new password. For maximum security, a password should be different from the old one. LISÄÄ TÄHÄN!!!

Menu box

- **Home:** The front page that is shown when you enter this site.
- **Representations:** Your finished representations are here too. After login you can see your own representations and the representations of others. It is possible to see the representations of others.
- **News:** The news about the infoportal. The newest news is at the top and the oldest news is at the bottom.
- **My info:** Shows your personal page and the news below it. It is possible to see the news of others.

Kuvio 37. Käyttäjän ohjeet

8.10 Sisältö

Käyttäjän toiminto, jota käytetään esityksissä käytettävien sivujen sisällön lisäämiseen, muokkaamiseen ja muuhun hallintaan. Sisällöllä tarkoitetaan tekstejä, kuvia ja videoita.

Sivujen yläosassa on linkkejä (kuvio X), joiden kautta pääsee nopeasti lisäämään haluamaansa sisältöä, esimerkiksi videoita.



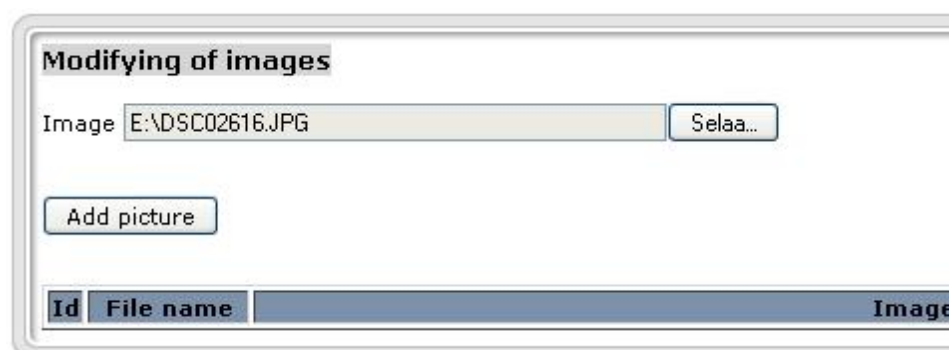
Kuvio 38. Pika-linkit sisällön hallinnoimiseen

Tekstien lisäämisessä käytetään PHP-Nuken mukana tulevaa kätevää (avoimena lähdekoodina jaettavaa) TinyMCE-editoria, jolla voi kirjoittaa tavallista tekstiä tai sen saa halutessaan muotoiltua HTML:ksi (kuvio 40). Lisättyjä tekstejä voi jälkeensä muokata ja poistaa.



Kuvio 39. Tekstien lisäämistä TinyMCE-editorilla

Kuvien lisäämisestä ei saa yhtä monimutkaista kuin tekstien. Kun kuva lisätään, ”Selaa...” -napilla haetaan tietokoneelta/muistitikulta tai muulta vastaavalta lisättävä kuvatiedosto ja painetaan ”Add image” -nappia (kuvio 41). Lisättyjä kuvia voi jälkeensä muokata, poistaa ja katsoa oikeassa koossa, mikä tapahtuu kuvaa klikkaamalla.



Kuvio 40. Kuvien lisäämistä

Videoiden lisääminen toimii samalla tavalla kuin kuvien, etsitään kuvatiedosto ja lisätään nappia klikkaamalla (kuvio 42). Videoitakin voi muokata, poistaa ja katsoa, mutta lisäksi niitä voi ladata sivuilta takaisin kovalevylle tms. laitteelle.



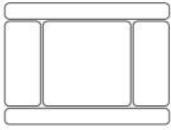
Kuvio 41. Videoiden lisäämistä

8.11 Sivut

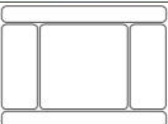





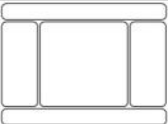


Käyttäjän toiminto, jolla hallitaan esityksissä käytettäviä sivuja. Sivun lisäämiseksi sille annetaan nimi, valitaan sivun rakenne ja sijoitetaan aikaisemmin luotu sisältö sivulle (kuvio 43).

Add a new page

Title

Structure 

Available pages

Title	Structure	Actions
Sivu		 
Sivu2		 
Sivu3		 

Kuvio 42. Esitysten sivujen tekemistä

Kun rakenne valitaan, sisällön sijoittamista varten avautuu ruutuja, joissa on valittu rakenne ja käytettävät tekstit, kuvat ja videot (kuvio 44). Kuvia ja videoita voi halutessaan katsella oikeassa koossaan. Sisältöä lisätään sivulle valitsemalla ensin sijoitettava asia klikkaamalla sen edessä olevaa ympyrää. Sitten painetaan sen osan ”Add” -nappia, johon valittu sisältö halutaan sijoittaa, jolloin ”laatikkoon” ilmestyy sisältöä kuvaava teksti. Se on kaksiosainen. Ensimmäinen osa kertoo sisällön tyyppin (teksti, kuva vai video) ja toinen osa sisällön tunnuksen (”id”), joka näkyy valitun sisällön rivillä viimeisessä sarakkeessa. Sisällön poistaminen tapahtuu valitsemalla ”laatikosta” jokin lisätty sisältö ja painamalla ”Remove” -nappia. Laatikossa olevien asioiden järjestystä voi muuttaa napeilla ”Up” ja ”Down”. Tällä on paljon merkitystä, koska sivua katseltaessa laatikossa ylimpänä oleva sisältö näytetään ensin ja alimpana oleva viimeisenä. Lisättyjä sivuja voi myöhemmin muokata ja poistaa.

Page

The screenshot shows a web page editor interface. At the top, there is a 'Page' section with a central control panel containing 'Add', 'Remove', 'Up', 'Text 13', and 'Down' buttons. Below this are three content boxes, each with its own 'Add', 'Remove', 'Up', and 'Down' controls. The middle box also has 'Text 18' and 'Text 19' labels. Below the content boxes is another central control panel with 'Add', 'Remove', 'Up', and 'Down' buttons. At the bottom, there are two tables. The first table is titled 'Text' and lists content items with their IDs. The second table is titled 'Image' and lists an image with ID 37.

	Text	Id
<input type="radio"/>	ghjyyu!!!	14
<input type="radio"/>	<u>JOHDANTO</u>	18
<input type="radio"/>	<u>Otsikko</u>	13
<input type="radio"/>	Linkki Linkki2	5
<input checked="" type="radio"/>	Tämä on aiheen johdanto.	19

	Image	Id
<input type="radio"/>		37

Kuvio 43. Sisällön lisäämistä sivulle

8.12 Esitykset

Käyttäjän toiminto. Tätä käytetään esitysten hallintaan (kuvio 45). Esityksen tekemiseksi, sille annetaan nimi ja lisätään valmiiksi tehtyjä sivuja niin monta kuin haluaa. Sama sivu voidaan lisätä esitykseen vain kerran. Kun sivu valitaan, sen nimi ilmestyy alla olevaan ”laatikkoon”. Sen voi poistaa laatikosta ”Remove” -napilla ja sen järjestystä esityksessä voi muuttaa napeilla ”Up” ja ”Down”. Esitykselle annetut tiedot voi poistaa kerralla painamalla ”Reset” -nappia. Esitys lisätään käyttöön napilla ”Add representation”. Sitä voi sen jälkeen muokata, poistaa ja katsella.

Add a new representation

Title

Choose pages

<input type="radio"/>	Sivu	
<input type="radio"/>	Sivu2	
<input checked="" type="radio"/>	Sivu3	

Available representations

Id	Title	Actions
26	Esitys	

Kuvio 44. Esitysten tekemistä

8.13 Äänestäminen

Käyttäjän toiminto. Tällä toiminnolla voi antaa oman äänensä infoportaalia koskeviin mielipidekyselyihin, katsoa senhetkisiä äänestystuloksia ja katsoa aikaisempia mielipidekyselyitä (kuvio 46).

Äänestäminen tapahtuu valitsemalla omaa mielipidettään parhaiten vastaava vaihtoehto ja painamalla ”Vote” -nappia. Äänestystulokset näkee ”Results” -linkin kautta ja aikaisemmat mielipidekyselyt ”Polls” -linkin kautta.

Survey

Koska opinnäytetyö loppuu?

Nyt
 Huomenna
 Joskus
 Ei koskaan
 En tiedä
 Ei ole alkanutkaan

[Results](#)
[Polls](#)
[Votes 2](#)

Kuvio 45. Mieliidekysely

8.14 Esityksen käynnistäminen

Käyttäjän toiminto, jolla olemassa olevat esitykset voi käynnistää niin, että se on kätevä, eikä sitä tarvitsisi hakea turhan kauan, jos on esimerkiksi vieraita ihmisiä katsomossa. Käyttäjän tarvitsee vain kirjautua sisään käyttäjätunnuksellaan ja valita käynnistettävä esitys (kuvio 47).

Title

Esitys

Kuvio 46. Käyttäjän lisäämät esitykset ovat listassa

Kun esitys on käynnistetty, sitä käytetään normaalisti vain yhdellä, oikealle osoittavalla nuolen muotoisella napilla, joka avaa esityksen seuraavan sivun. Edelliselle sivulle pääsee toiseen suuntaan osoittavalla ”nuolella”. Sivujen välillä voi vaihdella suoraan painamalla halutun sivun nimen mukaista linkkiä.

Sivu → [Sivu2](#)



Kuvio 47. Esi-
tyksen ohjaus-
toiminnot

Eesityksen sivu näkyy sellaiseksi kuin se on tehty (kuvio 49), mutta tämän kirjoitus-
hetkellä siinä olisi parantamisen varaa.

<u>Otsikko</u>		
Linkki Linkki2 Linkki Linkki2	<u>JOHDANTO</u> Tämä on aiheen johdanto.	

Kuvio 48. Eesityksen sivu

8.15 Uutiset

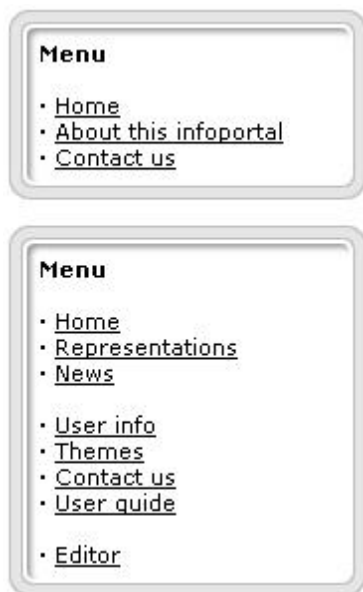
Käyttäjän toiminto, joka näyttää ylläpitäjien lähettämiä, infoportaalia koskevia artik-
keleita. Ne ovat aikajärjestyksessä uusimmasta vanhimpaan. Katso kuvio 50.



Kuvio 49. Infoportaalin uutiset

8.16 Vierailijoiden ja käyttäjien toimintojen käyttäminen

Ylläpitäjä voi käyttää vierailijoiden ja käyttäjien toimintoja. Ylläpitäjän ei tarvitse olla ”god admin”. Vierailijoiden käyttöliittymä on kuvion 51 ylemmässä osiossa ja käyttäjien käyttöliittymä alemmassa.



Kuvio 50. Vierailijoiden ja käyttäjien käyttöliittymät ylläpitäjän näkökulmasta

8.17 Äänestysten hallinta

Ylläpitäjän toiminto, jolla muokataan, poistetaan ja tehdään uusia mielipidekyselyitä (kuviot 51 ja 52). Kyselyitä poistetaan napilla ”Delete polls”, jolloin avautuu uusi ruutu. Siinä näkyy lista kyselyistä, joista valitaan poistettava kysely ja painetaan listan alla olevaa nappia. Samalla tavalla toimii kyselyiden muokkaaminen napilla ”Edit polls”. Uuden kyselyn lisäämistä varten annetaan sille nimi, kyselyn kieli, vaihtoehdot ja lopuksi lisätään kysely napilla ”Create poll”. Tämä kysely tulee näkyviin etusivulle, joten käyttäjät tulevat vastaamaan siihen, mikä kannattaa ottaa huomioon kyselyä tehdessä.

Create a New Poll
[\[Delete Polls | Edit Polls \]](#)

Poll Title:

Language: ▼

Please enter each available option into a single field

Option 1:

Option 2:

Option 3:

Option 4:

Option 5:

Option 6:

Option 7:

Option 8:

Option 9:

Option 10:

Option 11:

Option 12:

Kuvio 51. Uusien mielipidekyselyiden tekeminen

8.18 Uutisten hallinta

Ylläpitäjän toiminto, jota käytetään käyttäjille tarkoitettujen uutisten muokkaamiseen, poistamiseen ja uusien luomiseen. Uuden artikkelin (uutisen) voi määrittää näkyväksi heti tai johonkin tiettyyn aikaan.

Muokkaamisen ja poistamisen voi tehdä kahdella eri tavalla: ”laatikon” avulla tai käyttämällä nappeja uutisen perässä viimeisessä sarakkeessa. Laatikkoon kirjoitetaan käsiteltävän uutisen ”Id”, valitaan viereisestä ”alasetolaatikosta” haluttu toiminto ja painetaan nappia ”Go!”. Katso kuvio 53.

Last 20 Articles

Story ID:

Id	Title	Language	Actions
16	Artikkeli 2	english	 
15	Artikkeli 1	english	 
8	Artikkeli 3	english	 

Kuvio 52. Uutisten muokkaaminen ja poisto

Uutisen lisäämiseksi on oma lomake (kuvio 54). Sitä varten annetaan uutisen otsikko, kieli, lyhyt versio uutisesta, kokonainen versio uutisesta, uutiseen liittyviä huomioita ja uutisen julkaisuaika. Lopuksi valitaan alasetolaatikosta joko uutisen esikatselu tai kuitataan valmiiksi. Sitten painetaan nappia ”Ok!”.

Title:

Language:

Story Text:

B I U ABC | |

Extended Text:

B I U ABC | |

(Are you sure you included an URL? Did you test it for typos?)

Notes:

B I U ABC | |

Schedule News: Yes No

Publish on: Day: Month: Year: @ Hour: : : 00

Now is: May 26, 2011 @ 15:55:49

Kuvio 53. Uutinen lisätään tällä lomakkeella

8.19 Yhteydenotot

Ylläpidon toiminto, jolla kommunikoidaan käyttäjien kanssa (kuvio 55). Tavallinen toiminnon käyttötapa on lukea viesti, vastata siihen ja lopuksi poistaa se. Jos viesti on asiaton tai roskaposti, sen voi poistaa suoraan tai vaikka ryhtyä jatkotoimenpiteisiin lähettäjän kanssa. Tiettyä viestiä koskevat toiminnot ovat viestin viimeisessä sarakkeessa. Viestit on jaettu käsittelemättömiin ja arkistoituihin viesteihin.


Handle contacts				
Sent	Sender	Category	Subject	Actions
23.5.2011 klo 14:14.25	testi.henkilö@gmail.com	Feedback	Sivujen hyöty?	 

Archived contacts				
Sent	Sender	Category	Subject	Actions
23.5.2011 klo 14:10.52	testi.henkilö@gmail.com	Question about infoportal	Mikä php-nuke on?	 

Kuvio 54. Ylläpidon saamat viestit

8.20 Ylläpitäjien hallinta

Varsinaisen ylläpitäjän ("God Admin") toiminto. Tällä määritetään, ketkä ylläpitävät infoportaalia varsinaisen ylläpitäjän lisäksi. Kaikki ylläpitäjät näkyvät listassa, jossa niiden käyttäjätunnusten tietoja voidaan muokata tai poistaa (kuvio 56).

Edit Admins		
Admin ID	Languages	Functions
admin (<i>God Admin*</i>)	all	 
apuri	english	 

*(GOD account can't be deleted)

Kuvio 55. Sivujen ylläpitäjiä hallitaan tämän kautta

Uusi ylläpitäjä luodaan erillisellä lomakkeella (kuvio 57). Järjestelmä ei toimi niin, että olemassa olevalle käyttäjätunnukselle annetaan ylläpitäjän oikeudet, vaan ylläpitäjälle tehdään oma tunnus. Siihen tarvitaan ylläpitäjän nimi, käyttäjänimi, sähköpostiosoite, käytettävä kieli, hallinnoitavat moduulit, salasana ja painetaan nappia "Add Author". Ylläpitäjälle voi antaa luvan hallinnoida melkein samat oikeudet kuin varsinaiselle ylläpitäjälle valitsemalla kohdan "Super User". Kaikkia lomakkeen kohtia ei välttämättä tarvitse täyttää heti, mutta se kannattaa tehdä kuitenkin jossain vaiheessa.

Add a New Administrator

Name: (required, can't be changed later)

Username: (required)

Email: (required)

Language: ▼

Permissions:

<input type="checkbox"/> Contact us	<input type="checkbox"/> Images	<input type="checkbox"/> News
<input type="checkbox"/> Pages	<input type="checkbox"/> Representations	<input type="checkbox"/> Surveys
<input type="checkbox"/> Texts	<input type="checkbox"/> Videos	<input type="checkbox"/> Your Account

Super User

WARNING: If Super User is checked, the user will get full access!

Password (required)

Kuvio 56. Lomake uuden ylläpitäjän lisäämiseen

8.21 Sivujen ominaisuuksien muuttaminen

Ylläpitäjän toiminto, jonka tarkoitus on mahdollistaa sivujen ominaisuuksien muuttaminen, kuten sivujen nimi, www-osoite, käyttäjän salasanan vähimmäispituus. Tämä sivu on jaettu eri osiin riippuen siitä, mihin asiaan jokin ominaisuus liittyy.

Ensin näkyy sivun tärkeimmät asiat (kuvio 58): sivujen nimi, infoportaalin Internet-osoite, sivujen luontipäivämäärä, varsinaisen ylläpitäjän sähköpostiosoite, sivujen alkuperäinen teema, vierailijoiden ja käyttäjien oletuskieli ja tapa, jolla sivuilla ovat ajankohdat muotoillaan (esimerkiksi kellonajat).

General Site Info	
Site Name:	<input type="text" value="Information portal"/>
Site URL:	<input type="text" value="http://localhost:443/xampp/htdocs/xampp/"/>
Site Start Date:	<input type="text" value="March 2010"/>
Administrator Email:	<input type="text" value="joku@hotmail.com"/>
Default Theme for your site:	<input type="text" value="3D-Fantasy"/> ▼
Select the Language for your Site:	<input type="text" value="English"/> ▼
Locale Time Format:	<input type="text" value="en_US"/>

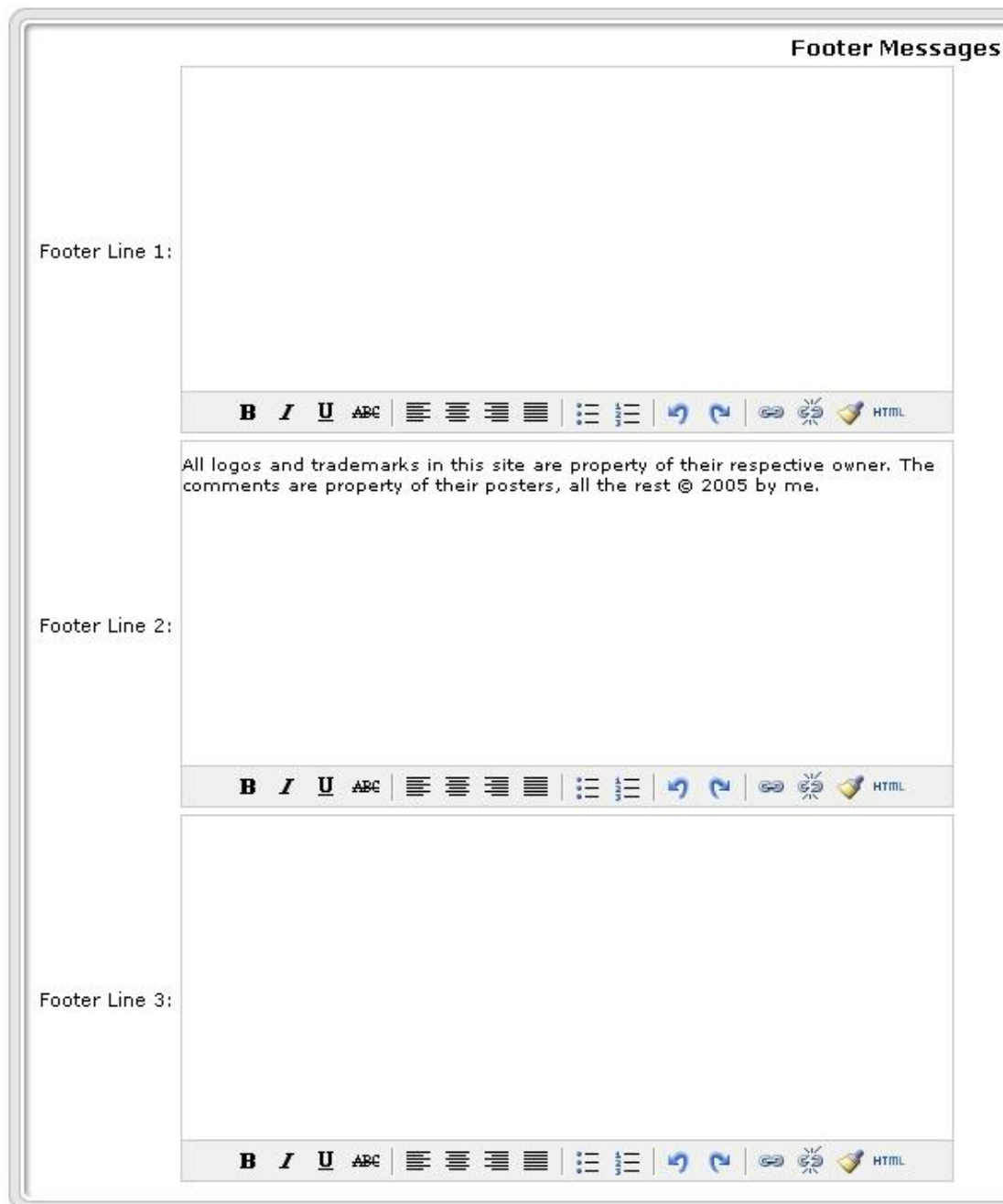
Kuvio 57. Sivujen yleisiä asetuksia

Toisessa kohdassa (kuvio 59) määritellään monikielisyyteen liittyviä asioita. Tässä voi päättää, ovatko sivut yhdellä kielellä ja jos ovat, näkyvätkö eri kielet maan lippuina vai pelkästään alasetolaatikkona, jossa kielet ovat tekstinä. Tästä on sitä enemmän hyötyä, mitä enemmän eri kieliä on, sillä selain avaa kauemmin sivua, jossa on kuvia.

Multilingual Options	
Activate Multilingual features?	<input checked="" type="radio"/> Yes <input type="radio"/> No
Display flags instead of a dropdown box?	<input checked="" type="radio"/> Yes <input type="radio"/> No

Kuvio 58. Sivujen kieliasetuksia

Seuraava kohta (kuvio 60) koskee sivujen alareunaa. Se koostuu kolmesta rivistä, joissa oleva teksti kirjoitetaan infoportaalin käyttämällä TinyMCE-editorilla. Kuviossa ylimpänä oleva teksti näkyy sivujen alareunassa ylimpänä, toinen alimpana, jne.



Kuvio 59. Tekstit, jotka näkyvät alimpana sivuilla

Tämä seuraava osa (kuvio 61) määrittelee sivujen RSS-syötteen ominaisuuksia. Infoportaalin syötteen näyttävät käyttäjilleen ja ylläpitäjilleen sivujen uutiset. Se on kätevää, jos ei aina muista sisäänkirjautumisen jälkeen katsoa uutisia. Tässä osassa kerrotaan, mikä syötteen aihe on ja millä kielellä se on. Yleensä kieleksi on hyvä laittaa sama kuin sivujen kieleksi.

Backend Configuration

Backend Title:

Backend Language:

Kuvio 60. Syötteiden asetukset

Sivujen lataamisen nopeuttamiseksi ylläpitäjän on mahdollista poistaa ylläpitovalikon kovalinkkien kuvat ja jättää vain tekstilinkit. Lomake näkyy kuviossa 62.

Graphics Options

Graphics in Administration Menu? Yes No

Kuvio 61. Ylläpitäjien toimintojen linkkien kuvat voidaan poistaa

Tämä osa (kuvio 63) määrittelee käyttäjätunnusten salasanojen vähimmäispituuden, joka voi olla 3, 5, 8 tai 10 merkkiä.

Users Options

Minimum users password length:

Kuvio 62. Käyttäjätunnusten salasanojen vähimmäispituus

Viimeinen osa (kuvio 64) on käyttäjiä varten. Ylläpitäjät eivät ole vastuussa käyttäjänsä tuottamasta sisällöstä, mutta käyttäjien kannalta on hyvä, että mahdolliset huonon tavan mukaiset sanat ovat poissa. Olisi noloa löytää esim. kiro sanoja omasta julkisesta esityksestään vasta silloin, kun on yleisöäkin. Siksikin sivuilla on oma sensurointijärjestelmä, joka korvaa huonon tavan mukaiset sanat halutulla tavalla (kuviossa korvaamisen merkintä on ”*****”). Sensuroitavan alueen laajuus valitaan alasetelaatikosta. Lopuksi kaikki asetusten muutokset tallennetaan napilla ”Save Changes”.



Censure Options

Censor Mode: Match anywhere in the text

Replace Censored Words with: *****

Save Changes

Kuvio 63. Sensuroinnin asetukset

8.22 Käyttöohjeet

Ylläpitäjän toiminto, joka auttaa ylläpitäjiä pääsemään alkuun tehtävässään. Ohjeissa (kuvio 65) selitetään kaikki tärkeimmät asiat ylläpidon toiminnoista. Ohjesivu on samankaltainen kuin käyttäjän käyttöohjeiden sivu. Se siis jakaantuu kahteen osaan: sisällysluetteloon ja ohjeisiin. Sisällysluettelon otsikot ovat linkkejä niiden ohjeisiin.

Contents

Introduction

Administrator's front page

Administration Menu

- Backup DB
- Blocks
- Edit Admins
- Ip Ban
- Modules
- Optimize DB
- Preferences
- Logout / Exit

Modules Administration

- Contacts
- Images
- News
- Pages
- Representations
- Polls/Surveys
- Texts
- Videos
- Users
- Logout / Exit

Guide

Introduction

This is for administrators of this web site. This will help to understand what c

Administrator's front page

When administrator logs in, he/she will see a page, administrator's page. Thi

Originally, at the left side are the boxes "administration", "waiting", and two
 - "Administration" consists of a link to logout and to access the administrato

Kuvio 64. Ylläpitäjien ohjeet

8.23 Käyttäjien hallinta

Ylläpitäjän toiminto. Tätä käytetään infoportaalin käyttäjien hallintaan. Käyttäjää muokataan ja poistetaan kuvion 66 mukaisella toiminnolla. Käyttäjän käyttäjänimi pitää tietää, jotta sen voi kirjoittaa ”laatikkoon”. Seuraavassa kohdassa valitaan, poistetaanko vai muokataanko käyttäjän tietoja. Lopuksi painetaan nappia ”Ok!”, jolloin suoritetaan toimenpide.

Edit User

Username:

Kuvio 65. Käyttäjätunnusta muokatakseen pitää sen käyttäjänimi

Sivun toisessa osassa (kuvio 67) on uuden käyttäjän lisääminen. Käyttäjä lähettää tiedot ylläpidolle ja ylläpitäjä lisää käyttäjän järjestelmään tämän lomakkeen kautta. Lomakkeeseen täytetään kaikki kohdat annettujen tietojen mukaan ja painetaan nappia ”Add User”.

Add a New User

Username (required)

Email (required)

Password (required)

Real name (required)

Phone (without country number) (required)

Company (required)

Address (required)

Zip code (required)

City (required)

Country (required)

Kuvio 66. Uuden käyttäjän lisääminen

8.24 Käyttäjien sisältöjen hallinta

Ylläpitäjän toiminto. Tällä hallitaan käyttäjien luomaa sisältöä. Toimintoon kuuluu siten tekstien, kuvien ja videoiden hallinta, jotka on jaettu selvyyden vuoksi erillisiksi toiminnoiksi ylläpidon sivuille.










Näille kolmelle toiminnolle on yhteistä se, että jokaista varten on toiminto etsi (kuvio X), jolla voidaan hakea jonkun tietyn käyttäjän luomaa sisältöä. Etsinnässä voidaan käyttää käyttäjän käyttäjänimeä tai vaihtoehtoisesti sähköpostiosoitetta (kuvio 68). Etsintä aloitetaan klikkaamalla nappia ”Ok!”.

Search by email.

Kuvio 67. Käyttäjän luomaa sisältöä etsitään näin

Kun nappia on painettu, alapuolelle ilmestyy löydetty sisältö. Toiminnosta riippuen ne ovat joko tekstejä, kuvia tai videoita. Kuviossa 69 on esimerkkinä kuvien hallinta.

Jos ei haeta tietyn käyttäjän sisältöä, alapuolelle ilmestyykin kaikkien käyttäjien sisällöt. Niitä voidaan sitten muokata ja poistaa. Videoita ja kuvia voidaan lisäksi katsella niiden alkuperäisessä koossa.

Owner	Id	File name	Image	Actions
Admin	43	Contact_us_er.jpg		 
demo@infoportal.fi	52	Demo_image1.gif		 
demo@infoportal.fi	53	Demo_image2.gif		 

Kuvio 68. Kuvien hallintalomake

8.25 Moduulien hallinta

Ylläpidon toiminto, jolla infoportaalin moduuleja voidaan hallita (kuvio 70). Tärkeimmät asiat, joita tällä toiminnolla voi tehdä, on infoportaalin alkusivun vaihtaminen, moduulin näkymisen määrittäminen tietyille käyttäjille (vierailijat, käyttäjät tai ylläpito) ja moduulin poistaminen käytöstä (esimerkiksi sen päivittämisen ajaksi). Infoportaalin alkusivulla oleva moduuli kuvataan sinertävällä värillä, kuten kuviossa 70 näkyy. Alkusivun moduuli määritetään painamalla halutun moduulin kohdalla olevaa kuvaa, jossa on paperi ja avain.

Title	Custom Title	Status	Visible to	Group	Functions
Contact_us	Contact_us		All Visitors	None	
Guest_guide	Guest_guide		All Visitors	None	
Guide	Guide		All Visitors	None	
Home	Home		All Visitors	None	
Images	Images		All Visitors	None	
News	News		All Visitors	None	
Pages	Pages		All Visitors	None	
Portal_demonstration	Portal_demonstration		All Visitors	None	
Representations	Representations		All Visitors	None	
Surveys	Surveys		All Visitors	None	
Texts	Texts		All Visitors	None	
User_home	User_home		All Visitors	None	
Videos	Videos		All Visitors	None	
Your_Account	Your_Account		All Visitors	None	

Kuvio 69. Moduulien hallintalomake

8.26 Osioiden hallinta

Ylläpitäjän toiminto. Tätä käytetään sivujen toimintojen muokkaamiseen omalta osaltaan. Tällä muutetaan kaikkea sivujen osioihin liittyvää.

Sivulle avautuu lista, jossa on kaikki käytettävät osiot (kuvio 71). Yleisimmin osioiden toimintaa muutetaan vaihtamalla osion sijaintia sivuilla ja ryhmää, joka osiota voi käyttää. Muita osiokohtaisia toimintoja ovat sen poistaminen käytöstä tai käyttöön ottaminen ja kielen vaihtaminen, jotta se näkyy esimerkiksi käyttäjän valitsemasta kielestä huolimatta. Lisäksi kaikki hakemiston "blocks" sisältämät ja infoportaalien kautta tehdyt osiot voi poistaa tai niitä voi muokata ja osiota voi katsella etukäteen, jotta sen saa ennen julkaisua näyttämään sellaiselta kuin haluaa. Osioita, joiden "Type" on "SYSTEM", ei voi poistaa.

Title	Position	Weight	Type	Status	Visible to	Language	Functions
Modules	▷ Center Up ◁	1	FILE		All Visitors	all	
Search	▷ Center Down ◁	1	FILE		All Visitors	all	
Administration	◁ Left	1	SYSTEM		Administrators Only	all	
Who's Online	◁ Left	1	FILE		All Visitors	all	
Languages	◁ Left	2	FILE		All Visitors	all	
Random Headlines	◁ Left	3	FILE		All Visitors	all	
User's Custom Box	◁ Left	4	SYSTEM		All Visitors	all	
Login	◁ Left	5	FILE		All Visitors	all	
User Info	◁ Left	6	FILE		All Visitors	all	
Menu	◁ Left	7	FILE		Anonymous Users Only	all	
Menu	◁ Left	8	FILE		Registered Users Only	all	
Categories Menu	Right ▷	1	FILE		All Visitors	all	
Survey	Right ▷	2	FILE		Registered Users Only	all	
Big Story of Today	Right ▷	3	FILE		All Visitors	all	
Old Articles	Right ▷	4	FILE		All Visitors	all	
Information	Right ▷	5	HTML		All Visitors	all	

Kuvio 70. Sivujen käytettävissä olevat osiot

Uuden osion lisäyksessä (kuvio 72) pitää ottaa huomioon, että osion sisältämät asiat voidaan luoda eri tavoilla. Infoportaalissa yleisin tapa on tehdä moduulihakemistoon tarpeelliset tiedostot, jonka jälkeen moduulin nimi haetaan ”alasetolaatikosta” kohdassa ”Filename”. Näin tehtäessä, kohtia ”Title”, ”RSS/RDF file URL” ja ”Content” ei tarvitse täyttää, koska niitä käytetään, kun moduulin sisältö luodaan muilla tavoilla. Toinen tapa on hakea sisältö RSS (tai RDF) syötteiden vaatima tiedosto joko tietystä Internet-osoitteesta tai valmiiksi haetuista Internet-osoitteista, jotta ylläpitäjän ei välttämättä tarvitse hakea kaikkien syötteiden osoitteita itse. Kolmas ja viimeinen tapa on antaa osiolle otsikko kohtaan ”Title” ja täyttää kohta ”Content”. Tämä kohta on se, joka osiossa tulee näkymään. Tällä tavalla on kätevää tehdä esimerkiksi sellaiset osiot, joissa on vain linkkejä vaikka moduuleihin.

Kun sisältö on valmiina, osiolle valitaan paikka, johon se sijoitetaan sivuille. Määritetään, minkä kielen valinneet sivujen käyttäjät näkevät tämän osion. Jos on aivan varma, että osio toimii oikein, se kannattaa ottaa käyttöön heti valitsemalla ”Yes” kohdassa ”Activate?”. Osion voi määrittää toimimaan niin monta päivää kuin haluaa, minkä jälkeen päätetään, poistetaanko osio käytöstä vai tuhotaanko se kokonaan. Toiseksi viimeisen kohta liittyy RSS-syötteisiin. Siihen merkataan, kuinka usein syötteiden tarjoama tieto tarkistetaan. Tämä on hyödyllinen asetus, koska joillakin sivuilla tieto päivittyy harvoin ja joillakin useammin. Viimeisessä kohdassa valitaan ryhmä, joka voi käyttää tätä osiota. Lopuksi klikataan nappia ”Create Block”, jolloin osio lisätään ja sitä voi hallita tavalliseen tapaan.

Add a New Block

Title:

RSS/RDF file URL: Custom [Setup]
(Select Custom and write the URL or just select a Site from the list to grab news headlines)

Filename: None (Select a custom Block to be included. All other fields will be ignored)

Content:

B *I* U ABC ☰ ☰ ☰ ☰ ☰ ☰ ☰ ↶ ↷ 🌐 📄 HTML

If you fill the URL the content you write will not be displayed!

Position: Left

Language: English

Activate? Yes No

Expiration: days

After Expiration: Deactivate

Refresh Time: 1 Hour (Only for Headlines)

Who can View This? All Visitors

Kuvio 71. Uuden osion lisääminen sivujen käyttöön

8.27 Kiellettyjen IP-osoitteiden hallinta

Ylläpitäjän toiminto, jolla voi estää tiettyjen IP-osoitteiden pääsy infoportaaliin (lomake kuviossa 73). Toiminto mahdollistaa jopa tietyn ryhmän laitteita ottamasta yhteyttä sivulle, mutta yleensä estetään vain tietty IP-osoite, jos tarvitsee. Eston syyksi kannattaa kirjoittaa oikea syy, koska sitten sen muistaa myöhemminkin. Eston voi myöhemmin purkaa tai sitä voi muokata.

Ban a new IP Address

. . .

Reason

Character * is accepted as the last number, but this will ban a complete Class C network. Use it carefully and only in extreme cases.

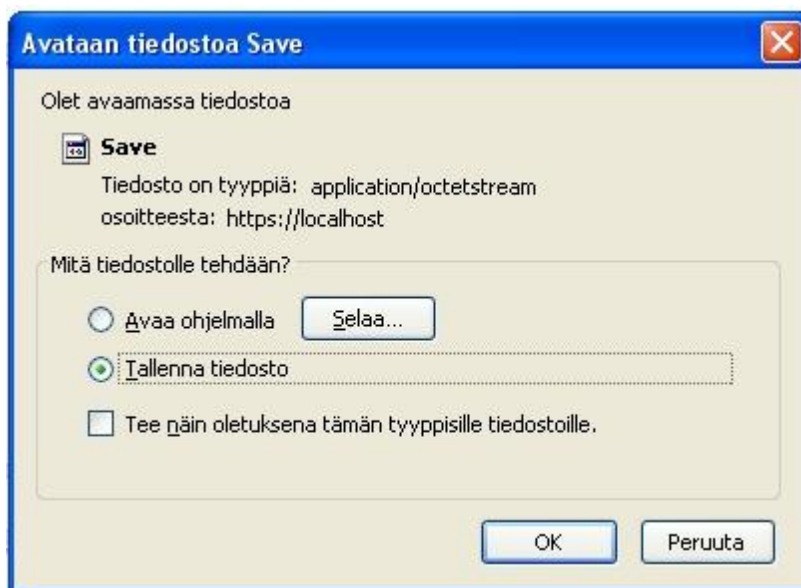
Banned IP Addresses

Banned IP Addresses	Reason	Ban Date	Functions
100.100.100.*	Because...	2011-05-27	 

Kuvio 72. Laitteilta voidaan estää sivujen käyttö tällä lomakkeella

8.28 Tietokannan kopiointi

Ylläpitäjän toiminto, jolla infoportaalin tietokannassa olevat taulut ja niissä oleva tieto saadaan tallennettua kovalevylle tms. laitteelle tiedostona. Tietokantatiedoston tallentaminen ja avaaminenkin vaihtelee selaimesta riippuen, mutta kuvio 74 näyttää idean.



Kuvio 73. Tietokantatiedosto voidaan tallentaa esim. kovalevylle

Tiedostolla ei ole päätettä (esimerkiksi ”txt”), mutta sen voi lukea tekstinkäsittelyohjelmalla, koska sen sisältö on tekstiä. Tiedoston alussa lukee tietokannan nimi. Sen jälkeen tietokannan taulut ja niiden sisältö esitetään MySQL-koodina. Jos tietokanta tuhoutuu jostain syystä, tuon MySQL-koodin voi hyödyntää esimerkiksi jollain ohjelmalla, joka luo tietokannan tiedostossa olevaa koodia (kuviossa 75) käyttämällä.

```

# =====
#
# Database saved : sisalto
# On 05-23-2011 at 15:47 by !
#
# =====
# -----
#
# Table structure for table 'contact_categories'
#
CREATE TABLE contact_categories (
  id bigint(20) NOT NULL auto_increment,
  f_category varchar(30) NOT NULL,
  e_category varchar(30) NOT NULL,
  viewer int(1) NOT NULL,
  PRIMARY KEY (id)
);
#
# Dumping data for table 'contact_categories'
#
INSERT INTO contact_categories VALUES ( '1', 'Kysymys infoportaalista', 'Question about infoportal', '2');
INSERT INTO contact_categories VALUES ( '2', 'Käyttäjätunnuksen tilaus', 'Order of an account', '1');
INSERT INTO contact_categories VALUES ( '3', 'Palaute', 'Feedback', '0');

```

Kuvio 74. Tietokantatiedoston koodi

8.29 Tietokannan optimointi

Ylläpitäjän toiminto, joka auttaa säästämään palvelimen tallennustilaa. Pienemmillä sivuilla tästä ei ole erityisemmin hyötyä, mutta isommilla sivuilla on, koska tietokannan sisältämä tieto vaihtuu paljon käyttäjien suuren määrän vuoksi. Optimointi ei aiheuta kuitenkaan taulujen tietojen muokkausta, joten se ei ole uhka sivujen toiminnalle. Kuviossa 76 on juuri tehty tietokannan optimointi.

nuke_users	6.961 Kb	Already optimized	0 Kb
nuke_users_temp	2.121 Kb	Already optimized	0 Kb
pages_frames	2.85 Kb	Already optimized	0 Kb
representations_pages	3.142 Kb	Already optimized	0 Kb
testit	2.043 Kb	Already optimized	0 Kb
user_images	2.707 Kb	Already optimized	0 Kb
user_pages	2.289 Kb	Already optimized	0 Kb
user_representations	2.074 Kb	Already optimized	0 Kb
user_texts	2.605 Kb	Already optimized	0 Kb
user_videos	2.523 Kb	Already optimized	0 Kb

Optimization Results			
Total Space Saved: 0.723 Kb			
You have run this script: 14 times			
21.957 Kb saved since its first execution!			

Kuvio 75. Tietokannan optimoinnin tulokset

9 TESTAUS

Infoportaalin testaus toteutettiin v-mallin mukaan, mutta ei aivan tarkasti mallin mukaan. Testaus suoritettiin pääasiassa toteutuksen jälkeen, mutta moduulitestaus tapahtui vähän poikkeavasti. V-mallissahan verrataan testaustuloksia projektin suunnitteluvaiheeseen. Testaus etenee pienistä yksityiskohdista suurempiin niin, että ensimmäinen testausvaihe on moduulien testaus ja viimeinen kolmesta testausvaiheesta on koko järjestelmän/ohjelmiston testaus.

9.1 Moduulitestaus

Moduulien toimimisen testaus sulautui suurimmalta osaltaan toteutusvaiheeseen, koska ohjelmoinnin tuloksia piti testata todella usein ja syytä olikin. Toteutuksessa käytettiin vain Mozilla Firefoxia, joten infoportaalin sivutkin tehtiin ja testattiin vain yhdellä selaimella. Tämä aiheutti kohtalaisen suuriakin ongelmia sivujen toimivuudelle, koska Internet Explorer ei ymmärtänyt varsinkaan JavaScript-ohjelmointikielen kaikkia komentoja samalla tavalla kuin Mozilla Firefox. Vaikeinta oli kuitenkin keksiä uusi tapa korjata ongelma, kun sen syy oli joissain tapauksissa

selaimen ohjelmointivirheissä. Ohjelmoinnin aikainen testaus olisikin kannattanut tehdä Mozilla Firefoxin sijasta Internet Explorerilla, koska sen kanssa oli suurimmat yhteensopivuusongelmat. Näiden ongelmien korjaaminen vei tavallista kauemmin, koska ensin piti selvittää ongelmien syyt internetin avulla tai itse ja sitten korjata koodi. Vielä varmempi, mutta enemmän työtä vaatinut tapa olisikin ollut testata moduuliin tehdyt muutokset kaikilla selaimilla, joilla sivujen olisi tarkoitus toimia.

Moduulien suunnitelmat olivat kuitenkin pääasiassa toimivia, vaikka jotkut yksityiskohdat muuttuivatkin.

9.2 Integroititestausta

Tästä vaiheesta ei ole mitään erityistä kerrottavaa, sillä kaikki näytti toimivan. Tässä vaiheessa PHP-Nuken käytöstä oli kyllä jonkin verran apua, koska modulaarinen ajattelu pysyi helpommin mielessä ja PHP-Nukekin ohjaa tekemään niin.

9.3 Järjestelmätestaus

Tämän vaiheen suoritin testaamalla kaiken vierailijan, käyttäjän ja ylläpitäjän roolissa. Yhteydenottoa en testannut niin kuin se oikeasti toimisi, koska toimintoon liittyy sähköpostien käyttö, eikä niiden virheellisyys liity infoportaalin sivuihin. Muutkin henkilöt ovat testanneet järjestelmän toimivuutta, eikä suurempia valituksien aiheita ole tullut esille. Käyttäjävälisyyksikin on tuntunut säilyneen tarpeeksi hyvin eri selaimienkin osalta, koska infoportaalin nykyisessä versiossa vain jotkut HTML-taulukkojen leveydet ja korkeudet ovat vähän outoja. Sivujen lisäyksen yhteydessä olevaa esikatselua ei tosin toimi, koska sen toimimaan saaminen olisi vaatinut vielä pidemmän ajan, eikä siihen ollut aikaa siinä tilanteessa. Esikatselun puuttuminen on kuitenkin vain pieni ongelma.

10 LOPPUTULOKSET

Koko tämä työ alkoi tutkimusongelman miettimisestä ja kaiken pitää perustua siihen. Toivoin löytäväni mahdollisimman kätevän tavan tehdä infoportaalin sivut, mutta en löytänyt täydellistä CMS-ohjelmistoa. Löysin kuitenkin PHP-Nuken, joka vaikutti tarpeeksi joustavalta, jotta voisin käyttää sitä tähän tarkoitukseen. Yksi vaatimus oli sekin, että CMS-ohjelmiston piti perustua PHP-kieleen.

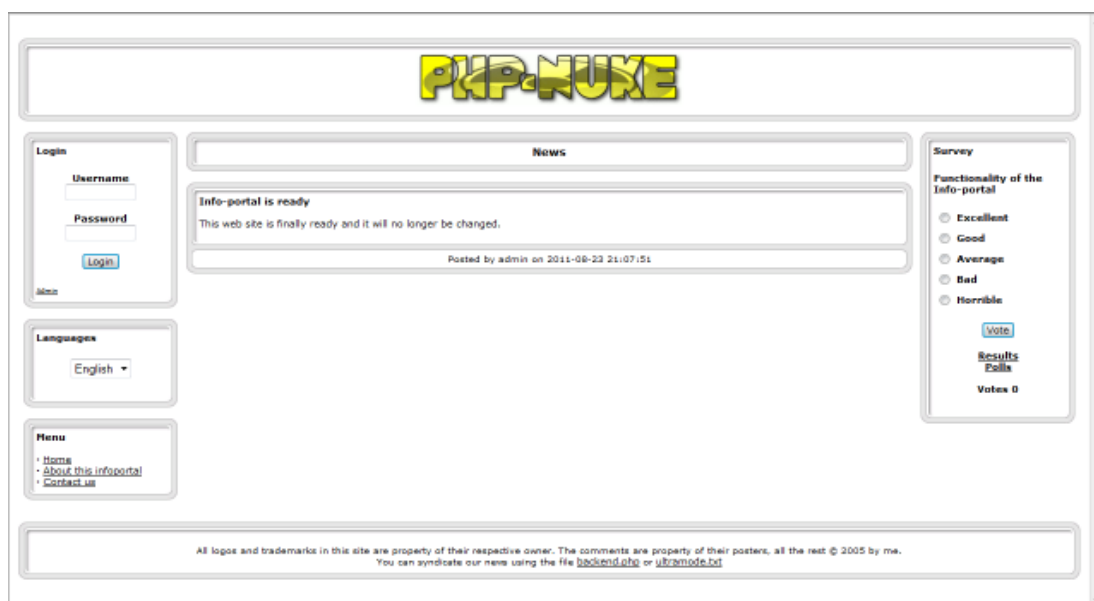
Nukella pystyin tekemään kaiken haluamani ja sain melko hyvin aikaan tutkimusongelman mukaisen järjestelmän, vaikka vaikeaa olikin. Koska työ perustui PHP-kieleen ja siten myös HTML-kieleen, minun piti opetella melkein kaikki kielet itse, varsinkin PHP, HTML ja JavaScript. Onneksi olin ohjelmoinut aikaisemminkin, mutta vain C-kielellä. Tämän lisäksi piti opetella PHP-Nuken järjestelmä, että pystyin hyödyntämään sitä.

Nuken avulla oli todella helppo saada sivujen rakenne näyttämään suunnilleen sellaiselta kuin halusi, joten tarkkaa suunnittelua siihen liittyen ei tarvittu. Suunnitteluksi riitti miettiä, millainen sivu olisi, jos osio olisi esim. vasemmalla puolella sivua. Alun perin PHP-Nuken mukana tulleet esimerkkisivut loivat varsinaisen perustan sivuille. Riitti, kun kokeili sivujen ulkonäköä vaihtamalla osioiden paikkoja. Muut suunnitelmat tein kirjasta löytämäni suunnittelumenetelmää soveltaen, koska tarkoitus oli edetä vaihejakomallin mukaan. Toteutus oli ylivoimaisesti vaikein osa työssä, koska minun piti opetella käyttämään useampaa kieltä. Internet oli jatkuvassa käytössä koko toteutusvaiheen. Järjestelmän testauksessa tuli ongelmia huomattessani, että Internet Explorer oli hankalin tapaus kolmesta suosituimmasta selaimesta, joiden kanssa infoportaalin piti olla yhteensopiva. Tämä aiheutti jonkin verran lisätyötä, kun piti korjailla Internet Explorerista aiheutuneita virheitä. Tästä johtuen sivujen esikatselu-toiminto jäi puuttumaan kokonaan, mikä ei ole kuitenkaan varsinainen ongelma. Käyttöönottovaihetta ei ollut tässä projektissa, kun PHP-Nuke piti asentaa jo projektin alkuvaiheissa. Alkuperäinen idea oli myös asentaa infoportaali internetin ilmaiselle palvelimelle, mutta ajan puutteen vuoksi en ehtinyt tekemään sitä.

10.1 Lyhyt esittely valmiista sivuista

Sivut toimivat periaatteeltaan niin kuin halusin, mutta joitakin toimintoja olisin halunnut saada käytännöllisemmiksi. Sivut ovat kuitenkin niin toimivat, että niistä on jotain hyötyäkin.

Kun sivuille tullaan, ensin aukeaa pääsivu (kuvio 77). Käyttäjät ja ylläpitäjät kirjautuvat Login-osion kautta. Kaikki ei-kirjautuneet ovat infoportaalin vierailijoita, jotka voivat kuitenkin kirjautua sisään, mutta vain demokäyttäjätunnusta käyttäen.



Kuvio 76. Pääsivu









Infoportaalin käyttö on maksullista, joten käyttäjäksi ei pääse aivan yksinkertaisesti. Vierailijan pitää painaa linkkiä ”Contact us” ja kirjoittaa viestiosaan omia tietojaan, jotka lähetetään ylläpitäjille (kuvio 78). Ylläpidon tarvitsemat tiedot löytyvät painamalla linkkiä ”about this infoportal”. Ylläpito voi lisätä käyttäjän ylläpidon toimintojen kautta. Muut tarvittavat toimenpiteet suoritetaan sähköpostin välityksellä. Uudeksi ylläpitäjäksi ei pääse kuka vain, joten siihen ei ole ohjeitakaan.

Contact us

Message category
Ordering of an account ▾

Subject

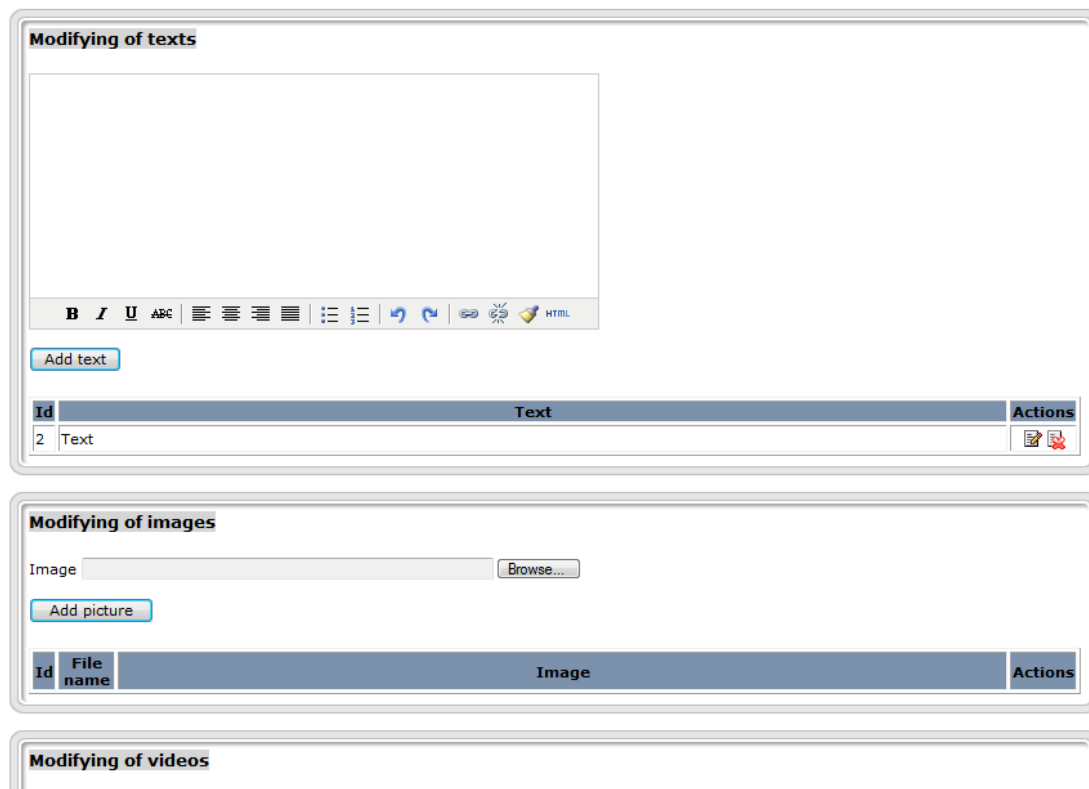
Message

B *I* U ABC |         HTML

Email (will be answered by us)

Kuvio 77. Tunnuksen tilaus

Kun vierailija kirjautuu sisään käyttäjäksi, käyttäjän tekemät esitykset näkyvät avautuvalla sivulla ja käyttöön tulee uusia toimintoja. Kun ylläpito kirjautuu sisään, hän näkee ylläpitäjän toiminnot. Käyttäjäksi kirjautunut voi vaihtaa salasanaansa vapaasti, mutta ei muita käyttäjätunnuksensa tietoja. Niiden muuttamiseen tarvitaan ylläpidon lupa. Esityksiä ja niihin liittyvät asiat löytyvät klikkaamalla linkkiä ”creation”. Tähän ja muuhunkin sivujen käyttämiseen löytyvät ohjeet käyttäjän valikosta. Esityksien tekoon tarvitaan tekstejä, videoita ja/tai kuvia (kuvio 79), jotka asetellaan itse tehdyille sivulle. Kuvat ja videot tallentuvat palvelimelle, mutta tekstit tallentuvat tietokantaan. Valmiit sivut laitetaan uuteen esitykseen haluttuun järjestykseen. Kaikkia näitä asioita voi muokata jälkeinpäinkin. Ylläpito voi hallita kaikkea sisältöä, joten käyttäjät eivät voi tehdä millaista sisältöä vain, koska liian sopimaton sisältö voidaan poistaa.



Kuvio 78. Tekstien, kuvien ja videoiden lisääminen on helppoa

Kun valmis esitys käynnistetään, selain avaa sen ensimmäinen sivun. Esityksen sivu- ja voidaan selata joko käyttämällä vasemmalla ylhäällä olevia nuolia tai nuolien yläpuolella olevia sivujen nimiä. Nuolien avulla pääsee seuraavalle ja edelliselle sivulle, kun taas sivujen nimiä painamalla selain avaa juuri halutun sivun.

11 TULEVAISUUDEN NÄKYMÄT

Kun aloitin tämän projektin, mielessä oli vähän sellainenkin ajatus, että työstä voisi olla hyötyä oikeassakin elämässä. Infoportaalin tämä ensimmäinen versio on kuitenkin liian yksinkertainen ja viallinen, joten siitä ei kehtaisi pyytää käyttömaksua. Järjestelmän ohjelmointi-ideat ovat kuitenkin sen verran hyviä, että järjestelmän laajentaminen ja parantaminen onnistuu kohtuullisella työmäärällä. Infoportaali vaatisi siis lisää joidenkin, tärkeidenkin, toimintojen kehittämistä, mutta ei nykyinenkään versio ole turha.

LÄHTEET

Wikipedia. 2009. Ohjelmistotuotanto. Viitattu 30.4.2010.

<http://fi.wikipedia.org/wiki/Ohjelmistotuotanto>

Märijärvi & Haikala. 2004. Ohjelmistotuotanto. Hämeenlinna: Karisto Oy.

Oulun Kauppaoppilaitos. Kehittämistyön vaiheet ja elinkaarimallit. Viitattu 6.5.2010.

http://www.okol.org/verkkokurssit/datanomi/tietojarjestelmien_kaytto_ja_kehittamin_en/johdatus_tietojarjestelmiin/kehittamistyon_vaiheet_ja_elikaarimallit/kehittamisty_on_vaiheet_ja_elikaarimallit.htm

Wikipedia. 2010. Moduuli. Viitattu 11.5.2010. <http://fi.wikipedia.org/wiki/Moduuli>

Wikipedia. 2010. V-malli. Viitattu 11.5.2010. <http://fi.wikipedia.org/wiki/V-malli>

Wikipedia. 2010. Portaali (internet). Viitattu 11.5.2010.

[http://fi.wikipedia.org/wiki/Portaali_\(internet\)](http://fi.wikipedia.org/wiki/Portaali_(internet))

Wikipedia. 2010. Sisällönhallintajärjestelmä. Viitattu 27.6.2010.

<http://fi.wikipedia.org/wiki/Sis%C3%A4ll%C3%B6nhallintaj%C3%A4rjestelm%C3%A4>

Wikipedia. 2010. XAMPP. Viitattu 28.6.2010. <http://en.wikipedia.org/wiki/XAMPP>

W3schools. 2010. Browser Statistics. Viitattu 28.6.2010.

http://www.w3schools.com/browsers/browsers_stats.asp

Wikipedia. 22.6.2010. Webmaster. Viitattu 30.6.2010.

<http://en.wikipedia.org/wiki/Webmaster>

Wikipedia. 28.6.2010. GNU General Public License. Viitattu 1.7.2010.

http://en.wikipedia.org/wiki/GNU_General_Public_License

Ibiblio. PHP-Nuke: Management and Programming. Viitattu 11.7.2010.

<http://www.ibiblio.org/pub/linux/docs/howto/other-formats/pdf/PHP-Nuke-HOWTO.pdf>

PHP-Nuke. Directory structure. Viitattu 3.9.2010.

http://phpnuke.org/modules.php?name=PHP-Nuke_HOWTO&page=directory-structure.html

PHP-Nuke. Module creation, administrator part. Viitattu 4.9.2010.

http://phpnuke.org/modules.php?name=PHP-Nuke_HOWTO&page=module-admin.html

Wikipedia. 2010. HTML. Viitattu 4.9.2010. <http://fi.wikipedia.org/wiki/HTML>

Wikipedia. 2010. Käyttöliittymä. Viitattu 6.9.2010.

<http://fi.wikipedia.org/wiki/K%C3%A4ytt%C3%B6liittym%C3%A4>

Wikipedia. Tietokanta. Viitattu 6.9.2010. <http://fi.wikipedia.org/wiki/Tietokanta>