

LAPPEENRANNAN TEKNILLINEN YLIOPISTO

Teknistaloudellinen tiedekunta

Tietotekniikan osasto

**KONFIGURAATIONHALLINTAJÄRJES-
TELMÄN SUUNNITTELU JA PROTOTYYPPI**
KANDIDAATINTYÖ

Työn aihe on hyväksytty 6.3.2013

Työn ohjaaja: Professori Jari Porras

Oki Nokso

TIIVISTELMÄ

Lappeenrannan teknillinen yliopisto
Teknistaloudellinen tiedekunta
Tietotekniikan osasto

Oki Nokso

Konfiguraationhallintajärjestelmän suunnittelu ja prototyyppi

Kandidaatintyö

2013

40 sivua, 10 kuvaa ja 2 liitettä

Tarkastaja: Professori Jari Porras

Hakusanat: konfiguraatio, konfiguraationhallinta, vaatimusmäärittely
Keywords: configuration, configuration management, requirements engineering

Yritykset tarvitsevat nykyisin paljon erilaisia asiakaskohtaisia ratkaisuja. Tämä on johtanut erilaisten konfiguraatioiden tarpeeseen yhdelle tuotteelle. Konfiguraatioiden takia tuotteeseen liittyvän tiedon määrä on kasvanut. Konfiguraatioihin liittyvää tietoa täytyy pystyä hallitsemaan. Hyvällä konfiguraationhallinnalla voidaan tehostaa yrityksen toimintaa. Hyvä konfiguraationhallinta vaatii myös yleensä jonkinlaisen työkalun eli konfiguraationhallintajärjestelmän.

Tässä kandidaatintyössä on suunniteltu konfiguraationhallintajärjestelmä eräälle tietotekniikan alan yritykselle. Yrityksessä ei ole aiemmin ollut käytössä konfiguraationhallintajärjestelmää. Konfiguraationhallinnan helpottamiseksi ja tehostamiseksi päätettiin luoda keskitetty järjestelmä. Järjestelmän avulla hallitaan yrityksen tuotteiden konfiguraatioita ja kerätään tietoja konfiguraatioihin tehdyistä muutoksista.

ABSTRACT

Lappeenranta University of Technology
Faculty of Technology Management
Department of Information Technology

Oki Nokso

Design and Prototype of a Configuration Management System

Thesis for the Degree of Bachelor of Science in Technology

2013

40 pages, 10 figures and 2 appendices

Examiner: Professor Jari Porras

Keywords: configuration, configuration management, requirements engineering

Companies need at present lots of different customer based solutions. This has led to need for different configurations on one product. Amount of product related information has grown because of configurations. Configuration related information has to be managed somehow. Company's operation can be enhanced by good configuration management. Good configuration management requires usually some kind of tool, in other words a configuration management system.

In this Bachelor's thesis, a configuration management system for an IT industry company is designed. In the company, there has not previously been used any configuration management system. To make configuration management easier and more efficient, a centralized system was decided to be created. With system, configurations of company's products are managed and information about made changes to configurations are collected.

SISÄLLYSLUETTELO

1 JOHDANTO	3
1.1 Tausta	3
1.2 Yritys.....	3
1.3 Tavoitteet ja rajaukset	3
1.4 Tutkimusongelma.....	4
1.5 Työn rakenne.....	5
2 KONFIGURAATIONHALLINTA.....	6
2.1 Konfiguraatio	6
2.2 Konfiguraationhallinnan tausta	7
2.3 Vaatimusmäärittely	8
2.4 Projektin tausta.....	10
2.5 Ratkaisutapa	10
3 KONFIGURAATIONHALLINTAJÄRJESTELMÄN SUUNNITTELU.....	12
3.1 Työn kuvaus	12
3.2 Työn vaiheet.....	12
3.3 Toteuttaminen	16
3.3.1 Arkkitehtuuri	18
3.3.2 Tietokanta.....	19
3.3.3 Käyttötapaukset.....	21
3.3.4 Käyttöliittymä	22
3.4 Ongelmat	25
3.5 Tulosten yhteenveto ja tulkinta	26
4 JOHTOPÄÄTÖKSET	28
LÄHTEET	29
Liite 1: Vaatimukset	
Liite 2: Käyttötapaukset	

LYHENTEET

EJB	Enterprise Java Beans
ICT	Information and Communications Technology
IEEE	Institute of Electrical and Electronics Engineers
Pk-yritys	Pieni ja keskisuuri yritys
RCS	Revision Control System
SCCS	Source Code Control System
XML	Extensible Markup Language

1 JOHDANTO

Tämä kandidaatintyö on toteutettu osana Propentus Oy:n konfiguraationhallintajärjestelmän toteutusprojektia. Kandidaatintyö käsittelee koko projektin osalta järjestelmän suunnitteluvaihetta ja prototyypin toteutusvaihetta.

1.1 Tausta

Konfiguraationhallintajärjestelmän toteutusprojektin taustalla ovat yrityksen tarpeet yksinkertaistaa, tehostaa ja valvoa konfiguraatiodostoihin tehtäviä muutoksia. Yrityksessä ei tällä hetkellä ole käytössä työkalua konfiguraationhallintaan. Konfiguraatit tehdään muokkaamalla suoraan xml-pohjaisia (Extensible Markup Language) konfiguraatiodostoja käyttäen tekstieditoria. Konfiguraation tekeminen on siis tällä hetkellä aikaa vievä ja virhealtis prosessi. Yrityksellä on ollut tarve konfiguraationhallintajärjestelmälle jo pidemmän aikaa, mutta aika ja resurssit eivät ole aiemmin riittäneet projektin toteutukseen.

1.2 Yritys

Propentus Oy, jolle konfiguraationhallintajärjestelmä suunnitellaan ja toteutetaan, on tietotekniikan alalla toimiva pk-yritys (pieni ja keskisuuri yritys). Yritys keskittyy palvelujen, prosessien ja tietojärjestelmien kehittämiseen asiakkailleen, suuremmille yrityksille. Yritys tarjoaa ohjelmistoja mm. käyttöoikeuksien ja identiteetinhallintaan.

1.3 Tavoitteet ja rajaukset

Kandidaatintyön ensijaisena tavoitteena on suunnitella yritykselle myöhemmin toteutettava konfiguraationhallintajärjestelmä. Toissijaisena tavoitteena on toteuttaa järjestelmän prototyypin niin lähelle lopullista järjestelmää, kuin vain on työlle varatun ajan puitteissa mahdollista. Työn lopputuloksena syntyy kaksi eri dokumenttia: konfiguraationhallintajärjestelmän toiminnallinen määrittely sekä kandidaa-

tintyön loppuraportti. Toiminnallinen määrittelydokumentti tulee ainoastaan yrityksen käyttöön.

Työssä toteutettava hallintajärjestelmän prototyyppi ei tule vielä käyttöön yrityksessä, eikä yrityksen asiakkailta. Prototyypin toteutuksen tarkoituksena on auttaa hahmottamaan paremmin lopullisen järjestelmän toimintaa sekä tutustuttaa kehitysympäristöön ja työkalujen käyttöön. Prototyypin avulla saadaan myös palautetta toiminnoista ja toiminnallisuudesta järjestelmän tulevilta käyttäjiltä. Prototyypin toteutuksen laajuutta ei ole rajattu tarkasti, vaan laajuus riippuu siitä, minkä verran aikaa jää käytettäväksi järjestelmän suunnittelun, toiminnallisen määrittelyn ja kandidaatintyön loppuraportin kirjoittamisen ohessa.

Tässä työssä ei toteuteta lopullista konfiguraationhallintajärjestelmää. Järjestelmän toteutus on tarkoitus suorittaa myöhemmin diplomityönä.

1.4 Tutkimusongelma

Kandidaatintyön tutkimusongelmana on selvittää, millainen konfiguraationhallintajärjestelmä yritykselle tulisi toteuttaa. Työssä pyritään siis vastaamaan seuraavaan tutkimuskysymykseen: **mitä ominaisuuksia ja toiminnallisuuksia vaaditaan järjestelmältä, jolla voidaan hallita yrityksen tuotteiden konfiguraatioita?**

Ominaisuuksien ja toiminnallisuuksien keräämisessä etsitään ensin kirjallisuudesta aiempia havaintoja hyvien konfiguraationhallintajärjestelmien ominaisuuksista. Tämän jälkeen kartoitetaan yrityksen tarpeet ryhmähaastattelujen (tai tässä tapauksessa enemmänkin ryhmäkeskustelujen) perusteella.

1.5 Työn rakenne

Kandidaatintyön rakenne jakautuu seuraavalla tavalla. Luvussa 2 eli teoriaosuudessa määritellään työssä käytetyt käsitteet, kerrotaan käsitteiden taustoista ja esitellään tarkemmin tutkimusongelma ja tapa, jolla ongelmaa lähdetään ratkaisemaan. Luvussa 3 eli toteutusosuudessa kuvataan työn vaiheet, toteuttaminen ja tulokset. Luvussa 4 on esitetty kandidaatintyön johtopäätökset. Johtopäätöksissä tarkastellaan, mikä oli lopputulos ja miten projekti toteutui.

2 KONFIGURAATIONHALLINTA

Tässä luvussa esitellään työssä käytettävät käsitteet. Luvussa 2.1 määritellään, mikä on konfiguraatio ja miksi konfiguraatioita tarvitaan. Luvussa 2.2 kerrotaan konfiguraationhallinnan taustoista: mistä konfiguraationhallinta on lähtöisin ja miten konfiguraationhallinta on muuttunut ajan myötä. Luvussa 2.3 kerrotaan vaatimusmäärittelystä: mikä on vaatimusmäärittelyn tarkoitus ja tavoite. Luvussa 2.4 esitellään tutkimusongelman taustat. Luvussa 2.5 esitellään, miten tutkimusongelmaa lähdetään ratkaisemaan.

2.1 Konfiguraatio

Yrityksillä on nykyisin tarve vastata yksityiskohtaisiin asiakasvaatimuksiin tuotteen erilaistamisen avulla. Tästä johtuen tarvitaan asiakaskohtaisia ratkaisuja. Yrityksen tarvitsee mahdollistaa asiakaskohtaiset muutokset tuotteeseen, jotta voidaan tyydyttää asiakkaiden erilaiset tarpeet. Nopea ratkaisu tähän ongelmaan on konfiguraatioiden mahdollistaminen tuotteisiin. Tällä tavalla pystytään pitämään muutoksista johtuvat kustannukset alhaisina. Konfiguraatio voidaan määrittellä seuraavasti: olemassa olevia komponentteja yhdistelemällä luodaan asiakkaan tarpeita vastaava tuote. (Kropsu-Vehkaperä et al. 2011)

Erilaisten konfiguraatioiden tarve syntyy normaalisti tuotteen eri versioiden takia, mutta myös eri laite- tai ohjelmistoympäristöihin tehtävistä variaatioista, tuotteen myynnistä osissa erilaisina konfiguraatioina sekä asiakaskohtaisten muutosten tekemisestä tuotteeseen. (Haikala & Mikkonen 2011) Tuotteen muuttaminen konfiguraation avulla on eri asia kuin uuden version tekeminen tuotteesta. Konfiguroitavasta tuotteesta voi esiintyä useita eri muunnoksia rinnakkain, kun taas uusi tuoteversio tehdään korvaamaan vanhemman version. Uusi tuoteversio sisältää yleensä korjauksia tai parannuksia tuotteeseen, kuten esimerkiksi paremman laadun tai tehokkuuden. (Kropsu-Vehkaperä et al. 2011)

2.2 Konfiguraationhallinnan tausta

Konfiguraationhallinta on lähtöisin 1950-luvulta avaruusteollisuudesta, kun avaruusalustuotannossa havaittiin ongelmia puutteellisesti dokumentoitujen ohjeiden ja suunnittelumuutoksien seurauksena. Vuosikymmeniä myöhemmin ohjelmistoissa alkoi olla samoja haasteita muutostenhallinnan kannalta. Huomattiin, että samoja tekniikoita voitaisiin hyödyntää minkä tahansa tekstimuotoisen ohjelmiston hallinnassa. Ohjelmistoista kuitenkin tuli nopeasti verkossa toimivia ja hallinta voitiin hoitaa automaattisella järjestelmällä. Tämän seurauksena vanhoihin ohjelmiston konfiguraationhallintajärjestelmiin liittyvä tietämys ei enää pätenyt. (Estublier et al. 2005)

Ohjelmiston konfiguraationhallinta ilmestyi esiin taas 1970-luvun lopulla. Käytetyimmät työkalut olivat: SCSS (Source Code Control System), RCS (Revision Control System), Make ja Sablime. Konfiguraationhallintajärjestelmät kohdistuivat tiettyyn toiminnallisuuteen ja keskittyivät versionhallintaan tai tukemaan build- prosessia, jossa lähdekooditiedostojen perusteella luodaan suoritettava ohjelma. Pian edelliset toiminnallisuudet integroitiin yksittäisiin ohjelmistojen konfiguraationhallintajärjestelmiin seuraavien päätoiminnallisuuksien kanssa: mahdollisuus hallita ohjelmistotuotteen luomiseen käytettäviä tiedostoja, mahdollisuus seurata muutoksia tiedostoihin, niistä seuraavien versioiden kannalta ja mahdollisuus tukea suoritettavan ohjelman koostamista lähdetiedostoista. Pohjimmiltaan tämä keskittyminen pätee vielä nykypäivänäkin. (Estublier et al. 2005)

Konteksti, jossa ohjelmistojen konfiguraationhallintajärjestelmät toimivat on muuttunut huomattavasti. Aluksi järjestelmillä hallittiin kriittisiä ohjelmistoja yhden henkilön toimesta keskustietokoneella. Järjestelmien tarve oli hallita versiointia ja tukea build-prosessia. Tämän jälkeen ohjelmistojen konfiguraationhallintajärjestelmät muuttuivat tukemaan ensisijaisesti käyttäjäryhmien tekemää, suuren mittakaavan kehitystä ja ylläpitoa UNIX-järjestelmässä. Järjestelmien tarve muuttui työympäristön hallinnaksi. Nykyisillä ohjelmistojen konfiguraationhallintajärjestelmillä hallitaan eri ohjelmistoilla tehtyä, useiden käyttäjien tekemää, hajautetuissa ympäristöissä tapahtuvaa ja useilla eri laitteilla tehtyä kokonaisuutta. (Estublier et al. 2005)

Dart (1991) määrittelee konfiguraationhallintajärjestelmän tärkeät toiminnallisuusalueet

- Komponentit: järjestelmä tunnistaa, luokittelee, säilyttää ja pääsee käsiksi komponentteihin, joista tuote koostuu.
- Rakenne: järjestelmä osaa kuvata tuotteen arkkitehtuuria käyttäjälle.
- Rakentaminen: järjestelmä tukee tuotteen ja sen osien rakentamista
- Tarkastaminen: järjestelmä säilyttää tarkastuspolun tuotteesta ja sen prosesseista, jotta palauttaminen edelliseen tilaan on mahdollista.
- Kirjaaminen: järjestelmä kerää tietoa tuotteesta ja prosessista.
- Hallinta: järjestelmä hallitsee, milloin ja miten muutoksia tehdään.
- Prosessi: järjestelmä tukee tuotteen kehittymisen hallintaa.
- Tiimi: järjestelmä vastaa tuoteperheiden kehityksestä ja ylläpidosta, konfliktitilanteiden ratkaisemisesta ja käyttäjien yhteistoiminnan toimivuudesta. (Dart 1991)

Tuotteen konfiguraatioiden tehokas hallinta on edelleen haastavaa, vaikka konfiguraationhallinta ei enää olekaan uusi toimintatapa. Konfiguraatioiden mahdollistamisen seurauksena tuotteeseen liittyvän tiedon määrä on kasvanut valtavasti, mikä vaikeuttaa tuotehallintaa ja tuotetukea. Konfiguraationhallinnan nähdään antavan suuren määrän potentiaalia tuotekehityksen parantamiseen ja toiminnan tehostamiseen erityisesti ICT-sektorilla (Information and Communications Technology) toimivissa yrityksissä. Useat tutkimukset pitävät jonkinlaista työkalua välttämättömänä tuotteen konfiguraatioiden tehokkaassa luomisessa ja hallinnassa. (Kropsu-Vehkaperä et al. 2011) Tuotteenhallinta ja muutosten tekeminen asiakkaan virheraporttien ja muutospyyntöjen perusteella voi olla työläs prosessi ilman kunnollista konfiguraationhallintaa. (Haikala & Mikkonen 2011)

2.3 Vaatimusmäärittely

Vaatimus on jotain, mitä tuotteella pystyy tekemään, tai ominaisuus, joka tuotteella tulee olla. Vaatimukset jaotellaan yleensä kolmeen eri luokkaan:

- Toiminnallinen vaatimus (functional requirement), esimerkiksi käyttäjän on kirjaututtava sisään tunnuksillaan päästäkseen käyttämään järjestelmää.
- Ei-toiminnallinen vaatimus (non-functional requirement), esimerkiksi käyttöliittymä toteutetaan ulkonäöltään yhtenäiseksi yrityksen muiden tuotteiden kanssa.
- Reunaehto (constraint), esimerkiksi järjestelmä on toteutettava java-kielellä eclipse-ympäristössä.

Järjestelmän tarpeita ja liiketoiminnallisia tavoitteita analysoidessa saadaan tuloksena asiakasvaatimuksia. Asiakasvaatimus tulee suoraan asiakkaan tarpeista ja kuvaamiseen ei yleensä tarvita muuta kuin sovellusalueen sanastoa. Ohjelmistovaatimus toteuttaa asiakasvaatimuksen. Ohjelmistovaatimukset ovat yleensä ohjelman toimintoja, jotka määrittelevät, miten kyseinen asiakasvaatimus tuodaan toteutettavassa ohjelmistossa esille käyttäjälle. Ohjelmistovaatimukset kuvautuvat lopulta toteutustasolla joukoksi teknisiä vaatimuksia. Tärkeä ominaisuus vaatimuksille on jäljitettävyyden jäljitettävyydellä tarkoitetaan, että asiakasvaatimuksen elinkaarta voidaan seurata määrittelystä toteutukseen asti (eteenpäin jäljitettävyyden). Toisaalta elinkaarta täytyy pystyä seuraamaan myös takaperin eli toteutuksesta takaisin asiakasvaatimukseen (taaksepäin jäljitettävyyden). (Haikala & Mikkonen 2011)

Ohjelmistotuotannon tuotekehityksen ensimmäinen vaihe on vaatimusmäärittely, jossa selvitetään, millainen toteutettavan järjestelmän tulee olla ja miten järjestelmän tulee toimia (Viitanen 2005). Vaatimusmäärittelyssä pyritään selvittämään järjestelmän asiakasvaatimukset mahdollisimman perusteellisesti. Vaatimukset kartoitetaan, analysoidaan ja dokumentoidaan. Vaatimusten kartoittamisessa käytetään esimerkiksi tulevien käyttäjien ja muiden sidosryhmien haastatteluja ja aivoriihiä. Vaatimusmäärittelyn tuloksena on yleensä dokumentti, toiminnallinen määrittely. Dokumentti sisältää yleensä sekä asiakas- että ohjelmistovaatimukset. (Haikala & Mikkonen 2011)

Vaatimuksia määriteltäessä tulee ottaa huomioon järjestelmän toiminnallisuus, tehtävien suorituskyky, käyttöliittymän suunnittelu ja toiminnallisuuden sekä ylläpidon toteutettavuus. Vaatimukset on hyvä priorisoida tärkeyden mukaan. Priorisoinnilla voidaan painottaa, mitkä vaatimukset on ehdottomasti saatava toteutettua kyseiseen versioon ja

mitkä taas voidaan esimerkiksi ajan puutteen vuoksi jättää toteutettavaksi seuraavaan versioon. (Viitanen 2005)

2.4 Projektin tausta

Konfiguraationhallintajärjestelmän toteutusprojektin taustalla ovat yrityksen tarpeet yksinkertaistaa, tehostaa ja valvoa konfiguraatitiedostoihin tehtäviä muutoksia. Konfiguraatioita tehdään tällä hetkellä ilman erillistä, keskitettyä työkalua, suoraan konfiguraatitiedostoja muokkaamalla. Konfiguraatitiedostojen etsiminen, lukeminen ja muokkaaminen ovat siis tällä hetkellä aikaa vieviä ja työläitä prosesseja. Muokkaaminen on kaiken lisäksi erittäin virhealtista. Konfiguraatioita tehdessä täytyy muokkauksia usein tehdä moneen eri tiedostoon. Lisäksi tiedostot ovat xml-muodossa, jolloin muokattavien kohtien löytäminen voi olla hankalaa. Muutoksien tekeminen suoraan xml-pohjaiseen tiedostoon aiheuttaa myös helposti virheitä. Nykyisessä tilanteessa virheiden jäljittäminen voi olla työlästä virhetilanteiden sattuessa.

Konfiguraationhallintajärjestelmän tarkoitus on helpottaa konfiguraatioprosessia ja vähentää konfiguraatioihin käytettyä työmäärää. Järjestelmän avulla pystytään seuraamaan konfiguraatitiedostoihin tehtyjä muutoksia ja voidaan tarvittaessa palauttaa tiedostot aikaisempaan tilaansa virhetilanteiden varalta. Järjestelmän avulla voidaan myös rajoittaa eri käyttäjien oikeuksia muokata konfiguraatitiedostoja. Järjestelmä myös seuraa käyttäjän toimintaa ja tallentaa tietoja käyttäjän suorittamista toimista. Seurannan tarkoitus on helpottaa ja yksinkertaistaa virheiden jäljittämisprosessia. Lisäksi yrityksen paremman tuottavuuden kannalta on tärkeää saada selville, onko virhetilanne johtunut yrityksen vai asiakkaan työntekijän suorittamista muutoksista konfiguraatitiedostoihin.

2.5 Ratkaisutapa

Kandidaatintyö toteutetaan seuraavissa vaiheissa. Aluksi tutustutaan konfiguraationhallintaan ja vaatimusmäärittelyyn lähdeaineiston perusteella. Tämän jälkeen määritellään järjestelmän tarpeet. Seuraavaksi kartoitetaan, analysoidaan, dokumentoidaan ja validoidaan vaatimukset sekä toteutetaan järjestelmän prototyyppi. Tämän jälkeen luodaan

malli järjestelmästä. Mallissa kuvataan järjestelmän arkkitehtuuri, käyttötapaukset ja käyttöliittymä. Vaatimukset ja malli dokumentoidaan toiminnalliseen määrittelydokumenttiin. Viimeisenä vaiheena kirjoitetaan kandidaatintyö lopulliseen muotoonsa.

3 KONFIGURAATIONHALLINTAJÄRJESTELMÄN SUUNNITTELU

Tässä luvussa kuvataan työn vaiheet, toteuttaminen, työssä kohdatut ongelmat ja työn tulokset sekä yhteenveto tuloksista.

3.1 Työn kuvaus

Työssä suunnitellaan yritykselle konfiguraationhallintajärjestelmä ja toteutetaan järjestelmän prototyyppi. Järjestelmällä hallitaan yrityksen tuotteiden konfiguraatioita. Järjestelmän avulla muokataan yrityksen tuotteiden konfiguraatioita eli tehdään muokkaukset konfiguraatitiedostoihin ja voidaan tarvittaessa seurata ja jäljittää tehtyjä muokkauksia.

3.2 Työn vaiheet

Työ suoritetaan seuraavissa vaiheissa:

1. Konfiguraationhallintaan ja vaatimusmäärittelyyn tutustuminen lähdeaineiston perusteella
2. Järjestelmän tarpeiden määrittely
3. Vaatimusten kartoittaminen, analysointi, dokumentointi ja validointi sekä prototyyppi järjestelmästä
 - a. Alustavat asiakasvaatimukset ja käyttötapaukset
 - b. Palaute sidosryhmiltä
 - c. Päivitetyt asiakasvaatimukset ja käyttötapaukset
 - d. Alustavat ohjelmistovaatimukset, abstrakti malli järjestelmästä ja prototyyppi
 - e. Palaute sidosryhmiltä
 - f. Päivitetyt ohjelmistovaatimukset ja prototyyppi
 - g. Vaatimusten priorisointi
 - h. Vaatimusten dokumentointi
 - i. Vaatimusten validointi

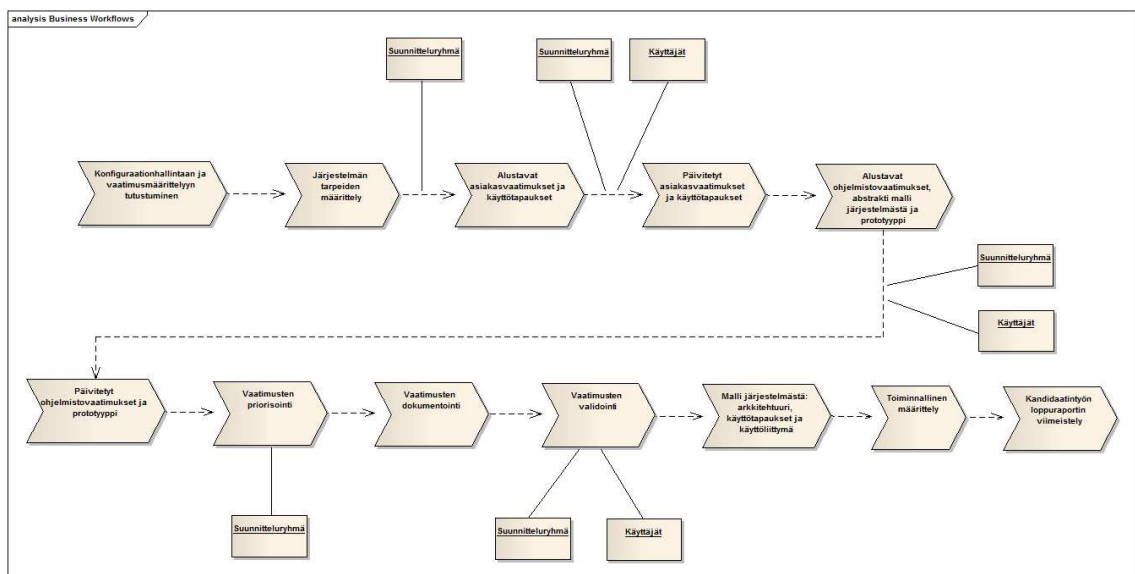
4. Malli järjestelmästä: arkkitehtuuri, käyttötapaukset ja käyttöliittymä
5. Toiminnallisen määrittelydokumentin kirjoittaminen
6. Kandidaatintyön loppuraportin kirjoittaminen

Seuraavassa kuvassa (kuva 1) on esitetty työvaiheiden alkuperäinen aikataulu.

Vaihe/Viikko	47	48	49	50	51	52
1	█					
2		█				
3			█			
4				█		
5		█	█	█	█	
6	█	█	█	█	█	█

Kuva 1: Työvaiheiden aikataulu

Sidosryhmiä on kaksi, suunnitteluryhmä ja käyttäjät. Suunnitteluryhmään kuuluu tämän työn tekijän lisäksi seuraavat henkilöt yrityksessä: työn ohjaaja, järjestelmien pääarkkitehti, järjestelmien käytettävyyssarkkitehti ja kehityspäällikkö. Käyttäjät ovat järjestelmän ensimmäisen version tulevat käyttäjät, yrityksen palveluksikkö (palvelupäällikkö ja järjestelmäasiantuntijat) sekä testauspäällikkö. Seuraavassa kuvassa (kuva 2) on esitetty työvaiheet ja sidosryhmien panos työhön.



Kuva 2: Työvaiheet ja sidosryhmien panos

Aluksi tutustutaan konfiguraationhallintaan ja vaatimusmäärittelyyn lähdeaineiston avulla. Lähdeaineiston etsinnässä hyödynnetään työhön liittyviä avainsanoja. Avainsanoja ovat konfiguraatio, konfiguraationhallinta, vaatimusmäärittely ja prototyyppi.

Seuraavaksi aloitetaan järjestelmän tarpeiden määrittely suunnitteluryhmän välisissä keskusteluissa. Järjestelmän tarpeen määrittäminen on erittäin tärkeä vaihe järjestelmän kehittämisessä. Jos järjestelmän tarkoitus ei ole tiedossa, niin voi olla epäselvää, millainen järjestelmä lopulta tullaan kehittämään. Lisäksi on mahdotonta päätellä, tuleeko kehitettävä järjestelmä täyttämään käyttäjien tarpeet. Järjestelmän tarpeiden määrittelyn tavoitteena on saada selville, miten järjestelmän pitää toimia ja mitä järjestelmän on tarkoitus tehdä. (Hull et. al. 2011)

Tämän jälkeen aloitetaan vaatimusten kartoittaminen. Vaatimusten kartoittamisessa on tärkeää, että ymmärretään kehitettävän järjestelmän sovellusalue eli nykyinen tilanne ja ongelmat. On myös tärkeää tunnistaa vaatimusten lähteet eli sidosryhmät ja käyttäjät. (Aurum & Wohlin 2005) Tässä tapauksessa tärkein sidosryhmä on järjestelmän tulevat käyttäjät.

Vaatimusten lähteiden tunnistamisen jälkeen kartoitetaan vaatimukset sidosryhmiltä käyttämällä tekniikoita ja lähestymistapoja. Alustavien asiakasvaatimusten ja järjestelmän käyttötapausten luomisessa käytetään introspektiota, jossa suunnitteluryhmä luo alustavat asiakasvaatimukset sen perusteella, mitä luulevat sidosryhmien ja käyttäjien haluavan ja tarvitsevan järjestelmältä (Aurum & Wohlin 2005). Käyttötapaukset johdetaan asiakasvaatimuksista. Käyttötapaukset ovat hyvä tapa saada sidosryhmät miettimään järjestelmän toimintaa. (Hull et al. 2011) Alustavat asiakasvaatimukset ja käyttötapaukset annetaan sidosryhmille arvioitaviksi. Sidosryhmiltä saadun palautteen perusteella tehdään tarvittavat muutokset. Muokattujen asiakasvaatimusten perusteella luodaan alustavat ohjelmistovaatimukset ja abstrakti malli järjestelmästä sekä toteutetaan järjestelmän prototyyppi. (Aurum & Wohlin 2005) Abstrakti malli luo suunnitteluryhmälle keskustelujen pohjan. Mallilla varmistetaan, että sidosryhmillä on yhteinen ymmärrys ehdotetusta ratkaisusta. Abstrakti malli myös varmistaa sidosryhmille, että järjestelmän kehittäminen etenee oikeaan suuntaan. (Hull et al. 2011)

Perinteisen tuotekehitysmallin yksi heikkous on, että toimiva järjestelmä on saatavilla vasta tuotteen elinkaaren lopulla, yleensä vasta järjestelmätestausvaiheessa. Tällöin voi olla jo liian myöhäistä ryhtyä korjaaviin toimiin, jos huomataan jokin suuri puute tai ongelma. Lisäksi ohjelmistovaatimuksia ei voida testata kunnolla ennen kuin toimiva järjestelmä on mahdollista esitellä loppukäyttäjille. Käyttäjien on myös usein hankalaa hahmottaa järjestelmän toimintaa paperilta. Näiden ongelmien ehkäisemiseksi järjestelmästä kannattaa kehittää prototyyppi. Evoluutioprototyypin etuna on, että voidaan testata tiettyjä järjestelmän osia, kuten esimerkiksi käyttöliittymää ja keskeisten algoritmien tehokkuutta, oikeellisuutta ja tarkkuutta. Prototyypin avulla saadaan myös aikaista palautetta käyttäjiltä järjestelmän käytöstä. Prototyypillä voidaan myös tietyissä määrin mitata lopullisen järjestelmän suorituskykyä. (Thayer & Dorfman 2000) Alustavat ohjelmistovaatimukset, abstrakti malli sekä prototyyppi annetaan sidosryhmille arvioitaviksi ja palautteen perusteella tehdään tarvittavat muutokset (Aurum & Wohlin 2005).

Seuraavaksi priorisoidaan vaatimukset. Priorisointi on tärkeää, koska joskus vaatimuksia voidaan joutua karsimaan aikataulun, budjetin, resurssien ja laadun konfliktien takia. Priorisoinnissa vertaillaan vaatimuksen hyötyä suhteessa hintaan. Vaatimusten priorisointi tapahtuu tässä tapauksessa sidosryhmien välisissä neuvotteluissa. Järjestelmän vaatimukset priorisoidaan niiden tärkeyden ja toteutukseen kuluvan ajan perusteella. (Aurum & Wohlin 2005) Käytettävä tekniikka on IEEE standardin 830-1998 suosittelema ryhmittely, jossa vaatimukset jaotellaan kolmeen ryhmään (kriittinen, normaali, valinnainen). (IEEE 1998)

Kun vaatimukset on saatu priorisoitua, vaatimukset dokumentoidaan toiminnalliseen määrittelydokumenttiin. Tämän jälkeen suoritetaan validointi katselmoimalla dokumentti sidosryhmien kanssa. (Haikala & Mikkonen 2011) Validoinnissa varmistetaan, että sidosryhmät hyväksyvät järjestelmän vaatimukset ja vaatimukset ovat yrityksen käytäntöjen ja standardien mukaisia. (Paetsch et al. 2003)

Kun vaatimukset on saatu validoitua, luodaan malli järjestelmästä. Mallissa kuvataan järjestelmän lopullinen arkkitehtuuri, lopulliset käyttötapaukset ja lopullinen käyttöliit-

tymä. (Hull et al. 2011) Järjestelmän arkkitehtuuri, käyttötapaukset ja käyttöliittymä dokumentoidaan toiminnalliseen määrittelydokumenttiin.

3.3 Toteuttaminen

Lähtökohtana yrityksessä ei ollut käytössä minkäänlaista konfiguraationhallintatyökalua tuotteilleen. Projektin minimivaatimuksiksi asetettiin tuotteiden konfiguraationhallinnan siirtäminen keskitettyyn järjestelmään. Järjestelmän tulee siis hallita yrityksen tuotteiden konfiguraatioita. Kandidaatintyön osalta vaatimuksiksi asetettiin järjestelmän suunnittelu ja prototyypin toteutus.

Järjestelmän tarpeiden määrittely suoritettiin suunnitteluryhmän kesken. Tarpeiksi lisättiin konfiguraatioiden muokkaaminen, lisääminen ja poistaminen sekä jonkinlainen seuranta tehdyistä muutoksista.

Järjestelmän tarpeiden perusteella alettiin määritellä alustavia asiakasvaatimuksia ja kehittää järjestelmän prototyyppiä. Vaatimuksia ja prototyypin tilaa katselmoitiin aina tietyin väliajoin suunnitteluryhmän kesken. Vaatimuksia lisättiin ja tarkennettiin aina tarpeen mukaan. Alustavista vaatimuksista johdettiin lopulta alustavat käyttötapaukset. Tämän jälkeen alustavat vaatimukset ja käyttötapaukset annettiin kommentoitaviksi sidosryhmille. Järjestelmän vaatimukset on esitetty liitteessä (liite 1).

Sidosryhmiltä saadun palautteen perusteella päivitettiin vaatimuksia ja lisättiin vaatimukset T16, T18, T19 ja T21. Tämän jälkeen luotiin abstrakti malli järjestelmästä. Abstrakti malli ja prototyyppi esiteltiin suunnitteluryhmän jäsenille. Palautteen perusteella lisättiin vaatimukset T17 ja T20. Tämän jälkeen abstrakti malli ja prototyyppi esiteltiin tuleville käyttäjille. Saadun palautteen perusteella lisättiin vielä vaatimus T3 ja päivitettiin vaatimukset sekä paranneltiin prototyyppiä.

Vaatimukset priorisoitiin tärkeyden ja toteutukseen kuluvan ajan arvion perusteella. Priorisoinnissa käytetty tekniikka on IEEE standardin 830-1998 suosittama ryhmitteily, jossa vaatimukset jaotellaan kolmeen ryhmään: kriittinen, normaali, valinnainen

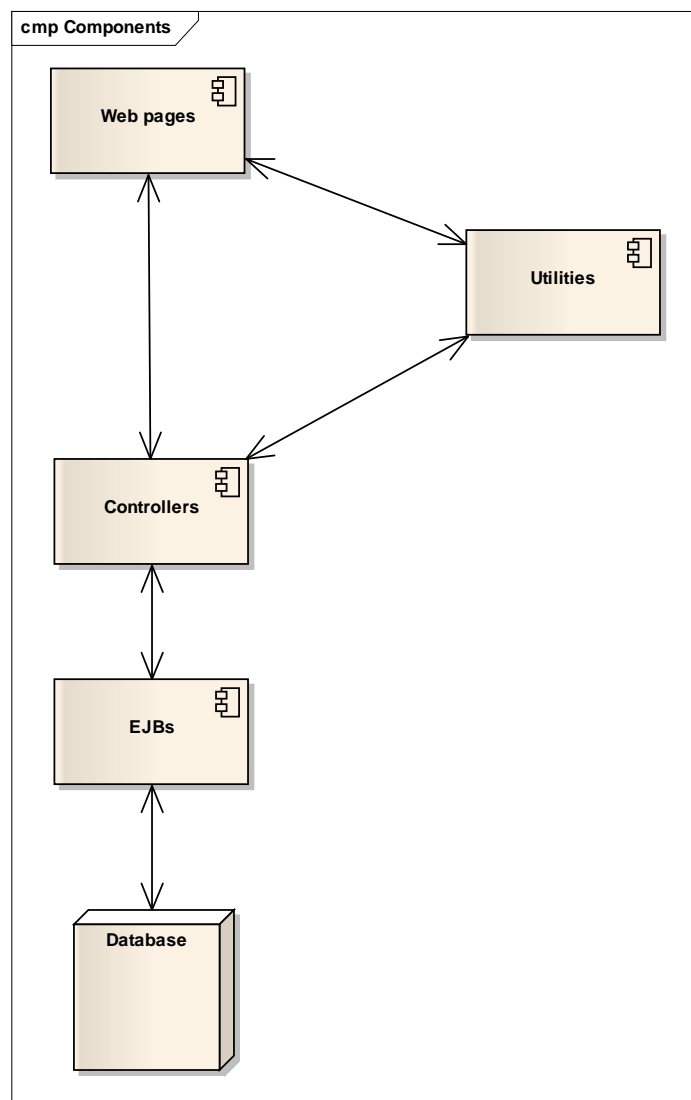
(IEEE 1998). Kriittiseksi merkityt vaatimukset on ehdottomasti saatava toteutettua järjestelmään, jotta järjestelmä olisi hyödyllinen käyttäjille. Normaalille prioriteetille asetetut vaatimukset ovat järjestelmän käytön kannalta tärkeitä, mutta eivät täysin välttämättömiä. Valinnaisiksi merkityt vaatimukset voidaan tarvittaessa (esimerkiksi ajanpuutteen takia) jättää toteutettavaksi tuleviin versioihin.

Priorisointi hyväksyttiin ensin suunnitteluryhmän jäsenillä ja sen jälkeen vaatimukset dokumentoitiin järjestelmän toiminnalliseen määrittelyyn. Tämän jälkeen toiminnallinen määrittelydokumentti validoitiin sidosryhmien kesken. Sidosryhmät hyväksyivät järjestelmän vaatimukset ja lisäksi todettiin, että vaatimukset noudattavat yrityksen käytäntöjä.

Validoinnin jälkeen luotiin lopullinen malli järjestelmästä. Mallissa on kuvattu järjestelmän arkkitehtuuri, käyttötapaukset ja käyttöliittymä. Malli dokumentoitiin toiminnalliseen määrittelyyn. Järjestelmän toiminnallinen määrittely valmistui (ilman lopullista käyttöliittymäsuunnitelmaa) 21.12. eli allekirjoittaneen osalta aikataulun mukaisesti. Käyttöliittymäsuunnitelmasta vastaava käytettävyysarkkitehti toimitti käyttöliittymäsuunnitelman myöhemmin.

3.3.1 Arkkitehtuuri

Seuraavassa kuvassa (kuva 3) on esitetty konfiguraationhallintajärjestelmän arkkitehtuuri. Käyttäjällä on pääsy järjestelmän toimintoihin web-sivujen (Web pages) kautta. Sivut käyttävät tietokantayhteyttä vaativiin toimintoihin joko suoraan kontrollikerrosta (Controllers) tai epäsuoraan työkalukerroksen (Utilities) välityksellä. Kontrollikerros käyttää hyväksi EJB-kerroksen (Enterprise Java Beans) palveluja. EJB-kerros suorittaa kaikki tietokantatoiminnot (haut, lisäykset, päivitykset ja poistot).



Kuva 3: Arkkitehtuuri

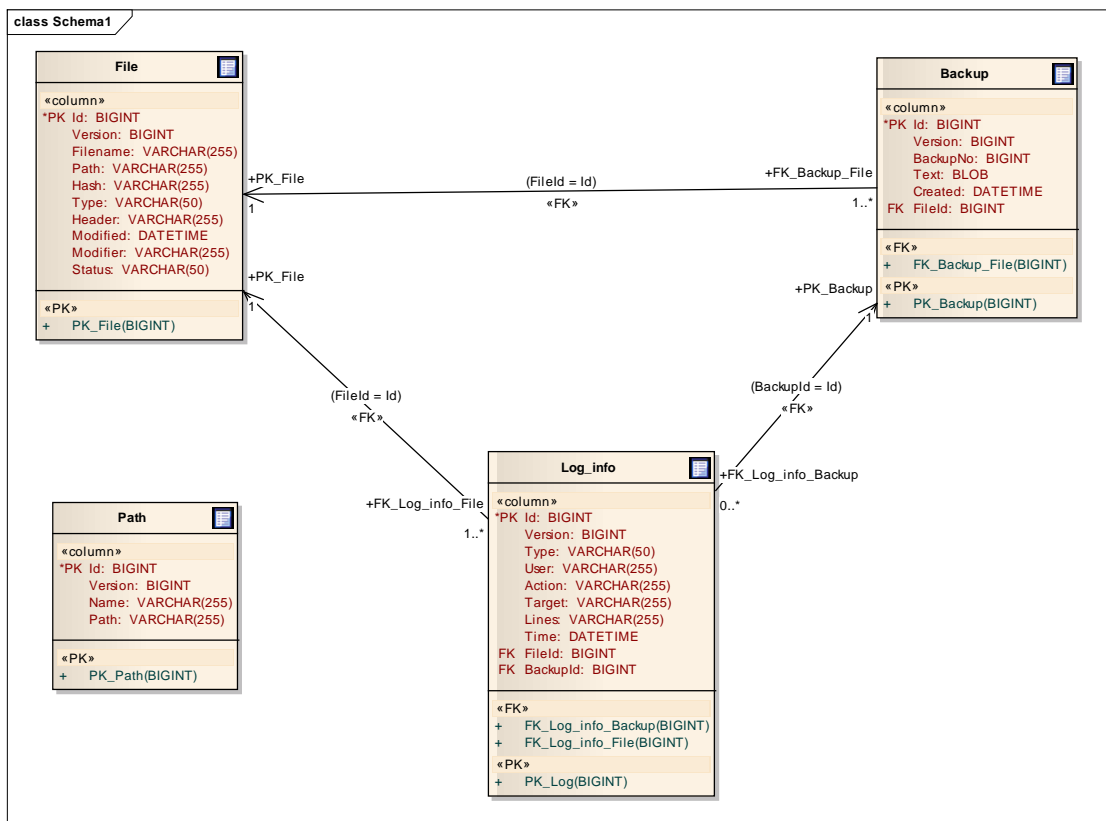
3.3.2 Tietokanta

Seuraavassa kuvassa (kuva 4) on esitetty konfiguraationhallintajärjestelmän käyttämän tietokannan taulut. File-tilusta löytyy tiedot konfiguraatiotiedostoista: tiedoston nimi (Filename), hakemistopolku (Path), josta tiedosto löytyy, tiedostosta laskettu tiiviste (Hash), konfiguraatiotyyppi (Type), kuvaus sisällöstä (Header), viimeisin muokausajankohta (Modified), viimeisin muokkaaja (Modifier) ja tila (Status).

Backup-tilussa on tiedot konfiguraatiotiedoston aiemmista versioista: varmuuskopion järjestysnumero (BackupNo), sisältö (Text), luontiajankohta (Created) ja tiedoston tunnus (FileId), jolla Backup-tilun tiedot linkitetään File-tiluun.

Log_info-tilusta löytyy lokitiedot: lokitiedon tyyppi (Type), toiminnon suorittanut käyttäjä (User), suoritettu toiminto (Action), toiminnon kohde (Target), muokattujen rivien numerot (Lines), lokitiedon luontiajankohta (Time), tiedoston tunnus (FileId), jolla Log_info-tilun tiedot linkitetään File-tiluun, ja varmuuskopion tunnus (BackupId), jolla Log_info-tilun tiedot linkitetään Backup-tiluun.

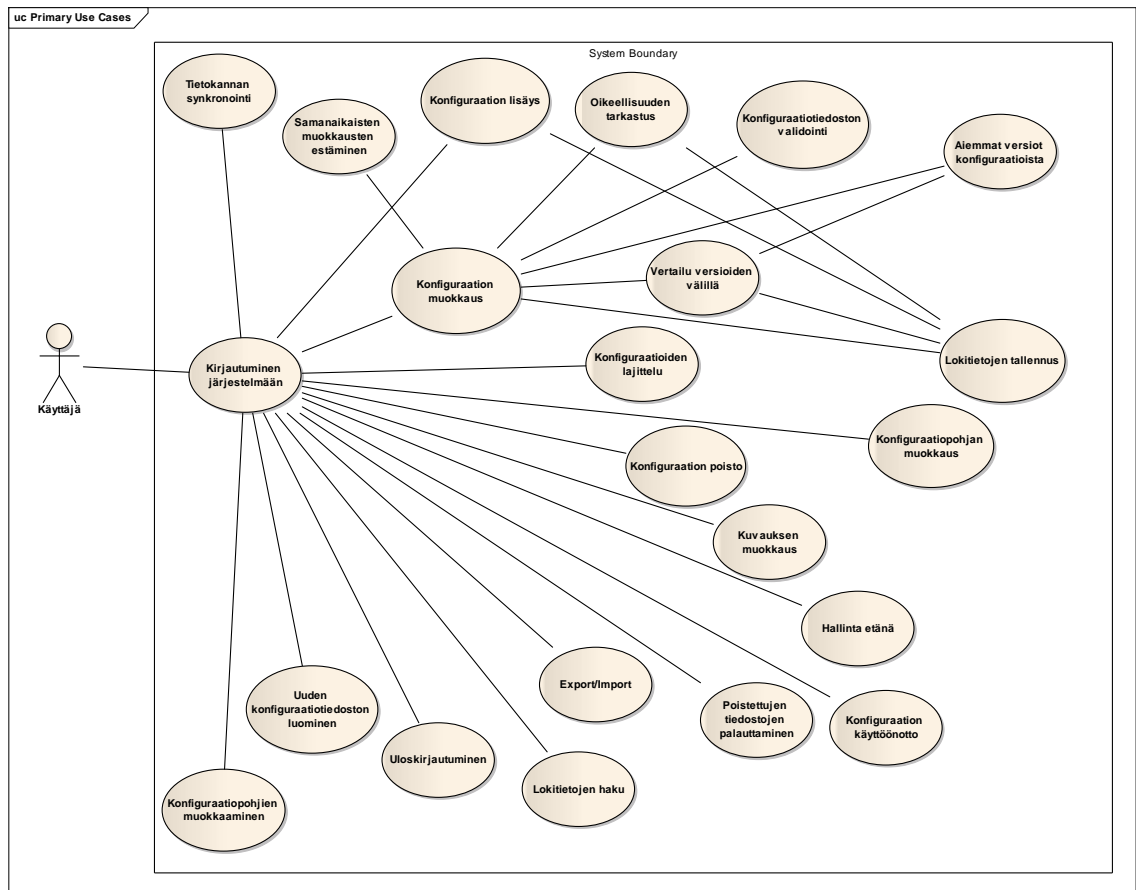
Path-tilusta löytyy hakemistopolkutiedot: hakemiston nimi (Name) ja polku (Path). Taulussa on tiedot hakemistojen sijainneista (tuotteiden konfiguraatiotiedostot ja sovel-luspalvelimen konfiguraatiotiedostot).



Kuva 4: Tietokannan taulut

3.3.3 Käyttötapaukset

Seuraavassa kuvassa (kuva 5) on esitetty järjestelmän käyttötapaukset. Käyttötapaukset on selitetty sanallisesti liitteessä (liite 2).



Kuva 5: Käyttötapaukset

3.3.4 Käyttöliittymä

Seuraavassa kuvassa (kuva 6) on esitetty konfiguraationhallintajärjestelmän etusivu, joka näytetään käyttäjälle kirjautumisen jälkeen. Etusivulla käyttäjä näkee järjestelmän tilatietoja, konfiguraatitiedostot ja tietoja konfiguraatitiedostoista.

The screenshot displays the PROPENTUS configuration management interface. The top navigation bar includes 'PROPENTUS' and 'Properties: United Identity Configuration Manager'. Below the navigation bar, there are tabs for 'KONFIGURAATIO-TIEDOSTOT', 'LOKITIEDOT', and 'JÄRISTELMÄN HALLINTA'. The main content area is divided into several sections:

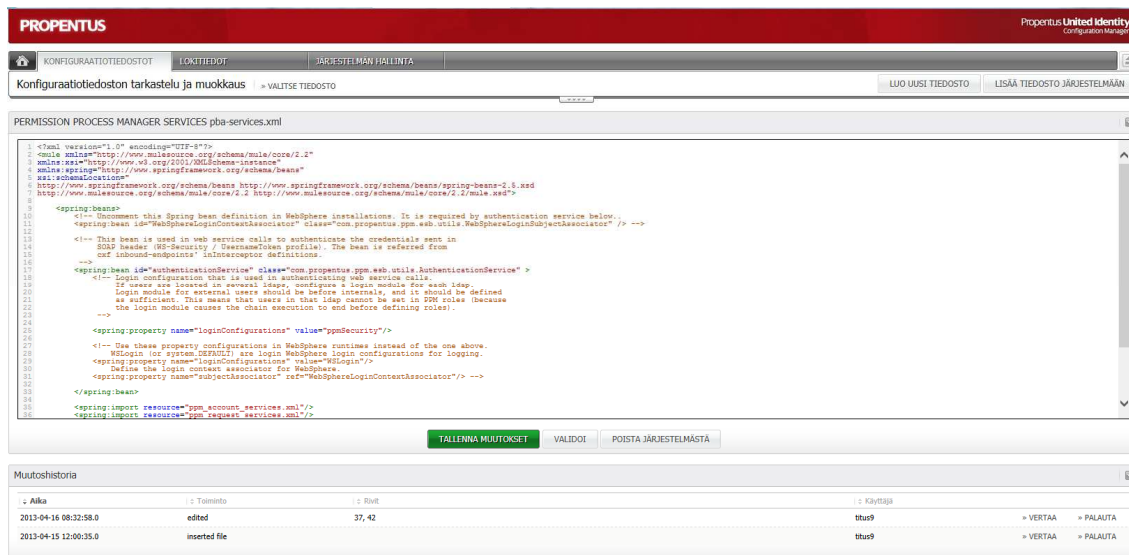
- Toiminnot:** Includes buttons for 'LISÄÄ TIEDOSTO JÄRISTELMÄÄN' and 'HAE LOKITIIETÖJÄ'.
- Järjestelmän tila:** A table showing system status information:

Kokonaismuisti	4079MB	Vapaa muisti	2351MB
Kokonaislevytila	232GB	Vapaa levytila	184GB
Käyttöjärjestelmä	Windows 7	Käyttöjärjestelmän versio	6.1
Prosessori	Intel® Family 6 Model 23 Stepping 6, GenuineIntel	Prosessorien lukumäärä	2
- Employment Manager, konfiguraatitiedostot:** A section for configuration files.
- External Identity Manager, konfiguraatitiedostot:** A table listing configuration files with columns for filename, description, creation date, and user:

Tiedosto	Kuوص	Muokattu	Muokkaja
application_parameters.xml	When an employment is created, it gets this status value (id of one of PPHs statuses)	2013-03-21 14:20:53.0	ttus9
ext-services.xml	This file configures only parts that vary from installation to installation (such as URLs).	2013-03-21 14:20:53.0	ttus9
- Permission Process Manager, konfiguraatitiedostot:** A section for configuration files.
- Permission Process Manager Services, konfiguraatitiedostot:** A section for configuration files.
- Permission Process Manager Provisioning, konfiguraatitiedostot:** A section for configuration files.
- Application Server, konfiguraatitiedostot:** A section for configuration files.
- Poistetut konfiguraatitiedostot:** A section for removed configuration files.

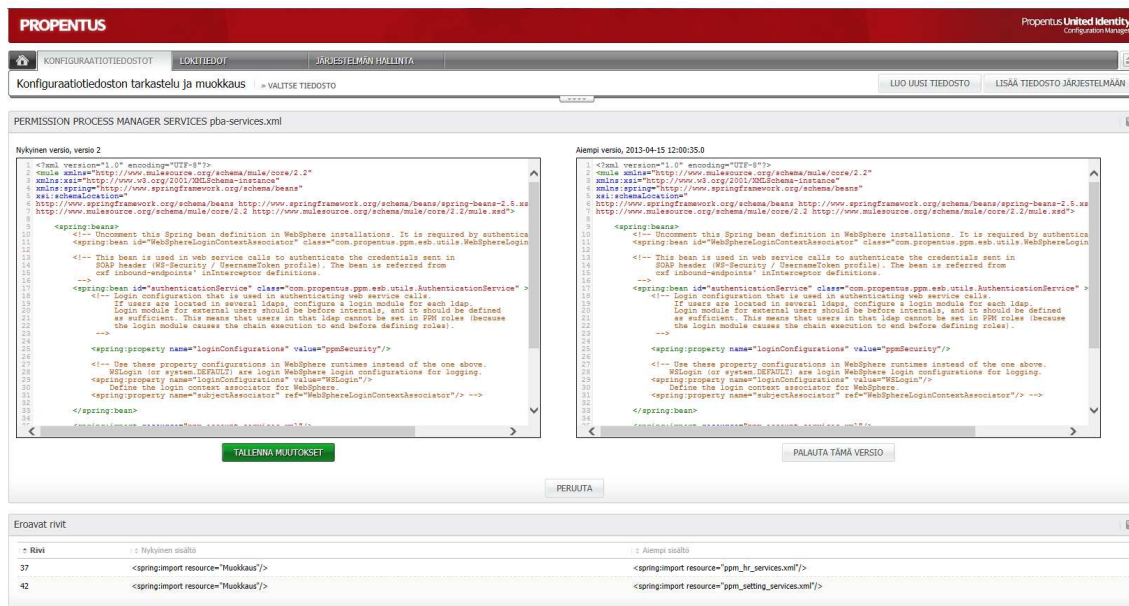
Kuva 6: Etusivu

Seuraavassa kuvassa (kuva 7) on esitetty konfiguraatiodostojen muokkaussivu. Muokkaussivulla käyttäjälle esitetään tiedoston sisältö ja tietoja tiedostoon tehdyistä muutoksista.



Kuva 7: Muokkaussivu

Seuraavassa kuvassa (kuva 8) on esitetty muokkaussivu, kun käyttäjä on valinnut vertailun. Käyttäjää voi vertailla eroja tiedoston nykyisen version ja aiemman version välillä.



Kuva 8: Muokkaussivu (vertailu)

Seuraavassa kuvassa (kuva 9) on esitetty lokitietojen hakusivu. Tällä sivulla käyttäjä voi hakea lokitietoja eri kriteerien perusteella.

PROPENTUS

Propentus United Identity Configuration Manager

KONFIGURAATIOTIEDOSTOT LOKITIEDOT JÄRJESTELMÄN HALLINTA

Lokitietojen haku ja tarkastelu PIILOTA HAKU

Haku

Tyyppi Tiedosto

Käyttäjätunnus Päivämäärä

HAE TYHJENNÄ HAKU

Lokitiedot

Aika	Tyyppi	Toiminto	Kohde	Rivit	Käyttäjä
2013-03-21 14:22:27.0	CHANGE	edited	EMPLOYMENT MANAGER domain_parameters.xml	5	blus9
2013-03-21 14:21:55.0	CHANGE	edited	EMPLOYMENT MANAGER domain_parameters.xml	5	blus9
2013-03-21 14:20:53.0	CHANGE	inserted file	EMPLOYMENT MANAGER domain_parameters.xml		blus9

Propentus Configuration Manager | Version 1

21.3.2013 © 2013 Propentus. All rights reserved.

Kuva 9: Lokitietojen hakusivu

Seuraavassa kuvassa (kuva 10) on esitetty järjestelmän hallintasivu. Hallintasivulla käyttäjä näkee järjestelmän tilatiedot ja asetukset. Hallintasivulla käyttäjä voi alustaa järjestelmän, hakea konfiguraatitiedostoja palvelimelta ja viedä konfiguraatitiedostot palvelimelle.

PROPENTUS

Propentus United Identity Configuration Manager

KONFIGURAATIOTIEDOSTOT LOKITIEDOT JÄRJESTELMÄN HALLINTA

Järjestelmän hallinta

Järjestelmän tila

Kokonaismuisti	4079MB	Vapaa muisti	2341MB
Kokonaislevytila	232GB	Vapaa levytila	184GB
Käyttöjärjestelmä	Windows 7	Käyttöjärjestelmän versio	6.1
Prosessori	Intel® Family 6 Model 23 Stepping 6, GenuineIntel	Prosessorien lukumäärä	2

Järjestelmän asetukset

Konfiguraatitiedostot	C:\Configs\	Sovelluspalvelimen properties-tiedostot	C:\jboss\jboss-4.2.3.GA\server\default\conf\props\
login-config.xml-tiedosto	C:\jboss\jboss-4.2.3.GA\server\default\conf\		

MUOKKAA ALUSTA

Vie ja tuo konfiguraatioita

Hae konfiguraatopaketti C:\Configs\

HAE

Vie konfiguraatopaketti

Browse...

VIE

Propentus Configuration Manager | Version 1

21.3.2013 © 2013 Propentus. All rights reserved.

Kuva 10: Järjestelmän hallintasivu

3.4 Ongelmat

Konfiguraationhallintajärjestelmän suunnittelussa ja prototyypin toteutuksessa ilmeni jonkin verran ongelmia, mutta mitään ylitsepääsemätöntä hankaluutta ei onneksi esiintynyt. Projektin alkuvaiheessa vaikeuksia aiheutti se, että yrityksessä oli vain erittäin pääpiirteinen suunnitelma siitä, mitä järjestelmällä pitäisi pystyä tekemään. Järjestelmän tarpeet ja toiminnallisuudet tarkentuivat aina vähitellen suunnitteluryhmän kesken pidettyjen katselmointien myötä. Järjestelmän suunnitteluprosessia hankaloitti myös se, että allekirjoittaneella ei ollut aiempaa tietämystä ja kokemusta konfiguraatioista, eikä niiden hallinnasta. Konfiguraatioiden käyttötarkoituksen ja muokausprosessin ymmärtäminen vei jonkin verran aikaa.

Yhden toiminnallisen vaatimuksen määrittelyssä oli vaikeuksia löytää ratkaisu, joka tyydytti kaikkia osapuolia. Konfiguraation muokkaamistavan valinnassa oli hankalaa päästä yksimielisyyteen siitä, kumpi on järkevämpi vaihtoehto: tekstieditori vai kehittyneempi käyttöliittymä, jossa olisi mahdollisuus valita vain tiettyjä arvoja listasta. Lopulta jouduttiin päättämään eräänlaiseen kompromissiratkaisuun, jotta järjestelmän toteuttaminen olisi varmasti mahdollista annetun ajan puitteissa. Järjestelmän ensimmäisessä versiossa muokkaamistavaksi valittiin tekstieditori. Lisäksi konfiguraatiodietoista luodaan skeemat. Muokkauksen yhteydessä tiedosto validoidaan skeeman mukaan, jotta vältytään virheiltä. Kehittyneempi käyttöliittymä, jossa arvot valitaan listoista, laitettiin vaatimuksena alimpaan prioriteettiiluokkaan ja ominaisuus jää monimutkaisuutensa takia suurella todennäköisyydellä toteutettavaksi järjestelmän tuleviin versioihin.

Oman haasteensa suunnitteluprosessiin toi myös se, että suunnitteluryhmän muilla jäsenillä oli välillä todella kiire muiden tehtäviensä parissa, minkä takia katselmointeja jouduttiin muutaman kerran siirtämään myöhempään ajankohtaan. Myös kandidaatintyön viimeistely siirtyi myöhemmäksi verrattuna alkuperäiseen aikataulusuunnitelmaan, koska käytettävyyssarkkitehti oli liian työllistetty muissa projekteissa ja lopullinen käyttöliittymäsuunnitelma ei valmistunut ajoissa.

3.5 Tulosten yhteenveto ja tulkinta

Kokonaisuutena työ onnistui ihan hyvin. Konfiguraationhallintajärjestelmän suunnitteluprojekti saatiin päätökseen pääosin aikataulun mukaisesti. Ainoa suurempi poikkeus aikataulusuunnitelmaan oli se, että kandidaatintyön viimeisteleminen siirtyi pari kuukautta myöhemmäksi, jotta järjestelmän lopullinen käyttöliittymäsuunnitelma ehti valmistua käytettävyyssarkkitechdin toimesta. Tulevat käyttäjät kokevat järjestelmän tarpeelliseksi ja yrityksessä ollaan sitä mieltä, että järjestelmä tulee vähentämään konfiguraatioihin käytettyä aikaa ja työmäärää. Järjestelmä tullaan ottamaan aluksi käyttöön yrityksen testiympäristössä. Tulevaisuudessa järjestelmä tullaan liittämään yrityksen muihin tuotteisiin ja toimitetaan myös asiakkaille. Työssä suunniteltu järjestelmä päätettiin toteuttaa kevään 2013 aikana.

Suunnitteluprojektin aikana havaittiin, että suurin haaste järjestelmän toteutuksessa tulee olemaan monimutkaisimpien konfiguraatioprosessien ymmärtäminen, jotta kehittyneemmän käyttöliittymän toteuttaminen näiden konfiguraatioiden muokkaamista varten olisi mahdollista. Toisaalta taas tekstieditorin käyttäminen konfiguraation muokkauksessa mahdollistaa käyttäjän tekemät virheet. Xml-syntaksin ja -skeeman tarkastuksilla vältytään kuitenkin suurimmalta osin virheiltä.

Yleisesti vaatimusmäärittely on todella tärkeä vaihe ohjelmistosuunnittelussa projektin onnistumisen kannalta. Tässä projektissa vaatimusmäärittelyvaihe ei ollut niin kriittinen kuin yleensä, koska asiakkaana toimii yritys itse. Vaatimuksia voidaan siis muuttaa vielä toteutusvaiheessa. Tämä toki lisää jonkin verran työmäärää, mutta toisaalta kehittää myös tässä tapauksessa toteuttajan ohjelmointitaitoja, mikä tulee olemaan jatkossa arvokasta yrityksen kannalta. Laajemmissa projekteissa kannattaa myös harkita jonkin vaatimustenhallintaohjelman käyttöä, jotta vaatimusten pitäminen ajan tasalla ei käy liian työlääksi. Tässä projektissa excel-taulukon käyttö vaatimusten listaamiseen oli riittävä ratkaisu.

Prototyypin toteutus eteni hyvällä tahdilla ja varsinaisen järjestelmän toteutusta on ollut helppo jatkaa prototyypin pohjalta. Prototyypin toteutukseen käytetty lähestymistapa,

jossa vähitellen edetään kohti lopullista järjestelmää ja osa toiminnallisuuksista korvataan täysin uudella toteutuksella, oli tässä työssä järkevä ratkaisu, koska tarkoituksena oli tutustuttaa työkalujen käyttöön ja kehittää ohjelmointitaitoja. Muissa tapauksissa myös prototyypin suunnitteluun kannattaa käyttää hyvin aikaa, jos toteutusvaiheessa halutaan välttyä turhalta työltä. Tämän prototyypin osalta jotkin keskeiset toiminnallisuudet ovat menneet täysin uusiksi työn edetessä ja kokonaistyömäärä on kasvanut.

Kun verrataan suunnitellun järjestelmän ominaisuuksia ja Dartin (1991) määrittelemiä konfiguraationhallintajärjestelmän tärkeimpiä toiminnallisuusalueita, huomataan, että kaikki kohdat toteutuvat järjestelmässä jollain tavalla.

- Komponentit - järjestelmä tunnistaa, luokittelee, säilyttää ja pääsee käsiksi komponentteihin, joista tuote koostuu: tässä tapauksessa järjestelmä ei hallitse suoraan tuotteen komponentteja, vaan konfiguraatiodietoja, jotka vaikuttavat komponentteihin.
- Rakenne - järjestelmä osaa kuvata tuotteen arkkitehtuuria käyttäjälle: tässä tapauksessa järjestelmä ei kuvaa suoraan tuotteen arkkitehtuuria, mutta kuvaa ohjeituksen avulla konfiguraatioihin tehtyjen muutosten vaikutusta tuotteeseen.
- Rakentaminen - järjestelmä tukee tuotteen ja sen osien rakentamista: hallittavilla konfiguraatiodietoilla määritetään, mitkä osat tuotteesta ovat käytössä.
- Tarkastaminen - järjestelmä säilyttää tarkastuspolun tuotteesta ja sen prosesseista, jotta palauttaminen edelliseen tilaan on mahdollista: tässä tapauksessa säilytetään konfiguraatiodietojen aiemmat versiot, jolloin palauttaminen on mahdollista.
- Kirjaaminen - järjestelmä kerää tietoa tuotteesta ja prosessista: lokitietoja kerätään tuotteisiin tehtävistä muutoksista.
- Hallinta - järjestelmä hallitsee, milloin ja miten muutoksia tehdään: käyttöönoton jälkeen kaikki muutokset on tarkoitus tehdä järjestelmän kautta.
- Prosessi - järjestelmä tukee tuotteen kehittymisen hallintaa: uusi versio tuotteesta tuottaa aina uudet mallipohjat konfiguraatiodietoille.
- Tiimi - järjestelmä vastaa tuotepuheiden kehityksestä ja ylläpidosta, konfliktitilanteiden ratkaisemisesta ja käyttäjien yhteistoiminnan toimivuudesta: konfiguraatiodietojen useampi samanaikainen muokkaus on estetty järjestelmässä.

4 JOHTOPÄÄTÖKSET

Kandidaatintyön tuloksena on suunniteltu yrityksen tarpeet toteuttava konfiguraationhallintajärjestelmä. Järjestelmän avulla voidaan hallita ja seurata yrityksen tuotteisiin kohdistuvia konfiguraatioita. Järjestelmän on tarkoitus korvata täysin yrityksessä tällä hetkellä käytössä olevat konfiguraatioprosessit.

Konfiguraationhallinta vaatii paljon teknistä osaamista ja yrityksen prosessien sekä tietojärjestelmien täydellistä ymmärtämistä. Tästä johtuen työ oli erittäin haasteellinen. Lisää haastetta työhön toi se, että yrityksessä ei ollut käytössä minkäänlaista konfiguraationhallintajärjestelmää. Yrityksellä oli vain hyvin pääpiirteinen suunnitelma siitä, mitä järjestelmän avulla tulisi voida tehdä. Järjestelmän suunnittelu jouduttiin aloittamaan siis melkein tyhjästä. Vaikeusastetta lisäsi myös se, että allekirjoittaneella ei ollut aiempaa kokemusta konfiguraatioista eikä niiden hallinnasta.

Järjestelmän toiminnallisen määrittelyn ja prototyypin perusteella saatu palaute oli positiivista. Sidosryhmät olivat hyvin mukana projektissa ja saatu palaute oli hyödyllistä ja tarpeellista. Yrityksessä ollaan vahvasti sitä mieltä, että käyttöönoton jälkeen järjestelmä on hyödyllinen ja tulee helpottamaan ja parantamaan konfiguraationhallintaa.

Lopullisen järjestelmän toteutusprojekti on jo käynnistynyt. Toteutusprojektin on suunniteltu valmistuvan kevään 2013 aikana. Toteutettavia ominaisuuksia on vielä reilusti ja saattaa olla, että osa ominaisuuksista joudutaan siirtämään tuleviin versioihin.

LÄHTEET

Aurum, A. & Wohlin, C. 2005. **Engineering and Managing Software Requirements**, Berlin, Heidelberg: Springer-verlag. ISBN: 978-3-540-28244-0.

Dart, S. 1991. **Concepts in Configuration Management Systems**. *SCM '91 Proceedings of the 3rd international workshop on Software configuration management*. ACM: New York, USA. 1-18.

DeBellis, M. & Haapala, C. 1995. **User-Centric Software Engineering**. *IEEE Expert*, Vol. 10, Issue 1, 34-41.

Estublier J., Clemm G., Leblang D., Tichy W., Van Der Hoek A., Conradi R. & Wiborg-Weber D. 2005. **Impact of Software Engineering Research on the Practice of Software Configuration Management**. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, Vol. 14, Issue 4, 383-430.

Haikala, I. & Mikkonen, T. 2011. **Ohjelmistotuotannon käytännöt**. 12. uud. p. Helsinki: Talentum.

Hull, E., Jackson, K. & Dick, J. 2011. **Requirements Engineering**. London: Springer. ISBN: 978-1-84996-405-0.

IEEE Std. 830-1998. 1998. ISBN: 0-7381-0448-5

Kropsu-Vehkaperä, H., Haapasalo, H., Jaaskelainen, O. & Phusavat, K. 2011. **Product Configuration Management in ICT Companies: The Practitioners' Perspective**. *Technology and Investment*, 2, 273-285.

Paetsch, F., Eberlein, A. & Maurer, F. 2003. **Requirements Engineering and Agile Software Development**. *Enabling Technologies: Infrastructure for Collaborative Enterprises*. WET ICE 2003. ISBN: 0-7695-1963-6.

Thayer, R. & Dorfman, M. 2000. **Software Requirements Engineering**. Los Alamitos, Ca: *IEEE Computer Society Press*. ISBN: 978-1-118-15667-4.

Viitanen, Johanna, 2005. **Vaatimusmäärittely käyttäjakeskeisessä tuotekehityksessä – lähtökohtia verkkopohjaisen oppimisympäristön suunnitteluun**. Diplomityö, Teknillinen korkeakoulu.

Liite 1. Vaatimukset

Toiminnalliset vaatimukset

Viite	Vaatimus	Kuvaus	Prioriteetti
T1	Konfiguraatioiden lajittelu	Järjestelmän avulla käyttäjä pystyy lajittelemaan konfiguraatiot haluamansa kriteerin perusteella. Järjestelmä lajittelee konfiguraatitiedostot automaattisesti konfiguraatiotyypin perusteella. Käyttäjä voi lajitella tiedostot haluamaansa järjestykseen eri kriteerien perusteella kunkin konfiguraatiotyypin sisällä.	Normaali
T2	Konfiguraatitiedoston muokkaus tekstipohjais- ta editoria käyttämällä	Käyttäjä pystyy muokkaamaan konfiguraatitiedoston sisältöä järjestelmän käyttöliittymästä integroitua tekstieditoria käyttäen.	Kriittinen
T3	Konfiguraation muok- kaus edistyneemmällä käyttöliittymällä	Käyttäjä pystyy muokkaamaan konfiguraation arvoja järjestelmän käyttöliittymästä. Järjestelmä huolehtii muutosten tekemisestä konfiguraatitiedostoihin	Valinnainen
T4	Konfiguraatioiden lisäys	Käyttäjällä on mahdollisuus lisätä uusia konfiguraatitiedostoja. Käyttäjä vie tiedoston järjestelmän kautta palvelimelle ja järjestelmä lisää automaattisesti tiedot tietokantaan.	Kriittinen
T5	Konfiguraatioiden pois- to	Käyttäjä pystyy poistamaan käytössä olevia konfiguraatitiedostoja. Poistettaessa tiedosto merkitään käytöstä poistetuksi, mutta säilytetään edelleen konfiguraatihakemistossa.	Kriittinen
T6	Kirjautuminen	Käyttäjän on kirjaututtava sisään tunnuk- sillaan päästäkseen käyttämään järjes- telmää. Kirjautuminen tapahtuu LDAP:in kautta.	Kriittinen
T7	Käyttäjäryhmät	Käyttäjäryhmät määrittävät kenellä on oikeus tarkastella ja muokata konfiguraa- tioita sekä tarkastella lokitietoja.	Valinnainen
T8	Lokitietojen tallentami- nen	Järjestelmä seuraa konfiguraatioihin teh- tyjä muutoksia ja tallentaa tiedot (kuka muokkasi, mitä ja milloin) lokitietoihin.	Kriittinen
T9	Lokitietojen hakeminen	Käyttäjällä on mahdollisuus hakea järjes- telmän tallentamia lokitietoja eri kriteerien perusteella.	Kriittinen
T10	Valmiit konfiguraa- tiopohjat	Järjestelmässä on valmiit konfiguraa- tiopohjat yleisimmille käyttötapauksille. Uutta konfiguraatiota luodessa järjes- telmä luo automaattisesti valmiin konfigu- raatiopohjan käyttäjälle.	Valinnainen

(jatkuu)

Liite 1. (jatkoa)

T11	Konfiguraatiopohjien muokkaaminen	Käyttäjällä on mahdollisuus muokata, lisätä ja poistaa valmiita konfiguraatiopohjia.	Valinnainen
T12	Kieli	Järjestelmän käyttöliittymän kielivaihtoehdot ovat Suomi ja Englanti.	Kriittinen
T13	Kuvaus konfiguraatiotiedostosta (header)	Järjestelmässä on näkyvillä kuvaus jokaisesta konfiguraatiotiedostosta. Kuvaus sisältää tietoja konfiguraatiotiedostosta ja ohjeita muutoksien tekemiseen.	Normaali
T14	Kuvauksen muokkaaminen	Käyttäjällä on mahdollisuus tehdä muutoksia konfiguraatiotiedoston kuvaukseen.	Normaali
T15	Aiemmat versiot konfiguraatioista	Järjestelmä säilyttää tietyn määrän aiempia versioita konfiguraatiotiedostosta mahdollista palauttamista varten. Versiot tallennetaan tietokantaan.	Kriittinen
T16	Konfiguraation ottaminen käyttöön muokkaamisen jälkeen	Konfiguraation muokkaamisen jälkeen konfiguroitavan järjestelmän uudelleenkäynnistys tai ilmoitus uudelleenkäynnistyksestä järjestelmävalvojalle, jotta muokkaukset tulevat voimaan.	Normaali
T17	Konfiguraation oikeellisuuden tarkastus	Konfiguraatiota avattaessa muokattavaksi järjestelmä tarkastaa, onko konfiguraatiota muokattu järjestelmän ohi.	Normaali
T18	Konfiguraationhallinta etänä	Järjestelmän avulla konfiguraationhallinta on mahdollista etänä. Yrityksestä voidaan muokata asiakkaan järjestelmässä olevia konfiguraatioita.	Valinnainen
T19	Export/Import	Käyttäjä voi hakea konfiguraatiotiedostopakettin omalle työasemalleen palvelimelta tai viedä konfiguraatiopakettin omalta työasemaltaan palvelimelle järjestelmän avulla.	Normaali
T20	Konfiguraatiotiedoston validointi	Konfiguraatiotiedosto validoidaan skeeman perusteella aina muokkauksen yhteydessä. Jos tiedosto ei ole skeeman mukainen, tehtyjä muutoksia ei tallenneta tiedostoon. Jos tiedostosta ei ole olemassa skeemaa, validointi suoritetaan xml-syntaksin perusteella.	Normaali
T21	Vertailu konfiguraation aiemman ja nykyisen version välillä	Järjestelmän avulla on mahdollista vertailla konfiguraatiotiedoston aiemman ja nykyisen version sisältöjen eroja.	Normaali
T22	Tietoa ympäristöstä	Järjestelmän käyttöliittymässä näkyville palvelimelle käyttäjä on kirjautuneena (palvelimen nimi tai ip-osoite).	Normaali
T23	Sovelluspalvelimen konfiguraatiotiedostot	Järjestelmän avulla käyttäjä pystyy muokkaamaan myös sovelluspalvelimen konfiguraatiotiedostoja.	Normaali

(jatkuu)

Liite 1. (jatkoa)

T24	Kopio nykyisestä konfiguraatitiedostosta	Järjestelmä säilyttää kopion tiedoston sisällöstä tietokannassa mahdollista palauttamista varten.	Normaali
T25	Tiedoston palauttaminen	Käyttäjä voi palauttaa tiedoston aiemman version tai kopioon nykyisestä versiosta virhetilanteessa.	Normaali
T26	Tietoa järjestelmästä	Järjestelmä esittää käyttäjälle tietoa palvelimen tilasta (vapaa levytila, vapaa muisti) ja palvelimen kokoonpanosta (esim. prosessorin teho).	Valinnainen
T27	Tietokannan synkronointi	Järjestelmä vertaa konfiguraatiohakemiston sisältöä ja tietokannan sisältöä aina kirjautumisen yhteydessä ja päivittää mahdolliset muutokset tietokantaan.	Normaali
T28	Uuden konfiguraatitiedoston luominen	Järjestelmän avulla voidaan luoda uusia konfiguraatitiedostoja.	Normaali
T29	Konfiguraation samanaikaisten muokkauksen estäminen	Järjestelmä pitää huolen siitä, että päällekkäiset muokkaukset eivät ole mahdollisia.	Normaali
T30	Käytöstä poistettujen tiedostojen tarkastelu	Käyttäjä voi halutessaan nähdä listan poistetuista tiedostoista.	Valinnainen
T31	Poistetun tiedoston palauttaminen	Käyttäjä voi ottaa poistetun tiedoston uudelleen käyttöön.	Valinnainen
T32	Käytössä olevat osat	Käyttäjälle ilmenee suoraan järjestelmän käyttöliittymästä, mitkä järjestelmät (PPM, EXT, EM) ovat käytössä ja siten myös konfiguroitavissa.	Normaali
T33	Järjestelmien tila	Järjestelmä esittää käyttäjälle tietoa konfiguroitavien järjestelmien tiloista (esimerkiksi PPM:n versio).	Normaali

Ei-toiminnalliset vaatimukset

Viite	Vaatimus	Kuvaus	Prioriteetti
E1	Järjestelmän ulkonäkö	Järjestelmä tulee toteuttaa ulkonäöltään yhtenäiseksi Propentuksen muiden tuotteiden kanssa.	Normaali

Reunaehdot

Viite	Vaatimus	Kuvaus	Prioriteetti
R1	Järjestelmän toiminta	Järjestelmä toimii web-palveluna Internet-selaimen kautta.	Normaali
R2	Ohjelmointikieli	Järjestelmä toteutetaan Java-kielellä.	Normaali
R3	Ympäristö	Järjestelmä toteutetaan Eclipse-ympäristössä.	Normaali
R4	Sovelluskehys	Järjestelmä toteutetaan wicket-sovelluskehysten päälle.	Normaali

Liite 2. Käyttötapaukset

Nimi: Konfiguraatitiedoston lisäys

Osallistajat: Käyttäjä.

Tuloehdot: Käyttäjä on kirjautunut järjestelmään.

Kuvaus: Käyttäjä valitsee konfiguraation lisäyksen käyttöliittymästä. Käyttäjä valitsee lisättävän konfiguraation tyyppin. Käyttäjä valitsee lisättävän konfiguraatitiedoston omalta työasemaltaan. Järjestelmä lisää tiedoston palvelimelle konfiguraatiotyyppin mukaiseen hakemistoon ja lisää tiedot tietokantaan.

Lopputulokset: Uusi konfiguraatitiedosto on lisätty järjestelmään.

Nimi: Konfiguraation muokkaus (tekstieditori)

Osallistajat: Käyttäjä

Tuloehdot: Käyttäjä on kirjautunut järjestelmään.

Kuvaus: Käyttäjä valitsee muokattavan konfiguraatitiedoston käyttöliittymästä. Järjestelmä esittää tiedoston sisällön käyttäjälle. Käyttäjä tekee tietoihin haluamansa muutokset tekstieditorissa. Järjestelmä luo varmuuskopion vanhasta sisällöstä mahdollista palauttamista varten. Vanha sisältö tallennetaan tietokantaan. Järjestelmä tallentaa käyttäjän tekemät muutokset konfiguraatitiedostoon. Järjestelmä kirjaa tiedon käyttäjän tekemästä muokkauksesta lokitietoihin.

Lopputulokset: Käyttäjän tekemät muutokset on tallennettu tiedostoon.

Nimi: Konfiguraation muokkaus (kehittyneempi käyttöliittymä)

Osallistajat: Käyttäjä

Tuloehdot: Käyttäjä on kirjautunut järjestelmään.

Kuvaus: Käyttäjä valitsee muokattavan konfiguraatitiedoston käyttöliittymästä. Järjestelmä hakee tarvittavat tiedot konfiguraatitiedostosta ja esittää tiedot käyttäjälle. Käyttäjä tekee tietoihin haluamansa muutokset valitsemalla uudet tiedot alavetovalikoista. Järjestelmä luo varmuuskopion vanhasta sisällöstä mahdollista palauttamista varten. Vanha sisältö tallennetaan tietokantaan. Järjestelmä tallentaa käyttäjän tekemät muutokset konfiguraatitiedostoon. Järjestelmä kirjaa tiedon käyttäjän tekemästä muokkauksesta lokitietoihin.

Lopputulokset: Käyttäjän tekemät muutokset on tehty konfiguraatioon.

Nimi: Konfiguraatioiden lajittelu

Osallistajat: Käyttäjä.

Tuloehdot: Käyttäjä on kirjautunut sisään järjestelmään.

Kuvaus: Järjestelmä lajittelee konfiguraatiot automaattisesti konfiguraatiotyyppien mukaisesti. Käyttäjä voi lajitella kunkin konfiguraatiotyyppin sisältämät tiedostot haluamansa kriteerin mukaiseen järjestykseen. Käyttäjä valitsee haluamansa lajittelukriteerin. Järjestelmä lajittelee konfiguraatiot kriteerin mukaiseen järjestykseen ja esittää tuloksen käyttäjälle.

Lopputulokset: Konfiguraatiot on esitetty käyttäjälle käyttäjän haluamassa järjestyksessä.

(jatkuu)

Liite 2. (jatkoa)

Nimi: Konfiguraation poisto

Osallistujat: Käyttäjä.

Tuloehdot: Käyttäjä on kirjautunut järjestelmään.

Kuvaus: Käyttäjä valitsee konfiguraation poiston käyttöliittymästä. Järjestelmä merkitsee tiedoston käytöstä poistetuksi, mutta säilyttää edelleen tiedoston palvelimella mahdollista palauttamista varten.

Lopputulokset: Konfiguraatiotiedosto on merkitty käytöstä poistetuksi.

Nimi: Kuvauksen muokkaus

Osallistujat: Käyttäjä.

Tuloehdot: Käyttäjä on kirjautunut järjestelmään.

Kuvaus: Käyttäjä tekee haluamansa muutokset kuvaukseen muokkaussivulla. Järjestelmä tallentaa tehdyt muutokset. Järjestelmä kirjaa tiedon muutoksista lokitietoihin.

Lopputulokset: Käyttäjän tekemät muutokset on tallennettu.

Nimi: Konfiguraation käyttöönotto

Osallistujat: Käyttäjä.

Tuloehdot: Käyttäjä on kirjautunut järjestelmään.

Kuvaus: Käyttäjä valitsee muokatun konfiguraation ottamisen käyttöön käyttöliittymästä. Järjestelmä käynnistää palvelimen uudelleen tai lähettää tiedon uudelleenkäynnistyksen tarpeesta järjestelmävalvojalle.

Lopputulokset: Tehdyt muutokset on otettu käyttöön.

Nimi: Export/Import

Osallistujat: Käyttäjä.

Tuloehdot: Käyttäjä on kirjautunut järjestelmään.

Kuvaus: Käyttäjä voi hakea konfiguraatiotiedostopakettien kohdejärjestelmästä omalle työasemalleen tai viedä konfiguraatiotiedostopakettien omalta työasemaltaan kohdejärjestelmään järjestelmän käyttöliittymästä.

Lopputulokset: Tiedostopaketti on haettu kohdejärjestelmästä tai viety kohdejärjestelmään.

Nimi: Lokitietojen haku

Osallistujat: Käyttäjä.

Tuloehdot: Käyttäjä on kirjautunut järjestelmään.

Kuvaus: Käyttäjä valitsee lokitietojen haun käyttöliittymästä. Järjestelmä suorittaa tietokantahaun käyttäjän antamien kriteerien perusteella. Järjestelmä esittää haun tulokset käyttäjälle.

Lopputulokset: Käyttäjä näkee haluamansa lokitiedot.

(jatkuu)

Liite 2. (jatkoa)

Nimi: Oikeellisuuden tarkastus

Osallistajat: Järjestelmä.

Tuloehdot: Käyttäjä on valinnut konfiguraation muokkauksen.

Kuvaus: Järjestelmä laskee hashin muokattavasta tiedostosta ja vertaa tulosta tietokannassa olevaan. Jos hashit eivät täsmää, on tiedostoa muokattu järjestelmän ohi. Järjestelmä tallentaa tiedon lokiin ja ilmoittaa virheestä käyttäjälle.

Lopputulokset: Tiedoston oikeellisuus on tarkastettu.

Nimi: Konfiguraatitiedoston validointi

Osallistajat: Järjestelmä.

Tuloehdot: Käyttäjä on muokannut konfiguraatiota.

Kuvaus: Järjestelmä validoi tiedoston joko skeeman tai syntaksin perusteella, riippuen siitä, onko tiedostolle olemassa skeemaa. Jos validointi ei mene läpi, muutoksia ei tallenneta ja käyttäjälle ilmoitetaan virheestä.

Lopputulokset: Konfiguraatitiedosto on validoitu.

Nimi: Aiemmat versiot ja kopio konfiguraatiosta

Osallistajat: Käyttäjä.

Tuloehdot: Käyttäjä on muokannut konfiguraatiota.

Kuvaus: Järjestelmä säilyttää aiemmat versiot tiedostoista ja kopion tiedoston nykyisestä versiosta. Aiemmat versiot ja kopio tallennetaan tietokantaan. Käyttäjän tehdessä muutoksia tiedostoon tallennetaan aina aiempi versio ja kopio tietokantaan mahdollista palauttamista ja vertailua varten.

Lopputulokset: Aiempi versio ja kopio tiedostosta on tallennettu tietokantaan.

Nimi: Vertailu versioiden välillä

Osallistajat: Käyttäjä.

Tuloehdot: Tiedostosta/konfiguraatiosta on olemassa aiempi versio.

Kuvaus: Käyttäjä valitsee vertailun käyttöliittymästä konfiguraation muokkauksen yhteydessä. Järjestelmä hakee aiemman version tietokannasta ja esittää nykyisen ja aiemman version käyttäjälle.

Lopputulokset: Käyttäjällä on vertailtavana nykyinen ja aiempi versio konfiguraatiosta.

(jatkuu)

Liite 2. (jatkoa)

Nimi: Lokitietojen tallentaminen

Osallistajat: Järjestelmä.

Tuloehdot: Käyttäjä on suorittanut toiminnon, josta tallennetaan lokitietoja.

Kuvaus: Järjestelmä tallentaa lokitiedon käyttäjän suorittamasta toiminnosta tietokantaan.

Lopputulokset: Lokitieto on tallennettu tietokantaan.

Nimi: Kirjautuminen järjestelmään

Osallistajat: Käyttäjä.

Tuloehdot: Käyttäjällä on voimassaolevat LDAP-tunnukset järjestelmän käyttöä varten ja oikeus (rooli) hallita konfiguraatioita.

Kuvaus: Käyttäjä saapuu järjestelmän kirjautumissivulle. Hän syöttää käyttäjätunnuksensa ja salasanaansa käyttöliittymään. Järjestelmä tarkastaa syötteiden oikeellisuuden. Käyttäjä ohjataan järjestelmän pääsivulle. Järjestelmä hakee konfiguraatioiden nimet tietokannasta ja esittää ne käyttäjälle lajiteltuna kriteerin mukaan.

Poikkeukset: Käyttäjän syöttämä tunnus ja salasana eivät täsmää.

Lopputulokset: Käyttäjä on kirjautunut järjestelmään.

Nimi: Uloskirjautuminen järjestelmästä

Osallistajat: Käyttäjä.

Tuloehdot: Käyttäjä on kirjautunut järjestelmään.

Kuvaus: Käyttäjä valitsee uloskirjautumisen käyttöliittymästä. Järjestelmä kirjaa käyttäjän ulos ja ohjaa käyttäjän järjestelmän kirjautumissivulle.

Lopputulokset: Käyttäjä on kirjautunut ulos järjestelmästä.

Nimi: Tietokannan synkronointi

Osallistajat: Järjestelmä.

Tuloehdot: Käyttäjä kirjautuu järjestelmään.

Kuvaus: Käyttäjän kirjautuessa järjestelmään järjestelmä käy läpi konfiguraatitiedostohakemiston ja lisää löytyneet uudet tiedostot automaattisesti järjestelmään.

Lopputulokset: Tietokanta on synkronoitu konfiguraatitiedostohakemiston kanssa.

Nimi: Uuden konfiguraatitiedoston luominen

Osallistajat: Käyttäjä.

Tuloehdot: Käyttäjä on kirjautunut järjestelmään.

Kuvaus: Käyttäjä valitsee uuden konfiguraatitiedoston luomisen käyttöliittymästä. Järjestelmä luo tiedoston mallipohjan perusteella ja avaa tiedoston muokattavaksi käyttäjälle. Käyttäjän tallentaessa muutokset lisätään luotu tiedosto järjestelmään.

Lopputulokset: Uusi konfiguraatitiedosto on luotu.

(jatkuu)

Liite 2. (jatkoa)

Nimi: Samanaikaisten muokkausten estäminen

Osallistujat: Järjestelmä

Tuloehdot: Käyttäjä on muokannut konfiguraatiota.

Kuvaus: Järjestelmä estää tiedoston useamman samanaikaisen muokkauksen tarkkailemalla muokkauksen aloitusajankohtaa ja tietokantaan merkittyä kopion tallentamisajankohtaa.

Lopputulokset: Samanaikaiset muokkaukset on estetty.

Nimi: Konfiguraatiopohjien muokkaaminen

Osallistujat: Käyttäjä.

Tuloehdot: Käyttäjä on kirjautunut järjestelmään.

Kuvaus: Käyttäjä valitsee konfiguraatiopohjan muokkaamisen käyttöliittymästä. Käyttäjä tekee muutokset konfiguraatiopohjaan.

Lopputulokset: Muutokset on tallennettu konfiguraatiopohjaan.

Nimi: Konfiguraationhallinta etänä

Osallistujat: Käyttäjä.

Tuloehdot: Käyttäjä on kirjautunut järjestelmään ja käyttäjä on järjestelmän pääkäyttäjä.

Kuvaus: Järjestelmän pääkäyttäjällä on mahdollisuus hallita ja muokata konfiguraatitiedostoja palvelimella etäyhteyden läpi.

Nimi: Käytöstä poistettujen tiedostojen tarkastelu.

Osallistujat: Käyttäjä.

Tuloehdot: Käyttäjä on kirjautunut järjestelmään.

Kuvaus: Käyttäjä valitsee poistettujen tiedostojen tarkastelun käyttöliittymästä. Järjestelmä näyttää listan poistetuista tiedostoista käyttäjälle.

Lopputulokset: Poistetut tiedostot on listattu käyttäjälle.

Nimi: Käytöstä poistetun tiedoston palauttaminen.

Osallistujat: Käyttäjä.

Tuloehdot: Käyttäjä on kirjautunut järjestelmään.

Kuvaus: Käyttäjä valitsee poistetun tiedoston palauttamisen käyttöliittymästä. Järjestelmä lisää tiedoston järjestelmään.

Lopputulokset: Käytöstä poistettu tiedosto on otettu takaisin käyttöön.