

Olli Norola

PROJEKTIHALLINTA- OHJELMISTON KÄYTTÖÖNOTTO

Mipro Oy

Opinnäytetyö
Tietotekniikka


Toukokuu 2011




MIKKELIN AMMATTIKORKEAKOULU

Mikkeli University of Applied Sciences

KUVAILULEHTI

 <p>MIKKELIN AMMATTIKORKEAKOULU Mikkeli University of Applied Sciences</p>	Opinnäytetyön päivämäärä 13.5.2011	
Tekijä(t) Olli Norola	Koulutusohjelma ja suuntautuminen Tietotekniikan koulutusohjelma	
Nimeke Projektinhallintaohjelmiston käyttöönotto		
Tiivistelmä Mipro Oy on suomalainen rautatieturvallisuuden ja vesi- ja lämpöhuollon järjestelmiin erikoistunut järjestelmätoimittaja. Yritys työllistää noin 70 henkeä, päätoimipiste on Mikkelissä ja sivutoimipiste löytyy Oulusta ja uusimpana myös Slovakiasta. Työn idea lähti Mipron tarpeesta uudistaa projektin- ja resurssinhallintaansa. Tavoitteena oli saada käytön työkalu, jonka avulla voi helposti nähdä kaikkien meneillään olevien projektien tilanteen, seurata niiden edistymistä ja tarkkailla henkilöiden kuormitusta. Tuntienkirjausjärjestelmän integraatio otettiin myös tavoitteeksi. Mipron selvitysten ja taustatutkimusten jälkeen projektinhallintaohjelmiston toimittajaksi valittiin kotimainen ohjelmistotalo ja heidän tuotteensa. Demoversio ohjelmasta luotiin Miprolla pidetyn määrittelypäivän jälkeen. Minun tehtäväni oli toimia Mipron puolelta henkilönä, joka testaa demojärjestelmää, syöttää sinne tietoa, raportoi ongelmista ja toimii yhteyshenkilönä toimittajan suuntaan asiakaskohtaistamistyön merkeissä. Räättälöinnin jälkeen suoritettiin ohjelmiston käyttöönotto Miprolla ja sisäinen koulutus henkilöstölle. Työssä on teoriaosuudessa käyty läpi projektimuotoista työskentelyä, web-sovellusten toimintaa ja tekniikkaa sekä hieman myös web-sovelluksen testausta. Lopussa on käyty läpi itse ohjelmiston käyttöönottoon liittyvät asiat, ohjelmiston käyttö ja ylläpitoon liittyviä toimia.		
Asiasanat (avainsanat) Projektinhallinta, projektinhallintasovellukset, web-sovellukset		
Sivumäärä 45 sivua	Kieli Suomi	URN
Huomautus (huomautukset liitteistä)		
Ohjaavan opettajan nimi Timo Mynttinen	Opinnäytetyön toimeksiantaja Mipro Oy	

DESCRIPTION

 <p>MIKKELIN AMMATTIKORKEAKOULU Mikkeli University of Applied Sciences</p>		Date of the bachelor's thesis 13 May 2011	
Author(s) Olli Norola		Degree programme and option Information Technology	
Name of the bachelor's thesis Introduction of project management software			
Abstract Mipro Oy is a Finnish company specializing in safety related systems for railways and automation systems for water works. The company employs around 70 employees. Mipro has headquarters in Mikkeli and branch offices in Oulu and latest in Slovakia. Most of the work is done in projects in Mipro and they have a project number, project manager and a budget. The target of this bachelor's thesis was to introduce new project management software because Mipro needed to renew their project management. The main requirement for the new software was the ability to monitor project progress and the workload of employees. The integration of work hour software was also an important need. My bachelor's thesis was divided in to two sections in which the first part introduced with the theory of project management and technology of web-applications. The second part included the introduction report of Mipro's new project management software. After Mipro had made background surveys and explored different situations a product of a Finnish software company was chosen. Demo version of the software was determined according to Mipro's software requirements specification (SRS). My role was to test the demo version, report problems and development proposals to the manufacturer. After cooperation the software was ready to use in Mipro. Before final introduction I arranged training sessions at Mipro. The Most essential document of this study was the manual for Mipro's internal use.			
Subject headings, (keywords) Project management, project management systems, web-applications			
Pages 45 pages	Language Finnish	URN	
Remarks, notes on appendices			
Tutor Timo Mynttinen		Bachelor's thesis assigned by Mipro Oy	

SISÄLTÖ

1	JOHDANTO	2
2	MIPRO OY	3
3	PROJEKTITYÖSKENTELY	4
3.1	Projektin määritelmä.....	4
3.2	Projektionnin hyödyt ja ongelmakohdat.....	4
3.3	Projektinhallinta.....	5
4	WEB-SOVELLUKSET JA NIIDEN TEKNIIKAT.....	6
4.1	Webin kerrosmallit	6
4.2	Web-sovelluksen rakenne.....	7
4.3	Webin staattisuus ja dynaamisuus	9
4.4	Web-sovelluksen testaus.....	10
4.5	Web-käytettävyys	12
4.6	Microsoft .NET-arkkitehtuuri ja ASP.NET.....	13
4.7	Pilvipalvelut.....	15
4.8	Virtualisointi	17
5	PROJECT ROUTER.....	18
5.1	Ohjelmiston demojärjestelmä	18
5.1.1	Mipron vaatimusmäärittely.....	18
5.1.2	Demojärjestelmän testaus	20
5.1.3	Testauksen tulosten tarkastelu ja toimenpiteet	21
5.2	Siirto Mipron palvelimelle.....	22
5.3	Ohjelmiston siirto Miprolle ja sisäinen käyttöönotto	25
5.4	Ohjelman käyttö.....	27
5.4.1	Kirjautuminen ja etusivu.....	27
5.4.2	Projektisalkku	29
5.4.3	Asetukset.....	30
5.4.4	Projektin luominen.....	30
5.4.5	Omat tehtävät	33
5.5	Tuntienkirjausjärjestelmän integraatio	34
5.6	Tietokantojen varmistukset ja palautus	35
5.7	Käytettävyyden tarkastelu	41

6 PÄÄTÄNTÖ	43
LÄHTEET	44

LYHENTEET JA KÄSITTEET

ADO.NET	Microsoftin luokkakirjasto tietokantayhteyksille
ASP.NET	WWW-sivujen luomiseen tarkoitettu palvelinpuolen ohjelmointimenetelmä, joka perustuu .NET arkkitehtuuriin.
ASP	Active Server Pages. ASP.NETin edeltäjä.
ASP	Application Service Provider
CLR	Common Language Runtime.
CIL	Common Intermediate Language
CLI	Common Language Infrastructure
HTTP	Hypertext Transfer Protocol
HTML	WWW:n hypertekstin merkintäkieli
IIS	Internet Information Services
.NET	Microsoftin Windows-pohjainen ohjelmistokomponenttikirjasto
RDBMS	Relational Database Management System
SAAS	Software as a service.
SRS	Software Requirements Specification
SQL	Structured Query Language
VMWARE	Virtualisointiohjelmistoja tarjoava yritys
Web	ks. WWW
Web-selain	Web-sivujen lukemiseen tarkoitettu ohjelmisto
WWW	World Wide Web. Internetin hypertekstijärjestelmä.
W3C	World Wide Web Consortium

1 JOHDANTO

Projektimuotoinen työskentely on nykyisin erittäin yleistä työelämässä. Projekti auttaa hallitsemaan tavoitteita ja helpottaa työn etenemisen seuraamista. Projektin ideana on hakea hyötyä ryhmästä töissä, jotka yksilötyönä eivät onnistu tai ovat kannattamattomia.

Mipro Oy:ssä suurin osa töistä tehdään projektimuodossa. Projektinhallintaan ei ole kuitenkaan ollut yhtä keskitettyä tapaa tai työkalua, vaan projektien etenemistä ja resurssointia on hoidettu monin eri keinoin.

Idea työhön lähti Mipron tarpeesta uudistaa projektinhallintaansa. Syksyllä 2010 sovimme opinnäytetyön tekemisestä ja sovimme työn aloitusajankohdaksi vuoden 2011 alun. Työn tavoitteena oli saada uusi projektinhallintaohjelmisto käyttöön yrityksessä. Mipron seulonnan ja tarjouspyynnön jälkeen sopivimmaksi sovellukseksi varmistui kotimaisen ohjelmistotalo Improlity Oy:n Project Router.

Joulukuussa 2010 pidimme Miprolla määrittelypäivän, jossa kävimme läpi Mipron tarpeet projektinhallintaohjelmistoa ajatellen. Paikalla oli Improlityn edustaja, muutama Mipron edustaja sekä minä. Sovimme että Improlity räätälöi sovelluksen Mipron tarpeita ajatellen ja vuoden 2011 alussa demojärjestelmä saatiin käyttöön Miprolla. Sovellus sovittiin pidettäväksi Improlityn palvelimilla siihen asti, kunnes se on käyttövalmis Miprolle. Siirto Mipron palvelimelle tapahtui helmikuussa ja Mipron sisäinen käyttöönotto suoritettiin kevään 2011 aikana. Tuntienkirjausjärjestelmän integraation käytännön toiminnan kuvausta ei ole kuvattu tässä työssä, sillä integraation toteutuminen venyi pitkälle keväeseen 2011.

Työn teoriaosuudessa on käyty läpi projektimuotoista työskentelyä, web-sovellusten toimintaa ja käytettävyyttä. Lopussa on kerrottu Project Routerin käyttöönotosta, käytöstä ja teknisestä toteutuksesta. Myös käytettävyyttä on arvioitu teoriaosuudessa esitettyjen määritelmien mukaan.

Työssä esiintyvät palvelimien nimet on muutettu tietoturvasyistä.

2 MIPRO OY

Mirpo Oy on vuodesta 1980 toiminut mikkeliäinen rautatieturvallisuuteen sekä vesi- ja lämpöhuollon hallintaan liittyviin järjestelmiin erikoistunut järjestelmätoimittaja. Yritys perustettiin nimellä Mikkelin Prosessiohjaus Ky ja alkuvaiheessa sen toimialana olikin prosessiteollisuuden sekä vesihuollon automaatio. Helmikuussa 1991 toiminimeksi vaihtui Mipro. Yrityksen liikevaihto on noin 10 miljoonaa euroa ja henkilöstöä on noin 70. Yrityksen päätoimipaikka on Mikkelin Rantakylässä ja sivutoimipisteet löytyvät Oulusta sekä uusimpana myös Slovakiasta.

1990-luvun alussa Mipro alkoi panostamaan voimakkaasti rautatiejärjestelmien kehittämiseen ja ensimmäinen tasoristeyksen ohjausjärjestelmä toimitettiin asiakkaalle vuonna 1995. Tähän päivään tultaessa Mipro on rautatieliikenteen turvallisuuteen liittyvien järjestelmien markkinajohtaja Suomessa. Ohjausjärjestelmätuotteista mainittakoon MiSO TCS ja MiSO CTC.

Vesilaitoksen toiminta käsittää usein veden hankinnan, jakeluverkoston, viemäriverkoston ja jätevedenpuhdistuksen. Kaukokäyttöjärjestelmien avulla maantieteellisesti kaukana olevien kohteiden prosessien ohjaus ja säätö voidaan hoitaa kustannustehokkaasti ja keskitetysti. Mipro on toimittanut vesihuollon ohjausjärjestelmiä jo 1980-luvulta lähtien. Mipron tuotteita vesi- ja energiahuollolle ovat mm. MiSO Puhdistamo ja MiSO Kaukokäyttö. Vesipuolen järjestelmissä hyödynnetään paljon langatonta tiedonsiirtoa.

Mipro tarjoaa elinkaaren palveluita tuotteilleen myös takuuajan loputtua. Ylläpitopalvelu mitoitetaan asiakkaan tarpeiden mukaan ja siihen voidaan sisällyttää esim. ympärivuorokautinen päivystyspalvelu, joka takaa järjestelmän jatkuvan käynnissäpidon.

Miprolla projektimuotoista työskentelyä sovelletaan miltei kaikkiin töihin. Toimitusprojekti lähtee käyntiin usein sillä, että asiakas hyväksyy Mipron tarjouksen ja projektille asetetaan projektipäällikkö. Projektipäällikkö suunnittelee tehtävät ja aikataulut tarjouksen mukaisesti. Aloituspäivästä projektiryhmä käy läpi projektisuunnitelman ja aikataulun. Projektin edetessä pidetään projektipalavereja. Projektilla on aina suunniteltu alkua- ja loppupäivä, budjetti ja projektipäällikkö. Miprolla on myös erilaisia

tiimejä (esim. ohjelmistokehitystiimi), jonka jäsenet koostuvat oman alansa asiantuntijoista. Tiimillä on aina tiimiesimies, joka jakaa töitä tiiminsä jäsenille.

3 PROJEKTITYÖSKENTELY

3.1 Projektin määritelmä

Alun perin latinasta peräisin oleva sana projekti tarkoittaa ihmisten ja resurssien joukkoa, jotka on tilapäisesti koottu suorittamaan tiettyä tehtävää. Projektille tyypillisiä piirteitä ovat mm. tavoite, elinkaari, itsenäinen kokonaisuus, ryhmätyöskentely, vaiheistus, ainutkertaisuus, muutos ja riski. Vaikka projekti on terminä hyvin yleisesti käytetty, sen lopputulokset ovat hyvin vaihtelevia. Projektin lopputuloksena voi olla niin konkreettinen tuote kuin myös ratkaisu johonkin vallitsevaan ongelmaan.

Projektia varten perustetaan aina projektiorganisaatio. Projektiorganisaatioon kuuluu mm. projektipäällikkö ja projektiryhmä. Projektipäällikön tehtävänä on vastata projektin johtamisesta, päätöksenteosta ja yhteydenpidosta organisaation ylemmille tasoille. Projektiryhmä koostuu ihmisistä, joille jaetaan tehtäviä heidän erikoisosaamisensa mukaan. Projektipäällikkö on projektiorganisaatiossa projektiryhmäläisten työnjohdollinen esimies. /1, s. 18 – 21./

3.2 Projektionnin hyödyt ja ongelmakohdat

Projektimuotoisella työskentelyllä saavutetaan paljon etuja. Projektille asetettujen selkeiden tavoitteiden takia voimavarat ja erikoisosaaminen on helppo kohdentaa oikeille ihmisille. Selkeän organisaatorakenteen avulla jokainen projektiryhmän jäsen ymmärtää oman paikkansa ja tehtävänsä. Projektointi mahdollistaa myös sen, että vastuuta voidaan jakaa normaaleista käytännöistä poiketen. Uusien haasteellisten tehtävien myötä työntekijän työmotivaatio lisääntyy. Improvisointi ja totutuista rutiineista poikkeaminen mahdollistuvat, kun huomio kiinnitetään tuloksiin eikä siihen kuinka ne saavutetaan. /1, s. 63./

Projekti voi helposti myös epäonnistua, mikäli projektinhallintaan ja sen menetelmiin ei kiinnitetä riittävästi huomiota. Useimmiten projektiryhmän jäsenet ovat alansa asiantuntijoita, joten projektin epäonnistuminen johtuu harvemmin ammattitaidon puutteesta. Huono organisointi ja puutteellinen suunnittelu ovat omiaan pilaamaan muuten pätevien ihmisten työn. Valmisteluvaiheessa projektin kustannukset ja hyödyt pitäisivät olla tarkasti selvillä. Kattavalla esiselvityksellä voidaan selvittää, kannattaako työtä ylipäättään tehdä projektimuotoisesti. Projekti on suunnitteluvaiheessa osattava rajata hyvin, jotta tiedetään mitä kaikkea tuleva projekti käsittää. Yllättävät muutosehdotukset ja lisätyöt venyttävät helposti projektin aikataulua ja muuttavat jo suunniteltua budjettia. Kesken projektin voidaan toki tehdä lisätoita ja muutoksia, mutta tällöin kaikkien osapuolten on oltava selvillä, kuinka nämä vaikuttavat projektin aikatauluun ja kustannuksiin. Mikäli huonosti suunniteltu projekti on saatu käyntiin, mutta sen tulevaisuus näyttää synkältä esim. motivaation ja resurssien puutteen takia, pitää olla uskallusta keskeyttää projekti ja katsoa tilannetta perusteista alkaen uudelleen. Tällaista tilannetta voidaan verrata uhkapelurin toimintaan. Uhkapeluri jatkaa pelaamistaan jotta voittaisi takaisin jo häviämänsä rahat. Peliä jatketaan sen toivossa, että onni kääntyisi. Näin kuitenkin harvemmin tapahtuu. Myös henkilöriiriidat, epärealistiset tavoitteet, puutteelliset aikataulut ja välitavoitteiden puuttuminen ovat omiaan aiheuttamaan projektin epäonnistumisen.

/1, s. 41 - 53./

3.3 Projektinhallinta

Projektinhallinnan ajatus on hyvin vanha. Ilman projektinhallintaa Egyptin pyramidit ja Rooman vesijohtoverkostot olisivat jääneet rakentamatta. Projektipäälliköiden työnkuvat olivat jo tuolloin samankaltaisia kuin nykyään. Projektipäälliköt ovat pyrkineet soveltamaan teknologiaa oman aikansa ongelmiin. /2, s. 2./ Projektinhallinnan alkuvaiheen osa-alueita ovat suunnittelu ja toimeenpano. Myöhempiä osa-alueita ovat päätöksenteko, ohjaus, koordinointi, valvonta, suunnan näyttäminen ja ihmisten johtaminen. /1, s. 30./

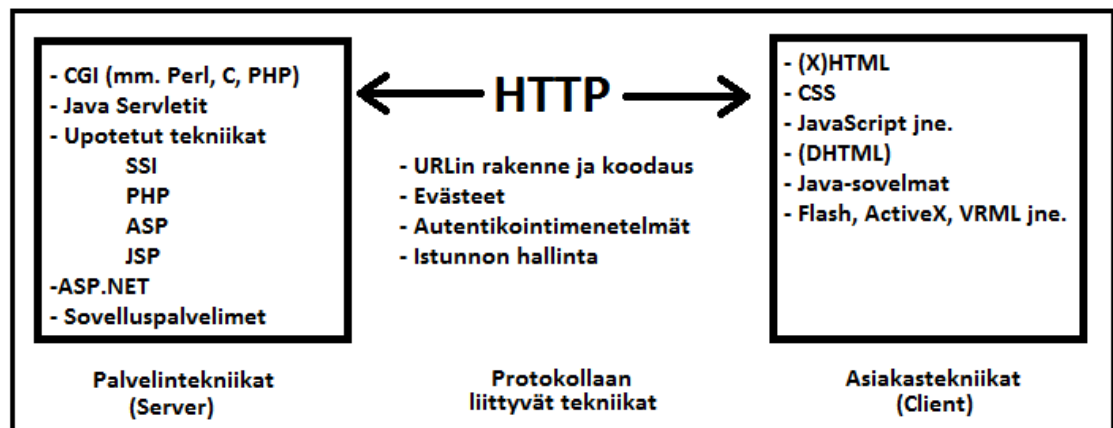
Projektinhallinta on riippuvainen organisaation rakenteesta, kulttuurista ja projektin tavoitteista. Se voi olla siis hyvin epämuodolinen rooli, eli vastuuta ei ole tarkasti

määritelty vaan projektin hoitoa tekee ”kuka kulloinkin kerkeää” tai vaihtoehtoisesti niin, että projektipäälliköt on tarkoin määritelty. Selvästi määritellyn projektipäällikön puuttuminen voi olla myös haitallista. Kun vastuu johdosta jaetaan löyhästi usealle eri ihmiselle, on vaarana että henkilökohtaiset mieltymykset johtavat siihen että työn eteneminen ja laatu kärsii. /2, s 9-10./

4 WEB-SOVELLUKSET JA NIIDEN TEKNIIKAT

4.1 Webin kerrosmallit

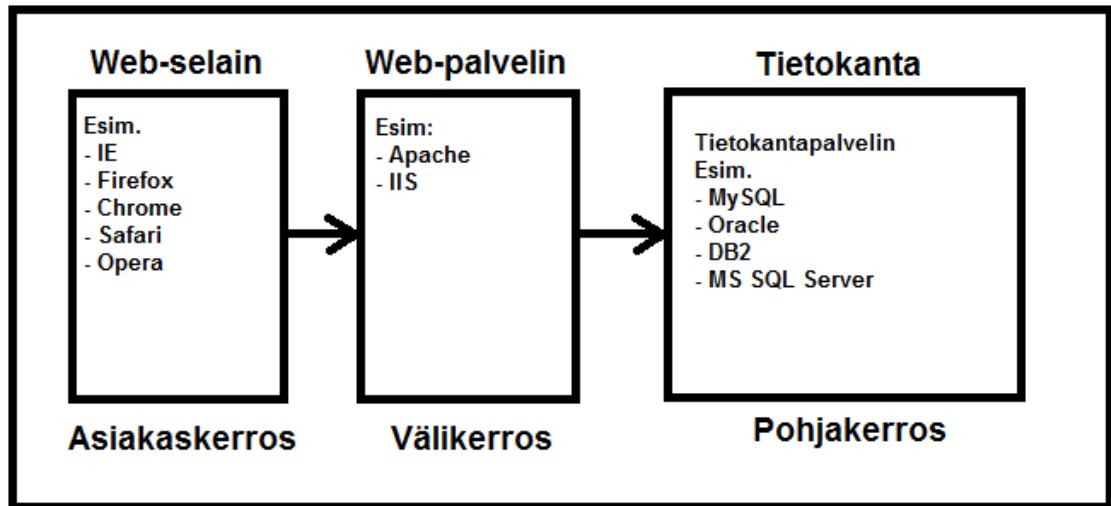
Webin toimintaa voidaan kuvata kuvan 1 mukaisella kaksiosaisella palvelin – asiakas-mallilla. Asiakasohjelma, tyypillisesti web-selain, on ohjelma, joka lähettää palvelimelle erilaisia pyyntöjä käyttäjän toimesta. Palvelinohjelma vastaa noihin pyyntöihin esim. lähettämällä selaimelle web-sivun. Sekä pyyntö että vastaus lähetetään HTTP-protokollaa hyödyntäen. /3, s. 2./



KUVA 1. Asiakas–palvelin-malli webissä. /3, s. 2./

Suosituin web-palvelinohjelmisto on Apache n. 58 prosentin osuudella kaikista palvelimista. Toisena tulee Microsoftin IIS, jonka osuus palvelinohjelmistoista on noin 22 prosenttia (tilanne lokakuussa 2010) /4/. Web-palvelinohjelmisto tarvitsee toimiakseen palvelinkäyttöjärjestelmän. Linuxilla on useita palvelinkäyttöjärjestelmiksi sopivia jakeluita, Microsoftin tuote on Windows Server -tuoteperhe.

Kun web-sovellus hyödyntää jotakin tietokantaa, ei se voi hyödyntää asiakas-palvelin-mallia sellaisenaan, koska selaimissa ei useimmiten ole mahdollisuutta toimia suoraan tietokantapalvelimen asiakkaana. Tällöin voidaan puhua ns. kolmikerrosmal-lista (three-tier model, kuva 2), jossa web-selaimen ja tietokantapalvelimen välissä käytetään välikerrosta, joka toimii palvelimena web-asiakkaille ja asiakkaana tieto-kantapalvelimelle. /3, s. 254./



KUVA 2. Kolmikerrosmalli. /3, s. 254./

4.2 Web-sovelluksen rakenne

Web-sovellus on verkon kautta jaettava ohjelmisto, jolla on web-käyttöliittymä. Web-sovellus voi toimia joko julkisen internetin yli tai pelkästään yrityksen omassa sisä-verkossa, intranetissä. /5./

Web-sovelluksen voidaan katsoa rakentuvan seuraavista osista:

- Käyttöliittymästä (käyttäjän web-selaimessa)
- Sovelluslogiikasta (web-palvelimella)
- Käyttöliittymän ja sovelluslogiikan välisestä viestinnästä

/3, s. 6./

Käyttöliittymä sijaitsee web-selaimessa asiakaspuolella. Se rakentuu minimissään x(html)-elementeistä ja web-selaimen ominaisuuksista. Lisäksi voidaan käyttää erilaisia elävöittämistekniikoita, kuten esimerkiksi Flash, CSS tai JavaScript. Erilaisia asiakastekniikoita hyödyntämällä voidaan saada dynaamisuutta web-sovelluksen käyttö-

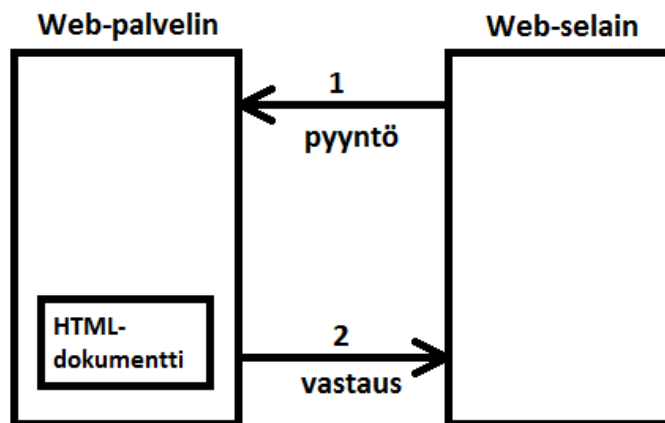
liittymän hallintaan. Sovelluslogiikka suoritetaan useimmiten palvelintekniikoiden avulla, koska järjestelmän tiedot eivät ole käyttäjien koneilla. Usein web-sovellukset joudutaankin integroimaan tietokantoihin. Perinteisesti on käytetty SQL-kieleen perustuvia relaatiotietokantoja. Sovelluslogiikan toteuttamiseen on runsaasti eri tekniikoita, esim. PHP, JSP ja ASP.NET. Käyttöliittymän ja sovelluslogiikan välinen kommunikointi tapahtuu HTTP-protokollan avulla. /3, s. 6-8./

Yhteenkuuluvaa ja tallennettavaa tietokokoelmaa voidaan kutsua tietokannaksi. Tietokannalle on myös ominaista, että tiedon rakenne ei ole riippuvainen sitä käsittelevistä välineistä. Tietokantojen tietomäärien käsittelyyn on usein tehokkaat välineet, joiden avulla haluttu tieto löydetään nopeasti ilman, että jokainen rivi jouduttaisiin käymään läpi. Myös yhteiskäyttö on tietokannalle ominaista, tiedot eivät vahingoitu vaikka niitä käsitelisi yhtäaikaaisesti useampi käyttäjä. Tiedon säilyvyys on myös tärkeä vaatimus tietokannalle. Tieto pitää voida varmuuskopioida luotettavasti vaikka se olisikin käytössä koko ajan. /3, s. 252./

Eniten käytetyt tietokannat perustuvat relaatiomalliin. Relaatiotietokantojen käsittely tapahtuu nykyisin useimmiten SQL-kielellä (Structured Query Language). SQL-standardi mahdollistaa, että eri valmistajien tietokantoja voidaan käsitellä samalla tavalla. Relaatiomallissa tietoalkiot muodostavat relaatioita. Henkilötaulu relaatiosta voisi ilmetä, että tiedot etunimi, sukunimi ja osoite kuuluvat yhteen. Usein relaatioita kutsutaan tauluiksi. Useimmiten relaatiotietokanta pitää sisällään useita tauluja. Taulun yhtä eri tiedoista muodostuvaa riviä kutsutaan tietueeksi. Relaation määrittäviä tietoja kutsutaan kentiksi. Relaatiomallin määritelmän mukaan yksikään tietue ei saa olla identtinen toisen tietueen kanssa, näin ollen kahden samanlaisen rivin olemassaolo on mahdotonta. Aiemmin luvussa 4.1 käsitelty asiakas-palvelin malli on yksinkertaisin tapa käyttää tietokantaa. Asiakasohjelma lähettää SQL-kyselyn tietokantapalvelimelle, joka suorittaa kyselyn ja palauttaa tuloksena vastauksen asiakkaan ruudulle. Näin tietokantaa voi käyttää useat eri sovellukset yhtä aikaa. Tunnettuja tietokannan hallintajärjestelmiä (RDBMS, Relational Database Management Systems) ovat mm. Oracle, MySQL, DB2 ja Microsoftin SQL Server. /3, s. 252-254./

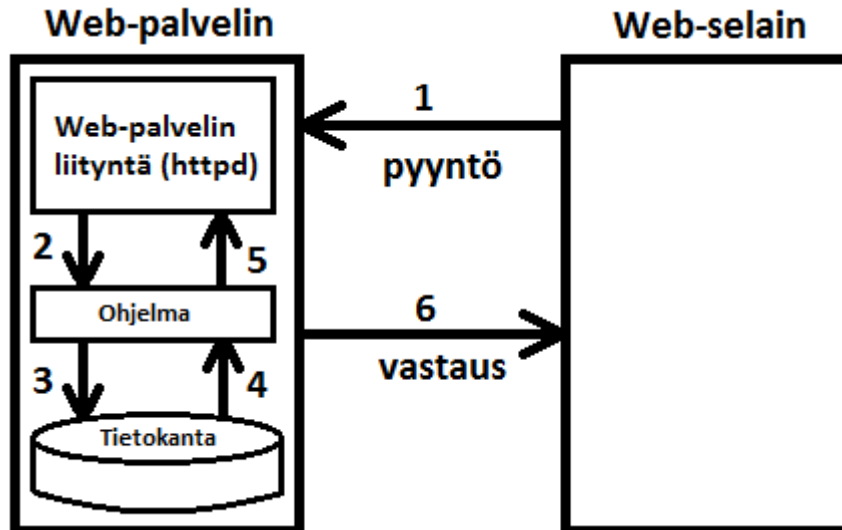
4.3 Webin staattisuus ja dynaamisuus

Perinteisesti web-sivut ovat olleet staattisia eli muuttumattomia. Kuvassa 3 on havainnollistettu tämä periaate. Linkkejä klikkailemalla selain lähettää pyyntöjä web-palvelimelle, joka vastaa html-dokumentilla. Staattisen sivun muuttaminen vaatii sen, että dokumentti on julkaistava web-palvelimille uudestaan. Staattisten web-sivujen muuttaminen voi olla pahimmillaan todella työlästä ja hankalaa. Staattiset web-sivut sopivat kuitenkin hyvin sellaisiin sivuihin, joita tarvitsee päivittää harvoin ja vuorovaikutusmahdollisuutta ei tarvita. /3, s. 3./



KUVA 3. Staattinen web-dokumentti. /3, s. 3./

Web-sivut voivat sisältää myös muuttuvaa tietoa. Erilaisten tietojärjestelmien tiedot ovat poikkeuksetta muuttuvia, muutenhan olisi edes turha puhua ATK:sta (automaattinen tietojenkäsittely). Tällaiset sivut ja järjestelmät vaativat ajan tasalla pysyäkseen helppokäyttöistä päivitettävyyttä. Staattisiin html-dokumentteihin ei helpot päivitykset onnistu. Näiden tarpeiden pohjalta on syntynyt tekniikoita, joiden avulla voidaan luoda web-sivuja osittain tai kokonaan dynaamisesti. Jos dynaaminen web-sivusto (kuva 4) on toteutettu jotain tiettyä sovellusaluetta ajatellen, voidaan puhua web-sovelluksesta. Koska web-sovellukset toimivat käyttäjän päätösten mukaan, puhutaan usein myös interaktiivisista web-sivustoista. Web-sovellus voi sisältää toki myös staattisiksi luokiteltuja sivuja. /3, s. 4./



KUVA 4. Dynaaminen web-dokumentti. /3, s. 4/

4.4 Web-sovelluksen testaus

Web-sovelluksen testauksessa on huomioitava seuraavia asioita:

- Testikohteiden valinta ja priorisointi
- Testauksen raportointi
- Testauksenaikainen seuranta
- Testiympäristön luominen
- Käytettävyys
- Yksikkötestaus
- HTML-koodin laadun varmistaminen
- Latausaikojen testaaminen
- Hyväksymistestaus
- Tietoturvan testaaminen

/6./

Testikohteiden valinnassa pitää osata priorisointia. Jos esimerkiksi web-sovelluksen tietoturvasuus on erittäin tärkeää, kannattaa myös testaus suunnitella niin että pääpaino on kriittisimmillä kohteilla. Web-sovellus voi myös olla täysin julkinen ja se ei sisällä näin mitään tietoturvan kannalta kriittistä informaatiota. Tällöin testauksen pääpaino voidaan asettaa esim. visuaaliseen ulkoasuun. /6./

Myös testauksen raportointi pitää hoitaa hyvin. Testauksen alussa pitää tietää kuinka, miten ja minne raportointi tehdään. Kuka tarvitsee testauksen perusteella syntyneitä

raportteja? Sovitaanko testauksen aikaisia palavereja ennakkoon vai vasta tarpeen tullen? Testauksen aikainen seuranta on myös tärkeää, esim. jos testausta suorittaa useampi henkilö. Testauksessa tehdyistä löydöksistä pitää raportoida niin, että kaikki testaukseen osallistuvat saavat tiedon. /6./

Testiympäristö kannattaa pitää erillään varsinaisesta tuotantoympäristöstä, jopa sijoittamista eri palvelimille kannattaa miettiä. Käytettävyydestestauksessa pitää tarkastella sovellusta loppukäyttäjän silmin. Kuinka helppoa navigointi ohjelmistossa on? Ymmärtääkö käyttäjä helposti mitkä ovat tärkeimmät toiminnot juuri hänelle? Ovatko eri sivujen ulkoasut yhtenevät keskenään (fontit, värit jne.)? /6./ Luvussa 4.5 web-käytettävyyttä käsitellään tarkemmin. Yksikkötestauksessa sovelluksesta testataan pientä osaa, voidaan keskittyä esimerkiksi testaamaan, että tiedot tallentuvat tietokantaan, kun Tallenna-painiketta painetaan sivulla. /6./

Web-sivun HTML-merkintäkieli (Hyper Text Markup Language) on kieli, jolla web-palvelin vastaa käyttäjän pyyntöihin. Selaimet tulkitsevat HTML:ää ja näin muodostavat sivun käyttäjän selaimeen. W3C (World Wide Web Consortium) on kansainvälinen yhteisöjen ja yritysten yhteenliitymä, joka kehittää ja ylläpitää WWW:n suosituksia. Koska web-sovelluksen vaatimuksena saattaa olla, että sen on toimittava mistäpäin maailmaa tahansa WWW:n yli, on tärkeää että html-koodi on virallisten suositusten mukaista. W3C:n suosituksia ovat mm. CSS, HTML, HTTP, XML ja XHTML. /6, 7./

W3C:n tavoitteena on ylläpitää suosituksia ja suuntaviivoja, joiden avulla varmistetaan WWW:n tulevaisuus ja yhteneväisyys. /8./ W3C tarjoaa ilmaisen W3C Markup Validation Service validaattorin (<http://validator.w3.org/>), joka tarkistaa web-dokumenttien (kuten HTML ja XHTML) rakenteen ja vertaa sitä virallisiin suosituksiin. Tarkastettavan koodin voi syöttää antamalla suoran URL-osoitteen, lataamalla tiedoston omalta tietokoneelta tai kirjoittamalla koodin suoraan tekstikenttään. Sovellus käy nopeasti koodin läpi ja raportoi mahdollisista poikkeamista suosituksiin. /9./

Pienet visuaaliset muutokset saattavat vaikuttaa ratkaisevasti web-sovelluksen latausaikoihin. Tämän vuoksi on tärkeää testata latausaikoja jo hyvissä ajoin. Ihmiset

vihaavat tunnetusti latausaikoja. Yleisesti ottaen kaikkien ohjelman ominaisuuksien avautuminen pitäisi kestää vähemmän kuin 15 sekuntia. /6./

Hyväksymistestauksen ideana on loppukäyttäjän avulla varmistaa, että sovellus palvelee käyttötarkoitustaan. Tehokas tapa on luoda käyttäjille lopullisen tuotteen kaltainen testausympäristö. Nykyään maailma on täynnä taitavia hakkereita, joten tietoturvaan ei voi kiinnittää huomiota liian paljon web-sovellusten kohdalla. Web-sovellusten tietoturvan suunnittelu ja varmistus pitäisi antaa ammattilaisten suoritettavaksi. Tietoturvan säilymisestä täytyy pitää huolta läpi sovelluksen eliniän, huolimatta ihmisten ja tekniikan vaihtuvuudesta. /6./

4.5 Web-käytettävyys

Web-sovelluksen käytettävyyttä voidaan tarkastella Jacob Nielsenin määritelmän kautta. Sen mukaan käytettävyys on minkä tahansa sovelluksen tai laitteen käyttökelpoisuutta. Riittämätön käytettävyyden taso tekee sovelluksen tai laitteen käytön mahdottomaksi. Nielsenin määritelmän mukaan käytettävyys jakaantuu viiteen osaluokkaan: opittavuuteen, tehokkuuteen, muistettavuuteen ja miellyttävyyteen /10, s. 28/.

Web-palvelun tai sovelluksen opittavuus alkaa siitä kun käyttäjä avaa sen selaimensa ensimmäisen kerran. Mikäli ensivaikutelma on huono eikä käyttäjä saa vastauksia haluamiinsa asioihin, hän siirtyy toiseen palveluun. Tämän takia on kiinnitettävä huomiota ensimmäisen käyttökerran helppouteen eli opittavuuteen. Mikäli ihminen oppii käyttämään sovellusta, hän haluaa saada siitä kaiken tehokkuuden irti. Sovelluksen oikopolkujen ja linkkien avulla käytön tehokkuutta voidaan parantaa; ihmiset aloittavat harvoin oma-alotteisesti optimoida toimintaansa. Muistettavuudella tarkoitetaan sitä, että sovelluksen suunnittelu on pysyvää ja näin ollen käyttö muistuu helposti mieleen pitkienkin taukojen jälkeen. Koska ihmisellä on todistetusti hyvä kuvamuisti, on myös visuaalinen ulkoasu todistetusti hyväksi muistettavuuden kannalta. Virheetömyydellä tarkoitetaan, että sovellus ei saa ohjata käyttäjää tekemään virheitä. Sivusto pitää suunnitella niin, että käyttäjän näkyville pomppaavat virheilmoitukset ovat viimeinen keino. Sovelluksen tulee antaa käyttäjälle aktiivisesti visuaalista palautetta, jotta virhetilanteilta vältytään. Esimerkiksi haku-painiketta painettaessa kursorin viereen ilmestyy tiimalasi, joka helpottaa käyttäjää ymmärtämään, että sovellus lataa ha-

kutuloksia eikä suinkaan jumiutunut. Viidennellä osa-alueella, miellyttävyydellä, tarkoitetaan sitä, että sivusto tai sovellus ei loukkaa käyttäjiä tai johdata heitä tarkoituksella harhaan. /10, s. 28 – 31./

Toinen suosittu määritelmä käytettävyydelle on ISO 9241-11:n (Kansainvälinen standardoimisliitto, standardi näytepääteyöskentelystä, ohjeita käytettävyyteen) määritelmä käytettävyydestä mitattava ominaisuutena. /10, s.27./

ISO 9241-11 toteaa että käytettävyys riippuu aina käyttäjästä ja hänen tekemisen tyypistä, käytettävistä työvälineistä ja käyttäjän ennakkotiedoista. Määritelmän mukaan käytettävyyttä voidaan mitata seuraavissa asioissa: /10, s. 31./

- tehokkuus
- taloudellisuus
- miellyttävyys

Tehokkuudella voidaan mitata kuinka paljon parempaan lopputulokseen uudella tuotteella voidaan päästä, verrattuna siihen että oltaisiin pysytty entisessä tilanteessa tai valittu kilpaileva tuote. Taloudellisuutta voidaan mitata sillä, kuinka paljon kustannuksissa on säästetty ajallisesti tai rahallisesti. Miellyttävyyden mittausta perustuu samoihin asioihin kuin Nielseninkin määritelmässä. ISO 9241-11:n määritelmä sopii paremmin uuden ja vanhan vertailuun, kun taas Nielsenin määritelmä auttaa uuden sovelluksen tärkeimpien ominaisuuksien tunnistamisessa. /10, s. 31 – 32./

4.6 Microsoft .NET-arkkitehtuuri ja ASP.NET

.NET Framework (usein myös pelkkä .NET) on Microsoftin luoma sovellusalusta, ohjelmointiympäristö ja ajoympäristö, joka on suunnattu erityisesti Windows-pohjaisille yrityssovelluksille. Vuonna 2002 julkaistiin .NETin ensimmäinen versio, Microsoft halusi luoda ohjelmointiympäristön joka olisi erittäin laaja-alainen. Yksi .NETin tärkeimmistä ominaisuuksista on ollut alusta asti riippumattomuus käytetyistä ohjelmointikielistä. Suosittuja .NET-yhteensopivia ohjelmointikieliä ovat C#, Visual Basic.NET (VB.NET), Delphi for .NET, IronRuby sekä C++. .NET-sovellukset pyö-

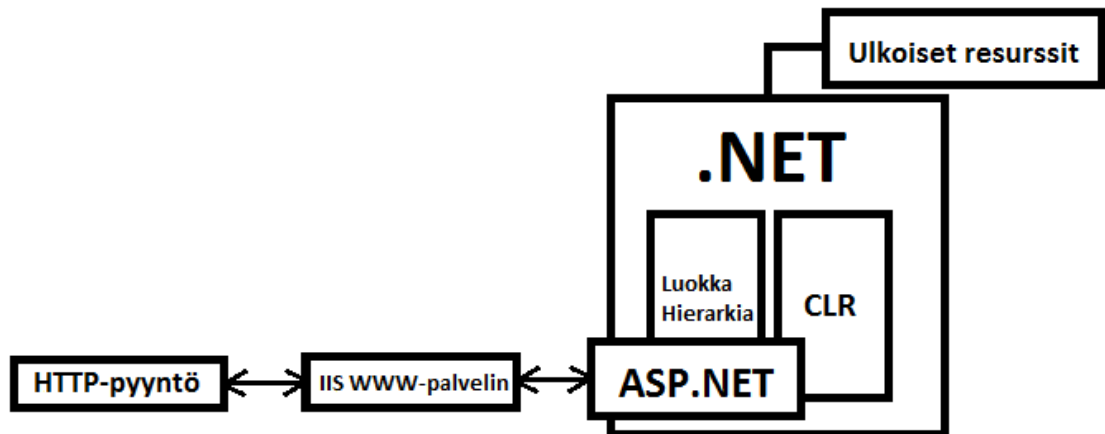
rivät virtuaalikoneen alaisuudessa, jota kutsutaan nimellä CLR (Common Language Runtime). Kun ohjelmoija kirjoittaa koodia esim. C#-kielellä ja kääntää koodin kääntäjällään, tuloksena ei ole perinteistä koodia, vaan virtuaaliselle suorittimelle tehtyä välikoodia CIL:ää (Common Intermediate Language). Aiemmin sitä kutsuttiin nimellä MSIL (Microsoft Intermediate Language). CIL ja CLR kuuluvat CLI:een. CLI on Microsoftin avoin ympäristö ja se muodostaa .NET Frameworkin ytimen. Ilman .NET-ajoympäristöä (runtime) .NET-pohjaisia sovelluksia ei saa käyntiin. Ajoympäristö on maksuton ohjelmistopaketti ja kenen tahansa ladattavissa Microsoftin sivuilta. /11, s. 24 - 25./

Tässä työssä esiintyvä projektinhallintaohjelmisto on toteutettu .NETin web-sovellusten tekniikalla, ASP.NETillä. Se muistuttaa hiukan edeltäjäänsä ASP:tä (Active Server Pages), mutta ajonaikainen toiminta on täysin erilaista. ASP.NETin tarkoitus on tehdä web-sovellusten kehittäminen mahdollisimman helpoksi. /11, s. 27./

ASP.NET-sovelluksien rakenne voi vaihdella, mutta yleisellä tasolla ne noudattavat kolmikerrosmallia (three-tier model) tai monikerrosmallia (n-tier model). /12, s. 58./

Vaikka teoriassa .NET ja ASP.NET ovat alustariippumattomia, tarvitaan ASP.NET sovellusten ajamiseen Microsoftin web-palvelin, IIS (Internet Information Services). Windows Server 2008 käyttää IISin versiota 7.0. IIS asennetaan Windows Server 2008:lle lisäämällä web-palvelimen rooli. Roolin valinnan jälkeen valitaan roolin palvelut, esim. ASP.NET-palvelu. /11, s. 379-380./

ASP.NET-sovellukset suoritetaan .NET-runtime ympäristössä kuvan 5 mukaisesti. Asiakkaan selaimelta lähtevä HTTP-pyyntö käsitellään IIS-palvelimessa. IIS-tunnistaa pyynnön kuuluvaksi ASP.NET-ympäristölle tiedostomuodon perusteella (esim. aspx). IIS antaa ASP.NETille toimintapynnön, jolloin käynnistetään sovelluskoodin suoritus. Suorituksen aikana hyödynnetään .NET-luokkahierarkiasta pyydettyjä resursseja, haetaan ulkoiset resurssit, käännetään rutiini CLR-välikielelle ja suoritetaan ohjelman pyytämä toiminto. Palvelimelle syntyy proxy-luokka, jota hyödynnetään myöhemmissä pyynnöissä. Lopulta valmiiksi käännetty tulos palautetaan pyynnön esittäjälle HTML-muodossa www-palvelinohjelmiston avulla. /12, s. 59 – 60./



KUVA 5. ASP.NET sovelluksen suorittaminen. /12./

Useimmiten web-sovellukset tarvitsevat toimintaansa tietokantoja. .NETissä tietokantoja käsitellään ADO.NET nimisen tekniikan avulla, joka mahdollistaa pääsyn erilaisiin tietokantatyyppeihin, kuten tärkeimpiin SQL-tietokantoihin (MS SQL Server ja Oracle). /11, s. 27/

4.7 Pilvipalvelut

Pilvipalvelut ovat nykypäivän IT-maailmassa kasvava trendi ja niiden avulla monista vanhoistakin konsepteista yritetään ”pilvipesun” avulla tehdä kiinnostavia tuotteita. Mainostoimistojen pilvipesu on aiheuttanut sen, että pilvi sotketaan asioihin, joihin se ei millään tavalla kuulu. Perinteinen ASP-palvelu (Application service provider) sotketaan usein pilveen. /13./

Pilvi-sana on peräisin jo 1980-luvulta, jolloin puhelinoperaattorit kuvasivat asiakkaan ja operaattorin laitteiden rajapintaa pilvisymbolilla. Tapa siirtyi tietoliikennelaitteiden valmistajille, jotka alkoivat kuvaamaan verkkoja pilvisymbolilla. Pilvipalvelulle ei ole vielä syntynyt mitään vakiintunutta määritelmää ja sitä on vaikea määrittellä standardien avulla. /14, s. 10./

Konsulttiyhtiö Accenture määrittelee pilvipalvelut seuraavasti: ”Pilvipalvelut ovat palveluntarjoajan IT-resurssien, kuten ohjelmistojen, laitteiston tai palvelujen, dynaamista tarjoamista asiakkaiden käyttöön verkon välityksellä.” /15, s. 16-17./

Yhdysvaltojen elinkeinoministeriön alaisuudessa toimiva NIST (National Institute of Standards and Technology (NIST) määritelmän mukaan pilvipalvelun ominaispiirteitä

ovat itsepalvelullisuus, pääsy palveluihin eri päätelaitteilla, resurssien yhteiskäyttö, nopea joustavuus ja käytön tarkka mittaaminen. Microsoftin määritelmä pilvipalveluille on samankaltainen. Ominaispiirteitä ovat dynaaminen laskentainfrastruktuuri, palvelukeskeinen lähestymistapa, käyttöön perustuva laskutus, automatisoitu hallinta ja itsepalvelu. /15, s.17./

NIST on esittänyt neljä tapaa pilvipalveluiden käyttöönottoon:

- yksityinen pilvi
- yhteisöllinen pilvi
- julkinen pilvi
- hybridipilvi

/15, s. 19./

Kansainvälinen IT-yritysten (mm. IBM ja Motorola) organisaatio Jericho Forum määrittelee pilvipalvelun ulottuvuudet seuraavasti:

1. sisäinen – ulkoinen
2. suljettu – avoin
3. rajattu – ei rajattu
4. itse tuotettu – ulkoistettu

/15, s. 19./

Pilvipalvelumalleja on kolme. Infrastruktuurille (IaaS, Infrastructure as a Service), sovellusalustalle (PaaS, Platform as a Service) ja sovelluksille (SaaS, Software as a Service). IaaS on tietotekniikkainfrastruktuurin eli fyysisten laitteiden tarjoamista palveluna. PaaS mallilla tarkoitetaan sovelluskehitysympäristön tarjoamista palveluna. Malleista tunnetuin, SaaS, tarjoaa sovelluksen palveluna perinteisen ostamisen, omistamisen ja ylläpitämisen sijaan. SaaS-palvelun käyttöliittymä on usein web-selain. /15, s.22./

SaaS-palvelussa yritys maksaa käytöstä käytön määrän mukaan perinteisen lisenssi-pohjaisen maksun sijaan. Monikäyttäjäisyys (multitenancy) on myös SaaS-palvelulle yleinen piirre, eli samaa sovellusta käyttää laaja asiakaskunta. Laaja asiakaskohtaistamistyö ja räätälöinti ei ole tavallista, vaan palvelulle on ominaista ”bulkkimaisuus”. /15, s. 29./

Web-käyttöliittymä ja toimivuus verkon kautta ovat Project Routerin tunnusmerkit pilvipalvelulle, mutta määritelmiä vastaan sotivia piirteitäkin löytyy. Käyttöön perustuva laskutus ei toteudu, sillä Project Routerista on maksettu ostamalla siihen lisenssi.

Hinta on sama käytön määrästä riippumatta. Myöskään ulkoistettu ylläpito ei toteudu kokonaisuudessaan, vaikka ohjelmiston toimittajalla etäyhteys ohjelmistoon onkin. Koska ohjelmisto jo alussa tiedettiin sijoitettavaksi Mipron palvelinhuoneeseen, tiedettiin, että käyttöönotossa tarvitaan järjestelmävastaavan ammattitaitoa niin palvelimen asennuksessa kuin etäyhteyden luomisessa. Näin ollen Project Routeria ei voida mielestäni pitää pilvipalveluna, vaikka se osan pilven tunnusmerkeistä täyttääkin.

4.8 Virtualisointi

Virtualisoinnilla tarkoitetaan tekniikkaa, jolla yksi fyysinen resurssi, kuten palvelin, käyttöjärjestelmä, sovellus, tallennusväline tai verkko voi toimia useampana loogisena resurssina tai useammat fyysiset resurssit yhtenä loogisena resurssina. Virtualisointi mahdollistaa paremman käyttöasteen laitteistoille ja ohjelmistoille, säästää fyysistä tilaa palvelinhuoneessa, nopeuttaa käyttöönottoa, laskee sähkölaskua ja helpottaa ongelmatilanteista ylipääsemistä. /15, s. 47 – 48./

Virtualisointiohjelmistoja tarjoava VMwaren mukaan viisi tärkeintä syytä virtualisointiohjelmiston käyttöön ovat:

1. Laitteistoresurssien parempi hyödyntäminen. Yhdelle fyysiselle laitteelle voidaan sijoittaa paljon toimintoja.
2. Kustannussäästöt laitteistossa ja hallintatöissä. Fyysisten laitteiden vähentyessä laskee sähkö- ja jäähdytyskulut. Tehokkaiden ja keskitettyjen hallintatyökalujen myötä myös hallinta helpottuu.
3. Laitteistojen ja sovellusten saatavuus paranee tehokkaan varmuuskopioinnin ja ongelmatilanteista palautumisen ansiosta.
4. Joustavuus ja dynaamisuus vallitsevan tarpeen mukaan.
5. Työpöytäympäristön helppous. Käyttäjät pystyvät avaamaan virtuaaliset työpöytätyökymät joko paikallisesti tai etäyhteyksien avulla.

/16./

5 PROJECT ROUTER

Project Router on kotimaisen ohjelmistotalo Improly Oy:n tuote yritysten projektinhallintaan ja resurssointiin. Tässä kappaleessa on esitelty käytännön työ alkuvaiheen demojärjestelmän testauksesta aina käyttöönottoon asti.

5.1 Ohjelmiston demojärjestelmä

Project Routerin demojärjestelmä saatiin käyttöön 12.1.2011. Demojärjestelmä toimi Improlytyn palvelimilta julkisen internetin yli. Noin kuukauden ajan sovellusta testattiin kaikilta osin demojärjestelmän avulla ja vioista sekä muutostarpeista raportoitiin toimittajaa. Tuon kuukauden aikana järjestin myös sisäisiä palavereja mm. projekti-päälliköille ja tiimiesimiehille, joissa he saivat tuoda oman näkemyksensä ja tarpeensa sovelluksen suhteen esiin.

5.1.1 Mipron vaatimusmäärittely

Projektinhallintaohjelmiston toimittajalle oli Mipron toimesta lähetetty alustavaa vaatimusmäärittelyä jo lokakuussa 2010, josta kävi ilmi pääpiirteissään Mipron projektityöskentelyn periaatteet. Joulukuussa 2010 pidimme Miprolla varsinaisen määrittelypäivän yhdessä ohjelmiston toimittajan kanssa. Koska toimittajalle oli toimitettu alustavaa selvitystä Mipron projektinhallinnan piirteistä jo aiemmin, kävi lopullisen vaatimusmäärittelyn (SRS) laatiminen määrittelypäivänä helpommin. Määrittelypäivänä saimme määriteltyä yksittäisen projektin (taulukko 1) ja tehtävän (taulukko 2) tiedot.

TAULUKKO 1. Projektin tietojen määrittely.

Kenttä	Tietotyyppi	Muuta	Salkku	Suodatus
Toimiala	Valintalista	Infra / TLJ	X	X
Projektilaji	Valintalista	Riippuvainen toimialasta	X	X
Projektin nimi	Valintalista	Koodi + Nimi	X	
Projektipäällikkö	Käyttäjävalinta		X	
Myyjä	Käyttäjävalinta			
Hinta/Budjetti	Numeroarvo	Salkkunäkymään oikeustaso	X	
Valmiusprosentti	Symboli	Ympyrägraafi, 0-100%, 10% välein	X	
Projektin status	Värisymboli	Vihreä, keltainen, punainen	X	
Suunnitellut tunnit	Ei muokattavissa	Hakee projektin ja sen aliprojektien suunnitellut tunnit		
Toteutuneet tunnit	Ei muokattavissa	Hakee projektin ja sen aliprojektien toteutuneet tunnit		
Projektin alku ja loppu	Päivämäärävalinta	Erilliset päivämäärävalinnat		
Vastaanotettu	Päivämäärävalinta			
Takuu päättynyt	Päivämäärävalinta			
Viimeksi päivitetty	-	Hakee käyttäjän, kellonajan ja päivämäärän		
Osaprojektit	Projektivalinta			

Salkku-valinnan omaavat rivit näkyvät siis projektisalkussa, jossa voi tarkastella kaikkia projekteja tai suodattaa toimialan tai projektilajin mukaan. Salkkuun määritellyt tiedot ovat siis projektin olennaisimmat tiedot.

TAULUKKO 2. Tehtävärivin tietojen määrittely.

Kenttä	Tietotyyppi	Muuta
Tehtävän tyyppi	Valintalista	Koodi +nimi. Riippuu projektilajista
Tehtävän kuvaus	Tekstikenttä	
Tehtävän vastuu	Käyttäjä/Tiimi	
Aikataulu	Päivämäärävalinta	Alku- ja loppupäivä
Suunniteltu kesto	Numero	Tuntiarvo
Toteutunut kesto	Numero	Tuntiarvo (myöhemmin tuntienkirjausjärjestelmästä)
Tulos	Tekstikenttä	
Kuittauspäivämäärä	Päivämäärä	Automaattisesti pvm. kun kuitattu

Projekti koostuu eri tehtävistä. Yksittäisen tehtävärivin kentät saatiin määriteltyä taulukon 2 mukaisesti.

Projektin tilat sovittiin seuraaviksi: prospekti, aktiivinen, vastaanotettu ja päättynyt. Projekti on prospekti silloin kun sen alkaminen on vasta tulevaisuudessa eikä varsinaista projektinnumeroakaan ole vielä määritelty hallinnon toimesta. Sekä aktiivinen että vastaanotettu tila tarkoittavat että projekti on käynnissä. Projekti on vastaanotettutilassa kun se luovutetaan asiakkaalle. Projekti on päättynyt-tilassa kun takuu-aika on ohi eikä ko. projektille kirjata enää kuluja. Mipron järjestelmien takuuajat ovat jopa useita vuosia.

5.1.2 Demojärjestelmän testaus

Miprolla pidetyn määrittelypäivän perusteella toimittaja loi ohjelmistosta demojärjestelmän, jonka saimme käyttöön tammikuun 2011 alussa. Demojärjestelmää alettiin tarkastella joulukuussa tehtyä vaatimusmäärittelyä vasten. Kun kaikki kohdat oli käyty läpi, voitiin todeta että toimittaja on onnistunut toteuttamaan miltei kaikki vaatimusmäärittelyn vaatimukset.

Vaatimusmäärittelyn tarkastamisen jälkeen pystyin aloittamaan varsinaisen ohjelman testauksen. Testauksessa yritin käydä läpi kaikki luvussa 4.4 mainitut web-sovelluksen testaukseen olennaisesti liittyvät kohdat, jotka asiakkaan näkökulmasta on mahdollista käydä läpi. Testauksessa ei ilmennyt suuria puutteita. Pääasiassa testaus keskittyi käytettävyyden testaukseen Mipron näkökulmasta.

Käytettävyydestestauksen aloitin luomalla järjestelmään käyttäjät, toimialat, projektilajit, tehtävätyypit Mipron projektityöskentelyrakenteen mukaisesti. Uuden käyttäjän luonnissa ei tarvinnut määrittää muuta kuin etunimi, sukunimi, sähköpostiosoite ja käyttäjän kielisyys (suomi/englanti).

Toimialat, projektilajit ja tehtävätyypit tehtiin identtisiksi tuntienkirjausjärjestelmän kanssa, jotta myöhemmin tapahtuva integrointi ehkä helpottuisi.

Loin järjestelmään projekteja oikeiden projektien tiedoilla; tiedot sain Outlook-tehtävistä sekä verkkolevyjen projektienseuranta taulukoista. Samalla kävin läpi aiemmin luotua vaatimusmäärittelyä. Loin demojärjestelmään heti alussa oikeat käyttä-

jät, eli koko Mipron henkilöstön. Lisäksi tein omia testikäyttäjiä käyttöoikeuksien testausta varten.

Demojärjestelmä testattiin 20.1.2011 viidellä yleisimmällä selaimella /17./

- Mozilla Firefox (versio 3.6.13)
- Internet Explorer (versio 8.0.7600.16385)
- Google Chrome (versio 8.0.552.237)
- Safari (Versio 5.0.3; 7533.19.4)
- Opera (Versio 11.00, koontikäännös 1156)

Sovelluksen toimintaa englanninkielisenä pääsi testaamaan, sen jälkeen kun projektilajeille ja tehtävätyypeille oli määritelty vieraskieliset vastineet. Kielisyyttä testattiin kaikilla mahdollisilla sivuilla, virheitä ei esiintynyt ja käännöstyö oli suoritettu hyvin. Demojärjestelmässä havaitut ongelmat ja kehitysideat kirjattiin dokumentinhallintajärjestelmään omaan taulukkoonsa, josta ilmeni ongelman tyyppi, raportointiaika toimittajalle ja ratkaisuaikaa. Tämä helpotti vikojen seuranta itselleni sekä esimiehilleni.

Ohjelman varmistusta ei pystytty testaamaan demojärjestelmässä, koska ohjelma toimi julkisen verkon yli toimittajan palvelimelta. Varmistuksen testaus päätettiin suorittaa, kun sovellus on siirretty Mipron omalle palvelimelle. Sovelluksen varmuuskopioinnista lisää luvussa 5.6.

5.1.3 Testauksen tulosten tarkastelu ja toimenpiteet

Testauksessa havaitut ongelmat ja kehitystarpeet kirjattiin erilliseen taulukkoon dokumentinhallintajärjestelmään, josta muutkin pystyivät niitä tarkkailemaan. Tammi-kuussa pidimme palaveria yhdessä Mipron tiimiesimiesten, projektipäälliköiden ja muun johdon kanssa liittyen ohjelman käyttöön. Palavereissa tuli ilmi paljon kehitysehdotuksia, joista raportoin ohjelmiston toimittajalle. Tiivis yhteistyö toimittajan kanssa muokkasi ohjelmistosta nopeasti Mipron näkökulmasta toimivan järjestelmän. Kuukauden testauksen ja räätälöinnin jälkeen sovellus saatiin sellaiseen kuntoon, että se oli valmis siirrettäväksi Miprolle.

5.2 Siirto Mipron palvelimelle

Sovitun mukaisesti ohjelmisto saatiin kuukaudessa räätälöityä vastaamaan Mipron tarpeita ja helmikuussa 2011 se pystyttiin siirtämään etäyhteydellä Mipron palvelimelle. Mipron vastuulle jäi luoda tarvittava virtuaalipalvelin Microsoft Server 2008 -käyttöjärjestelmällä ja avata palomuriin reikiä etäyhteyttä varten. Toimittajan vastuulle jäi tietokantojen asennus ja ohjelmiston toimintakuntoon saattaminen.

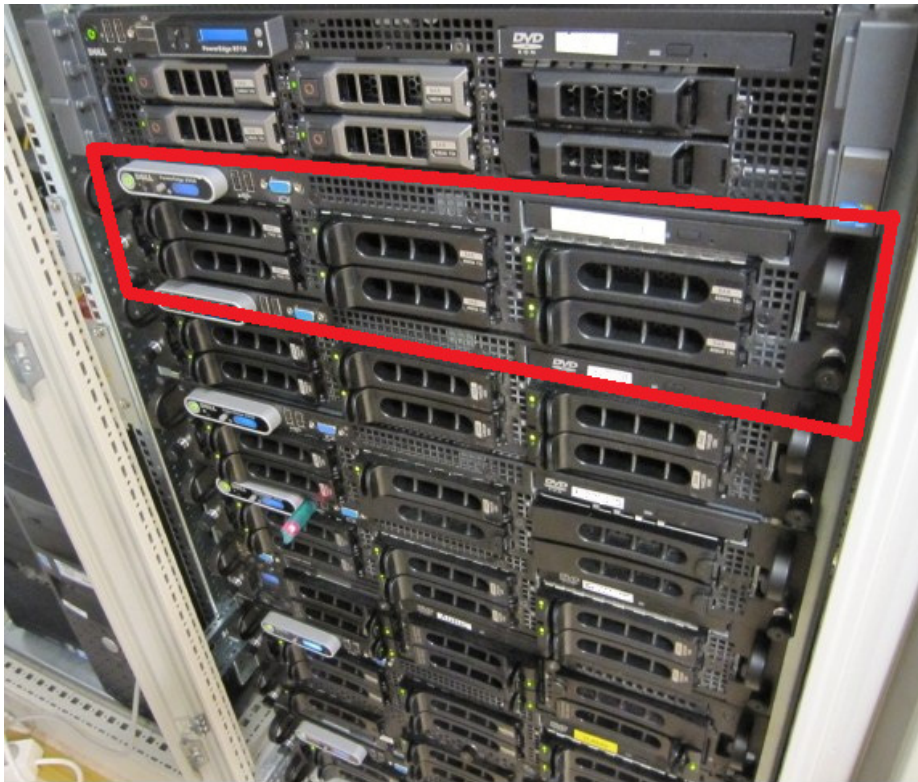
Miprolla on eri käyttötarkoituksiin erilaisia verkkoja talon sisällä. Kaikkien henkilökohtaiset työasemat ovat kytkettynä toimistoverkkoon. Toimistoverkkoon ei saa kytkeä omia henkilökohtaisia tai asiakkaiden laitteita. Toimistoverkon kautta ihmiset pääsevät tärkeimmille verkkolevyille, sisäiseen intranettiin ja julkiseen internettiin. Myös Project Router sijoitettiin toimistoverkkoon. MPLS-yhteyden kautta myös Oulun toimipisteen henkilöstöllä on pääsy toimistoverkkoon. Lisäksi Miprolta löytyy erilaisia testaus- ja valvomoverkkoja, joihin pääsy on erittäin rajattua. Niiden käyttö edellyttää perehdytyskoulutusta verkkojen käyttötarkoituksesta ja tietoturvaan liittyvistä seikoista. Testaus- ja valvomoverkoista ei ole pääsyä julkiseen internetiin.

Työn alussa pidettynä määrittelypäivänä oli jo sovittu, että Mipro hoitaa palvelimen siihen vaiheeseen, että Improlyty voi tehdä siirron palvelimelle ongelmitta. Palvelimen valmistelu suoritettiin yhdessä Mipron kokeneen järjestelmä vastaavan kanssa.

Project Router sijoitettiin Miprolla virtuaalipalvelimelle. Itse fyysinen isäntäpalvelin oli Mipron palvelinhuoneessa sijaitseva Dell PowerEdge 2950 III (kuva 7). Tällä palvelimella pyörii yli kymmenen virtuaalipalvelinta VMWare ESXi 3.5 ympäristössä. Isäntäpalvelimen hostname on miprosrv3. Isäntäpalvelin on ostettu Miprolle aikoinaan seuraavalla kokoonpanolla: /18./

- 2 x Quad-Core Xeon E5440 prosessorit, 2,8GHz 6MB välimuistilla
- 1333 MHz FSB-dataväylä
- 16GB keskusmuistia (4 x 4GB, 667MHz)
- 2 x 73GB kovalevyt SAS-väylällä (RAID1 konfiguroitu)
- 4 x 450GB kovalevyt SAS-väylällä (RAID5 konfiguroitu)

- PERC6/i SAS RAID ohjainkortti
- 2 x 750W virtalähde
- Broadcom NetXtreme II 5708 verkkokortti
- DVD-asema
- DRAC etähallintakortti (Dell Remote Access Controller)



KUVA 6. Isäntäpalvelin mipro srv3 Mipron palvelinhuoneessa.

RAID-tekniikoiden periaatteiden mukaisesti käytössä on siis kaksi kovalevyä; 73GB ja 1350GB (3 x 450GB) suuruiset kovalevyt.

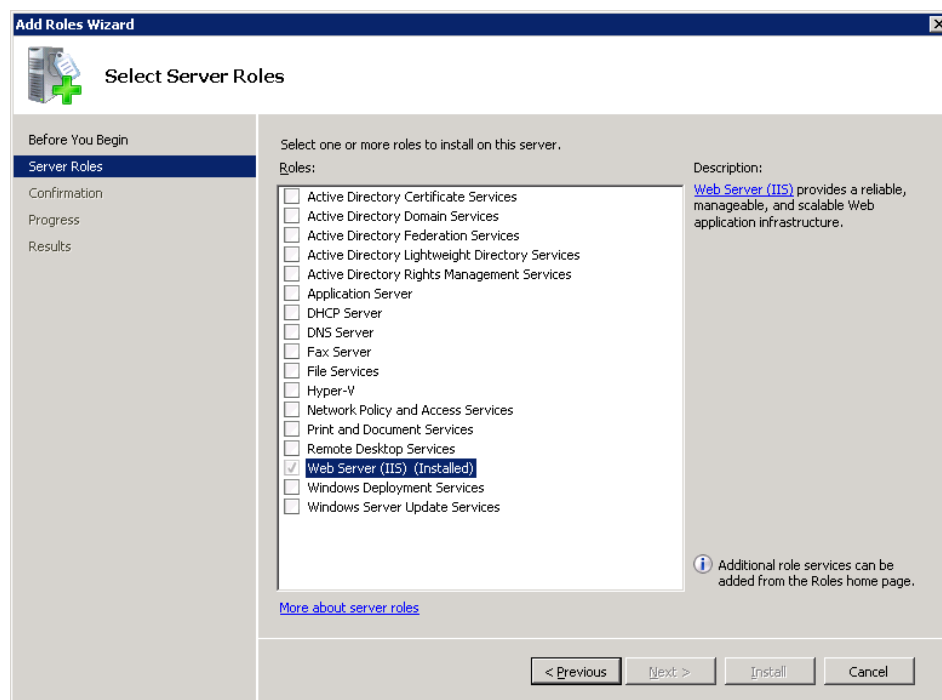
VMWare ESXi ympäristö on asennettu pienemmälle, 73GB kovalevylle. Virtuaalipalvelimet ovat suuremmalla kovalevyllä.

Isäntäpalvelimeen asennettuun VMWare ESXi ympäristöön otettiin yhteys VMware Infrastructure Clientillä. New Virtual Machine Wizardin avulla luotiin ympäristöön uusi virtuaalikone seuraavalla kokoonpanolla:

- Kaksi virtuaalista prosessoria
- 2048MB muistia
- DVD-asema
- Verkkokortti
- 60GB kovalevytilaa
- Käyttöjärjestelmä Windows Server 2008 R2 Standard 64-bit

Virtuaalikoneelle varattiin isäntäpalvelimen resursseista käyttöön prosessoritehoa 500MHz ja keskusmuistia 1024MB. Nämä resurssit ovat siis aina virtuaalikoneen käytössä isäntäkoneelta. Virtuaalikoneen luomisen jälkeen asennettiin käyttöjärjestelmäksi Windows Server 2008 normaaliin tapaan. 60GB kovalevy osioitiin ja alustettiin kahdeksi 30GB osioksi. Käyttöjärjestelmä asennettiin toiselle ja toinen jätettiin tallennustilaksi.

Kun Windows Server saatiin asennettua ja aktivoitua onnistuneesti, asennettiin VMware Tools, joka optimoi virtuaalikoneen ajurit ja parantaa suorituskykyä sekä käyttömukavuutta. Palvelimelle asennettiin myös tarvittava web-palvelimen (IIS) rooli (role). Roolin lisääminen onnistuu Add Roles Wizardin avulla. Roolin asennuksessa valittiin vielä asennettavat palvelut. Yhteensä web-palvelimen roolille asennettiin 23 palvelua (mm. ASP.NET).



KUVA 7. Windows Server 2008:lle asennettu yksi rooli (IIS).

Verkkokorttiin määriteltiin Mipron toimistoverkon IP-alueelle osuva vapaa IP-osoite jonka jälkeen kone nostettiin toimialueelle ja koneelle määritettiin nimeksi (hostname) srv3project.

Virtuaalikoneeseen asennettiin myös ESET-virustorjuntaohjelma, sama kuin Mipron muillakin työasemilla ja palvelimilla. Windowsin järjestelmäasetuksista sallittiin etätyöpöytäohjaimet toimialueen tietyille käyttäjille, jotta koneen etäkäyttö on mahdollista. Tietokantaa varten asennettiin Microsoftin SQL Server 2008 johon haettiin viimeisimmät päivitykset ja luotiin yksi SQL-instanssi. Tietokantojen luominen jätettiin ohjelmiston toimittajalle, jotta kaikki määrytykset menisivät varmasti oikein. Tietokantojen hallintaa varten asennettiin Microsoftin SQL Server 2008 Express Management Studio.

Asennuksen jälkeen järjestelmävastaava teki palomuriin NAT-säännön joka antaa palvelimelle oman julkisen IP-osoitteen. Ohjelmiston toimittajalta pyydettiin etäyhteydessä käytettävä IP-osoite ja tarvittavat portit, jotta palomuriin pystyttiin tekemään etäyhteyden mahdollistavat säännöt.

5.3 Ohjelmiston siirto Miprolle ja sisäinen käyttöönotto

Toimittaja suoritti ohjelmiston siirron Mipron palvelimelle etäyhteyden avulla. Käyttöönotto etäyhteyden kautta onnistui ilman suurempia ongelmia. Sopivaksi räätälöity demojärjestelmä kopioitiin kahden erillisen ympäristön pohjaksi. Kaikki ylläpidolliset listat (käyttäjät, projektilajit, tehtävätyypit, tiimit jne.) säilyivät, mutta kaikki projektit hävisivät, niin kuin aiemmin oli sovittu.

Ohjelmisto käsittää siis kaksi toistensa kaltaista, mutta erillistä ympäristöä.

Tuotanto (<http://srv3project/prod>)

Testaus (<http://srv3project/test/>)

Tuotantoympäristö on tarkoitettu ohjelman varsinaiseen käyttöön ja siellä ei enää testata mitään. Testausympäristö on luotu tätä varten, sinne voi luoda keksittyjä käyttäjiä ja projekteja. Testausympäristön avulla ylläpito voi myös testata järjestelmän palauttamista tietokantojen varmuuskopioista.

Tuotantoympäristön ylläpidolliset ja projektinhallintaan liittyvät asetukset piti saada ensimmäisenä määritettyä. Projektinhallinta-asetuksista tarkastettiin lähinnä, että projektitunnisteista ei löydy enää mitään demojärjestelmään viittaavia testitunnisteita. Myös englanninkielisten termien oikeellisuus varmistettiin projektilajien, projektitunnisteiden ja tehtävätyyppien osalta. Englanninkielisiä vastineita ei tarvinnut itse alkaa pohtimaan, vaan ne löytyivät Mipron virallisista dokumenteista.

Ylläpidollisten asetusten määrittäminen käytiin läpi yhdessä esimiesteni kanssa. Käyttäjryhmiä luotiin aluksi kuusi kappaletta. Kaikki Mipron työntekijät määritettiin kuulumaan ryhmään ”Henkilöstö” ja oikeudet määritettiin niin, että he pystyvät tarkastelemaan kaikkien ei-salaisten projektien projektinäkymiä. Muita ryhmiä olivat mm. ”projektipäälliköt” ja ”tiimiesimiehet”. Näille ryhmille määritettiin tarvittavat oikeustasot. Esimerkiksi projektipäällikkö-ryhmän jäsenet voivat luoda uusia projekteja projektisalkkuun, mutta eivät näe eri tiimien kuormitusta kuten tiimiesimiehet. Käyttäjryhmälle ”Järjestelmän ylläpitäjät” määritettiin luonnollisesti kaikki mahdolliset oikeudet niin projektinhallintaan kuin ylläpitoonkin liittyen.

Tiimit pystyin määrittämään pitkälle Mipron ajan tasalla olevan organisaatiokaavion perusteella. Kaavion mukaisten virallisten tiimien lisäksi luotiin ”ei-virallisia” tiimejä, jotta projektin salaiseksi määrittämisen yhteydessä projektiryhmään pystytään nopeasti valitsemaan esim. kaikki ns. ohjausryhmien jäsenet.

Myös malliprojektit projektien luonnin helpottamiseksi piti luoda tuotantoympäristöön. Tarvittavat malliprojektit määritin esimieheni ohjeiden mukaan. Mallivaraston malleihin luotiin ennakkoon ko. tarkoitukseen sopivat tehtävätyypit sekä milestonet. Mallipohjien tarkoitus on nopeuttaa projektipäällikön projektinluontia.

Ohjeiden viimeistely Mipron sisäistä käyttöä varten oli myös tärkeää ennen varsinaista sisäistä käyttöönottoa. Käyttöohjeesta tuli noin 20 sivuinen ja se käsittää erittäin seikkaperäisen ohjeistuksen siitä, kuinka ohjelmistoa Miprolla käytetään. Ohje tallennet-

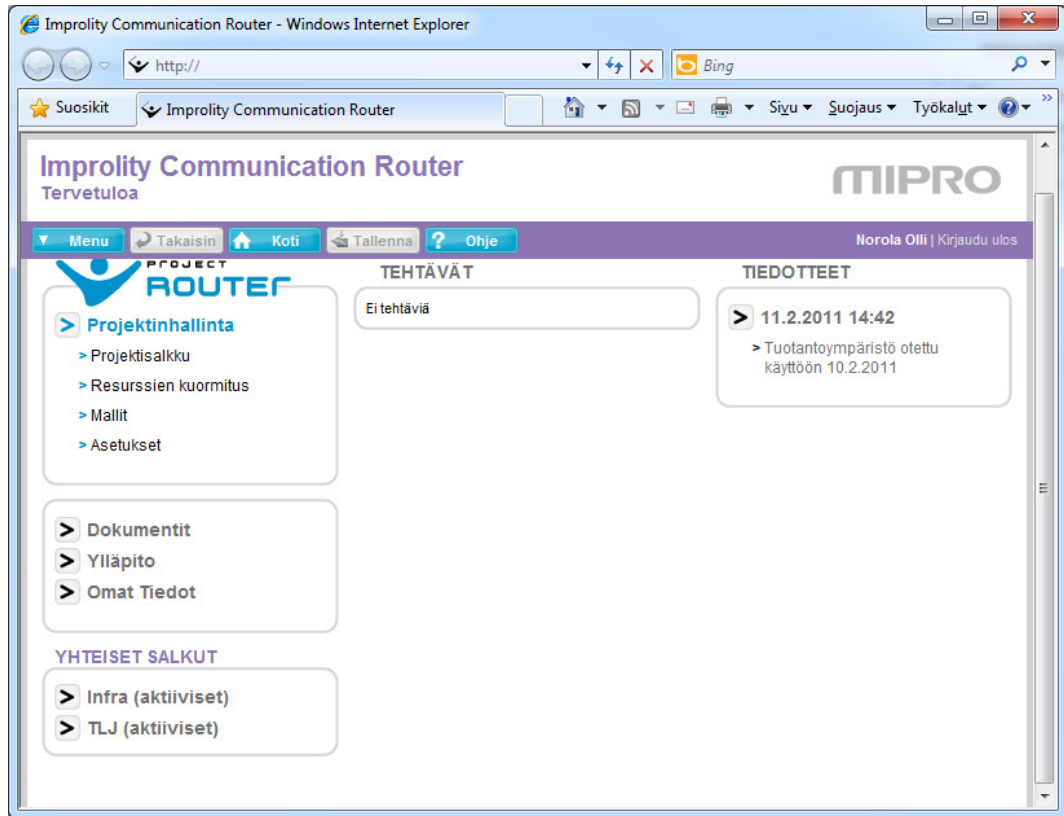
tiin dokumentinhallintaohjelmistoon, jotta se on kätevästi kaikkien saatavilla ja sen päivittäminen onnistuu helposti. Kyseinen ohje lisättiin ohjelmistoon myös ohjepainikkeen ”yleinen” valinnan taakse, eli se on avattavissa jokaiselta sivulta missä ko. painike on näkyvässä. Lisäksi tietyille sivuille lisättiin ohjepainikkeen taakse myös sivukohtaisia ohjeita. Maaliskuussa 2011 järjestin hallinnolle, projektipäälliköille ja tiimiesimiehille koulutuksia ohjelman käytöstä. Projektipäälliköille ja tiimiesimiehille järjestin kaksi koulutustilaisuutta, jotta jokainen ehtisi jompaankumpaan koulutukseen. Hallinnolle järjestin oman pienen koulutuksen, jossa käytiin läpi projektitunnisteiden, henkilöiden, ryhmien ja tiimien lisäykset ja poistot.

5.4 Ohjelmiston käyttö

Tässä luvussa käydään läpi Project Routerin tärkeimmät ominaisuudet ja käyttö.

5.4.1 Kirjautuminen ja etusivu

Käyttäjä löytää kirjautumissivulle intranetin etusivun linkin kautta. Käyttäjä autentikoidaan toimialueen mukaan automaattisesti Internet Exploreria käytettäessä. Muihin selaimiin pitää toimialueen tunnukset syöttää avautuvaan salasanaikkunaan. Vaihtoehtoisesti sovellukseen voi kirjautua myös paikallisella salasanalla, jonka voi määrittää ”Omat tiedot” sivulla.



KUVA 8. Etusivun näkymä.

Kuvan 8 mukaisella etusivulla näkyy keskellä projektien tehtävät, joista on itse vastuussa. Kuvassa tehtävien vähyys johtuu siitä, että en luonnollisesti ole mukana muussa kun itse luomassani testiprojektissa. Omien tehtävien sijoittaminen etusivulle on järkevää, sillä näin käyttäjä näkee omat tehtävänsä aina yhdellä vilkaisulla.

Oikeassa laidassa on tiedotteet, joita voi kirjoittaa vain ylläpitäjän oikeustason omaavat käyttäjät. Tiedotteiden avulla ylläpito voi tiedottaa käyttäjiä esim. huoltokatkoista ja tiedossa olevista ongelmista. Vasemmassa laidassa näkyy valikko, joka riippuu käyttäjän oikeustasosta. Tavallinen käyttäjä ei näe esim. ylläpitoa valikossaan. Toimintopalkki näkyy ohjelmistossa joka sivulla samanlaisena. Palkki sisältää kuvan 9 mukaiset painikkeet.



KUVA 9. Toimintopalkin painikkeet

Menu-painike avaa vetovalikon, jonka sisältä riippuu sivusta jossa ollaan. Vetovalikosta löytyy useimmilla sivuilla mahdollisuus tulostaa näkymä pdf-tiedostoon. Takai-

sin-painikkeella pääsee luonnollisesti takaisin edelliseen näkymään. Koti-painiketta klikattaessa palataan aina etusivulle. Tallenna-painike muuttuu aina tummaksi, kun esim. projektinäkymään tehdään tekstikenttään muutoksia. Painamalla sitä tiedot tallentuvat. Ohje-painiketta klikkaamalla tavalliselle käyttäjälle avautuu mahdollisuus lukea joko Mipron sisäiseen käyttöön laadittua laajaa ohjetta, tai mahdollisesti joillekin sivuille laadittuja sivukohtaisia ohjeita.

5.4.2 Projektisalkku

Salkkunäkymässä (kuva 10) projekteja voi suodattaa niiden tilan, toimialan tai projektityylin mukaan. Portfolio-näkymässä projektista näkyy kaikki sen tärkeimmät tiedot. Hinta ja työvarantoarvio voidaan määrittellä näkyväksi vain halutuille käyttäjäryhmille oikeustasojen avulla. RoadMap-näkymä on yksinkertaistettu näkymä, joka näyttää projekteista vain niiden sijoittumisen ajallisesti.

Työvaranto kuvaa sitä, kuinka paljon projektin budjetista on vielä käyttämättä.

Työvaranto = Hinta x (100% - valmiusprosentti)

Näkyvien projektien aakkosellinen lajittelu onnistuu tunnuksen, projektin nimen, tilan, projektipaällikön, myyjän, toimialan ja projektityylin mukaan.

Tilan	Edist.	Tunnus	Projektin nimi	Tila	Projektipaällikkö	Myyjä	Toimiala	Projektityyppi	Hinta / Budjetti	Työvarantoarvio	Portit												
											1	2	3	4	5	6	7	8	9	10	11	12	
<input type="checkbox"/>	●	vp2846	Salon Vesi, vesitalonliiketoiminta. Uusien osakaupunkien kaukokäyttö.	Aktiivinen			Infra	Infra Projektit															
<input type="checkbox"/>	●	vp2870	Jämsän Vesi liikelaitos. Automaatiojärjestelmien yhdistäminen ja päivittäminen	Aktiivinen			Infra	Infra Projektit															
<input type="checkbox"/>	●	vp2874	Salon Kaukolämpö Oy, Mittausasemien kaukovalvonta	Aktiivinen			Infra	Infra Projektit															
<input type="checkbox"/>	●	vp2875	Lahti Aqua Oy / Aqua Veikko Oy, Kariniemen Ilmastuksen saneeraus, SI-urakka	Aktiivinen			Infra	Infra Projektit															

KUVA 10. Projektisalkun portfolio-näkymä.

5.4.3 Asetukset

Project Routerista löytyy erikseen projektinhallintaan ja ylläpitoon liittyvät asetukset. Projektinhallinta-asetuksiin on pääsy hallinnolla ja ylläpitäjillä. Ylläpidollisiin asetuksiin on pääsy vain ylläpitäjän oikeustasolla.

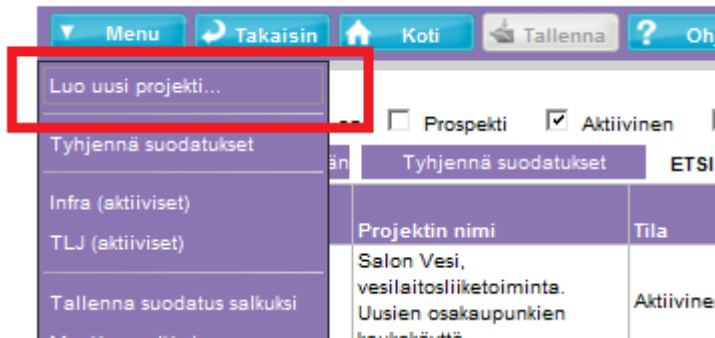
Projektinhallinta-asetuksista pääsee muokkaamaan Mipron projektinhallinnalle ominaisia ns. ”listoja”. Listoilla on kielisyys (englanti). Listoihin voidaan lisätä helposti rivejä. Toimiala-listasta löytyy Mipron päätoimialat TLJ (Turvallisuuden liittyvät järjestelmät) ja Infra (Vesihuollon järjestelmät). Projektilaji-listoilta löytyy molempien toimialojen projektilajit. Molemmilta toimialoilta löytyy esimerkiksi elinkaari-projektilaji. Projektitunniste-listalta löytyy kaikkien Project Routeriin syötettyjen projektien projektitunnisteet. Projektitunniste muodostuu koodista (esim. vs2880) ja nimestä (esim. Rääkkylän Vesilaitoksen valvomouudistus). Projektitunnisteen koodi on siis projektinumero, joka esiintyy kaikissa projektiin liittyvissä dokumenteissa. Tehtävä-listalta löytyy kaikkien projektilajien tehtävätyypit. Myös tehtävätyypit muodostuvat koodista (esim. 8A) ja nimestä (esim. Instrumentointisuunnittelu). Tehtävätyypin koodit luotiin yhteneviksi tuntienkirjausjärjestelmän kanssa, jotta integraatio järjestelmien välillä helpottuisi.

Järjestelmän ylläpitäjille tarkoitettua ylläpitoasetuksista pääsee muokkaamaan käyttäjiä, käyttäjäryhmiä, tiimejä ja tiedotteita. Käyttäjät-sivulta pääsee tarkastelemaan kaikkia järjestelmään luotuja käyttäjätunnuksia. Uusia käyttäjiä pystyy luomaan helposti, mutta vanhoja ei pysty poistamaan. Mikäli työntekijä lopettaa työnsä Miprolla, hänen tunnuksensa merkitään ei-käytössä olevaksi. Yksittäisen käyttäjän näkymässä voidaan tarkastella käyttäjän tiimejä ja käyttäjäryhmiä. Tiimit-sivulla voi tarkastella olemassa olevia tiimejä, lisätä uusia, poistaa vanhoja ja muokata tiimien jäseniä. Tiedotteet-sivulla ylläpitäjä voi kirjoittaa tiedotteita jotka näkyvät kaikkien käyttäjien etusivuilla (kuva 8).

5.4.4 Projektin luominen

Uuden projektin luominen aloitetaan projektitunnisteen luomisella. Uuden projektitunnisteen pääsee lisäämään projektinhallinta asetuksista. Kun projektitunniste on

luotuu, voidaan siirtyä etusivun kautta projektisalkkuun. Salkussa valitaan vasemman yläkulman vetovalikosta kuvan 11 mukainen ”Luo uusi projekti”.



KUVA 11. Uuden projektin luominen

Seuraavaksi voi valita alkaako tekemään projektia tyhjästä, mallivaraston pohjasta vai jos luodusta projektista. ”Tyhjä projekti” avaa projektinäkymän jossa kaikki kentät ovat täyttämättä. ”Kopioi mallivarastosta” avaa esitäytetyn pohjan ja ”Kopioi jo luodusta projektista” kopioi kaikki tiedot halutusta olemassa olevasta projektista.

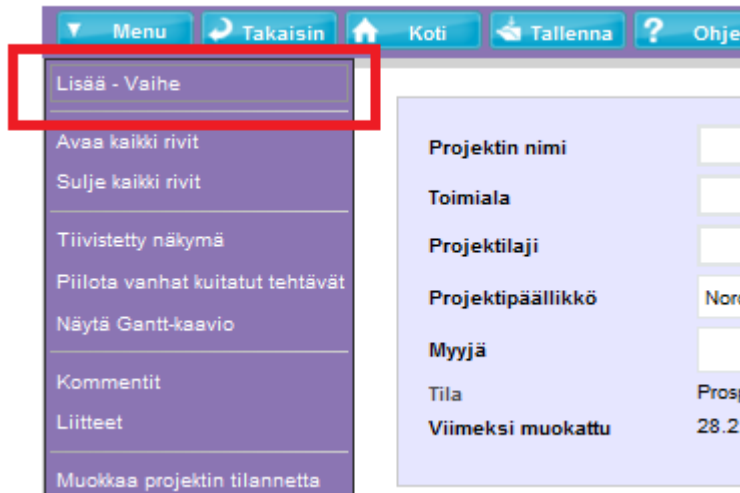
Valittaessa ”Tyhjä projekti” avautuva projektinäkymä on nimensä mukaisesti tyhjä. Projektin nimi-vetovalikosta löytää aiemmin tekemänsä projektitunnisteen. Projekti-päällikkö täyttää kaikki kentät siltä osin kuin hän tietää.

KUVA 12. Projektin perustiedot.

Tila-painiketta painamalla avautuu uusi ikkuna, josta projektin tilaksi voi valita prospekti, aktiivinen, vastaanotettu tai päättynyt. Näillä kaikilla voi suodattaa projektia projektisalkkunäkymässä. Toinen tilan määritelmä on sanallinen kuvaus; huono, kohdalainen tai hyvä. Näitä kuvaavat värit ovat punainen, keltainen ja vihreä. Projektin kokonaisedistymisen voi valita kymmenen prosenttiyksikön välein. Kokonaisedistymisen näkyy ympyrägraafina projektisalkussa ja projektinäkymässä. Projektipäällikkö

antaa projektin tilasta oman subjektiivisen näkemyksenä, jonka hän on muodostanut tehtävien edistymisen perusteella.

Jotta projektille voidaan asettaa tehtäviä pitää sille luoda ainakin yksi vaihe. Ensimmäisen vaiheen luominen onnistuu kuvan 13 mukaisesti valitsemalla vasemman yläkulman ”Valitse toiminto”-vetovalikosta ”Lisää vaihe”.



KUVA 13. Ensimmäisen päävaiheen lisääminen projektiin.

Päävaiheelle voi antaa haluamansa nimen, kuten esim. ”Vaihe 1”. Päävaiheelle voidaan asettaa edelleen uusia vaiheita tai tehtäviä Lisää alataso-painikkeen kautta.

Jos vaiheelle lisätään tehtävä, ilmestyy tehtäväriivi, johon määritellään tehtävän kuvaus, suunniteltu alku- ja loppuaika, vastuhenkilö ja suunniteltu kesto (Kuva 14). Tehtävän kuvaus-sarakkeeseen valitaan tehtävää kuvaava tehtävälaji. Tehtävälaji riippuu ylempänä määritetystä projektilajista. Ellei projektilajia ole määritelty, ei vetovalikossa näy tehtävätyyppejä. Vetovalikon alla on myös tekstikenttä, johon voi kirjoittaa itse tarkempaa kuvausta tehtävään liittyen.

Tila	Tehtävän kuvaus	Suunniteltu aika	Vastuhenkilö	Suun. / Tod. kesto (h)	Tulos	Kuittaus pvm.
	Tehtävät	1.3.2011 - 4.3.2011		15,00 h 0,00 h		
Toiminnot Lisää Sulje muokaus						
Tila	Tehtävän kuvaus	Suunniteltu alku- ja loppuaika	Vastuhenkilö	Suun. / Tod. kesto	Tulos	Kuittaus pvm.
	8A - Projektin hoito	01.03.2011 - 04.03.2011	Norola Olli	15,0 0,0		

KUVA 14. Projektille on luotu vaihe ”Tehtävät” ja sille yksi tehtävä.

Suunniteltu kesto lasketaan automaattisesti päivämäärien välisellä ajalla (7,5h/pv), mutta se voidaan muuttaa manuaalisesti. Vastuhenkilö kenttään projektipäällikkö määrittelee tehtävälle vastuuhenkilön projektiryhmästään. Vastuuhenkilökenttää napsauttamalla avautuu käyttäjalista, johon listautuu kaikki järjestelmän käyttäjät.

Tehtävän toiminnot-valikosta löytyy erilaisia toimintoja, jolla tehtäviä voi kopioida, leikata, poistaa tai lähettää huomautuksen vastuuhenkilölle. Toiminnot valikon lähetä sähköposti-toiminto lähettää tehtävälle määritellylle vastuuhenkilölle automaattisen sähköpostin, jossa häntä informoidaan tehtävästä. Tämä on hyödyllinen tapa muistuttaa tehtävien vastuuhenkilöitä siitä, että tehtävien etenemisestä pitää muistaa raportoida järjestelmään.

Tehtäväriivin alussa näkyvä ympyräkuvaaja kertoo värillään missä vaiheessa tehtävä on. Väri perustuu asetettujen alku- ja loppupäivämäärien suhteesta nykyhetkeen.

Valkoinen	tehtävän suunniteltu alku tulevaisuudessa
Keltainen	tehtävän suunniteltu loppu alle viikon päässä
Vihreä	tehtävän suunniteltu loppu yli viikon päässä
Punainen	tehtävän suunniteltu loppu menneisyydessä (tehtävä myöhässä)

5.4.5 Omat tehtävät

Kun projektipäällikkö on nimennyt projektihenkilön vastuuhenkilöksi johonkin tehtävään, ilmestyy kyseisen henkilön etusivulle ko. tehtävä. Klikkaamalla tehtävää pääsee muokkaamaan sitä. Tehtäväsi-välilehdellä vastuuhenkilö voi päivittää tehtävän etenemistä Tulos-tekstikenttään oman näkemyksenä mukaisesti, esim. ”Valmius 30%, FAT-testit viikolla 8”.

Kuittaamalla tehtävän, tehtävä häviää etusivun omista tehtävistä ja projektinäkylässä sen tila muuttuu kuitatuksi.

5.5 Tuntienkirjausjärjestelmän integraatio

Jo projektinhallintaohjelmiston toimittajaa valittaessa Miprolla otettiin huomioon, että integraatio tuntienkirjausjärjestelmään on mahdollinen. Mipron tuntienkirjausjärjestelmä on erään kotimaisen ohjelmistotalon SaaS-palvelu, jota voidaan käyttää internetin kautta mistä tahansa. Ideana oli siis, että tuntienkirjausjärjestelmään kirjatut toteutuneet tunnit pystyttäisiin tuomaan projektinhallintaohjelmistoon ja näin pystyttäisiin seuraamaan kuinka hyvin suunnitellut tunnit vastaavat toteutuneita.

Tuntienkirjausjärjestelmän integraatiopalaveri pidettiin Miprolla helmikuussa 2011. Paikalle saapui Project Routerin toimittajan edustaja, jolla oli kokemusta useista erilaisista intergraatioista. Palaverissa tehtiin määritelmä siitä, kuinka tunnit pystytään kohdistamaan oikealle tehtäviriville ohjelmistossa. Tuntien kohdennus päätettiin tehdä seuraavanlaisilla määritteillä:

- Projektinumero (esim. vp2331)
- Käyttäjätunnus (esim. mattim)
- Tehtävätyyppi (esim. 8A HW-testaus)

Ajatuksena oli, että tuntienkirjausjärjestelmän toimittajalta saataisiin tunneista jonkinlainen siirtotiedosto, joka sisältäisi tarvittavan informaation, ja joka sitten manuaalisesti Mipron hallinnon toimesta ajettaisiin Project Routeriin. Hahmotelma tarvittavasta siirtotiedoston rivin informaatiosta oli tämänkaltainen:

mattim;vp2869;8A;10

mattim	käyttäjätunnus
vp2869	projektinumero
8A	tehtävätyypin koodi
10	kumulatiivinen tunti-arvo

Päivämäärää ei otettu mukaan tuntirivin kohdistamiseen, koska Project Routeriin syötetty suunniteltu alku- ja loppupäivämäärä eivät välttämättä ole todellisia tai niitä ei ole välttämättä syötetty tehtävälle ollenkaan. Päivämäärän puuttuminen kohdistumisesta tarkoittaa siis sitä, että projektissa ei yhdellä henkilöllä voi olla yhden tyyppistä

tehtävätyyppejä kuin yksi kappale. Toteutuneet tunnit ovat siis kumulatiivinen summa koko projektin ajalta.

Myös mahdolliset virhetilanteet tuntien tuomisessa otettiin huomioon. Jos tuotu tunti-rivi (projektinnumero + käyttäjätunnus + tehtävätyppi) täsmää useammalle eri tehtävälle, käy niin että tunti-arvo kirjataan ensimmäiselle ylhäältä alas vastaantulevalle tehtäviriville. Jos tuntienkirjausjärjestelmästä taas tuodaan tunteja tehtävätyypille, jota ei projektista löydy, käy niin että Project Router lisää ko. tehtävän projektin ensimmäisen vaiheen alimmaksi tehtäväksi ja kirjaa tehtävän kuvaukseen ”Automaattisesti lisätty tehtävä”. Mikäli projektiin ei ole luotu ainuttakaan vaihetta, järjestelmä luo ensin vaiheen nimeltä ”Tehtävät” ja lisää tehtävän sen alle. Tieto automaattisen tehtävän lisäyksestä tulostuu raporttiin.

Jos tunteja yritetään tuoda käyttäjälle tai projektitunnukselle jota ei Project Routerista löydy, tulostuu raporttiin virheilmoitus josta asia ilmenee. Myös tuntien tuominen kuitatulle tehtävälle tai jo päättyneeseen projektiin ilmenee raportista. Lisäksi kuitatulle tehtävälle tunteja tuodessa ilmestyy tehtävirivin tuloskenttään teksti ”Tunteja lisättiin kuitattuun tehtävään, vanha arvo oli X”.

Tuntienkirjausjärjestelmän integraation käytännön toiminnan kuvausta ei ole kuvattu tässä työssä, sillä integraation toteutuminen venyi pitkälle toukokuulle 2011.

5.6 Tietokantojen varmistukset ja palautus

Kuten jo luvussa 4.2 mainittiin, tietokannan ominaisuuksiin kuuluu, että se voidaan varmistaa luotettavasti. Kumpikin ohjelmiston ympäristö (tuotanto ja testi) rakentuu kolmesta tietokannasta. Sovellus käsittää siis yhden tietokantainstanssin ja kuusi tietokantaa.

- pr_test_project_extras
- pr_test_project
- pr_test_main
- pr_project_extras
- pr_project
- pr_main

Tietokannan nimi rakentuu vapaasti valittavasti alkuliitteestä ja kiinteästä loppuosasta. Testiympäristön tietokannan nimen alkuliite on ”pr_test” ja tuotannon ”pr”.

x + ”_main” sisältää käyttäjät, käyttöoikeusryhmät, tiimit ym. sekä kaiken koko järjestelmään ja ohjelmistoalustaan liittyvän datan.

x + ”_project” sisältää projektinhallinnan yleisen datarakenteen, projektin metatiedot, vaiheet, tehtävät, projektinhallinta-asetusten listat yms.

x + ”_project_extras” on erikoistapauksiin varattu tietokanta, johon voi tehdä tuoterakenteesta poikkeavia taulurakenteita. Mipron tapauksessa tämä on täysin tyhjä, mutta löytyy koska se kuuluu tuotteeseen.

Koska ohjelmisto päätettiin sijoittaa Mipron omalle palvelimelle, jäi myös varmuuskopioiminen Mipron harteille. Varmistaminen päätettiin suorittaa ottamalla tietokannoista ajastetusti varmuuskopiot kolmesti päivässä toiselle palvelimelle, joka taas edelleen varmuuskopioidaan kahdesti päivässä nauhavarmistimelle. Tähän tarkoitukseen valittiin avoimeen lähdekoodiin perustuva ExpressMaint (v. 2.0.0.1), joka on yhteensopiva Microsoftin SQL Server 2008 ohjelmiston kanssa. Ohjelma on ainoastaan 56 kilotavun kokoinen exe-tiedosto, jota ajetaan erilaisten skriptien avulla.

Ohjelma kopioitiin paikallisesti samalle virtuaalikoneelle, kuin itse ohjelmisto ja sen tietokannat. Jokaiselle tietokannalle tehtiin skriptitiedostot ohjelmiston käyttöohjeen avulla. Skriptin luominen onnistuu muistion avulla. Skripti pitää muistaa tallentaa .cmd-tiedostopäätteellä, jotta se toimisi oikein suoritettaessa.

Alla yhden skriptitiedoston (pr_test_main_backup.cmd) sisältö:

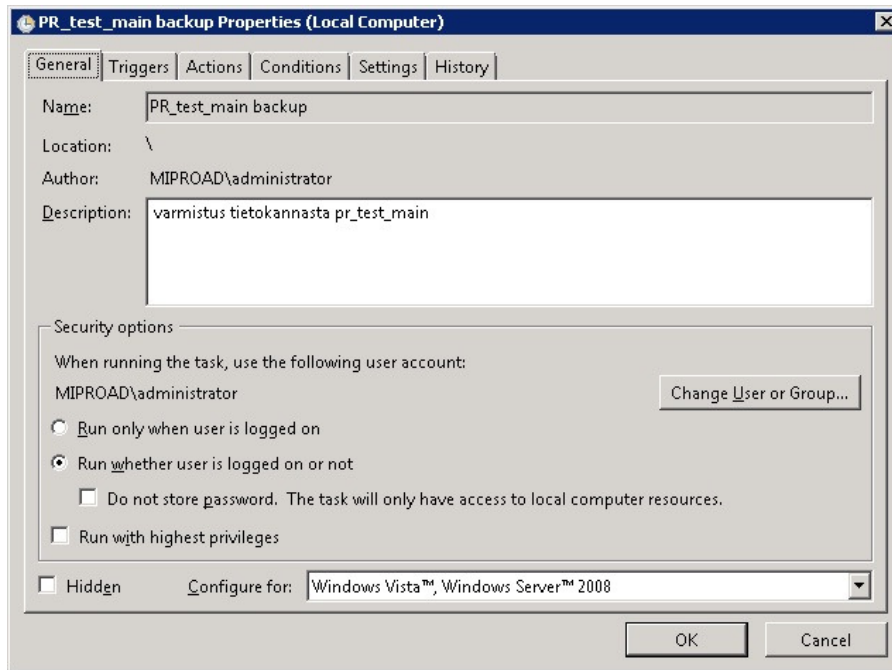
```
E:\SQL_Backup_scriptit\expressmaint.exe -S (local)\sqlexpress -D pr_test_main -T
db -R \\miprosrv6\srv3project_SQL_backups -RU WEEKS -RV 1 -B
\\miprosrv6\srv3project_SQL_backups -BU WEEKS -BV 1 -BF
$(DB)_$(DATE)_$(TIME)
```

Skriptin parametrien selitykset: /19./

- S** Kyseessä oleva SQL-server
- D** Tietokannan nimi
- T db** Varmuuskopio kokotietokannasta
- R** Polku jonne raportti varmuuskopioinnista kirjoitetaan
- RU** Aikayksikkö jonka verran raportti säilytetään
- RV** -RU määritteen aikayksiköiden määrä
- B** Polku jonne varmuuskopiot säilötään
- BU** Aikayksikkö jonka verran varmuuskopio säilytetään
- BV** -BU määritteen aikayksiköiden määrä
- BF** Varmuuskopion tiedostonimen muoto

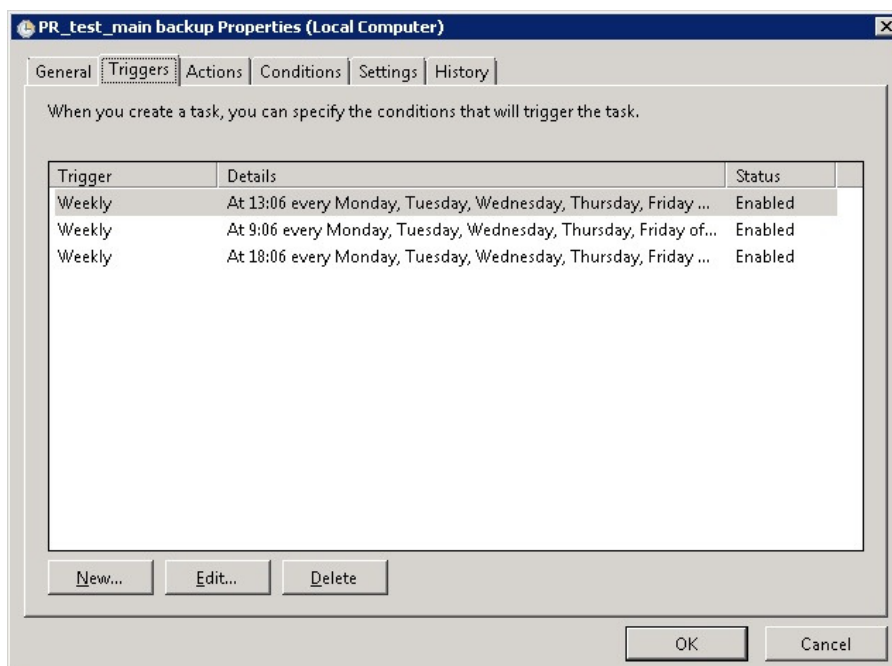
Komentokehotteeseen on siis määritelty, että se ajettaessa otetaan varmuuskopio (local)\sqlexpress-tietokantainstanssin koko pr_test_main-tietokannasta. Sekä varmuuskopiot, että raportit säilötään toisen palvelimen (miprosrv6) hakemistoon viikon ajaksi, jonka jälkeen ne tuhotaan. Selkeyden vuoksi varmuuskopion tiedostonimi on määritetty muodostumaan tietokannan nimestä, päivämäärästä sekä kellonajasta. Palvelin johon varmuuskopiot tallennetaan, varmuuskopioidaan ajastetusti edelleen nauhavarmistimelle kaksi kertaa päivässä Mipron muiden palvelinten tavoin.

Varmuuskopioinnin automaattisuus toimii Windows Serverin automaattisten tehtävien ajastuksella (Task Scheduler). Task Scheduleriin on siis tehty varmuuskopiointia varten kuusi tehtävää, koska kopioitavia tietokantojakin on kuusi. Seuraavassa aiemman pr_test_main-tietokannan ajastuksen asetukset Task Schedulerissa:



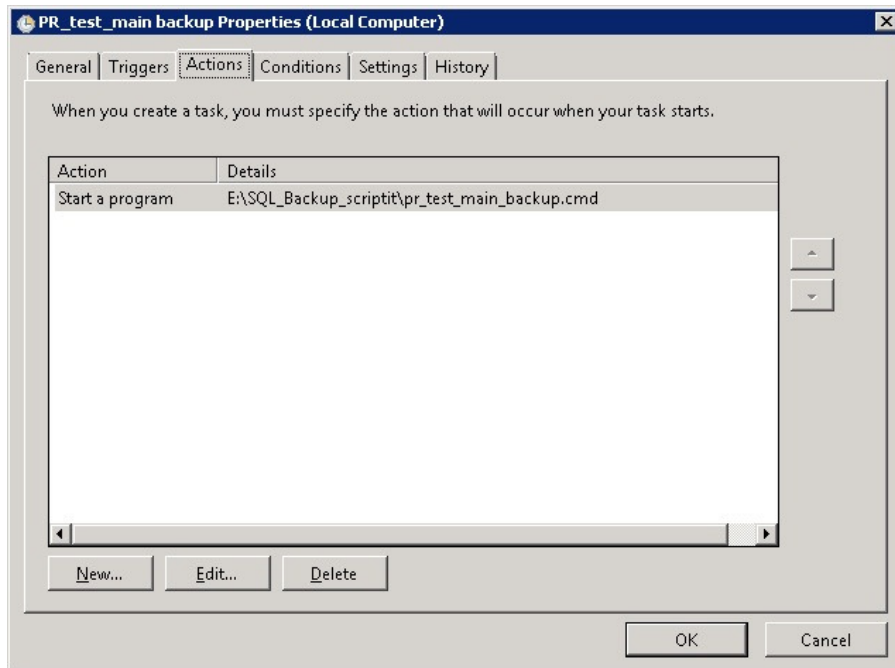
KUVA 15. Ajastetun tehtävän General-asetusvälilehti

General-välilehdelle pitää määritellä automaattisen tehtävän nimi ja kuvaus. Myös kohta ”Run whether user is logged on or not” pitää olla valittuna, jotta varmuuskopiointi tapahtuu riippumatta siitä onko virtuaalikoneeseen kirjaututtu sisään vai ei.



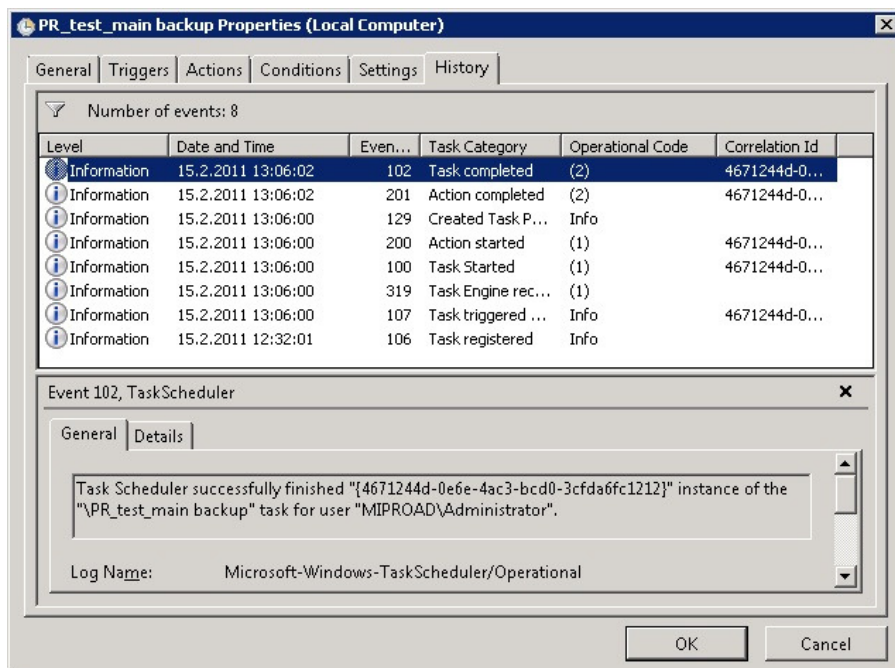
KUVA 16. Ajastetun tehtävän Triggers-asetusvälilehti.

Triggers-välilehdelle on määritelty, että tehtävä ajetaan automaattisesti kolme kertaa päivässä jokaisena arkipäivänä.



KUVA 17. Ajastetun tehtävän Actions-asetusvälilehti.

Actions-välilehdelle on lisätty suoritettavaksi toiminnoksi aiemmin esitelty skriptitiedosto pr_test_main_backup.cmd.

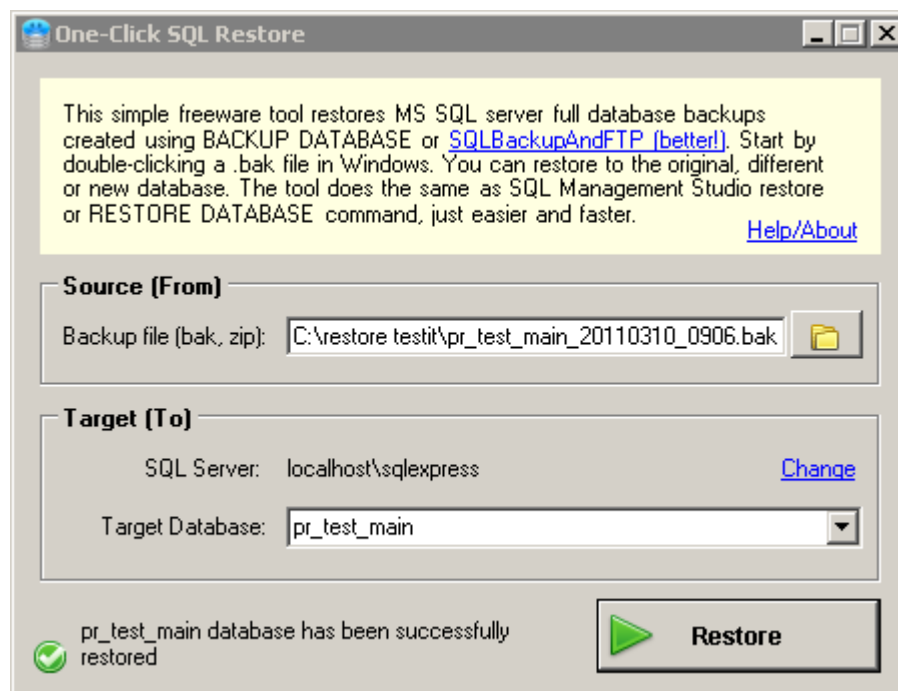


KUVA 18. Ajastetun tehtävän History-asetusvälilehti.

History-välilehdeltä voidaan tarkastella ajastetun tehtävän historiatietoja. Kuvasta voidaan havaita että tietokannan varmuuskopiointiin kuluu aikaa noin kaksi sekuntia. Tästä huolimatta tietokantojen automaattisten varmuuskopiointi-tehtävien välille on asetettu kahden minuutin viiveet, koska ohjelmiston käytön myötä tietokantojen koot kasvavat ja sitä myötä myös varmuuskopiointiajat suurenevät.

Järjestelmän palauttaminen varmuuskopiosta on aiheellista, jos on vaikkapa uhrannut aikaa projektin luomiseen ja jostain syystä onnistunut poistamaan projektin projektisalkusta. Tietokantojen palautus onnistuu varmuuskopiosta helposti One-Click SQL Restore ohjelman avulla. Ohjelma on ilmainen ja tarkoitettu MS SQL Serverin tietokantojen palauttamiseen varmuuskopioista.

Project Routerin tapauksessa varmuuskopio (bak-tiedosto) pitää käydä ensin hakemassa palvelimelta jonne ne on ExpressMaint ohjelman skriptitiedostossa määritelty tallennettaviksi. Virheen luonteesta riippuen valitaan joko ”_main” tai ”_project” tietokannan varmuuskopio sen ympäristön tietokannasta, joka halutaan palauttaa. Varmuuskopio kopioidaan Project Routerin palvelimelle haluttuun paikkaan. Tämän jälkeen voidaan avata Project Router palvelimella asennettu One-Click SQL Restore.



KUVA 19. Tietokannan palautus varmuuskopiosta.

Ensimmäiselle riville (Source) haetaan tietokannan varmuuskopio. Toiselle riville (Target) valitaan tietokanta, johon varmuuskopio halutaan ajaa. Restore-painike suorittaa toiminnon. Ohjelma kysyy vielä, että käynnistetäänkö tietokantapalvelu uudelleen, jotta palautus onnistuu.

Varmuuskopiointi voitiin testata helposti testiympäristön avulla. Testiympäristön testikäyttäjän käyttäjätietoihin tehtiin muutos, jonka jälkeen suoritettiin ”_main” – tietokannan palautus viimeisimmästä varmuuskopiosta. Palautuksen jälkeen voitiin havaita, että testikäyttäjän tiedot olivat muutosta edeltäneessä tilassa.

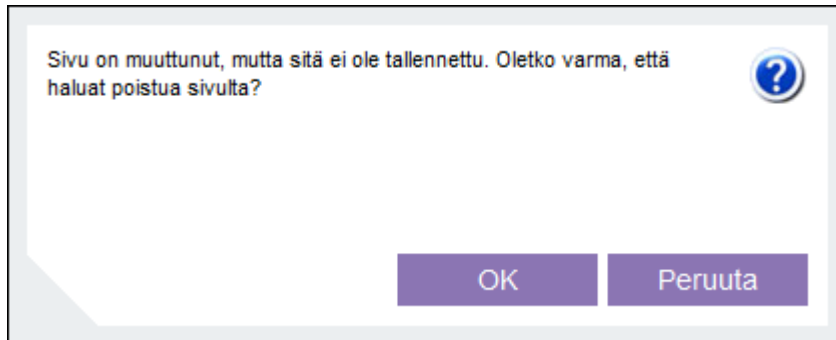
5.7 Käytettävyyden tarkastelu

Luvussa 4.5 esiteltiin käytettävyyden kaksi tunnettua määritelmää; Jacob Nielsenin määritelmä käytettävyydestä sekä standardoimisliiton määritelmä (ISO 9241-11:n käytettävyydestä mitattavana ominaisuutena.

Nielsenin määritelmään Project Routerin käytettävyyttä on hyvä verrata, koska ohjelmisto oli lähtökohtaisesti täysin uusi kaikille käyttäjille. Toistaalta Project Routeria voidaan katsoa myös ISO 9241-11:n määritelmän kautta, sillä kyllähän Miprolla projektin- ja resurssienhallintaa hoidettiin ennen Project Routeriakin menestyksekkästi.

Tarkastellaanpa ohjelmistoa ensin Nielsenin käytettävyyden määritelmän kautta. Opittavuus; kuinka helppoa sovelluksen käyttö on ensimmäisellä kerralla? Mielestäni sovelluksen käyttö oli alusta alkaen helppoa, sovelluksen selkeän ja yksinkertaisen web-käyttöliittymän käyttö tuntui heti hyvältä ja sulavalta. Tehokkuutta tarkasteltaessa nousee esiin sivuston vaakasuuntainen valikkopalkki joka löytyy joka sivulta. Sen sisältö on sama joka sivulla ja se sisältää oleelliset toiminnot, kuten linkin etusivulle ja edelliseen näkymään. Valikkopalkki siis lisää käytön tehokkuutta ja muistettavuutta. Virheettömyyttä tarkasteltaessa nousee esiin mm. tallenna-painikkeen toiminta. Kun jollekin sivulle tehdään tekstikenttiin muutoksia, vaihtaa tallenna-painike automaattisesti väriä vaaleasta tummaksi. Näin järjestelmä muistuttaa visuaalisesti

käyttäjää siitä, että kannattaisi tallentaa. Jos muutoksista huolimatta ei muista painaa tallenna-painiketta, sovellus muistuttaa käyttäjää kuvan 20 mukaisella ilmoituksella.



KUVA 20. Sovelluksen muistutus tallentamattomista tiedoista.

Myös mahdollisuus ohjeiden nopeaan tarkasteluun lisää virheettömyyttä. Ohje-painike esiintyy joka sivulla ja siitä saadaan auki joko laatimani yleinen ohje tai mahdollisesti sivulle kirjoitettu sivukohtainen pikaohje. Yleisen ohjeen tiedostoon on liitetty pikakuvake dokumentinhallintajärjestelmän uusimpaan versioon ohjeesta. Näin saatavilla oleva ohje on aina paikkansa pitävä. Viimeinen Nielsenin käytettävyyden tekijä, miellyttävyys, toteutuu mielestäni hyvin. Sivuston hillitty ja yksinkertainen ulkoasu miellyttää silmää. Sovelluksessa esiintyvä violetti väri on käyttäjille tuttu Mipron intranetistä ja muusta yrityksen materiaalista, näinkin pienellä muutoksella on saatu luotua käyttäjille ”turvallisuuden tunnetta”. Huolimatta web-käyttöliittymästä ohjelmiston käyttö on sujuvaa ja kaikki toiminnot käyvät nopeasti ilman suurempia viiveitä.

Standardoidun käytettävyyden (ISO 9241-11) näkökulmasta voidaan tarkastella tehokkuuden ja taloudellisuuden nykytilaa verrattuna tilaan ennen Project Routeria tai tilanteeseen, jossa olisi valittu jokin toinen projektihallintaohjelmisto. Verrattuna tilaan ennen Project Routeria tilanne parani selkeästi, kun käyttöön saatiin keskitetty työkalu projektihallintaan. Project Routerin myötä Outlook-tehtävistä pystyttiin luopumaan kokonaan. Taloudellisia kustannuksia uudesta ohjelmistosta toki syntyi, mutta ajan myötä ohjelmiston hyödyt luultavasti kattavat syntyneet kustannukset. Tehokkuuden ja taloudellisuuden vertailu tilanteeseen, jossa käyttöön otettu sovellus olisi ollut jokin muu kuin Project Router, on hyvin vaikeaa sillä päätös käytettävästä sovelluksesta tehtiin jo ennen varsinaisen työni aloitusta.

6 PÄÄTÄNTÖ

Tämän opinnäytetyön tavoitteena oli saada uusi projektihallintatyökalu käyttöön Mipro Oy:ssä. Työ alkoi vuoden 2010 lopulla ja kesti alkukevääseen 2011 asti. Työ onnistui tavoitteessaan, uusi ohjelmisto pystyttiin ottamaan käyttöön maaliskuun aikana.

Työ alkoi demojärjestelmän testauksesta ja kehittämisestä. Tiivis yhteistyö ohjelmiston toimittajan kanssa muokkasi ohjelmistosta Mipron tarpeita vastaavan. Demojärjestelmän aikana pidettiin paljon myös sisäisiä palavereita, joissa tulevat käyttäjät saivat tuoda esiin omia kehitysideoitaan. Toimittaja toteutti haluttuja muutoksia sopimuksessa mainitun asiakaskohtaistamistyön puitteissa.

Helmikuussa 2011 suoritettiin ohjelmiston siirto Mipron palvelimelle. Mipron vastuulle jäi luoda sopiva palvelin ohjelmiston käyttöön, toimittaja suoritti asennuksen etäyhteyden kautta. Ennen varsinaista käyttöönottoa oli tärkeää saada käyttäjärühmät ja niiden oikeudet lopulliseen muotoonsa. Tuotantoympäristöön vietiin kaikki avoimet toimitusprojektit. Käyttökoulutuksia järjestettiin useaan eri otteeseen Miprolla sisäisesti. Oulun toimiston väelle järjestin koulutustilaisuuden hyödyntäen videopuheluyhteyttä. Tuotantoympäristöstä erillinen testiympäristö helpotti uusia käyttäjiä kun he tutustuivat uuteen ohjelmistoon.

Työ antoi hyvän kuvan siitä, kuinka keskisuuressa yrityksessä hoidetaan uuden tietoteknisen järjestelmän käyttöönotto. Samalla tuli selväksi ASP.NET-sovellusten toimintaperiaatteet, web-käytettävyyden merkitys, testauksen tärkey sekä projektityöskentelyn tehokkuus käytännön työelämässä. Myös koulutustilaisuuksien pitäminen suurelle määrälle ihmisiä oli haastavaa, piti keksiä keinoja kuinka saada ihmiset oppimaan uuden järjestelmän käyttö, sillä tehdystä käyttöohjeesta huolimatta on mahdotonta saada kaikki sitä lukemaan.

Tuntienkirjausjärjestelmän integraatio tulee tapahtumaan kesään 2011 mennessä, minkä jälkeen ohjelmistosta tullaan saamaan kaikki hyöty irti. Koska työt Miprolla jatkuvat myös opinnäytetyön jälkeen, tulen ainakin alkuvaiheessa toimimaan ohjelmiston käyttötukena sekä yhteyshenkilönä ohjelmiston toimittajan suuntaan mahdollisissa ongelmatilanteissa. Tärkeimpänä dokumenttina työstä jäi laatimani n. 20-sivuinen käyttöohje Mipron sisäiseen käyttöön.

LÄHTEET

- 1 Ruuska, Kai. Pidä projekti hallinnassa. Helsinki: Talentum Media Oy. 2007.
- 2 Berkun, Scot. Projektinhallinnan taito. Jyväskylä: Gummerrus Kirjapaino Oy. 2006.
- 3 Rantala, Ari. Web-ohjelmointi. Jyväskylä. Docendo Finland Oy. 2005.
- 4 Netcraft. November 2010 Web Server Survey. <http://news.netcraft.com/archives/2010/11/05/november-2010-web-server-survey.html>. WWW-dokumentti. Luettu 10.2.2011.
- 5 Wikipedia. Web-application. http://en.wikipedia.org/wiki/Web_application. WWW-dokumentti. Päivitetty 18.4.2011. Luettu 20.4.2011.
- 6 Krishen Kota. Testing Your Web Application. http://www.adminitrack.com/articles/testing_web_apps.pdf. WWW-dokumentti. Luettu 4.3.2011.
- 7 W3C. W3C Standards. <http://www.w3.org/standards/>. WWW-dokumentti. Luettu 12.3.2011.
- 8 W3C. W3C Mission. <http://www.w3.org/Consortium/mission>. WWW-dokumentti. Luettu 12.3.2011.
- 9 W3C. About The W3C Markup Validation Service. <http://validator.w3.org/about.html>. WWW-dokumentti. Luettu 12.3.2011.
- 10 Parkkinen, Jarmo. Hyvään verkkopalveluun! – Käytettävyysoapas verkkoviestijöille. Tampere: Inforviestintä Oy. 2002.
- 11 Järvinen, Jani. Visual Studio 2008 –käsikirja. Jyväskylä: WSOYpro Oy. 2008.
- 12 Inkinen, Ville. ASP.NET. Jyväskylä: Docendo Finland Oy. 2003.
- 13 Pietilä, Antti. Oletko vetänyt pilveä viimeaikoina? WWW-dokumentti. Luettu 21.2.2011. <http://www.ohjelmistoyrittajat.fi/blogi/2010/05/10/oletko-vetanyt-pilvea-viime-aikoina/>.
- 14 Heino, Petteri. Pilvipalvelut. Hämeenlinna: Talentum Media Oy. 2010.
- 15 Salo, Immo. Cloud computing, palvelut verkossa. Jyväskylä: WSOYpro Oy. 2010.

- 16 VMware. Virtualization Basics.
<http://www.vmware.com/virtualization/why-virtualize.html>. WWW-dokumentti. Luettu 12.3.2011
- 17 Netmarketshare. Browser Market Share.
<http://marketshare.hitslink.com/report.aspx?qprid=0&qptimeframe=M&qpsp=145&qpnp=1>. WWW-dokumentti. Luettu 8.4.2011.
- 18 Tuhkanen, Aki. Henkilökohtainen tiedonanto 2.2.2011. Järjestelmävas-
taava. Mipro Oy. 2011.
- 19 Jasper Smith. Automating Database maintenance in SQL 2005 Express
Edition Part II. <http://www.sqldbatips.com/showarticle.asp?ID=29>.
WWW-dokumentti. Luettu 15.2.2011.