

Johanna Vainionpää

Ohjelmiston suunnittelun ja testauksen kehittäminen:

Case Gapassi -projekti

Opinnäytetyö

Kevät 2011

Tekniikan yksikkö, Seinäjoki

Tietojärjestelmäosaamisen koulutusohjelma, YAMK



SEINÄJOEN AMMATTIKORKEAKOULU

Opinnäytetyön tiivistelmä

Koulutusyksikkö: Tekniikan yksikkö, Seinäjoki
Koulutusohjelma: Tietojärjestelmäosaamisen koulutusohjelma

Tekijä: Vainionpää, Johanna

Työn nimi: Ohjelmiston suunnittelun ja testauksen kehittäminen: Case Gapassi -projekti

Ohjaaja: Salmenjoki, Kimmo

Vuosi: 2010

Sivumäärä: 70

Liitteiden lukumäärä: 3

Opinnäytetyön kehityshankkeena on tuottaa ohjelmistoprojektiin liittyvää dokumentaatiota Gapassi-ohjelmistoprojektiin Galileo Oy:n käytettäväksi. Gapassi-projekti käsittää tarjousten- ja tilaustenhallintasovelluksen kehityksen työasemaversiosta internet-selaimen kautta käytettäväksi web-sovellukseksi. Projekti kattaa sovellukseen kirjautumis- eli login-sovelluksen sekä selainkäyttöisen WebKAPU-sovelluksen toteuttamisen. Nimi juontaa juurensa työasemaversiosta KAPU. Työasemaversio on ollut käytössä jo vuodesta 1998. KAPU- ja WebKAPU-ohjelmistot on toteutettu ikkuna- ja oviteollisuuden käyttöön. Ohjelmiston esimerkkikäyttäjänä projektissa toimii ikkunoita ja ovia valmistava yritys Alavus Ikkunat Oy.

Osallistuminen Gapassi-projektiin alkoi, kun WebKAPU:sta oli beta-versio toteutettuna. Alavus Ikkunat Oy otti tämän ensimmäisen vaiheen web-sovelluksen testi-käyttöön. Ohjelmisto todettiin hyvin toimivaksi, joten seuraavana tavoitteena oli saada sovellukselle lisää testikäyttäjiä ja sitä kautta maksavia asiakkaita. Ensimmäisenä työnä projektissa aloitettiin käyttöohjeiden tekeminen käyttäjiä varten. Työhön kuului myös ohjelmiston mainos- ja messumateriaalin tekeminen.

Opinnäytetyö toteutettiin teknisestä näkökulmasta edistäen teknistä dokumentointia. Gapassi-projekti on edennyt ohjelmoijien kesken, joten dokumentointia ei juuri ole. Galileo Oy:n kanssa todettiin että on tärkeää saada vaatimusmäärittely-, suunnittelu- ja mallinnusdokumentteja tulevaisuutta ajatellen. Galileo Oy:llä on tarkoitus laajentaa ohjelmistoaan muillekin teollisuuden aloille, etenkin login-sovelluksen osuutta. Tulevaisuuden ja kasvun kannalta dokumentointi olisi erittäin arvokasta. Opinnäytetyössä käsitellään myös ohjelmiston testaus. Testitapausten avulla toteutetaan testaussuunnitelma, jota Galileo Oy voi käyttää hyväkseen ohjelmiston jatkokehityksessä.

Asiasanat: Gapassi-projekti, vaatimusmäärittely, suunnittelu, mallinnus, testaus

SEINÄJOKI UNIVERSITY OF APPLIED SCIENCES

Thesis abstract

Faculty: School of Technology

Degree programme: Information Systems Competence

Author: Vainionpää, Johanna

Title of the thesis: Development of planning and testing of a software: Case Gapassi-project

Supervisor: Salmenjoki, Kimmo

Year: 2010

Number of pages: 70

Number of appendices: 3

The aim of the thesis was to produce software project documentation for a software project called Gapassi to be used by a Finnish company, Galileo Oy. The Gapassi project includes an offer and order management software development from a workstation version to a version which can be used with web browser. The project includes login-software and a web browser implement called WebKAPU. It was named after the software workstation version KAPU which has been used since 1998. The pilot company of the project is Alavus Ikkunat Oy which produces windows and doors.

The Gapassi-project started when beta-versio of WebKAPU-software had been implemented. Alavus Ikkunat Oy took this first version of the web browser software for the test use. The Software was found to be working well so the next target was to get more test users, aka subsequent paying customers, for the Software. First, user manuals for the program users were produced, as well as marketing material, for example, for the production fair.

The aim of the thesis was to give a more technical perspective as well as to improve technical documentation. The Gapassi-project has only been advanced between the programmers so documentation does not exist. Requirements specification, planning and modelling documentation are very important in the future since Galileo Oy plans to expand its software production for other industrial sectors, especially this login-software part. Thinking of the future growth documentation will be very valuable. Thesis also concerns software testing. By means of a test case, a testing plan, which can be used for Galileo Oy's future development process, will be produced.

Keywords: Gapassi-project, requirements specification, planning, modeling, testing

SISÄLLYS

Opinnäytetyön tiivistelmä	2
Thesis abstract.....	3
Käytetyt termit ja lyhenteet.....	6
Kuvio- ja taulukkoluetelo	8
1 JOHDANTO	9
1.1 Toimeksiantajan ja opinnäytetyön aiheen esittely	9
1.2 Rajaukset opinnäytetyössä.....	11
2 GAPASSI-PROJEKTIN VAATIMUSMÄÄRITTELY	14
2.1 Taustaa Gapassi-projektille ja sen ohjelmistoille KAPU ja WebKAPU.....	15
2.2 Gapassi-projektin tarkoitus ja tavoitteet.....	16
2.2.1 Ohjelmiston suorituskyky ja tehokkuus	18
2.2.2 Ohjelmiston käytettävyys, toipuminen ja tietoturvallisuus.....	19
2.2.3 Ohjelmiston ylläpidettävyys ja operointi	21
3 GAPASSI-PROJEKTIN SUUNNITTELU	22
3.1 Tekninen toteutusluonnos.....	22
3.2 WebKAPU-ohjelmiston toiminta käyttöliittymässä	24
3.3 Login-sovelluksen ja WebKAPU:n käyttöliittymät	27
3.3.1 Kirjautumisvaihe eli ohjelmiston käyttäjä kirjautuu sisään WebKAPU- sovellukseen login-sovelluksen kautta	27
3.3.2 Asiakastietojen ja toimitustietojen syöttäminen järjestelmään.....	28
3.3.3 Tarjouksen rivit eli tarjoukseen syötetyt, asiakkaan pyytämät tuotteet riveinä	29
3.3.4 Oikean mallisten, väristen ja kokoisten ikkunoiden ja ovien syöttäminen järjestelmään	31
3.3.5 Hinnoittelu eli lasketetaan sovelluksella tarjouksen riveille eli tuotteille loppusumma, sekä loppusumma koko tarjoukselle.....	33
3.3.6 Tarjouksen tulostus asiakkaalle joko tiedostona tai paperille	34
3.4 Ohjelmiston tietojen tallennus ja tietokannat	35
4 LOGIN- ELI OHJELMISTOON KIRJAUTUMIS- JA WEBKAPU- SOVELLUKSEN MALLINNUS	37

4.1 Korkean tason toiminnallisuus: Tarjouslaskijan ja tehtaan toiminnan käyttötapauskaavio	39
4.1.1 Asiakas pyytää tarjouslaskijalta tarjouksen ikkunoista ja ovista	40
4.1.2 WebKAPU:ssa tarjouslaskijan toiminnan käyttötapauskaavio.....	41
4.1.3 Tehtaan toiminnan, esimerkiksi Alavus Ikkunat Oy:n, käyttötapauskaavio	42
4.2 Järjestelmän dynaaminen käyttäytyminen	43
4.2.1 Login-sovelluksen eli kirjautumis-sovelluksen kuvaus sekvenssikaaviona	44
4.2.2 Tarjouslaskijan ja WebKAPU-sovelluksen kuvaus sekvenssikaaviona	46
4.2.3 Login-sovelluksen eli kirjautumissovelluksen kuvaus aktiviteettikaaviona.....	47
4.2.4 Tarjouslaskijan ja WebKAPU-sovelluksen kuvaus aktiviteettikaaviona.....	48
4.3 Järjestelmän staattinen ja dynaaminen rakenne.....	49
4.3.1 Gapassi-projektin rakenne kuvattuna luokkakaaviona	50
5 GAPASSI-PROJEKTIN TESTAUSSUUNNITELMA JA TESTAUS	51
5.1 Login-sovelluksen ja WebKAPU:n testitapausten valinta.....	52
5.2 Gapassi-projektin testitapaukset.....	55
5.2.1 Testitapaus 1: Tarjouslaskija kirjautuu sisään ohjelmaan	56
5.2.2 Testitapaus 2: Tarjouslaskija syöttää asiakkaan henkilötiedot ja toimitustiedot järjestelmään.....	57
5.2.3 Testitapaus 3: Tarjouslaskija syöttää pyydetyn tarjouksen tuotteiden tiedot järjestelmään	59
5.2.4 Testitapaus 4: Tarjouksen tulostaminen.....	60
5.2.5 Testitapaus 5: Opittavuus, tehokkuus, muistettavuus, virheettömyys, miellyttävyys ja hyödyllisyys	61
6 JOHTOPÄÄTÖKSET	65
6.1 Gapassi-ohjelmiston jatkokehitys	67
6.2 Gapassi-projektin tulevaisuus	67
LÄHTEET	68
LIITTEET	70

Käytetyt termit ja lyhenteet

Alavus Ikkunat Oy	Ikkunoita pääasiallisesti valmistava yritys Alavudella, Etelä-Pohjanmaalla.
FireBird	Interbase SQL -palvelimesta muokattu vapaa SQL-palvelinohjelmisto.
Galileo Oy	Vuonna 1989 perustettu ohjelmistoalan yritys, jonka tuotantoa ovat mm. ikkuna- ja oviteollisuuden Hiku-ohjelmisto, KAPU-ohjelmisto ja karttasovellukset.
Gapassi	Projektinimi KAPU-, WebKAPU- ja login-sovellukselle.
KAPU	Ikkuna- ja ovi-tarjouslaskenta ja tilaustenhallinta ohjelmisto, jota kehitetty jo vuodesta 1999 alkaen.
Login-sovellus	Toteuttaa käyttäjän tunnistamisen ja yhdistämisen oikeaan sovellukseen.
MS SQL	Microsoft SQL Server -palvelinohjelmisto.
PDF	Portable Document Format –tiedostomuoto joka on käyttöjärjestelmäriippumaton Adoben kehittämä formaatti.
Tarjouslaskenta	Lasketaan tarjous asiakkaan pyytämien tuotteiden mukaisesti.
UML	Unified Modeling Language eli UML-mallinnus eli oliomallinnustekniikka.

WebKAPU KAPU-ohjelman web-käyttöliittymän sisältävä versio.

(Raisio, J. 2008.)

(Fowler, M. & Kendal, S. 2002.)

(Alavus Ikkunat Oy, 2008.)

Kuvio- ja taulukkoluetelo

KUVIO 1. Alavus Ikkunat Oy:n tuotantoa	10
KUVIO 2. Alavus Ikkunat Oy:n tehtaalla Alavudella	10
KUVIO 3. Tyypillinen vesiputousmalli	13
KUVIO 4. Toiminnot KAPU-ohjelmistossa	15
KUVIO 5. WebKAPU:n mahdollisia käyttäjiä	18
KUVIO 6. Gapassi-projektin komponenttien tekninen kuvaus	23
KUVIO 7. Tilasiirtymäkaavio: Tarjouslaskija, WebKAPU ja tietokanta	26
KUVIO 8. Kirjautuminen WebKAPU-ohjelmaan login-sovelluksen kautta	27
KUVIO 9. Asiakas- ja toimitustietojen syöttäminen järjestelmään	28
KUVIO 10. Tarjous sisältää monta riviä eli erilaisia ikkunoita	30
KUVIO 11. Tarjouksen sisältämät tuoterivit	28
KUVIO 12. Tuotetietojen syöttäminen tarjoukselle	32
KUVIO 13. Tarjouksen rivin laskenta eli tarjouksen tuotteiden hinnoittelu	33
KUVIO 14. Tarjouksen tallennus tai tulostus PDF-dokumentista	34
KUVIO 15. Tarjous- ja tarjousrivien-taulu	35
KUVIO 16. Eri UML-kaaviotyypit kuvattuna ohjelmistokehitysprosessin etenemisjärjestyksessä	37
KUVIO 17. Kaavioiden luokittelu	39
KUVIO 18. Käyttötapauskaavio: Asiakkaan toiminta	40
KUVIO 19. Käyttötapauskaavio: Tarjouslaskijan toiminta	41
KUVIO 20. Käyttötapauskaavio: Tehtaan toiminta	42
KUVIO 21. Sekvenssikaavio: Tarjouslaskija kirjautuu järjestelmään, Login-sovelluksen toiminta	44
KUVIO 22. Sekvenssikaavio: Tarjouslaskijan WebKAPU-sovelluksen työnkulku	46
KUVIO 23. Login-sovelluksen kuvaus aktiviteettikaaviona	47
KUVIO 24. Aktiviteettikaavio: Tarjouslaskijan ja WebKAPU-sovelluksen kuvaus aktiviteettikaaviona	49
KUVIO 25. Luokkakaavio: Gapassi-projektin kuvaus luokkakaaviona	50
KUVIO 26. Testauksen V-malli	52
TAULUKKO 1. Testitapaus 1: Tarjouslaskija kirjautuu sisään ohjelmaan	56
TAULUKKO 2. Testitapaus 2: Tarjouslaskija syöttää asiakkaan henkilötiedot ja toimitusehdot järjestelmään	57
TAULUKKO 3. Testitapaus 3: Tarjouslaskija syöttää pyydetyn tarjouksen tuotteiden tiedot järjestelmään	59
TAULUKKO 4. Testitapaus 4: Tarjouksen tulostaminen	60
TAULUKKO 5. Testitapaus 5: Opittavuus, tehokkuus, muistettavuus, virheettömyys, miellyttävyys ja hyödyllisyys	63

1 JOHDANTO

Ohjelmistot voivat olla joko työasemille asennettavia ohjelmistoja tai internet-selaimella käytettäviä ohjelmistoja. Selainkäyttöiset ohjelmistot tuovat vapauden käyttää ohjelmistoja mistä tahansa käsin, jos vain tietokoneella on internet-yhteys. Selainkäyttöisyys helpottaa ohjelmiston käyttäjää siten, että hän voi tehdä työtä toimistolla, kotona tai vaikkapa matkatessaan ulkomailla. Selainkäyttöisyys helpottaa myös it-ympäristön ja järjestelmän ylläpitoa, sekä järjestelmän sisältäminen yrityskohtaisten tuotetietojen ylläpitoa.

1.1 Toimeksiantajan ja opinnäytetyön aiheen esittely

Galileo Oy yritys toimii kahden diplomi-insinöörin ja yhden insinöörin voimin. Galileo Oy on toteuttanut ohjelmistoräätälöintiä ja tietotekniikkakonsultointia jo vuodesta 1988 lähtien. Vuodesta 1998 lähtien Galileo Oy on pääasiallisesti toteuttanut ohjelmistoja ovi- ja ikkunateollisuudelle. Muita osaamisalueita ovat muun muassa karttasovellukset, tietoliikenne ja tietoturvakonsultointi. Galileo Oy toteuttaa myös eri yritysten ohjelmistojen jatkokehitystä, sekä toteuttaa tehdasautomaatioita ja integrointia pääjärjestelmiin. (Raisio, J. 2008.)

Galileo Oy:n yksi tärkeistä yhteistyökumppaneista on Alavus Ikkunat Oy, joka toimii Alavudella valmistuen ikkunoita ja ulko-ovia. Galileo Oy:n ja Alavus Ikkunat Oy:n yhteistyö on jatkunut jo usean vuoden ajan. Opinnäytetyössä esimerkkyri-tyksenä suurelta osin on Alavus Ikkunat Oy:tä. Kuviossa 1 on ikkuna- ja ovi-esimerkki Alavus Ikkunat Oy:n tuotannosta. Ikkuna on T-mallinen ja avattava MSE/A- ikkuna. Uloimmissa laseissa on puitejako ja oikean puoleinen osa on varustettu tuuletusheloituksella. Ulko-oven tyyppi on Fasadi. Oveissa ovilehti on lämpöeristetty ja pintakäsittely on kovapintainen HDF-kuitulevy. Pintalevyn alla on alumiinijäykiste. (Alavus Ikkunat Oy 2009.)



KUVIO 1. Alavus Ikkunat Oy:n tuotantoa: Puualumiini-ikkuna ja Fasadi-niminen ulko-ovi. (Alavus Ikkunat Oy 2009.)

Alavus Ikkunat Oy työllistää tällä hetkellä noin 80 työntekijää, mutta yritys on kasvamassa. Kuviossa 2 on kuva Alavus Ikkunat Oy:n tuotantotiloista, jotka sijaitsevat Alavudella. Kuviossa on kyseessä kokoonpanolinjalla ikkunan sisäpuiteen asennuksesta karmiin. Ikkuna malli on MSE/A. Ikkuna on avattava, 2-puitteinen ja 3-lasinen. (Paalanen, A. 2009.)



KUVIO 2. Alavus Ikkunat Oy:n tehtaalla Alavudella. (Alavus Ikkunat Oy 2009.)

Galileo Oy on toteuttanut ohjelmiston nimeltä KAPU. Se on ollut käytössä muun muassa Alavuden Puunjalostustehtaalla jo ennen 2000-lukua. KAPU on työasemille erikseen asennettava ohjelmisto, joka on toteutettu ikkuna- ja oviteollisuuden käyttöön. Vuonna 2006 Galileo Oy aloitti Gapassi-projektin, jossa on tarkoitus kehittää edelleen aiemmin toteutettua KAPU-ohjelmistoa. Gapassi-projektissa on tavoitteena toteuttaa KAPU-ohjelmistosta web-selaimella käytettävä ohjelmisto, joka on nimeltään WebKAPU. Gapassi-projekti sisältää myös ohjelmaan kirjautumisen eli tietoturvallisen login-sovelluksen toteutuksen.

1.2 Rajaukset opinnäytetyössä

Opinnäytetyön kehittämishanke on Gapassi-projektin dokumentointi ja ohjelmiston testaus. Aloituvaiheessa ensimmäinen beta-versio WebKAPU:sta oli otettu testi-käyttöön Alavus Ikkunat Oy:llä. Gapassi-ohjelmistokehitysprojektissa kirjallinen määrittely on alusta alkaen jäänyt vähäiseksi Galileo Oy:llä. Ohjelmoijat keskittyivät lähinnä varsinaisen ohjelmointityön tekemiseen. Tämän vuoksi opinnäytetyön tärkeä rooli on toteuttaa dokumentointia tulevaa ohjelmiston kehitystä, ohjelmiston käyttäjiä ja uusia asiakkaita varten.

Gapassi-projektin edetessä projektissa oli eri vaihteita, kuten ohjelmiston testaus, ohjeistuksen tuottaminen loppukäyttäjille (Liite 3) ja markkinointi- ja messumateriaalin valmistaminen.

Opinnäytetyössä käsitellään Gapassi-projektin vaatimusmäärittely, suunnittelu, projektin teknisen toteutuksen kuvaukset sekä mallinnus. Opinnäytetyössä tehdään myös järjestelmän testaus ja sen myötä syntyy testaustulos-dokumentti käytettäväksi ohjelmoijille. Galileo Oy saa opinnäytteen myötä materiaalia, jota yritys voi käyttää tulevaa kehitystyötä ajatellen. Dokumentointia voidaan käyttää esimerkiksi ohjelmiston jatkokehityksessä sekä tuotettaessa dokumentteja asiakkaille. Dokumentointia voidaan käyttää hyväksi myös tilanteessa, jos Galileo Oy laajentaa ohjelmiston kehitystä tuotannon eri aloille. Dokumentointi voi olla apuna myös

tilanteissa, joissa uusia ohjelmoijia tulee tekemään työtä tämän projektin kanssa. Gapassi-projekti on vasta alkuvaiheessa ja suuri tavoite on työstää ohjelmistoa käytettäväksi paljon laajemmin eri teollisuuden aloilla. Ensimmäinen askel on kuitenkin varmistaa Gapassi-projektin toimivuus ikkuna- ja ovituotannon aloilla.

Dokumentointia Gapassi-projektista tähän mennessä on Jukka Raision tekemät hankehakemus-dokumentti (Liite 1) sekä pelkistetty tekninen toteutusluonnos (Liite 2). Nämä dokumentit ovat opinnäytetyön liitteinä.

Vaatimusmäärittely on opinnäytetyössä johdanto ohjelmiston vaatimuksiin, jossa paneudutaan myös hankkeen tarpeellisuuteen ja taustoihin. Vaatimusmäärittelyssä asetetaan ohjelmiston vaatimukset. Tämä kattaa myös asiakasvaatimukset eli millaista järjestelmää halutaan katsottuna asiakkaan näkökulmasta. Asiakkaan näkökulma on sinänsä jo tiedossa käytössä pitkään olleen KAPU-ohjelmiston kautta.

Suunnittelussa käydään läpi login-sovelluksen ja WebKAPU-ohjelmiston toiminta ensin tekniseltä kannalta ja sen jälkeen käyttäjänäkökulmasta katsottuna. Teknisellä toteutusluonnoskaaviolla havainnollistetaan ohjelmiston taustalla toimivat tekniset komponentit. Käyttäjänäkökulma käydään läpi ohjelmistosta otettujen näyttökuvien avulla. Näyttökuvia avataan tekstein ja esimerkkien avulla.

Mallinnus-osiossa lähdetään liikkeelle peruskäyttötapauskuvauksista. Niistä edetään tarkempiin kuvauksiin käyttäen mallinnuksen eri kuvaustekniikoita.

Testausvaiheessa laaditaan testaussuunnitelma ennalta suunniteltujen ja testattaessa esiin tulevien testitapausten kautta. Testausta toteutettaessa, testitulokset kirjataan Excel-taulukkoon. Testitapaukset ja testauksen tulokset käydään läpi ohjelmoijan kanssa, siten ettei mikään kohta jää epäselväksi.

Opinnäytetyötä viedään eteenpäin case-tutkimuksen tapaan eli tapaustutkimus tyyppisesti. Jokaisen aihealueen alussa kerrotaan kunkin aihealueen teoriasta,

johon työtä peilataan. Opinnäytetyössä käytetään muun muassa opinnäytetyöhön sopivalla ja sovelletulla tavalla apuna vesiputousmallia, jonka mukaisesti työ etenee rajatuilla alueilla. Kuviossa 3 on esitetty Haikalan ja Märijärven (2002,3) vesiputousmalli.



KUVIO 3. Tyypillinen vesiputousmalli. (Haikala & Märijärvi 2002, 3.)

Ylimpänä vesiputousmallissa on esitutkimus, joka opinnäytetyössä on johdantokappale. Tästä edetään vaatimusmäärittelyyn ja suunnitteluun. Toteutusvaihetta ei sinänsä käsitellä, sillä Gapassi-projektissa ohjelmiston toteutuksen on jo suorittanut Galileo Oy. Toteutuksesta jatketaan ohjelmistotestaukseen, testaustulosten kirjaamiseen ja tulosten eteenpäin saattamiseen.

2 GAPASSI-PROJEKTIN VAATIMUSMÄÄRITTELY

Projektin tarpeellisuus ja toteuttamiskelpoisuus kartoitetaan vaatimusmäärittelyssä, sekä siinä asetetaan tavoitteet ja vaatimukset (Haikala & Märijärvi 2002, 78). Määrittely jaetaan usein kahteen osaan, jotka ovat asiakasvaatimusmäärittely ja toteutettavan järjestelmän määrittely eli ohjelmistovaatimusten määrittely. Opinäytetyössä keskitytään lähinnä toteutettavan järjestelmän määrittelyyn, sillä asiakasvaatimusmäärittely on oleellisesti toteutettu jo aikaisemmassa ohjelmistossa eli KAPU -versiossa. Toki web-versio asettaa asiakasnäkökulmasta tiettyjä tarpeita, tavoitteita ja vaatimuksia, mutta ne ovat suurimmalta osin jo tiedossa.

Vaatimusmäärittely voi sisältää koko järjestelmän määrittelyn, ohjelmiston määrittelyn tai jonkin ohjelmiston osan määrittelyn. Tämän työn vaatimusmäärittelyssä käsitellään järjestelmän tekniset vaatimukset, ohjelmiston perusvaatimukset, sekä ohjelmiston muutamien osien tarkemman vaatimusmäärittelyn. Nämä osat ovat ohjelmaan sisään kirjautuminen eli login-sovellus, asiakas- ja toimitustietojen syöttäminen, sekä tarjouksen laskeminen. Näitä kolmea käyttötapausta tarkastellaan tarkemmin vaatimusmäärittely- sekä mallinnusosiossa.

Timo Lehtimäki (2006,13) kirjassaan ”Ohjelmistoprojektit käytännössä” kuvaa hyvin ohjelmistoprojektin määrittelyä. Kysyttäessä esimerkiksi kokeneilta projektipäälliköiltä, mikä on vaikeinta ohjelmistoprojekteissa, ei yleensä puhuta teknisistä ongelmista. Tekniset ongelmat ovat teknisten henkilöiden parasta osaamista ja niistä on totuttu selviämään. Sen sijaan usein nousee esille se, että on vaikeaa saada selville se, miten järjestelmän tulee toimia. Järjestelmän tulee toimia organisaatiossa samalla tavalla tai jopa paljon paremmalla tavalla. Järjestelmän toimivuuden vaatimukset voi määrittellä henkilö, joka on työskennellyt organisaatiossa useita vuosia. Projektipäällikön yhdessä projektiryhmän kanssa tulee osata tulkita koko tämä toiminta ja tuottaa toiminnot järjestelmällä. Se tuntuu varmasti vaikealta, tässä tulee pystyä yhdistämään projekti ja käyttäjän ajatukset toimivaksi kokonaisuudeksi. (Timo Lehtimäki 2006,13.)

2.1 Taustaa Gapassi-projektille ja sen ohjelmistoille KAPU ja WebKAPU

Galileo Oy on toteuttanut työasemaversioin tarjous- ja tilausten käsittelyohjelmasta Alavus Ikkunat Oy:lle. Ohjelmisto kattaa tarjouslaskennan ovi- ja ikkunatuotannosta, mukaan lukien tarjous- ja tilausvahvistukset, työsuunnitelut, oston ja alihankinnan, lähetystoiminnot, laskutulosteet ja raportoinnit (kuvio 4). Tämän ohjelmiston nimi on KAPU. Tämä Windows-sovellus asennetaan paikallisesti ja erikseen jokaisen käyttäjän koneelle. Taustalla pyörii tarpeen mukaan MS SQL- tai Firebird-tietokannat. (Raisio 2008.)



KUVIO 4. Toiminnot KAPU-ohjelmistossa.

KAPU-sovelluksesta on tarkoitus tehdä selainpohjainen web-versio eli WebKAPU. Pilottiyrityksenä toimii niin ikään Alavus Ikkunat Oy. Ohjelmalla on jo myös muutama muu käyttäjäyritys. Ohjelmaa voisivat käyttää yrityksen eri tarjouslaskijat ympäri Suomen, niin Alavuden tehtaalla kuin esimerkiksi jälleenmyyjät eri kaupungeissa. Ovi- ja ikkunatuotteiden valmistajia on Suomessa muutamia kymmeniä, mutta heidän allaan toimii satoja jälleenmyyjä, jotka voisivat hyödyntää sovellusta. Näin ollen käyttäjäkunta saattaisi parhaimmillaan olla varsin laaja.

WebKAPU:n kehittämiseen liittyy olennaisena osana myös login-sovelluksen eli kirjautumis-sovelluksen tietoturvallinen toteuttaminen. WebKAPU:n ja login-sovelluksen toteutushanke kulkee nimellä Gapassi.

Vastaavia web-sovelluksia on ikkuna- ja ovituotantomarkkinoilla muutamia. Esimerkkeinä mainittakoon markkinoilla olevat yritykset Joiner Oy ja DB-Manager Oy. Ne tarjoavat samankaltaista web-käyttöliittymätuotetta. Muita vastaavia ohjelmistojä löytyy yrityksiltä Vertex Systems Oy ja AtBusiness Oy. Galileo Oy on päässyt sen verran tutustumaan kilpailijoiden vastaaviin ohjelmistoihin, että tietää etteivät markkinoilla olevat ohjelmistot ole niin kattavia ja monipuolisia eivätkä niin hyvin räätälöitävissä kuin WebKAPU-sovellus. On tietenkin selvää, että eri yritysten ohjelmistojä on hankala päästä testaamaan sillä yritykset ovat kilpailijoita Galileo Oy:lle. Lisäksi ohjelmistojen käyttäjäyritykset ovat kilpailijoita Galileo Oy:n tärkeimmälle asiakkaalle eli Alavus Ikkunat Oy:lle. Omien käyttäjien kautta on saatu palautetta siitä, että KAPU-ohjelmisto on kilpailijoihin verrattuna selkeäkäyttöinen sekä käytössä nopea. Nämä kommentit tietysti perustuvat peruskäyttäjän kokemuksiin. Jukka Raisio kommentoi itse ohjelmistoaan seuraavasti: ”KAPU:n parmmuuteen, verrattuna muiden yritysten tuotteisiin, väitän syyksi rakennetta, jossa web-ohjelma on Windows-ohjelma. Ohjelman käyttöliittymästä on tuotu web-käyttäjälle rajattu näkymä, jonka toiminnallisuus on täsmälleen sama kuin työpöytä-versionkin. Kaikki työpöytäohjelman kehitystyö on suoraan hyödynnettävissä web-ohjelmassa. Kilpailijoilla web-laskentaohjelmat ovat erillisiä ohjelmistoprojekteja.” (Raisio 2008.)

2.2 Gapassi-projektin tarkoitus ja tavoitteet

Gapassi-projektin tavoitteena on työasemasovelluksen muuttaminen selainkäyttöiseksi web-versioksi. Näin muun muassa ohjelmiston erillinen päivitystarve poistuu. Tavoitteena on luoda ohjelmistokokonaisuus, joka rakentuu login- eli kirjautumis-sovelluksesta ja WebKAPU-sovelluksesta. Tavoitteena on rakentaa olemassa olevalle Windows-sovellukselle web-käyttöliittymä. Kun ohjelmisto on saatu toimi-

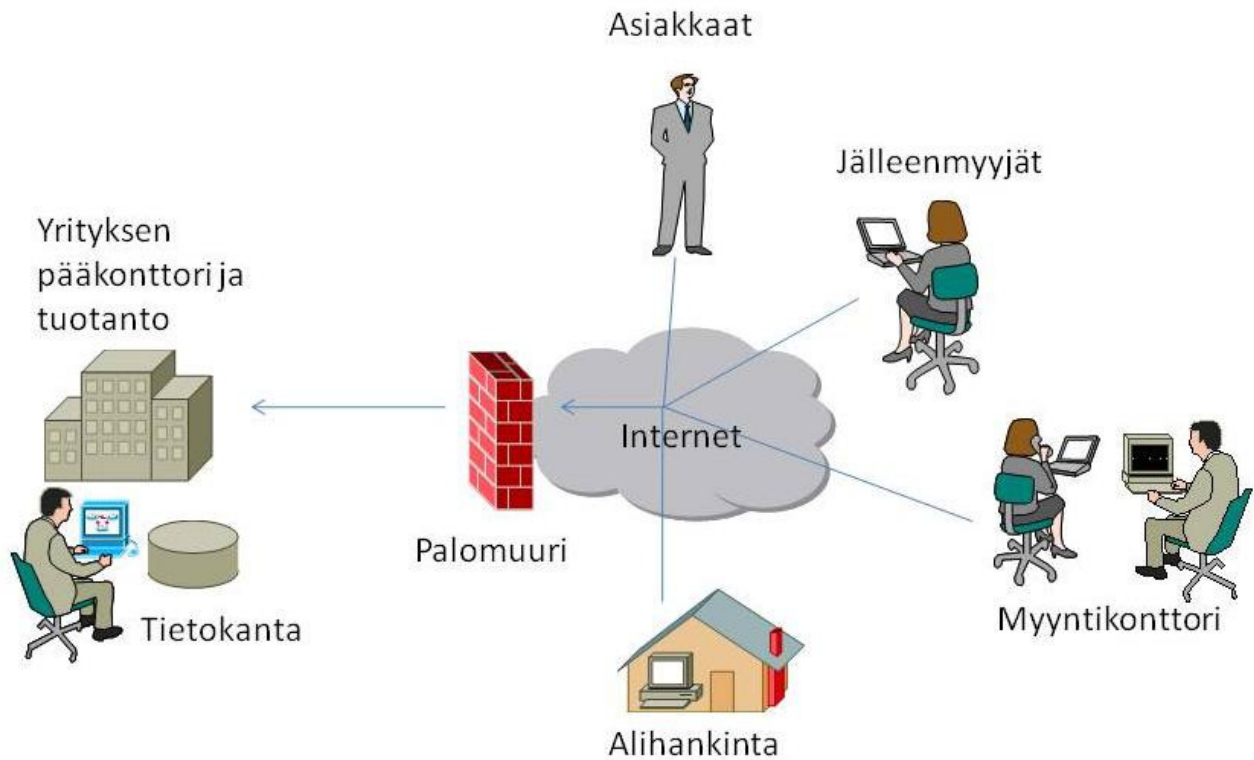
maan ikkuna- ja oviteollisuuden alalla, on projektin tavoitteena sovellukset muunneltavuus useille teollisuuden tahoille saman toimintamallin kautta.

Tavoitteena on myös luoda tietoturvallinen login-ympäristö, jolla voidaan tarjota samassa tietokannassa usealle myyjälle oma näkymä, kunkin käyttäjän tarpeiden mukaisesti. Esimerkiksi ovia myyvillä alihankkijoilla on vain ovitietokannan vaihtoehdot käyttöliittymässään. Tehtaalle voidaan tarjota hallinnointimahdollisuus käyttöliittymälle: toteutetaan esimerkiksi mahdollisuus siirtää tilausvahvistukset suoraan tehtaalle tuotantokantaan, jolloin vältetään eri välikäsien toiminnalta ja nopeutetaan prosesseja huomattavasti. Tuotteiden ja valikoiman muutokset päivittyvät kantaan joka kerta kun web-yhteys muodostuu. Toteutus ei vaadi lisenssejä kolmansilta osapuolilta.

Ohjelmistolla voi olla useita erilaisia tahoja käyttäjinään:

- Tehtaiden eri tehtävissä toimivat työntekijät: toimihenkilöt ja työtekijät tuotannossa
- Ympäri Suomen etätyöpisteissä toimivat myyjät ja muut henkilöt, esimerkkinä Alavus Ikkunat Oy:n myyjät Helsingin toimipisteessä
- Alihankkijat ympäri Suomen, esimerkiksi K-Rauta-kaupoissa tarjouksia laskevat henkilöt
- Ulkomaan toimipisteissä työskentelevät henkilöt
- Ulkomailla toimivat jälleenmyyjät, erityisesti Virossa ja Venäjällä (englannin kielellä)

Kuviossa 5. on kuvattu ohjelmiston ylläpitäjä (esimerkiksi tehdas) ja mahdolliset käyttäjäryhmät.



KUVIO 5. WebKAPU:n mahdollisia käyttäjiä.

2.2.1 Ohjelmiston suorituskyky ja tehokkuus

Ohjelmiston ei tule vaatia erityisiä suorituskykyominaisuuksia käytettävissä olevalta laitteistolta tai internet-yhteyden nopeudelta. Tämä siksi, että käyttäjiä saattaa olla missä tahansa yrityksessä ja missä vain Suomessa. Käyttäjillä on erilaisia IT-ympäristöjä ja tietokoneita. Internet-yhteys saattaa olla kiinteä, langaton tai muu yhteysmuoto, nopeudet saattavat vaihdella huomattavastikin. Tarkoitus on toteuttaa ohjelmisto, jonka käyttämiseen ei tarvita muuta kuin toimiva tietokone, selain ja internet-yhteys.

Ikkuna- ja ovitarjous koostuu riveistä. Rivi tarkoittaa yhdentyypistä tuotetta esimerkiksi pyöreää valkoista ikkunaa. Tarjouksessa saattaa olla jopa 200 erilaista tuotetta eli 200 erilaista riviä. Monen rivin hinnoittelulaskenta kestää eniten aikaa, koska rivikohtaisesti tallennetaan kantaan noin 200 riviä tietoja osista ja työtiedoista. Ominaisuuksia löytyy joka riville useita (lasipaketti, helat, värit jne), jopa sata. Näin esimerkiksi 200 riviä sisältävään tarjoukseen saattaa sisältyä jopa 40 000 eri osaa ja työtietoa. Laskenta ja tietokannan suorituskyky osoittautuu ohjelmiston kannalta kriittisimmäksi: miten nopeasti tietojen haku, käsittely ja tallennus onnistuvat tietokantaan.

2.2.2 Ohjelmiston käytettävyys, toipuminen ja tietoturvallisuus

Käytettävyys on erittäin tärkeä osa-alue ohjelmistoprojektissa. WebKAPU-ohjelmiston toteutuksessa kiinnitetään paljon huomiota helppokäyttöisyyteen, siten että käytettävyys olisi mahdollisimman hyvä. Tärkeässä roolissa on myös tarjouslaskentaosion nopea käyttäminen. Esimerkiksi rautakaupassa asiakas saattaa jäädä odottamaan että myyjä tekee hänelle tarjouksen paperille saman tien. Kun asiakas on saanut tarjouksen, hän mahdollisesti pyytää jotain muutoksia, tarjous korjataan ja tulostetaan uudelleen. Asiakasta tulee palvella nopeasti ja laadukkaasti. Tarjoustulosten tulee olla helpposelkoinen ja kattava. Ohjelmiston helppokäyttöisyydestä huolimatta toimintojen tulee kuitenkin olla kattavat. Ohjelmiston ulkonäkö ei ole tärkein asia, vaan helpposelkoinen käyttöliittymä. Ohjelmiston on oltava helppokäyttöinen myös sen vuoksi että käyttäjiä on ympäri Suomea ja jopa ympäri ulkomaita. Käyttäjille ei ole aina mahdollista järjestää koulutusta, siksi ohjelman tulee olla helppokäyttöinen, lisäksi ohjeistuksen tulee olla kattava. Jos kyseessä ulkomaalainen ohjelmiston käyttäjä, käytettävyyttä parantaa mahdollisuus kielivalintaan. Ohjelmistossa tulee olla kielivalinta ainakin englannin kieleen. Mahdollisesti tulisi olla valittavina myös ruotsin ja venäjän kieli.

Toipumisella tarkoitetaan sitä, miten ohjelmisto saadaan jälkeen toimimaan erilaisien virhetilanteiden jälkeen. Jos verkkoyhteys jostain syystä katkeaa tai ohjelmisto

kaatuu, tulee selain käynnistää uudelleen. Toipuminen tapahtuu suhteellisen yksinkertaisella toimenpiteellä. Toinen asia on, milloin käyttäjä on viimeksi tehnyt tallennuksen, eli miten paljon dataa ohjelma on ehtinyt tallentaa tietokantaan. Ongelmatilanteita ajatellen voidaan kyseenalaistaa seuraavia asioita: Onko käytössä jotain toista internet-yhteyttä ohjelmiston käyttämiseen? Mitä tapahtuu jos ohjelmiston käyttämä tietokanta vioittuu, onko varmuuskopiota olemassa? Miten nopeasti tietokanta saadaan palautettua varmuuskopiosta jälleen käytettäväksi? Mitä jos laitteisto hajoaa, esimerkiksi ohjelmistopalvelin tai kantapalvelin? Miten käy jos yrityksen internet-yhteys katkeaa, voidaanko WebKAPU:n kautta tehdä mitään? Olisiko yrityksellä hyvä olla varalla 3G:n kautta muodostettava yhteys tai ADSL-yhteys? Tulisiko toimiston kiinteä verkkoyhteys varmentaa käyttäen kahta eri operaattoria?

Ohjelmistojen tietoturvasuus on yksi tärkeimmistä asioista toteutettaessa ohjelmistoa. Ohjelmiston kehittäjien tulee olla varmoja siitä, että kohdeyrityksen on turvallista käyttää ohjelmistoa. On oltava varmoja ettei yrityksen tietoja pääse kukaan ulkopuolinen saamaan. Gapassi-projektissa tärkeä osuus on Login-sovelluksen toteuttaminen tietoturvasesti. Tietomurtoja ei saa päästä tapahtumaan. Vaikka tietoturva olisi ohjelmiston kehittäjien toimesta varmistettu, turvallisuusriskejä saattavat aiheuttaa erilaiset internetin haittasovellukset sekä virukset. Ohjelmiston käyttäjiä käyttäjäyrityksellä on myös vastuu siitä, että he pitävät järjestelmien ohjelmien versiot ja päivitykset tietoturvasella tasolla. Etenkin selainten versiot ja päivitykset tulevat olla ajan tasalla. Tärkeää on myös se että virustorjuntaohjelmit ovat ajan tasalla.

2.2.3 Ohjelmiston ylläpidettävyys ja operointi

WebKAPU-ohjelmisto on ylläpidettävissä joko Galileo Oy:n toimesta ja yrityksen itsensä ylläpidettävissä. Tämän voi kukin yritys itse määritellä, haluavatko ylläpidon vai ei. Useimmissa tapauksissa Galileo Oy hoitaa kaikki ohjelmiston ylläpitoasiat ja päivitykset. Ohjelmiston käyttäjäyrityksellä tulee olla perusmahdollisuudet ohjelmiston operointiin, kuten käyttäjätunnusten ja käyttöoikeuksien hallintaan.

Jos palvelimet ja tietokannat ovat Galileo Oy:n hallinnassa, yksi palomuurin takana oleva Windows-tietokone riittää asiakkaalle. Kaikki operointi voidaan toteuttaa Galileo Oy:n toimesta etäyhteydellä. Tällöin otetaan halutulla etäyhteysohjelmalla käyttäjän työpöytä hallintaan käyttäjän tämän salliessa. Näin ollen asiakasyritys voi olla missä päin vain Suomea tai maailmaa.

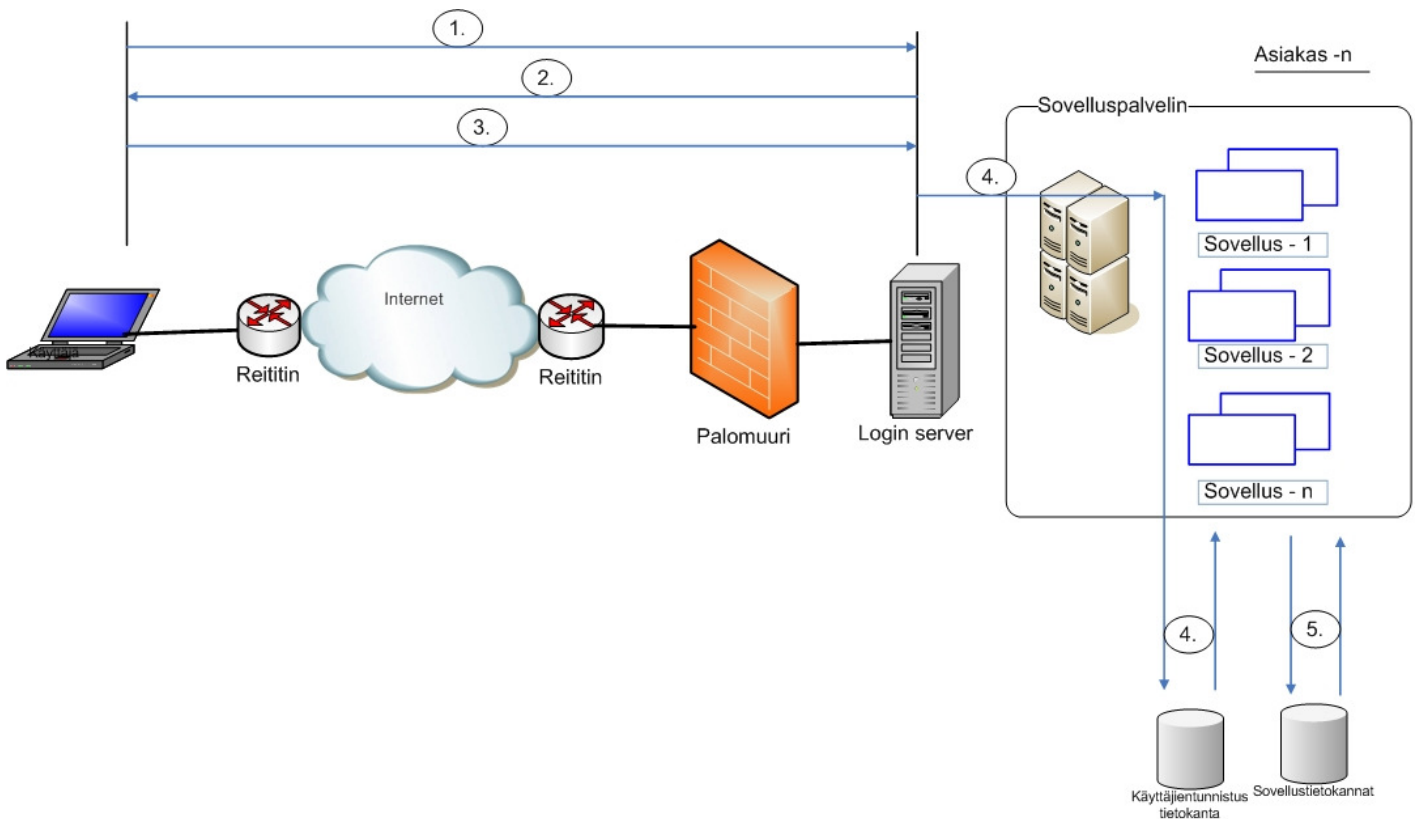
3 GAPASSI-PROJEKTIN SUUNNITTELU

Opinnäytetyössä suunnittelu kattaa Gapassi-projektin teknisen ympäristön kuvaamisen teknisen toteutusluonnoskuvan avulla. Tekninen toteutus sisältää sekä login-sovelluksen, että WebKAPU-ohjelmiston kuvauksen laitteistotasolla.

Toisena osana suunnittelua käydään läpi login-sovelluksen ja WebKAPU-ohjelmiston toimintaa käyttäjänäkökulmasta. Tarkastellaan vaihe vaiheelta miten käyttäjä etenee käyttäessään Gapassiin kuuluvia osa-alueita eli login-sovellusta ja itse sovellusta. Sovelluksesta katselmoidaan tarjouslaskenta-osio siten, että toteutetaan esimerkki tarjousta. Ohjelmiston toimintaa ja ulkonäköä havainnollistetaan ohjelmistosta otetuilla näyttökuvilla, joita avataan selitysten ja esimerkkien avulla.

3.1 Tekninen toteutusluonnos

Gapassi-projekti muodostuu kuvion kuudesta komponenteista. Kuviossa 6 Kannettavan tietokoneen kuva vasemmalla tarkoittaa käyttäjää eli esimerkiksi tarjouslaskijaa. Käyttäjä ottaa yhteyttä reitittimien, internetin ja palomuurin kautta. Tämän jälkeen saavutetaan login-sovellus eli jatketaan tunnistautumisen kautta WebKAPU-sovellukseen. Onnistuneen tunnistautumisen jälkeen saadaan yhteys valitulle sovelluspalvelimelle. Tämän jälkeen valitaan oikea sovellus ja saavutetaan yhteys oikeaan tietokantaan. Jukka Raisio on laatinut teknisen toteutusluonnoskuvan projektin alkuvaiheessa (Liite 2). Opinnäytetyössä tekninen toteutus (kuvio 6) piirretään auki hieman tarkemmalla tasolla ja täsmennetään tekstien avulla.



KUVIO 6. Gapassi-projektin komponenttien tekninen kuvaus.

Seuraavassa on kuvion 6 tekninen toteutus esitettyä sanallisessa muodossa:

1. Asiakas avaa http-yhteyden kirjoittamalla selaimen URL-kenttään <http://kapu.galileo.fi>
2. Login-palvelin lähettää asiakkaalle login- eli tunnistautumis-ikkunan, johon asiakas antaa käyttäjätunnuksen ja salasanan.
3. Asiakas kirjoittaa käyttäjätunnuksen ja salasanan.
4. Login-palvelin tarkistaa saatujen tunnusten oikeellisuuden käyttäjätietokannasta, joka määräytyy URL-osoitekentässä olevan asiakasosan mukaan (tässä esimerkissä kapu eli <http://kapu.galileo.fi> eli yrityksen Alavus Ikkunat Oy:n tunnus)

5. Jos tunnistautuminen on OK, asiakas saa yhteyden sovelluspalvelimeen haluamaansa sovellukseen. Sovellusta käytettäessä sovellus keskustelee tietokantapalvelimen kanssa. Asiakkaalla ei ole yhteyttä tietokantaan, ainoastaan sovellukseen, joka käyttää tietokantaa.

Login- ja Web-sovellus on käytettävissä missä tahansa, jos käytössä on tietokone, selain ja internet-yhteys. Alustavasti on testattu, että muun muassa Symbian- ja Linux-pohjaiset selaimet toimivat.

Tietokannat ovat FireBird-muodossa ja ovat siirrettävissä tiedostona ja "kanta-backup"-muodossa. Kanta voi olla myös Linux-palvelimella. Ohjelmisto on muuten aina Windows-ympäristössä.

3.2 WebKAPU-ohjelmiston toiminta käyttöliittymässä

WebKAPU-ohjelmisto koostuu seuraavista toiminnoista:

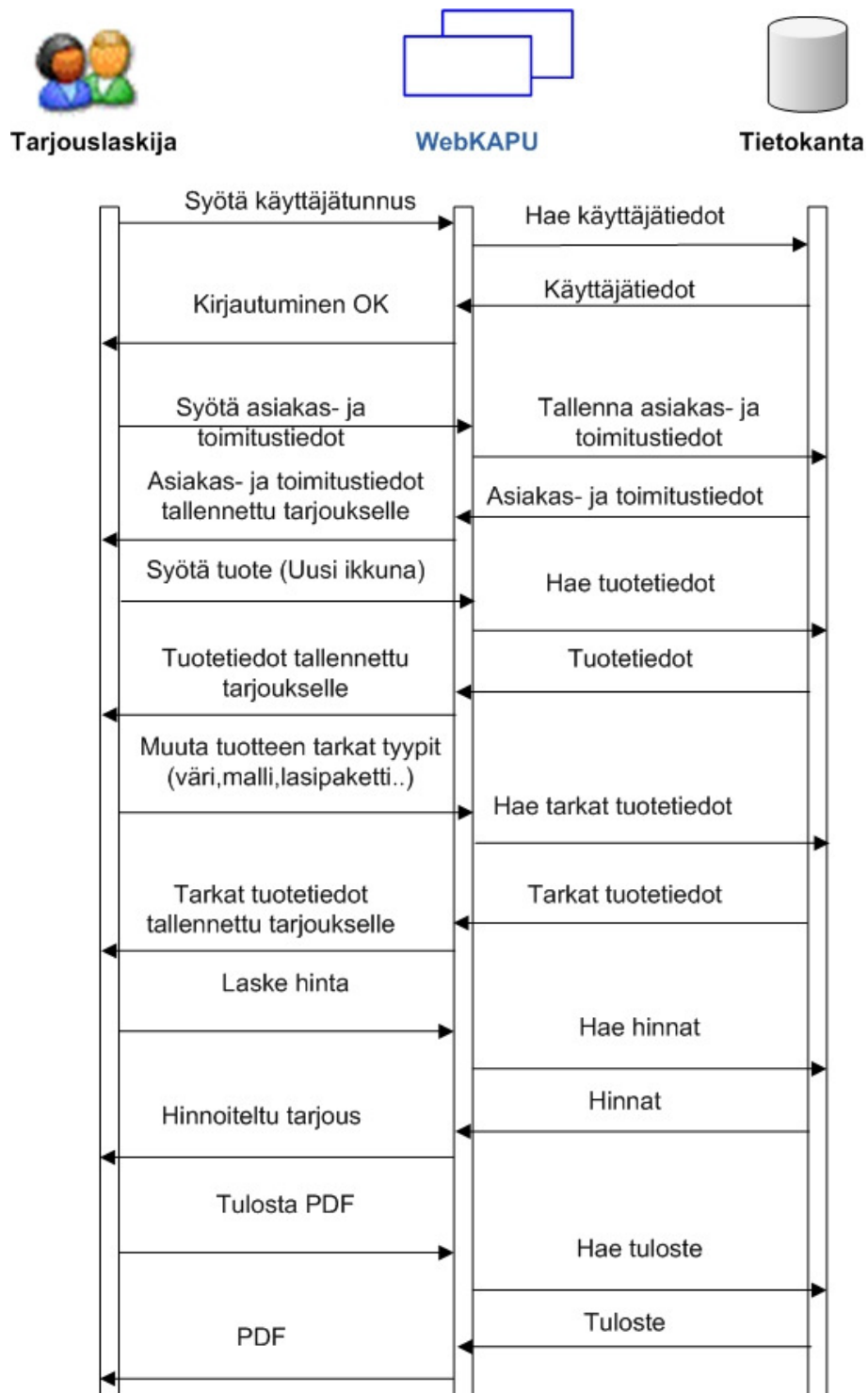
- Login-sovellus
- Henkilö- ja toimitustietojen syöttäminen tietokantaan
- Tarjouksen tietojen syöttäminen tietokantaan
- Laskenta eli ohjelma laskee tarjoukselle hinnan
- Tarjouksen tulostaminen
- Tilausvahvistuksen tekeminen
- Hinnoittelun tarkennus
- Käyttäjätietojen ylläpito: uusien käyttäjätunnusten luonti ja salasanojen vaihtaminen

- Tietojen siirto: WebKAPU käyttää "extranet-web-kantaa", yrityksen tuotantodata on "master-kannassa" eli tiedon siirto näiden kahden eri kannan välillä tapahtuu tekstimuodossa.

Opinnäytetyössä keskitytään kolmeen ensimmäiseen toimintoon eli käyttötapaukseen ja ne ovat: Login-sovellus, henkilö- ja toimitustietojen syöttäminen tietokantaan ja tarjouksen tietojen syöttäminen tietokantaan.

Login-sovelluksessa käyttäjä, esimerkiksi tarjouslaskija, kirjautuu oikeaan järjestelmään sisään. Oikea sovellus määräytyy URL-osoitteen ja käyttäjätunnusten mukaan. Kun asiakas on pyytänyt tarjouksen ikkunoista ja ovista, tarjouslaskija aloittaa tarjouksen laskemisen henkilö- ja toimitustietojen syöttämisestä. Näiden tietojen syöttämisen jälkeen aloitetaan tarjouksen syöttäminen järjestelmään. Kolmantena vaiheena syötetään pyydetyt tuotteet tarjoukselle.

Kuviossa 7 on edellä mainittujen kolmen käyttötapauksen tilasiirtymäkaavio. Toimijoina kaaviossa ovat ohjelmiston käyttäjä eli tarjouslaskija, WebKAPU-sovellus sekä tietokanta.



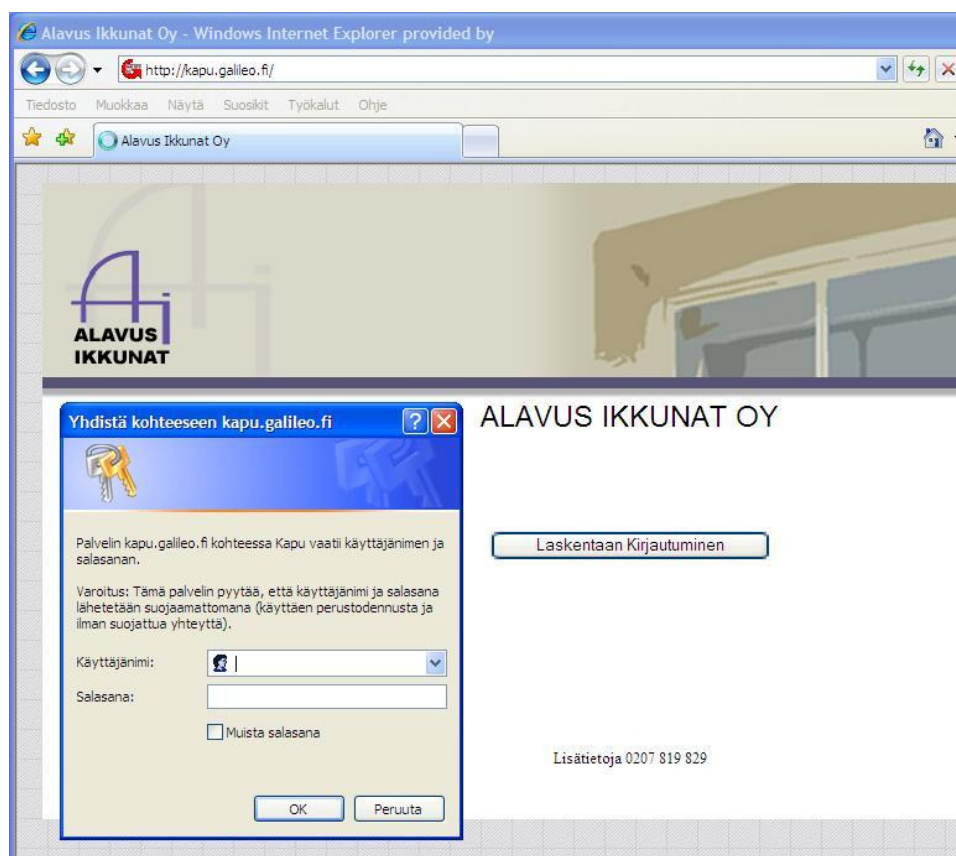
KUVIO 7. Tilasiirtymäkaavio: Tarjouslaskija, WebKAPU ja tietokanta.

3.3 Login-sovelluksen ja WebKAPU:n käyttöliittymät

Seuraavissa kappaleissa käydään läpi login- ja WebKAPU-sovelluksen käyttöliittymät. Kuvaruutukaappaukset ovat Alavus Ikkunat Oy:n ohjelmistosta. Alavus Ikkunat Oy:ltä saatiin testitunnukset heidän järjestelmäänsä ja lupa käyttää järjestelmää referenssinä muun muassa näyttökuvien muodossa.

3.3.1 Kirjautumisvaihe eli ohjelmiston käyttäjä kirjautuu sisään WebKAPU-sovellukseen login-sovelluksen kautta

Osoitteessa <http://kapu.galileo.fi/> kirjaututaan Alavus Ikkunat Oy:n tarjouslaskentaohjelmistoon login-sovelluksen avulla. Alavus Ikkunat Oy:n järjestelmän käyttäjätunnuksia hallinnoi henkilö heidän organisaatiossaan. Sivulla on puhelinnumero jonka kautta tunnuksia voi tiedustella.



KUVIO 8. Kirjautuminen WebKAPU-ohjelmaan login-sovelluksen kautta.

Kirjautuminen tapahtuu painamalla ”Laskentaan kirjautuminen”, jolloin aukeaa ikkuna ”Yhdistä kohteeseen kapu.galileo.fi”. Saatu käyttäjätunnus ja salasana kirjoitetaan niille varattuihin kenttiin. Tämän jälkeen valitaan ”OK”-painike ja tarjouslaskennan etusivu aukeaa eli sivu jolla näkyvät kaikki tarjoukset. URL-osoitteessa määritellään yritys, jonka ohjelmistoon ollaan kirjautumassa. Käyttäjätunnus ja salasana määrittelevät mihin sovellukseen ja näkymään käyttäjällä on oikeudet.

3.3.2 Asiakastietojen ja toimitustietojen syöttäminen järjestelmään

Asiakastietoja ei tallenneta tietokantaan siten että ne voitaisiin noutaa kannasta tarjousta tehdessä. Tämä johtuu siitä että varsin harvoin sama asiakas pyytää tarjousta ikkunoista tai ovista useaan kertaan. Tai jos pyytää niin harvoin esimerkiksi toimitusosoitetiedot ovat pysyneet ennallaan. Tästä syystä tietojen kantaan tallentamisesta ei ole hyötyä.

Kuvion 9 ikkunassa täytetään kaikki asiakkaaseen ja toimitukseen liittyvät tiedot, kuten toimitusehdot. Kaikkia tietoja ei ole pakko täyttää vielä tässä vaiheessa, vaan tietoja voidaan vielä täydentää ja tarkentaa, jos edetään tilaukseen saakka.

The screenshot shows a web browser window with the following content:

- Browser title: KAPU - Dokumenttiotsikko - Windows Internet Explorer provided by
- Address bar: http://kapu.galileo.fi/
- Page title: Paluu Riveille 54078/0 OTSIKKO TESTIKÄYTTÄJÄ Tulosta
- Form fields:
 - Merkki:
 - Yhteyshenkilö:
 - Asiakasnimi:
 - Firma:
 - Kieli: FIN
 - Osoite:
 - Osoite2:
 - Puh 2:
 - Posti:
 - Puh 1:
 - Fax:
 - Viite:
 - Y-tunnus:
 - Email:
 - Toimitusehto: FCA Vapaasti tehtaalla
 - Maksuehto: Sopimuksen mukaan
 - Toimitusaika: Sovitaan ennkseen Viikko: Vuosi:
 - Hinta: Rahni: Veroton Yht: Verollinen:
- Buttons: Oletiedot, Poista tarjous

KUVIO 9. Asiakas- ja toimitustietojen syöttäminen järjestelmään.

Ennen kuin ikkunassa edetään, on muistettava painaa Talleta-painiketta. Toiminto- ja tuottavat linkit ja painikkeet on sijoitettu kaikki sivun yläreunaan. Alareunassa on vain muutama painike, ja näitä painikkeita tarvitaan harvemmin. Alareunassa on esimerkiksi Poista tarjous -painike. Kyseinen painike on sijoitettu alareunaan sen vuoksi, ettei painettaisi vahingossa. Lisäksi kyseistä painiketta tarvitaan harvoin.

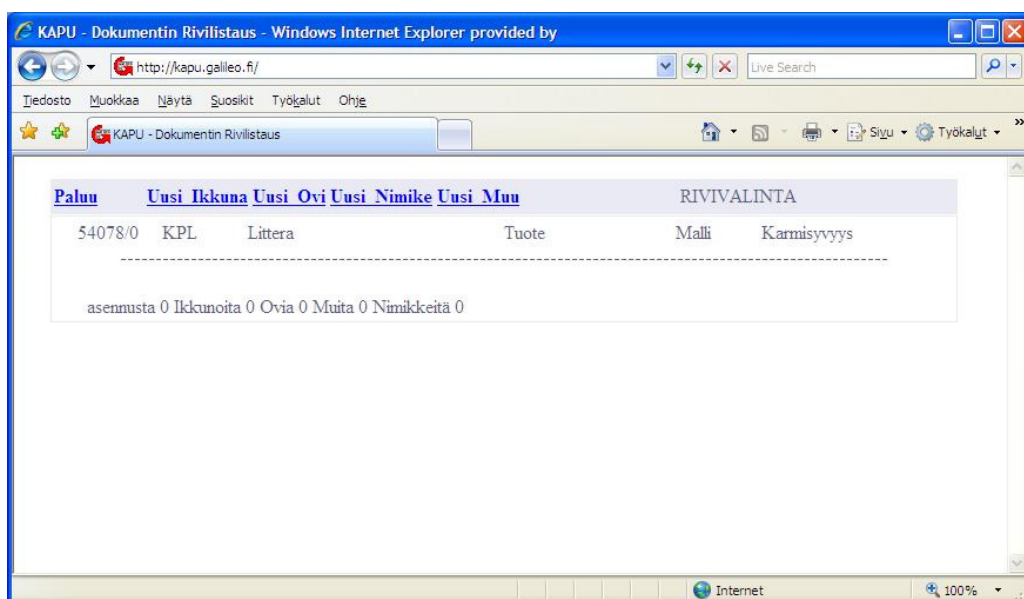
3.3.3 Tarjouksen rivit eli tarjoukseen syötetyt, asiakkaan pyytämät tuotteet riveinä

Asiakastietojen syöttämisen jälkeen avataan Riveille-ikkuna. Tällä näytöllä nähdään kaikki tarjouksen eri tuotteet eli tuoterivit. Jokaisen rivin taakse tallennetaan tuote. Yhdellä rivillä määritellään yhdenlainen tuote ja niiden tilausmäärä. Erilaiset tuotteet ovat siis kukin omalla rivillään. Rivinumerointi on toteutettu parillisina numeroina eli rivit etenevät numeroin 2, 4, 6, 8 ja niin edelleen. Numerointi on toteutettu näin siksi, että järjestelmä jättää varalle parittomat luvut niin, että niihin voidaan tarvittaessa jälkikäteen lisätä tuotteita. Tämä helpottaa sitä, että koko tarjousta ei tarvitse syöttää kokonaan uusiksi, jos huomataan että väliin tulee lisätä vielä ikkuna tai ovi. Isossa tarjouksessa voi olla esimerkiksi sata riviä, mikä tarkoittaa että tarjouksessa on 100 erilaista ikkuna- tai ovivariaatiota. Eroa erilaisissa riveissä eli tuotteissa ei tarvitse olla suuri, ero voi olla vaikkapa se että joihinkin ikkunoihin halutaan hyttyspuitteet tai vaikkapa kahva jolla ikkuna voidaan avata. Kuviossa 10 on kuvattu ikkunatarjouksen rakenne.



KUVIO 10. Tarjous sisältää monta riviä eli erilaisia ikkunoita.

Rivinäyttö-ikkunan kautta aloitetaan uusien tuotteiden syöttäminen joko ”Uusi Ikkuna”- tai ”Uusi Ovi” -linkin kautta.



KUVIO 11. Tarjouksen sisältämät tuoterivit.

Myös tässä näytössä (kuvio 11) on kaikki valikot sijoitettu yläreunaan. Jos tarjouksessa on esimerkiksi sata riviä erilaisia tuotteita, on esimerkiksi alareunaan sijoitettujen painikkeiden käyttäminen hidasta.

3.3.4 Oikean mallisten, väristen ja kokoisten ikkunoiden ja ovien syöttämisen järjestelmään

Rivi-näytöllä on tässä esimerkissä (kuvio 12) valittu linkki "Uusi Ikkuna", josta aukeaa kuviossa 12 esitetty ikkuna. Tässä näkymässä syötetään yksityiskohtaisesti kaikki ikkunan osat, kuten koko, värit, mallit, kätisyudet, kappalemäärät ja niin edelleen. Eri osien syöttäminen tapahtuu avaamalla valintalista ja valikoimalla sieltä oikea osa ikkunalle. Osa syöttökentistä on sellaisia, että niihin kirjoitetaan haluttu tieto, kuten leveys ja korkeus. Tieto tulee olla oikeassa muodossa, tai tietojen tallennus ei onnistu. Esimerkiksi leveys tulee olla senttimetreinä oikeassa sarakkeessa. Määrittäjiä on useita, joten tarjouslaskijan tulee olla tarkkana, jotta kaikki määrittäykset tulee huomioitua. Eri kenttien looginen sijainti ja liikkuminen kentissä tulee huomioida tarkasti kehitystyössä, sillä tarjouksen laskemisen tulee sujua jouhevasti.

"Identtinen"-painikkeen avulla voidaan luoda määritellystä ikkunasta identtinen eli toisin sanottuna kopio, jota on helppo tarvittaessa muokata. "Peilikuva"-painikkeen avulla voidaan luoda peilikuva samasta ikkunasta, eli tällöin esimerkiksi ikkunan kätisyys muuttuu päinvastaiseksi. Nämä toiminnot ovat erittäin hyviä käytännössä ja nopeuttavat tarjouksen syöttämistä. Tällä tavoin uuden ikkunan syöttämistä ei tarvitse aloittaa aivan alusta, vaan voidaan käyttää toisen ikkunan tietoja pohjatietona. Ikkunassa voi muuttua vaikka vain jokin pieni yksityiskohta ja muut tiedot pysyvät kuten kopioidussa ikkunassa.

The screenshot shows a web browser window titled "KAPU - Rivinäyttö - Windows Internet Explorer provided by". The address bar shows "http://kapu.galleo.fi/". The page content is a form for configuring a window. At the top, there are tabs: "Paluu", "Uusi Ikkuna", "Uusi Ovi", "Uusi Nimike", "Uusi Muu", and "Edellinen Seuraava". The current window is identified as "54078/0 RIVI 2".

The form includes several sections:

- Buttons:** Talleta, Rivin tekstit, LISÄT, Tarvikkeet, Oletustiedot, Identifinenc, Peilikuva.
- Basic Info:** Malli: A, Tuote: MSE/A_08, Tunnus: 12*12.
- Dimensions:** Karmis: 170, Kpl: 1, Ahv0%: 405.15, Leveys: 12, 1190, Korkeus: 12, 1190.
- Colors:** VäriRyhmä: Vakio, Sisäpintak: MA, SP+Kar sisäväri: RPV (valkoinen), Ulkopintak: MA, Karmin ulkoväri: RAL 9010.
- Options:** helaväri Sisä, helaväri Ulko, Sälekaihdin, Hyvtyyspuite.
- Prices:** Erikoislasi Sisä, Erikoislasi Sisä hinta: 0.00, Erikoislasi Ulko, Erikoislasi Ulko hinta: 0.00.
- Other:** Aukko 1 (checked), Karmi (checked), Sisäp. (checked), Ulkop. (checked), Sisajako, Ristikko, RIV, kiinnr. (checked), SP Muovisuoja, UP Muovisuoja.

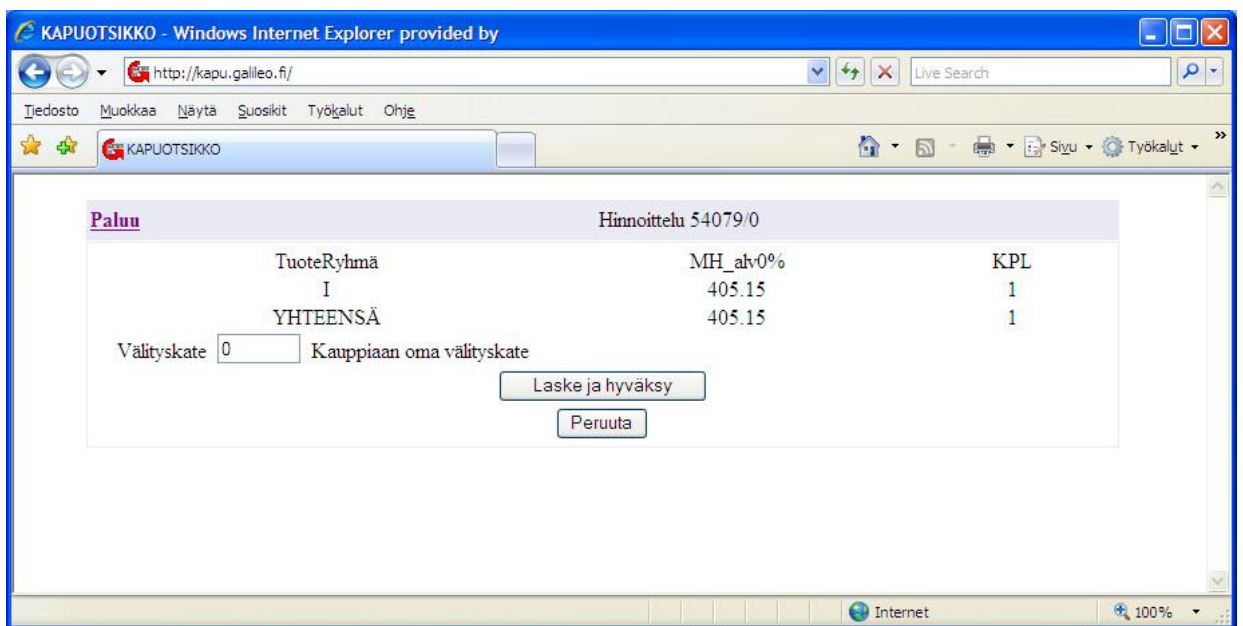
A technical drawing of a square window with a diagonal line is shown on the right. The drawing is labeled "MSE/A_08" and has dimensions "1190" for both width and height. A "Poista" button is located at the bottom right of the form.

KUVIO 12. Tuotetietojen syöttäminen tarjoukselle.

Kuvio 12 ikkunan linkkien ja painikkeiden asettelussa noudatetaan samaa sijoittelua kuin aiemmin, eli ne on sijoitettu ikkunan yläreunaan. Poista-painike on jälleen sijoitettu hieman kauemmas, eli alareunaan jottei sitä vahingossa painettaisi. Vaikka Poista-painikketta painaa vahingossa, ei se poista tuotetta saman tien vaan antaa varmistuskysymyksen "haluatko todella poistaa tämän ikkunan".

3.3.5 Hinnoittelu eli lasketetaan sovelluksella tarjouksen riveille eli tuotteille loppusumma, sekä loppusumma koko tarjoukselle

Tarjouksen rivien eli eri tuotteiden hinnat lasketaan Laskenta-ikkunassa (kuvio 13). Tämä vaihe saattaa kestää muutamia minutteja, varsinkin jos rivien määrä on ollut suuri. Kun laskenta on suoritettu, on tarjous valmis tulostettavaksi asiakkaalle.

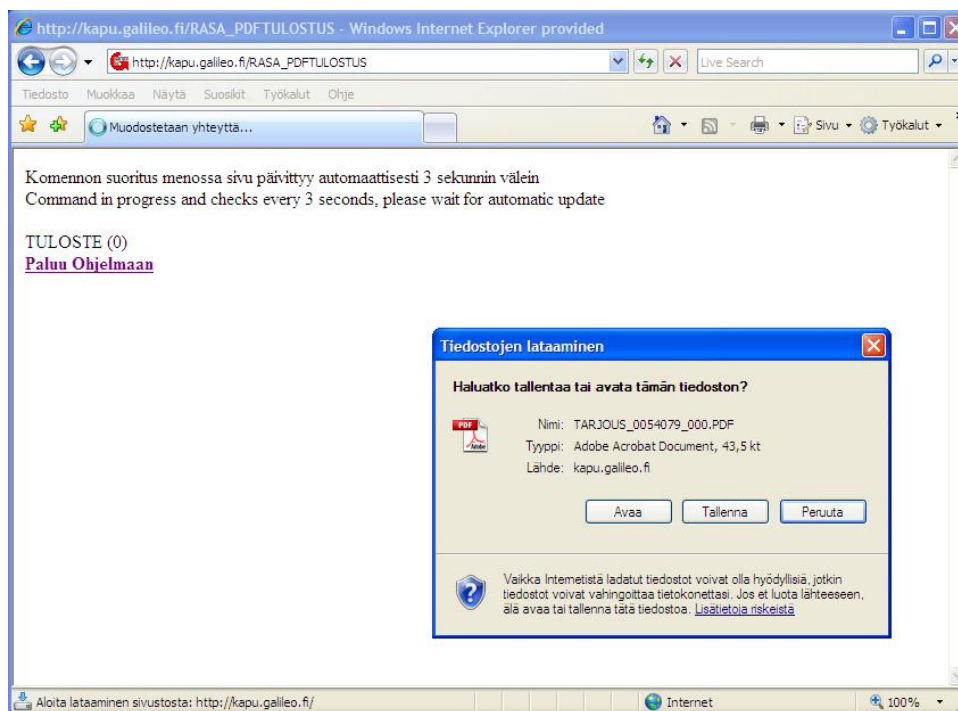


KUVIO 13. Tarjouksen rivin laskenta eli tarjouksen tuotteiden hinnoittelu.

Laskennan kaavat määritellään kunkin yrityksen katteiden mukaisesti. Määrittely voidaan tehdä myös myyjäkohtaisesti eli näin määräytyy vaikkapa kunkin myyjän ansaitsema provisio. Ohjelmiston käyttämisessä laskentavaihe on se vaihe, joka saattaa olla eniten aikaa vievä. Jos tarjouksessa on vaikkapa sata riviä erilaisia tuotteita, saattaa laskenta hieman kestää. Laskennan osuutta voidaan pitää kriittisimpänä ohjelmiston kuormittajana.

3.3.6 Tarjouksen tulostus asiakkaalle joko tiedostona tai paperille

Kun tarjous on valmis ja se halutaan tulostaa, valitaan ”Tulosta”-painike ”Dokumenttiotsikko”-sivulla eli sivulla, jossa syötettiin asiakastiedot. Tarjous tulostuu ja antaa käyttäjälle tiedoksi kuvion 14 ikkunan merkiksi siitä että tulostus on käynnissä. Tulostaminen kestää hetken, kunnes aukeaa ”Tiedoston lataaminen” –ikkuna (kuvio 14). Tällöin tarjous voidaan joko tallentaa esimerkiksi sähköpostilla lähettämistä varten tai avata paperille tulostamista varten. Käyttäjien koneilla vaatimuksena on siis PDF-dokumenttien katselumahdollisuus eli sen avulla tulostaminen PDF-muotoon (Portable Document Format) onnistuu. PDF-dokumenttia voidaan katsoa tietokoneen näytöllä, tulostaa paperille tai postittaa asiakkaalle sähköpostin liitteenä.



KUVIO 14. Tarjouksen tallennus tai tulostus PDF-dokumentista.

PDF-dokumentti on yleinen tiedostomuoto, sen käytössä ei liene ongelmia. Jos dokumentti ei aukea, koneelta puuttuu todennäköisesti Adobe Acrobat Reader -ohjelma, ohjelmiston versio on liian vanha tai koneella on joku tekninen vika. Adobe Acrobat Reader -ohjelmisto on vapaasti ladattavissa internetistä.

3.4 Ohjelmiston tietojen tallennus ja tietokannat

Vuosia käytössä olleen KAPU-sovelluksen pohjalta on lähdetty rakentamaan web-versiota eli WebKAPU:a. Kuten KAPU:ssa, myös WebKAPU:ssa tietokantoina käytetään MS SQL- tai FireBird-tietokantoja.

Perustiedot ja tarjoustiedot ovat omissa tietokannoissaan. Perustietokanta päivitetään yrityksen tuotantoversiosta eli jokaisella yrityksellä on oma perustietokantansa eli tuotetietokantansa. Perustietokanta sisältää kaikki yrityksen käyttämät ja määrittelemät tuotetiedot. Joka yrityksellä on myös oma tarjoustietokantansa. Niimensä mukaisesti tarjoustietokanta sisältää kaikki yrityksen tekemät tarjoukset. Tietokannat sijaitsevat web-palvelimissa tai välittömässä läheisyydessä.

Kuviossa 15 on ote SQL-tietokannasta ja tietokannassa olevasta tarjous-taulusta. Kuvassa on tarjousrivien taulu (All Database Aliases). ”Content of TARJOURSRI-VIT” -sarakkeissa on muutaman rivitiedon alku. Kenttiä tarjoustaulussa voi olla noin 80 ja rivin tarkennustauluissa vielä huomattavasti enemmän.

ID	ID_PART	ID_ROW	TUNNUS	TUOTE	MALLI	KARMIS	MLEVEYS	MKORKEUS	LKM1
50003	1	2	15'12	MSE/A	B4	170	15	12	
50003	1	4	12'12	MSE/A	B	170	12	12	
50003	1	6	12'14	MSE/A	A	170	12	14	
50003	1	8	6'12	MSE/A	A	170	6	12	
50003	1	10	8'6	MSE/A	A	170	8	6	
50003	1	12	6'6	MSE/A	A	170	6	6	
50003	1	14	9'21/12	AJU044LEV	A	170	12	21	
50003	1	16	9'21/14	IOUA/ELEM	A	170	13	21	
50003	1	18	12'12	MSE/A	E	170	12	12	
50003	1	20	9'21	AJU011	A	170	9	21	
50003	1	22	12'12	MSE/A	A	170	12	12	
50003	1	24	9'21	IPDA/ELEM	A	170	9	21	
50003	1	26		ASENNUS	A	0	0	0	
50004	0	2	12'12	MSE/A	A	170	12	12	
50004	0	4	Parvekeovi	IPDA/ELEM	A	150	9	21	
50004	0	6	9'21	IOUA/ELEM	A	210	9	21	
50004	0	8	9'21	IPDA/ELEM	A	150	9	21	
50004	0	10	9'21	IOUA/ELEM	A	210	9	21	
50004	0	12	9'21	IOUA/ELEM	A	210	9	21	
50004	0	14	9'21	IOUA/ELEM	A	210	9	21	
50004	0	16	9'21	IOUA/ELEM	A	210	9	21	
50004	0	18	9'21	IOUA/ELEM	A	210	9	21	
50004	0	20	9'21	IOUA/ELEM	A	210	9	21	
50004	0	22	9'21	IOUA/ELEM	A	210	9	21	
50004	0	24	9'21	IOUA/ELEM	A	210	9	21	
50004	0	26	9'21	IOUA/ELEM	A	210	9	21	
50006	0	2	9'6	MSE/A	A	130	9	6	
50006	0	4	12'14	MSE/A	C	130	16	14	
50006	0	6	12'14	MSE/A	F	130	9	6	
50006	0	8	9'21	IPDA/ELEM	A	150	14	21	
50006	0	10	9'21	IPDA/ELEM	A	170	14	21	
50006	0	12	12'12	MSE/A	G	170	12	12	
50007	0	2	12'12	MSE/A	A	170	12	12	
50011	0	2	12'12	MSE/A	A	210	12	12	
50011	0	4	Vessa	MSE/A	A	210	12	12	

KUVIO 15. Tarjous- ja tarjousrivien-taulu.

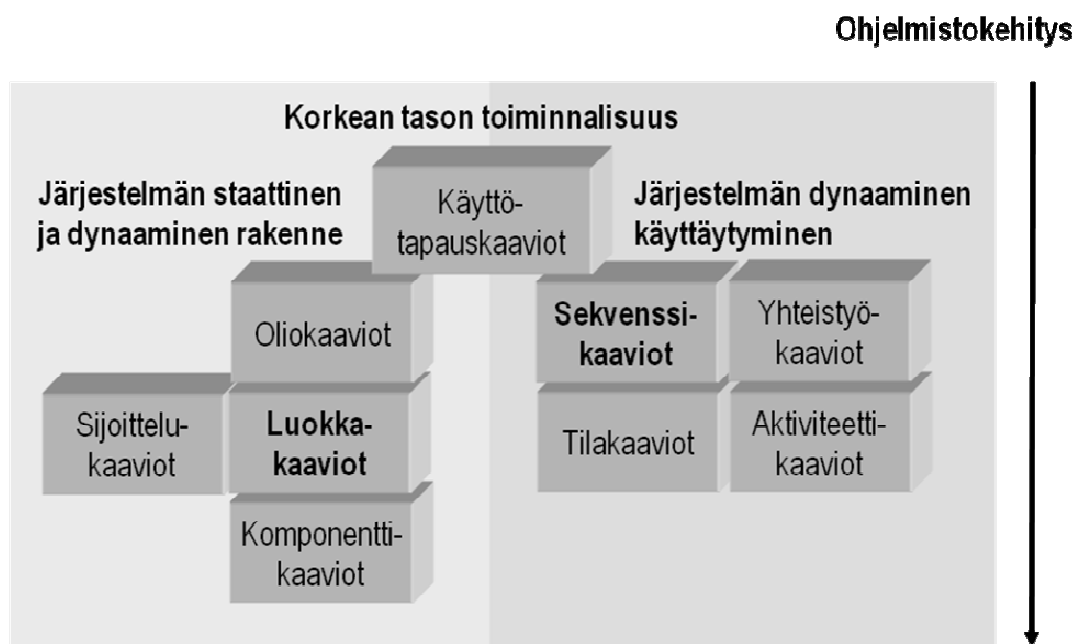
Tarjouksien ja tilausvahvistusten tietojen siirrot tehdään tekstitiedostoina, joissa ovat tauluittain tarjous- ja tilaus-dokumenttiin liittyvät taulukuvaukset ja rivit.

Tietokantoihin tallennettaessa vasteaikojen tulee olla lyhyitä. Esimerkiksi tarjouksen laskennassa on monta eri vaihetta ja eri ikkuna- tai oviosien noutamiskertoja tietokannoista on useita. Jos jokainen vaihe kestää sekunteja, on tarjouksen syöttäminen liian hidasta.

4 LOGIN- ELI OHJELMISTOON KIRJAUTUMIS- JA WEBKAPU-SOVELLUKSEN MALLINNUS

UML-mallinnus eli Unified Modeling Language (UML) syntyi 1990-luvun lopulla, kun kolme menetelmäkehittäjää yhdisti omia kaavioesityksiään universaaliksi notaatioksi. Kehittäjät olivat Grady Booch, James Rumbaugh ja Ivar Jacobson. UML:n kehittämisen on jatkossa ottanut tehtäväkseen Object Management Group eli OMG. Vuonna 1996 OMG julkaisi ehdotuksen UML 1.0 versioksi. Tämän pohjalta kehitys jatkui, ja tällä hetkellä UML versio on 2.0. (Koskimies 2000, 124.)

UML on ohjelmistojen mallien kuvaustapa eli graafinen suunnittelukieli. UML on laaja kieli eri kaaviotyypeineen (kuvio 16) ja sitä pidetäänkin UML:n heikkoutena. UML-esitystapojen laajuuden vuoksi UML:ää ei kannata ajatella ottavansa käyttöön täysin kattavasti. On hyvä ajatella ohjelmistokehitysympäristön tarpeita ja valikoida UML-esitystavat sen mukaan mikä parhaiten tukee järjestelmää. (Koskimies 2000, 124.)



KUVIO 16. Eri UML-kaaviotyypit kuvattuna ohjelmistokehitysprosessin etenemisjärjestyksessä. (Koskimies 2000, 124.)

UML sisältää yhdeksän erityyppistä kaavioesitystä:

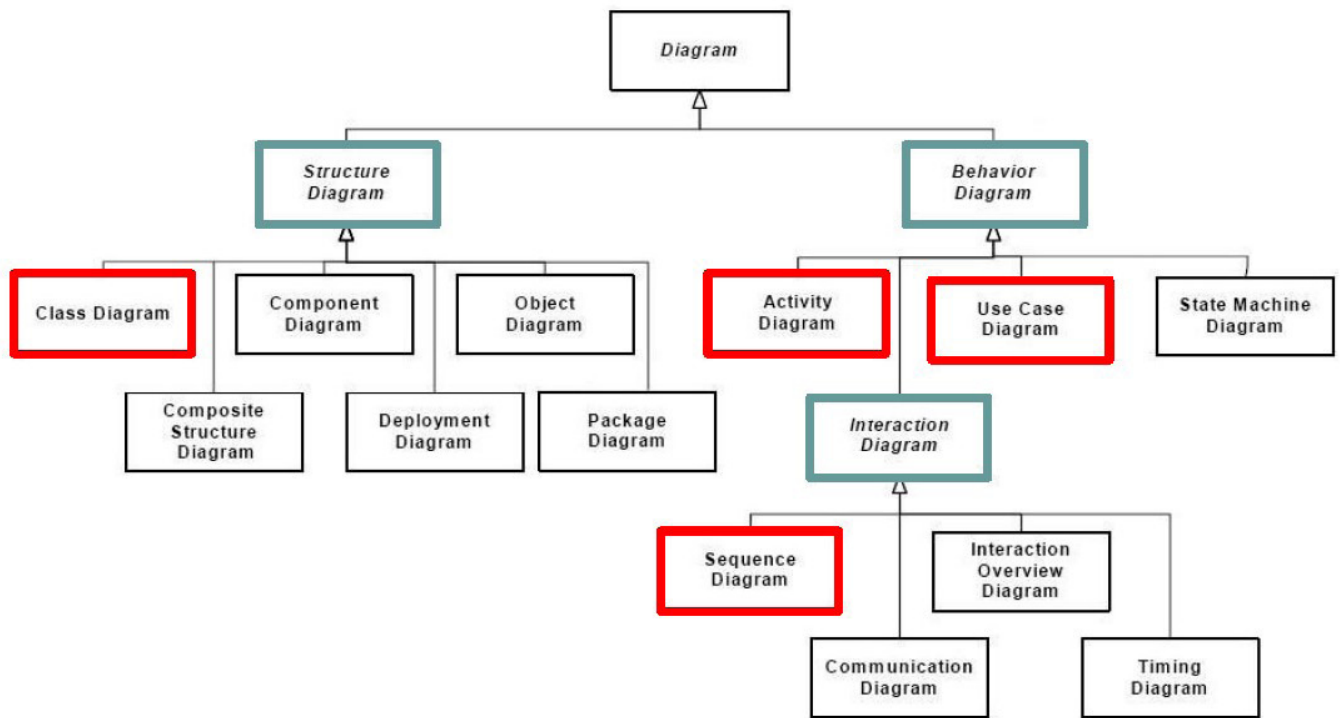
- Käyttötapauskaavio: kuvataan järjestelmän korkean tason toiminnallisuutta
- Oliokaaviot, luokkakaaviot, sijoittelukaaviot, komponenttikaaviot: kuvataan järjestelmän staattista ja dynaamista rakennetta
- Sekvenssi-, yhteistyö-, tila- ja aktiviteettikaaviot: kuvataan järjestelmän dynaamista käyttäytymistä (Koskimies 2000, 125.)

Vaikka eri kaaviotyyppejä on näin monta erilaista, ovat ne osittain päällekkäisiä ja raja eri kaavioiden välillä saattaa olla liukuva. Kuviossa 17. on eri kaaviotyypit ryhmiteltynä siten että aikaisemmin ohjelmistokehitysprosessissa käytettävät kaaviotyypit ovat kuvassa ylimpänä.

Kuviossa 17 on kuvattu UML 2.0 -version eri kaaviot, niitä on yhteensä 13 kappaletta ja kaaviotyyppejä on kolme erilaista:

- Staattista rakennetta kuvaavat:
 - Luokkakaavio (Class diagram)
 - Oliokaavio (Object diagram)
 - Komponenttikaavio (Component diagram)
 - Rakennekaavio (Composite Structure diagram)
 - Pakkauskaavio (Package diagram)
 - Sijoituskaavio (Deployment diagram)
- Käyttäytymistä kuvaavat:
 - Käyttötapauskaavio (Use Case diagram)
 - Aktiviteettikaavio (Activity diagram)
 - Tilakaavio (State machine diagram)
- Vuorovaikutusta kuvaavat:
 - Sekvenssikaavio (Sequence diagram)
 - Kommunikaatiokaavio (Communication diagram)
 - Ajoituskaavio (Timing diagram)
 - Vuorovaikutuskaavio (Interaction overview diagram)

(Koskimies 2000, 125.)



© Jukka Viljamaa, Juha Gustafsson

KUVIO 17. Kaavioiden luokittelu (Heikkinen 2010.)

4.1 Korkean tason toiminnallisuus: Tarjouslaskijan ja tehtaan toiminnan käyttötapauskaavio

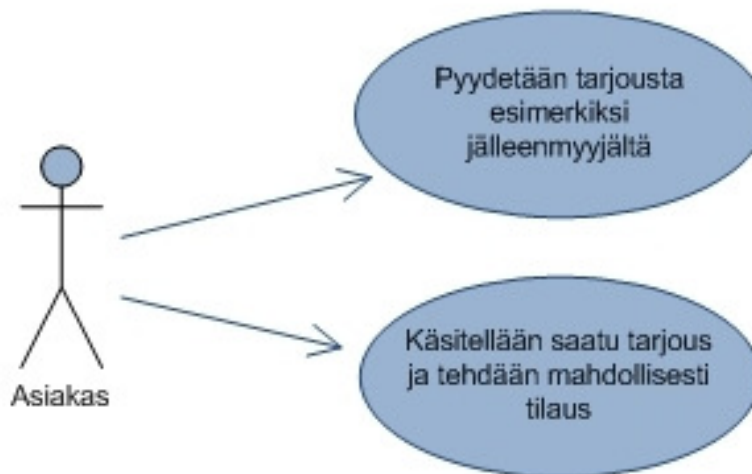
Käyttötapauskaavio eli Use Case Diagram kuvaa järjestelmän käyttötapausten väliset suhteet ja käyttötapauksiin osallistuvat järjestelmän ulkoiset toimijat. Käyttötapauskaavio kuvaa myös käyttötapauksiin osallistuvat järjestelmän ulkoiset toimijat. Käyttötapauskaaviota käytetään käyttöyhteyden ja vaatimusten korkean tason kuvaamiseen. Käyttötapaus eli Use Case on kuvaus järjestelmän ja käyttäjien välisestä vuorovaikutuksesta. (Koskimies 2000, 127.)

Käyttötapauskaavio kuvataan symboleilla, jotka ovat helposti ymmärrettäviä vaikka henkilö ei tietäisi mitään ohjelmoinnista. Käyttötapauskaavion on todettu olevan hyvin ymmärrettävä kommunikaatiomuoto juuri kuvissa käytettyjen symboleiden ymmärrettävyyden vuoksi. (Heikkinen 2010.)

Seuraavissa kappaleissa on esitetty yksinkertaiset käyttötapauskavioit tärkeimmistä järjestelmän toimijoista. Näitä ovat tarjouslaskija, asiakas sekä tehtaan toiminnot.

4.1.1 Asiakas pyytää tarjouslaskijalta tarjouksen ikkunoista ja ovista

Asiakas pyytää tarjouksen esimerkiksi yrityksen tarjouslaskijalta ovista ja ikkunoista. Kuviossa 18 on esitetty asiakkaan toiminta käyttötapauskaviona.



KUVIO 18. Käyttötapauskavio: Asiakkaan toiminta.

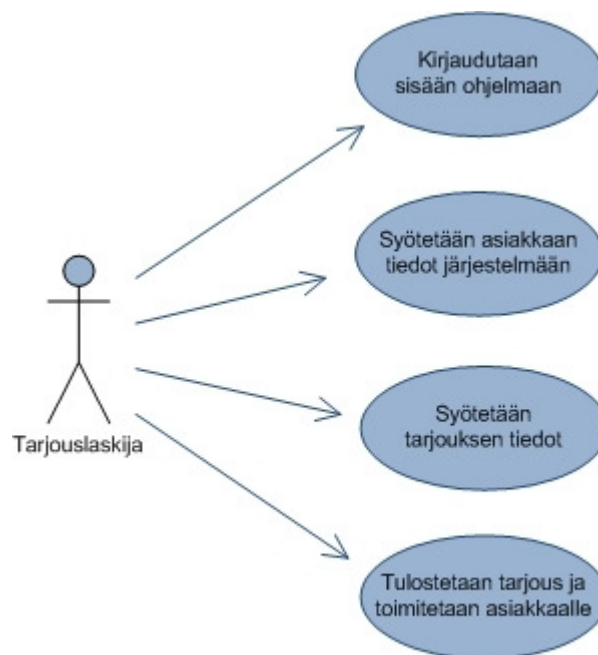
Kun asiakas on saanut tarjouksen hän joko hylkää sen, pyytää muutoksia tai hyväksyy tarjouksen. Kun asiakas on hyväksynyt tarjouksen, tarjouslaskija tekee siitä tilauksen.

4.1.2 WebKAPU:ssa tarjouslaskijan toiminnan käyttötapauskaavio

Tarjouslaskijan toiminnassa on seuraavat käyttötapaukset:

1. Tarjouslaskija kirjautuu sisään ohjelmaan
2. Tarjouslaskija syöttää asiakkaan henkilötiedot järjestelmään
3. Tarjouslaskija syöttää pyydetyn tarjouksen tiedot järjestelmään
4. Tarjouslaskija tulostaa tarjouksen järjestelmästä ja toimittaa asiakkaalle

Kuviossa 19 on piirretty käyttötapauskaaviona tarjouslaskijan toiminta.



KUVIO 19. Käyttötapauskaavio: Tarjouslaskijan toiminta.

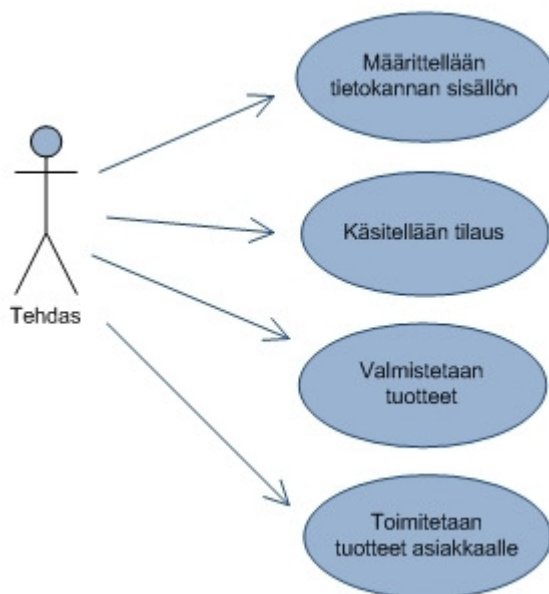
Tarjouksen laatiminen asiakkaalle edellyttää neljän edellä mainitun käyttötapauksen toteutumisen. Opinnäytetyössä keskitytään kolmeen ensimmäiseen eli teknisesti tärkeimpiin käyttötapauksiin.

4.1.3 Tehtaan toiminnan, esimerkiksi Alavus Ikkunat Oy:n, käyttötapauskäyttö

Tehtaan toiminnassa on seuraavat käyttötapauskäytöt:

1. Tehtaan määrittelee tuotteet ja tiedot tietokantaan
2. Tehtaan käsittelee tarjouslaskijan syöttämän tilauksen
3. Tehtaan valmistaa tilauksessa tilatut tuotteet
4. Tehtaan toimittaa asiakkaan tilaamat tuotteet asiakkaan määrittelemään kohteeseen

Kuviossa 20 on piirretty käyttötapauskäytönä tehtaan toiminta.



KUVIO 20. Käyttötapauskäyttö: Tehtaan toiminta.

Jotta asiakas saa tilaamansa tuotteet tehtaalta, edellyttää tilauksen toimitus tehtaalta kuvion 20 toimenpiteet.

4.2 Järjestelmän dynaaminen käyttäytyminen

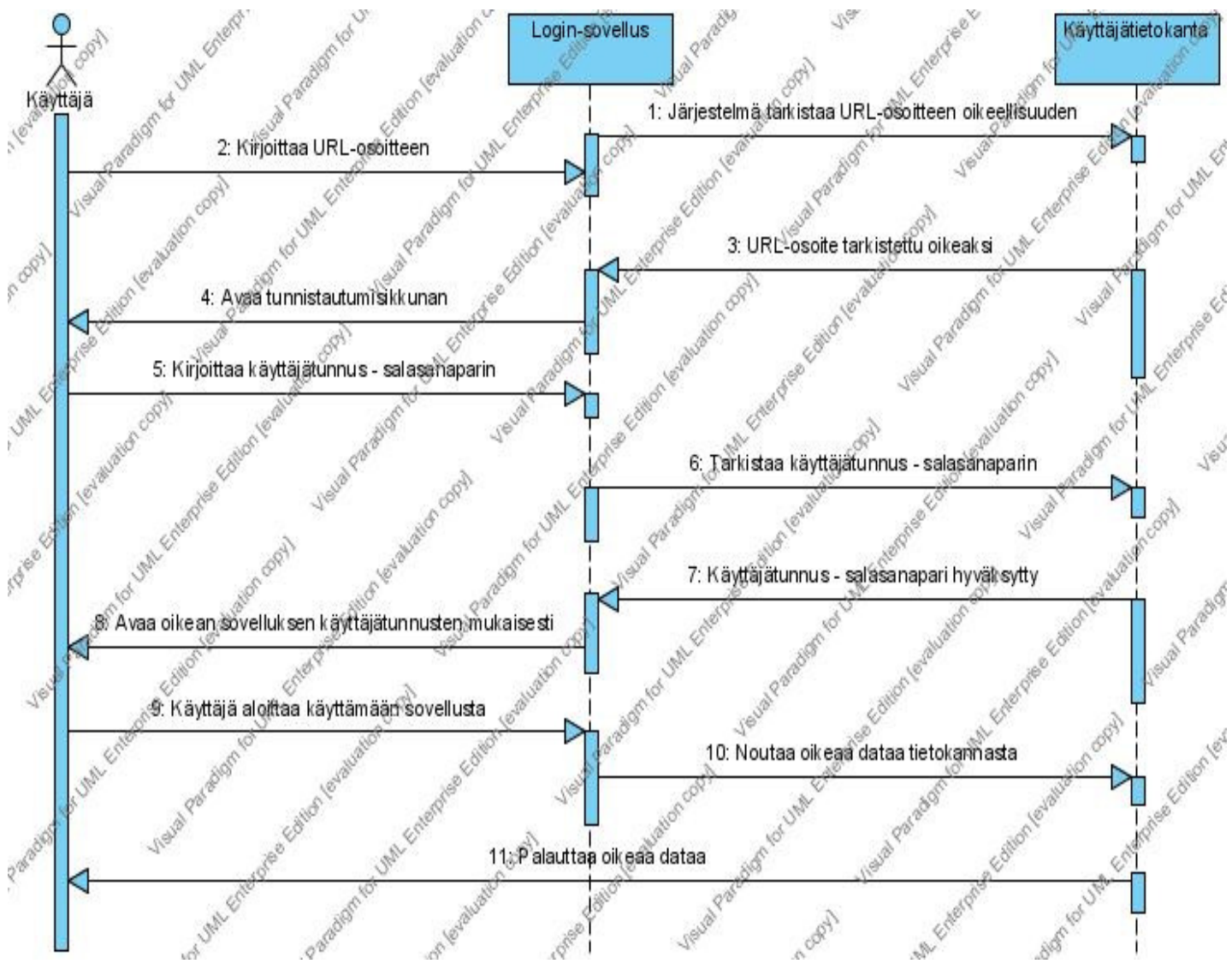
Järjestelmän dynaamista käyttäytymistä kuvaavat sekvenssi-, yhteistyö-, tila- ja aktiviteettikaaviot. Seuraavissa kappaleissa kuvataan Gapassi-projektin eri toimintoja havainnollistavin kaavioin.

Sekvenssikaavio eli Sequence Diagram muodostuu olioista ja niiden välisistä sanomista. Sekvenssikaaviolla kuvataan olijoukon välistä vuorovaikutusta, joka yleensä kuvaa tietyn käyttötapauksen toteutusta. Aika kulkee sekvenssikaaviossa ylhäältä alas. Vuorovaikutustapahtumat kulkevat vaakatasossa nuolilla kuvattuna sanoman lähettäjältä sanoman vastaanottajalle. Osallistujat kuvataan pystypalkkina. (Koskimies 2000, 130.)

Aktiviteettikaavion eli Activity Diagramin voidaan sanoa olevan muodollisesti luokkakaavion erikoistapaus, mutta sitä käytetään eri näkökulman kuvaamiseen. Se ei kuvaa olion reagoimista ulkopuolisiin tapahtumiin, vaan tietyn tehtävän sisäisen logiikan. Aktiviteettikaaviossa voi siis olla kuvattuna useampikin käyttötapaus. (Koskimies 2000, 133.)

4.2.1 Login-sovelluksen eli kirjautumis-sovelluksen kuvaus sekvenssikaaviona

Kuviossa 21 on kerrottu käyttäjän eli tarjouslaskijan tunnistautumisen kulku sekvenssikaaviona.



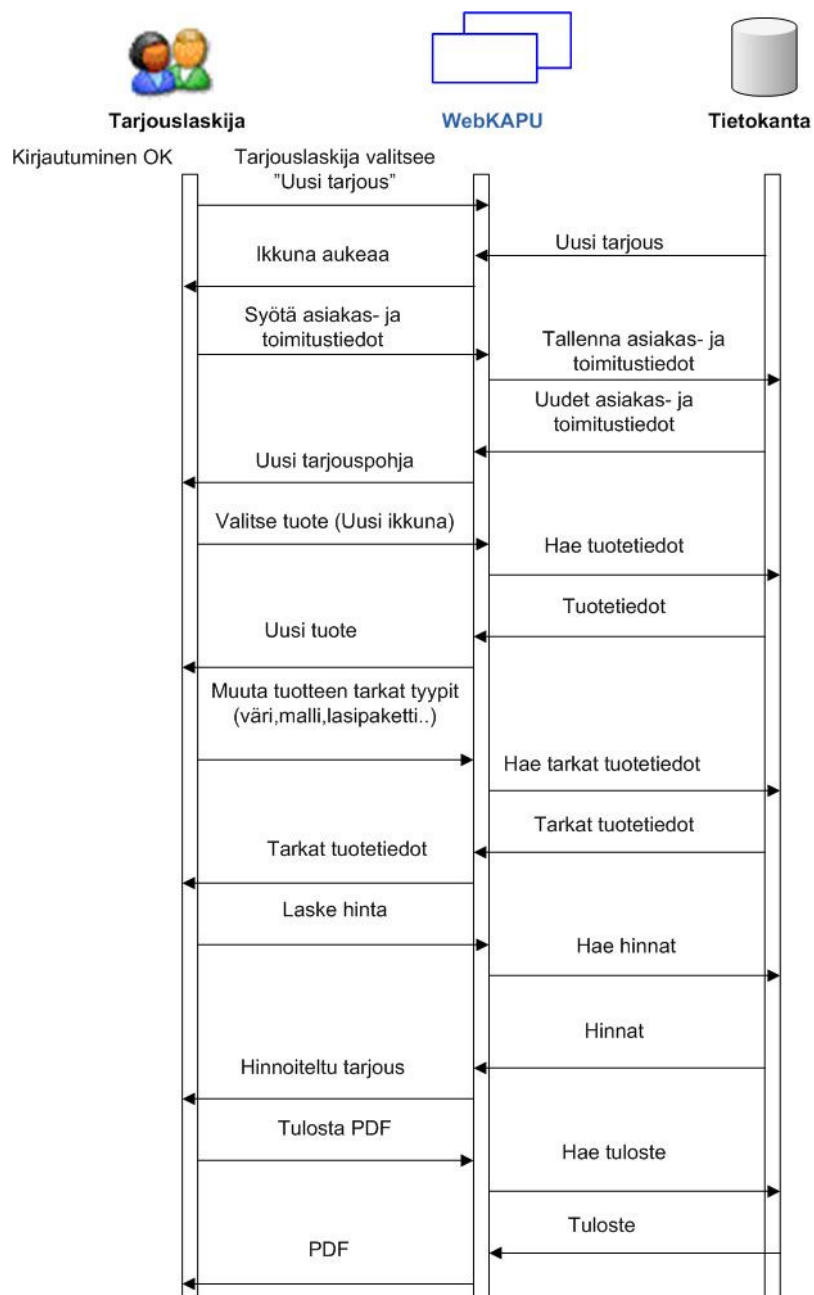
KUVIO 21. Sekvenssikaavio: Tarjouslaskija kirjautuu järjestelmään, Login-sovelluksen toiminta.

Seuraavassa on kuvattu sanallisesti kuvion 21 sekvenssikaavion kulku eli tarjouslaskijan kirjautuminen web-järjestelmään login-sovellusta käyttäen:

1. Järjestelmä tarkistaa URL-osoitteen oikeellisuuden
2. Käyttäjä kirjoittaa URL-osoitteen
3. URL-osoite on tarkistettu oikeaksi
4. Ohjelma avaa tunnistautumis-ikkunan
5. Käyttäjä kirjoittaa käyttäjätunnus-salasanaparin
6. Login-sovellus tarkistaa käyttäjätunnus-salasanaparin
7. Login-sovellus antaa palautteen että käyttäjätunnus-salasanapari on hyväksytty
8. Ohjelma avaa oikean sovelluksen käyttäjätunnuksen mukaisesti
9. Tarjouslaskija aloittaa sovelluksen käyttämisen
10. Ohjelma noutaa oikeaa dataa KAPU-sovelluksen tietokannasta
11. Ohjelma palauttaa oikeaa dataa tarjouslaskijan näkymään

4.2.2 Tarjouslaskijan ja WebKAPU-sovelluksen kuvaus sekvenssikaaviona

Kuviossa 22 on kuvattu sekvenssikaavio tarjouslaskijan ja WebKAPU-sovelluksen työnkulusta. Kaaviossa oletetaan että kirjautuminen järjestelmään on tehty onnistuneesti.

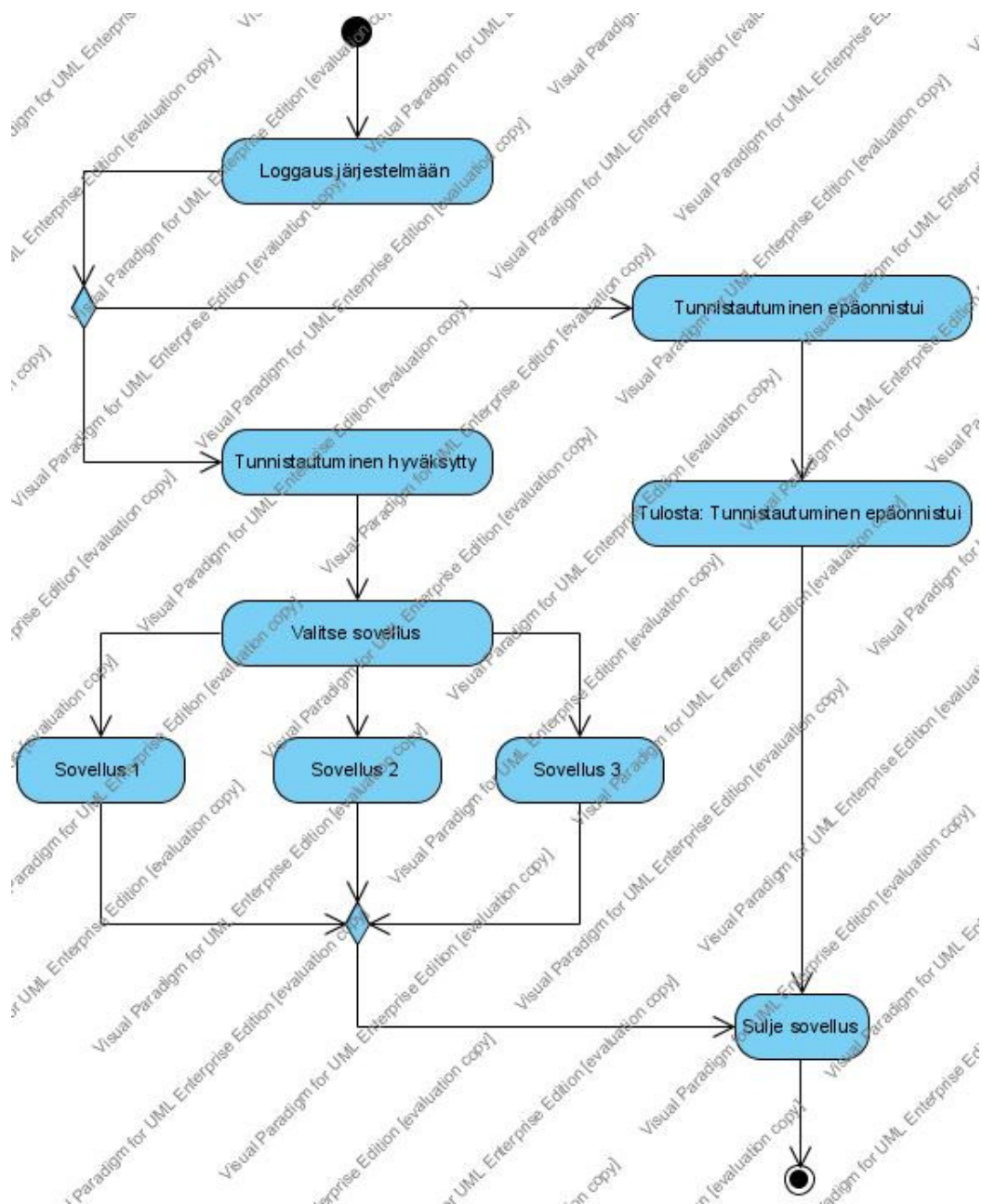


KUVIO 22. Sekvenssikaavio: Tarjouslaskijan WebKAPU-sovelluksen työnkulku.

Lopputuloksena syntyy tarjous, joka voidaan antaa asiakkaalle joko paperille tulostettuna tai sähköisenä vaikkapa liitteenä sähköpostissa.

4.2.3 Login-sovelluksen eli kirjautumissovelluksen kuvaus aktiviteettikaaviona

Kuviossa 23 on esitetty login-sovelluksen toiminta aktiviteettikaaviona. Aktiviteettikaaviossa musta pallo kuvaa lähtöpistettä. Nuolet kuvaavat tapahtuman kulkua ja pyöreäkulmaisissa laatikoissa on toiminnon tiedot.

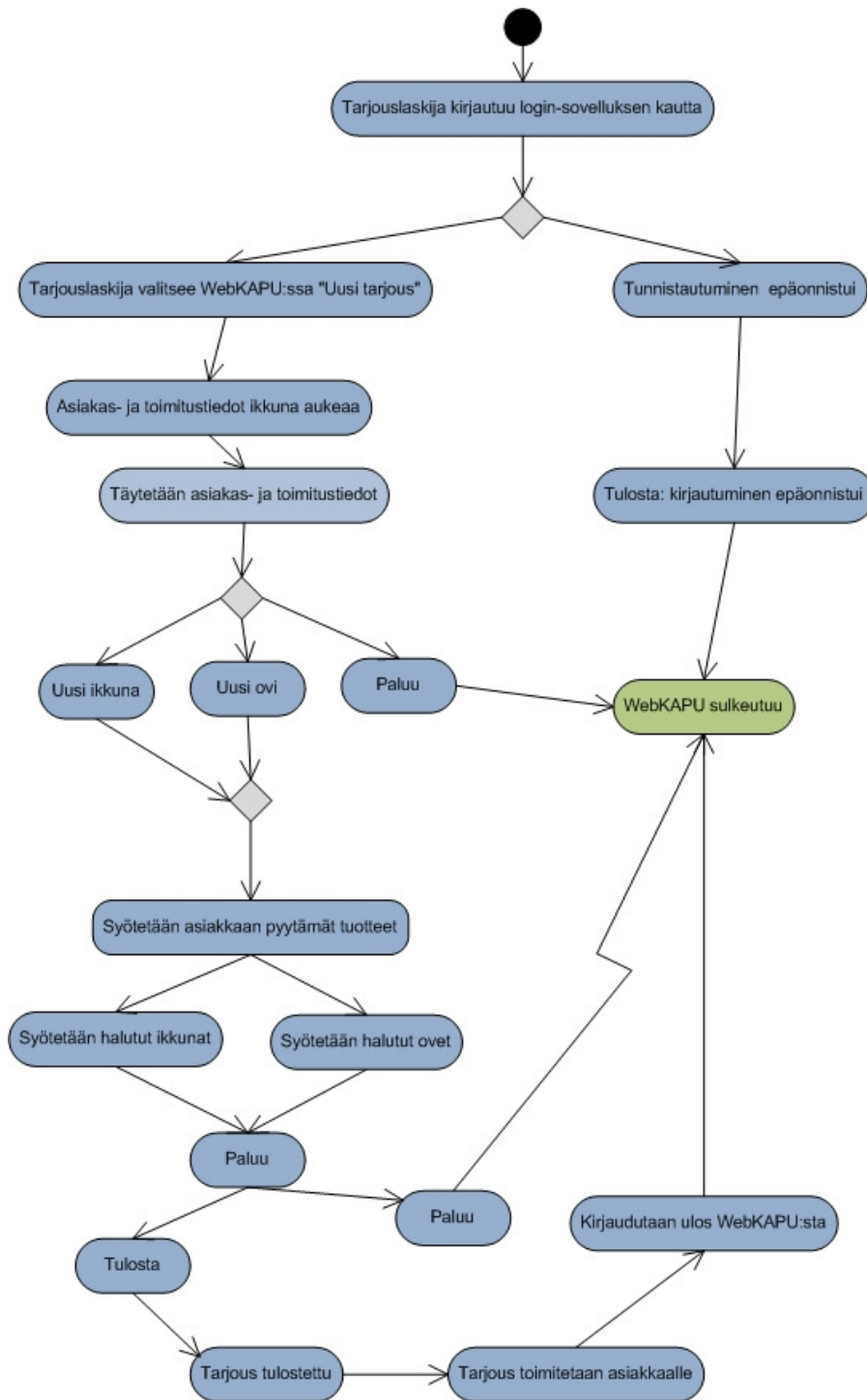


KUVIO 23. Aktiviteettikaavio: login-sovelluksen kuvaus aktiviteettikaaviona.

Aktiviteettikaaviossa kuviossa 23 ylin ympyrä kattaa toiminnon ”loggautuminen” eli järjestelmään sisään kirjautumisen. Kirjautuminen joko epäonnistuu tai onnistuu. Jos kirjautuminen epäonnistuu, antaa ohjelma ilmoituksen epäonnistuneesta kirjautumisesta ja sovellus sulkeutuu. Onnistuneen sisään kirjautumisen jälkeen aukeaa käyttäjätunnukselle määritelty sovellusvalikko. Valikossa käyttäjä voi valita minkä sovelluksen (esimerkiksi laskutus, tarjouslaskenta, tilaukset) avaa. Sovellusvaihtoehdot määräytyvät käytetyn URL-osoitteen, sekä käyttäjätunnuksen mukaan.

4.2.4 Tarjouslaskijan ja WebKAPU-sovelluksen kuvaus aktiviteettikaaviona

Aktiviteettikaaviossa lähdetään etenemään mustasta pallosta, joka on alkupiste. Kuvion 24 kaaviossa tarjouslaskija kirjautuu järjestelmään onnistuneesti tai vaihtoehtoisesti kirjautuminen ei onnistu. Onnistunut kirjautuminen etenee kaaviossa vasemmalla ja epäonnistunut kirjautuminen etenee kaaviossa oikealla. Onnistuneen kirjautumisen jälkeen tarjouslaskija aloittaa uuden asiakas- ja toimitustietojen syöttämisen järjestelmään.



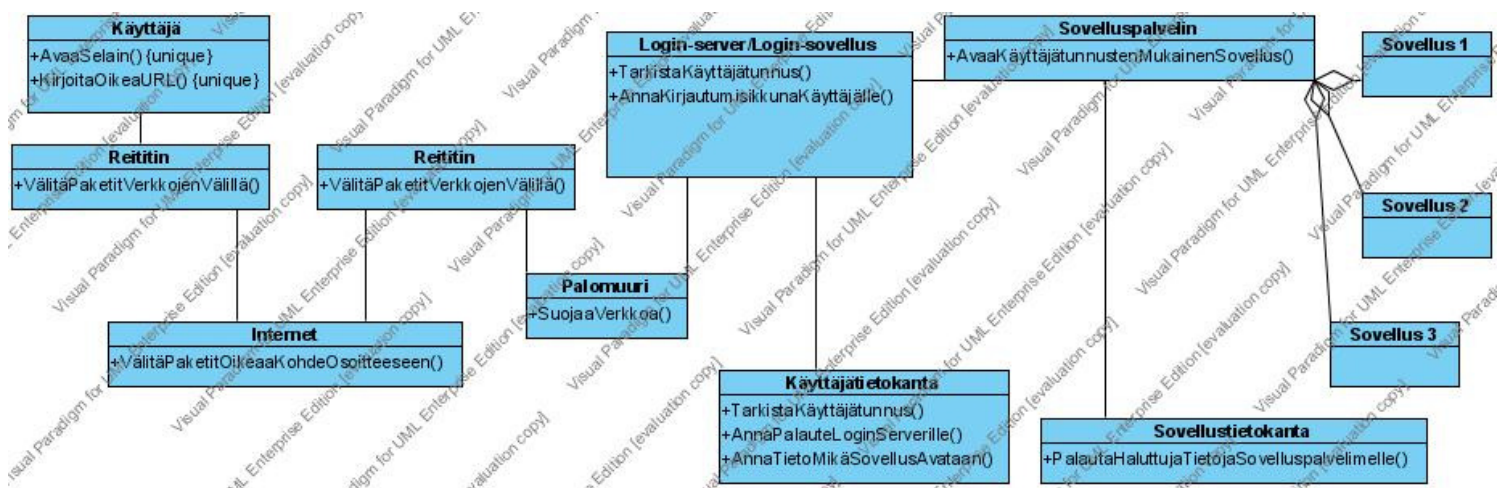
KUVIO 24. Aktiveettikaavio: Tarjouslaskijan ja WebKAPU-sovelluksen kuvaus aktiveettikaaviona.

4.3 Järjestelmän staattinen ja dynaaminen rakenne

Luokkakaavio eli Class Diagram kuvaa järjestelmän luokkia ja niiden välisiä suhteita. Luokkakaavio kuvaa järjestelmän staattisen rakenteen eli järjestelmään kuuluvia luokkia ja niiden välisiä suhteita. (Koskimies 2000, 128.)

4.3.1 Gapassi-projektin rakenne kuvattuna luokkakaaviona

Kuviossa 25 luokkakaaviossa on kuvattu käyttäjän eli tarjouslaskijan, laitteistojen, sovellusten ja tietokantojen suhteet.



KUVIO 25. Luokkakaavio: Gapassi-projektin kuvaus luokkakaaviona.

Kuviossa 25 edetään vasemmalta oikealle. Sinisissä laatikoissa tummennettu teksti kuvaa luokkaa. Viivat luokkien välillä kuvaavat luokkien välisiä suhteita.

5 GAPASSI-PROJEKTIN TESTAUSSUUNNITELMA JA TESTAUS

Testauksen työvaiheet ovat testauksen suunnittelu (testaussuunnitelma ja testitapaukset) testiympäristön luonti, testin suorittaminen ja tulosten tarkastelu. (Haikala & Märijärvi 2002, 281.) Opinnäytetyössä keskitytään testaussuunnitelman ja testitapauksien suunnitteluun ja toteutukseen. Tuloksena tehdään testitapaustaulukot. Ne toimitetaan edelleen katselmoitavaksi, kommentoitavaksi ja toteutettavaksi ohjelmoijille.

Testauksen tarkoituksena on löytää testattavan ohjelman virheet. Ohjelmiston testauksen yhteydessä testaus määritellään suunnitelmalliseksi virheiden etsimiseksi koko ohjelmaa tai sen osaa suorittamalla. (Haikala & Märijärvi 2002, 282.)

Kunnollinen ja kattava testaus parantaa myös asiakastyytyväisyyttä ja vähentää ylläpidon tarvetta. Huolellinen testaus takaa asiakkaan tyytyväisyyden jolloin myös ylläpidon työkuorma pienenee.

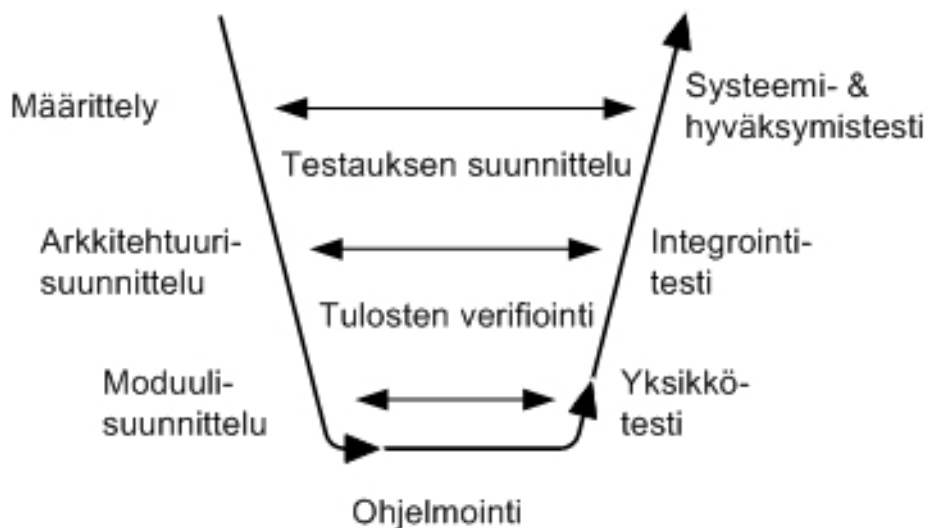
Ohjelmaa testataan myös siten, että pyritään etsimään pois virheet, joita tavallinen käyttäjä ei välttämättä koskaan huomaisi. Testaussuunnitelmassa kiinnitetään huomiota myös ohjelmiston käytettävyyteen, esimerkiksi liikkuminen eri ikkunoissa voi olla hankalaa, huonontaa käytettävyyttä. Kommentoitaessa annetaan palautetta jos testaajan mielestä jokin osa olisi hänen mielestä paremmin toteutettavissa. Jos käy niin että virheitä ei löytyisikään, osoittaa testaus sen, että ohjelmisto toimii hyvin.

Gapassi-projektin testauksen tarkoitus on tuottaa ohjelmoijille tärkeitä kommentteja login- eli kirjautumis-sovelluksesta, sekä WebKAPU:n toiminnasta. Kantaa otetaan niin ohjelman toimintaan eri näkymissä, sekä lisäksi ohjelmiston käytettävyyteen ja ulkonäköön.

5.1 Login-sovelluksen ja WebKAPU:n testitapausten valinta

Testitapausten valinnassa on kaksi peruslähestymistapaa: lasilaatikkotestaus (glass/white box testing) ja mustalaatikkotestaus (black box testing). (Haikala & Märijärvi 2002, 289.) Lasilaatikkotestauksessa tarkoitetaan tilannetta, jossa käytetään hyväksi tietoa ohjelman toteutuksesta, käytössä on ohjelman lähdekoodi. Mustalaatikkotestaus tarkoittaa sitä, että testitapaukset valitaan perehtymättä ohjelman toteutukseen. Opinnäytetyössä käytetään kuitenkin harmaalaatikkotestaus-ta (gray box testing). Tämä tarkoittaa sitä, että käytetään hyväksi tietoa ohjelman toteutusperiaatteista, kuitenkin niin, että testaajalla ei ole lähdekoodia saatavillaan. Tämä testaustapa on välimuoto lasilaatikko- ja mustalaatikkotestaukselle.

Kuviossa 26 on testauksessa yleisesti tunnettu V-malli, joka kuvaa testauksen testaustasot. Opinnäytetyössä Haikalan ja Märijärven V-malli kuva on piirretty uudelleen Microsoft Visio 2007 – ohjelmistolla. Testauksen suunnittelut V-mallissa tapahtuu testaustasoa vastaavalla suunnittelutasolla. Järjestelmätestaus suunnitellaan määrittelyvaiheessa. Integrointitestaus vastaavasti suunnitellaan suunnittelu-vaiheessa ja moduulitestaus modulisuunnittelu vaiheessa. (Haikala & Märijärvi 2002, 286.)



KUVIO 26. Testauksen V-malli eli testaustasot. (Haikala & Märijärvi 2002, 287.)

Opinnäytetyössä testitapaukset on johdettu suoraan käyttötapauksista, joten nämä kolme käyttötapausta ovat:

1. Tarjouslaskija kirjautuu sisään ohjelmaan.
2. Tarjouslaskija syöttää henkilö- toimitustiedot järjestelmään.
3. Tarjouslaskija syöttää pyydetyn tarjouksen tiedot järjestelmään

Näiden lisäksi testaan testitapaukset:

4. Tarjouksen tulostaminen.
5. Jacob Nielsenin viisi laatukomponenttia.

Testauksessa testataan toimintaa suunnitelluin testitapauksin, mutta myös suunnittelemattomien eli sellaisten tapausten kautta jotka tulevat vastaan testattaessa. Kaikkia testitapauksia ei ole mahdollista keksiä ennen testausta, vaan ne tulevat vastaan, välillä yllättäenkin, testauksen edetessä.

Testaussuunnitelmassa käydään läpi jokaisen ohjelman näytön syöttöruudut ja valintalaatikat. Testataan mitä lukuja, kirjaimia, tapoja syöttökentät sallivat ja kommentoidaan ne edelleen ohjelmoijalle katsottavaksi. Esimerkkinä kenttien testauksista mainittakoon @-merkin käyttäminen; vaatiiko sähköpostikenttä @-merkin, vai antaako ohjelma tallentaa tilaajan tiedot ilman kentän täyttämistä. Lisäksi on testattava pystyykö ohjelma käsittelemään ison kuormituksen. Esimerkkinä ohjelmiston kuormittamisesta on se, että järjestelmään syötetään iso tarjous, joka sisältää vaikkapa 200 erilaista tuotetta. Käynnistetään tarjouksen hinnoittelun laskenta ja katsotaan kestäkö ohjelmisto ison kuormituksen.

Ovi- ja ikkunatuotannossa on monia eri tuotemalleja ja monia eri mahdollisuuksia tuotteiden räätälöinnille. On siis erittäin tärkeää että eri mahdollisuudet on tarkoin määritelty ja helposti saatavilla. Tärkeää on myös rajata pois väärät erilaiset kombinaatiot siksi, että minimoidaan vääränlaisten ikkunoiden ja ovien syöttämismahdollisuus niin tarjous- kuin tilaus vaiheessakin. Jokin pieni määritys saattaa johtaa

siihen että asiakas saa vääränmallisen tai muuten vääränlaisen ikkunan, joka taas johtaa reklamaatioon ja isoihin kuluihin tehtaalle.

Yhtenä tärkeänä osa-alueena testauksessa käsitellään ohjelmiston käytettävyyden parantamista. On tärkeää että ohjelmiston käyttäminen olisi mahdollisimman yksinkertaista. Näin kannustetaan uusia käyttäjiä ja etenkin uusia yrityksiä ostamaan ohjelmisto käyttöön. Tavoitteena tietysti myös se, että käytettävyys olisi mahdollisimman hyvä, että tarjousten laskenta onnistuu tehokkaasti. Näin tarjous saadaan nopeasti tulostettua, toimitettua tarjouksen pyytäjälle ja pystytään solmimaan kaupat mahdollisimman nopeasti.

Omat haasteensa tuottavat eri yritysten erilaiset tuotteet ja tuotteiden sisältämät määritykset. Kaikki tieto tulee ensinnäkin kartoittaa tarkasti ja sitten vielä rakentaa ohjelmiston sisältö siten, että tiedot pystytään viemään järjestelmään juuri tarvittavassa muodossa. Hyvä puoli on se, että ikkuna- ja oviteollisuudessa tuotteet muistuttavat paljon toisiaan, riippumatta yrityksestä. Perusrakenteet ikkunoissa ja ovis- sa ovat samat eli tuotteet koostuvat samankaltaisista materiaaleista kuten maalien värit, lasipakettityypit, hyttyspuitteet, helat, ristikot ja niin edelleen. Siinä vaiheessa kun ohjelmistoa lähdetään toteuttamaan eri teollisuuden aloille, tuleekin kysymykseksi aivan erilaiset termit ja määrittelyt. Esimerkkinä tällaisesta erilaisuudesta voisi olla vaikka metalliteollisuus.

Testaus toteutetaan lähinnä kannettavilla tietokoneilla, joissa käyttöjärjestelmä on Windows XP ja Windows 7. Internet Explorer 8.0 toimii internet-selaimena. Kannettava tietokone on merkiltään HP (Hewlett Packard) ja malli on 6735b ja Pro-Book 6545b. Verkkoyhteytenä käytetään sekä ADSL- että makkula-verkkoyhteyttä.

5.2 Gapassi-projektin testitapaukset

Seuraavassa osiossa on kerätty testitapaukset taulukkomuotoon. Taulukot muodostavat testaussuunnitelman. Testitapaukset on johdettu suoraan käyttötapauksista. Taulukoissa toistuvat otsikot:

- Testitapauksen nimi: jokaiselle testitapaukselle on annettu kuvaava nimi
- Testauksen tarkoitus: mitä testauksessa testataan
- Toteutettu testaus: mitä testauksessa on tehty
- Testauksen tulos: kerrotaan saatu testaustulos
- Kehitysehdotuksia / huomioita: annetaan kehitysehdotus testauksen tuloksesta tai muuten jos halutaan antaa jotain huomiota testauksen tuloksesta

Testitapauksia eli taulukoita on yhteensä viisi. Ensimmäisessä testitapauksessa käyttäjä kirjautuu sisään sovellukseen ja siitä edetään testitapaukseen jossa tarjous tulostetaan. Viidentenä testaustapauksena käsitellään Jacob Nielsenin viisi laatu-komponenttia: Opittavuus, tehokkuus, muistettavuus, virheettömyys, miellyttävyys ja hyödyllisyys. (Nielsen 2000, 124.)

5.2.1 Testitapaus 1: Tarjouslaskija kirjautuu sisään ohjelmaan

Tarjouslaskija kirjautuu järjestelmään Login-sovelluksen eli WebKAPU-ohjelman kirjautumissovelluksen avulla.

Testitapauksen nimi: Tarjouslaskija kirjautuu sisään ohjelmaan oikean linkin kautta ja login-sovelluksen avulla.	Testauksen tarkoitus: Mitä virhetilanteita voi aiheutua kirjautumisessa ja miten käytettävyyttä voidaan parantaa.	
Toteutettu testaus:	Testauksen tulos:	Kehitysehdotuksia / huomioita:
URL-osoite hukassa; löytyykö osoite jostain? Millä tavoin?	Galileo Oy:n ja Alavus Ikkunat Oy:n sivuilla on yhteystiedot joista voi tiedustella osoitetta.	Molemmilla sivuilla voisi olla maininta että numerosta voi tiedustella oikeaa osoitetta.
Voiko WebKAPU-sovelluksen osoitteen löytää Google-hakukoneen avulla?	Haettaessa webkapu sanalla Google-hakukoneella, löytyy WebKAPU:n ohje ja sitä kautta Galileo Oy:n kotisivut josta voi kysyä oikeaa osoitetta.	Voisi jollain tavalla parantaa WebKAPU sanalla löytyviä hakuja, esimerkiksi Galileo Oy:n tai Alavus Ikkunat Oy:n sivuilla.
Mitä jos kirjoitetaan väärä URL-osoite?	Oikea osoite oltava täysin oikein kirjoitettu, muuten selain ei sitä löydä.	Yrityksille ohjaukset että osoitteeksi riittää vaikkapa webkapu ja kyseinen sana löytäisi oikean osoitteen.
Mitä jos ei ole käyttäjätunnusta?	Arvaamalla tunnuksia ei sisään ohjelmaan voi päästä.	Galileo Oy:n ja Alavus Ikkunat Oy:n sivuilla on yhteystiedot joista voi tiedustella käyttäjätunnusta.
Mitä jos on unohtunut käyttäjätunnus tai salasana?	Galileo Oy:n ja Alavus Ikkunat Oy:n sivuilla on yhteystiedot joista voi tiedustella käyttäjätunnusta ja salasanaa.	Automaattinen uuden salasanan pyyntö toiminto, jolloin tunnus ja uusi salasana toimitetaan suoraan käyttäjän omaan sähköpostiin.
Mitä jos kirjoitetaan salasana väärin monta kertaa, meneekö tunnus lukkoon tai muuta vastaavaa?	Tunnus ei mene lukkoon, mutta ohjelma ei myöskään anna palautetta missä mennään vikaan.	Ohjelmalle voisi tehdä määrittämisen joka ilmoittaisi onko käyttäjätunnus väärin vai salasana ja miten tulee toimia jos jompikumpi on väärin.
Mitä jos painetaan Peruutuspainiketta kirjautumisikkunassa?	Ohjelma avaa tyhjän, valkoisen sivun.	Paluu voisi olla takaisin kirjautumissivulle.
Muuta huomioitavaa:		Kehitysehdotuksia / huomioita:
Nyt kirjautumis-sivulla on aika vähän informaatiota ja sivun käyttäjän vaikea löytää esimerkiksi Alavus Ikkunoiden pääsivulle.		Voisiko pääsivulla olla linkki myös Alavus Ikkunat Oy:n sivulle? Voisiko pääsivulla olla linkki myös Galileo Oy:n sivulle?
Jos ohjelman käyttäjällä ei ole ohjetta WebKAPU ohjelmiston käyttöön.		Voisiko sivuilla olla linkki WebKAPU:n ohjedokumenttiin?

Jos henkilö löytää sivuille sattumalla ensimmäistä kertaa, ei sivuilla ole juuri mitään informaatiota mikä sivu ja kirjautuminen on kyseessä.		Voisiko sivuilla olla hieman enemmän informaatiota WebKA-PU:sta? Esimerkiksi lyhyt mainos voisi olla ohjelmistosta, sekä yhteystiedot.
---	--	--

TAULUKKO 1. Testitapaus 1: Tarjouslaskija kirjautuu sisään ohjelmaan.

5.2.2 Testitapaus 2: Tarjouslaskija syöttää asiakkaan henkilötiedot ja toimitustiedot järjestelmään

Tarjouslaskija syöttää asiakkaan henkilötiedot ja toimitustiedot järjestelmään.

Testitapauksen nimi: Tarjouslaskija syöttää asiakkaan henkilötiedot ja toimitustiedot järjestelmään.	Testauksen tarkoitus: Mitä virhetilanteita voi aiheutua tietojen syöttämisessä, miten käytettävyyttä ja ulkonäköä voidaan parantaa.	
Toteutettu testaus:	Testauksen tulos:	Kehitysehdotuksia / huomioita:
Kirjaudutaan järjestelmään.	Kirjaudutaan onnistuneesti sisään WebKAPU ohjelmaan. Aukeaa pääsivu jossa otsikko "DOKUMENTIT / KÄYTTÄJÄNIMI".	Vaikka selainikkunaa muuttaa erikokoiseksi, pysyy tarjoussivua mukavasti luettavana. Sivun skaalautuu hyvin eri kokoihin.
DOKUMENTIT-sivun informaatio.	DOKUMENTIT sivulla on listattu kaikki käyttäjän tekemät tarjoukset.	DOKUMENTIT sanana on hieman erikoinen. Mitä dokumentteja tarkoittaa? Voisiko olla enemmän esimerkiksi TARJOUKSET?
HAKU-kentän testaus. Hakuja voidaan tehdä nuolivalikon valinnoilla "Numero" ja "PVM".	"Numero" ja "PVM"; en saa kiinni logiikasta.	Haun toimintaa voisi tarkistaa ja ehkä laajentaa toimintoja.
Hakuja voidaan tehdä valinnoilla: tilaus, merkki, asiakas, päivä, yhteyshenkilö ja viite.	Haussa voidaan kertoa vai osa kentän tiedoista, esimerkiksi jos asiakas on Vainionpää, voidaan haku tehdä esimerkiksi tekstillä pää.	Haun jälkeen määritelty hakukenttä tyhjenee, hyvä ominaisuus. Hakua voisi kehittää esimerkiksi että useampaan hakukenttään voisi määritellä tietoa. Jos tilauskenttään määritellään vain muutama numero tilausnumerosta, ei haku toimi. Päivämäärä tulee olla oikeaan muotoon kirjoitettu, muuten haku ei toimi. Voisi olla esimerkki kirjoitettavasta muodosta.

Hakuja voidaan tehdä valinnoilla: tilaus, merkki, asiakas, päivä, yhteyshenkilö ja viite.	Haussa voidaan kertoa vai osa kentän tiedoista, esimerkiksi jos asiakas on Vainionpää, voidaan haku tehdä esimerkiksi tekstillä pää.	Haun jälkeen määritelty hakukenttä tyhjenee, hyvä ominaisuus. Hakua voisi kehittää esimerkiksi että useampaan hakukenttään voisi määritellä tietoa. Jos tilauskenttään määritellään vain muutama numero tilausnumerosta, ei haku toimi. Päivämäärä tulee olla oikeaan muotoon kirjoitettu, muuten haku ei toimi. Voisi olla esimerkki kirjoitettavasta muodosta.
Sivujen "Logout"-linkillä kirjaututaan ulos ohjelmasta.	Aukeaa Alavus Ikkunoiden sivu jossa kiitetään ohjelman käytöstä ja pyydetään sulkemaan selain.	Tässä ikkunassa voisi olla ohjelmaan uudelleen kirjautumisen mahdollisuus eli linkki takaisin kirjautumiseen. "Logout" voisi olla käännetty suomeksi eli "Kirjaudu ulos ohjelmasta" tai vastaava teksti. Huomiona: jos selainta ei suljeta ja mennään takaisin kirjautumissivulle, ohjelma kirjautuu sisälle ilman tunnusta ja salasanaa.
Uuden tarjouksen luonti aloitetaan napauttamalla linkkiä "Uusi tarjous".	Aukeaa henkilö- ja toimitustietojen täydennys ikkuna. Otsikkona sivulla lukee "OTSIKKO JA KÄYTTÄJÄNIMI". Sivulla ylhäällä lukee myös tarjouksen numero. Tämä numero syntyy juuri tässä vaiheessa kun "Uusi tarjous"-linkkiä napautetaan.	
Syötetään asiakas- ja toimitustiedot.		Kenttien sijoittelu, voisiko olla toisin / tosin sanoen toimisiko joku muu järjestys paremmin? Tarkennukset mitä kentät tarkoittavat esimerkiksi mitä muotoa tulee olla "merkki"? Onko sääntöjä? Mitä tarkoittaa "asiakasnimi"? "Posti", pitääkö olla postinumero sekä postitoimipaikka? Missä muodossa puhelinnumero tulee syöttää? jne.
Syötetään kenttien otsikoiden mukaiset tiedot.	Voiko valinnat olla tyhjiä, salliiko tyhjän kentän? Hyväksyykö ohjelma kenttiin sekä kirjaimia että numeroita? Esim postinumerokenttä, voiko olla kirjaimia?	Voisi tehdä ehtoja kenttiin eli jos ei ole syötetty oikeassa muodossa, ohjelma antaisi virheherjan ja ohjaisi oikeaan syöttötapaan. Esimerkkinä että sähköpostiosoitteessa tulee olla @-merkki, ellei ole ohjelma herjaa.
Pakolliset täytettävät kentät? Onko minimimäärää mitä tietoa tulee syöttää ennen tallennusta?	Antaako tallentaa vaikka henkilötiedot ovat vaillinaisesti täydennetty.	Voisiko pakolliset kentät olla esimerkiksi tummennetulla tai tähdellä merkityt?
Liikkuminen täydennettävissä kentissä.	Tab-painikkeella liikkuminen, eteneekö liikkuminen loogisesti eri valintalistoisissa?	Voisi miettiä parempaa järjestystä. Esimerkkinä osoitetiedot voisi olla ensin ja sitten puhelinnumerot. Nyt ne menevät sekaisin vuorotellen.
Syötetään tietoja kenttiin	Voiko kenttiin syöttää rajaton määrä tekstiä vai onko se rajattu?	Kentille on määritelty suhteellisen paljon merkkejä.

Poista-painike	Painettu Poista-painiketta; ohjelma esittää varmistussivun halutaanko tarjous todella poistaa. Valitaan joko linkki että poistetaan tai paluu linkistä palataan takaisin tarjoukseen.	Hyvä kun on vahvistussivu poistolle.
Muuta huomioitavaa:		Kehitysehdotuksia / huomioita:
Välilehdessä lukee RASA Dokumenttilistaus, mitä se tarkoittaa?		Voisiko olla joku muu nimitys, kuvaavampi esimerkiksi ajatellen uusia käyttäjiä?
Kenttien tietojen täydennys	Mitä tietoja kenttiin tulee syöttää?	Voisiko saada näkyviin tarkennuskommentin tietyistä kentistä? Esimerkiksi kun menee kentän päälle niin voisiko esiin tulla ohjeikkuna?
Oletiedot-linkki	Aukeaa "Valitse tarjoukselle oletustiedot". Ikkunassa voidaan syöttää koko tarjouksen kattavat oletustiedot kuten malli ja väri.	Voisi olla kuvaavampi nimi mitä tästä linkistä tapahtuu.

TAULUKKO 2. Testitapaus 2: Tarjouslaskija syöttää asiakkaan henkilötiedot ja toimitusehdot järjestelmään.

5.2.3 Testitapaus 3: Tarjouslaskija syöttää pyydetyn tarjouksen tuotteiden tiedot järjestelmään

Testitapauksen nimi: Tarjouslaskija syöttää pyydetyn tarjouksen tuotteiden tiedot järjestelmään.	Testauksen tarkoitus: Mitä virhetilanteita voi aiheutua tietojen syöttämisessä, miten käytettävyyttä ja ulkonäkö voidaan parantaa.	
Toteutettu testaus:	Testauksen tulos:	Kehitysehdotuksia / huomioita:
Klikataan linkkiä "Riveille".	Aukeaa ikkuna johon listautuu kaikki tarjouksen rivit eli kaikki tuotteet.	
Klikataan linkkiä "Uusi ikkuna" ja aloitetaan tarjouksen tietojen syöttäminen.	Aukeaa ikkuna jossa otsikkona on rivinumero esimerkiksi "Rivi 4". Rivi-ikkunassa on paljon syötettäviä kenttiä eli tietoja siitä millainen ikkuna halutaan. On nuolivalikoita, itse kirjoitettavia ja syötettäviä kenttiä.	
Liikkuminen kentissä tab-painikkeella.	Kentissä pystyy liikkumaan tab-painikkeella, joka nopeuttaa huomattavasti tietojen täydennystä.	Voisiko tab-painikkeella siirtymisjärjestystä parantaa? Esimerkiksi korkeus ja leveys valinnoissa ei olisi pakko käydä molemmissa mittayksiköissä vaan siinä kumpaan yleisemmin käytetään.

Kenttien täydentäminen. Voiko valinnat olla tyhjiä, salliiiko tyhjän kentän? Salliiiko minkälaiset tahansa mittayksiköt?	Jos esimerkiksi leveyden ja korkeuden syöttää väärässä muodossa, ohjelma antaa virheilmoituksen. Jos tiedot jättää syöttämättä antaa virheilmoituksen.	Jos pakollista tietoa jää syöttämättä, ohjelma ilmoittaa puuttuvasta tiedosta.
Nuolivalikoiden käyttäminen näppäimistöllä.	Nuolivalikoita voidaan käyttää näppäimistöltä käsin, jos tiedetään mitä valikot sisältää. Painamalla valikon vaihtoehdon ensimmäistä kirjainta, vaihtuu nuolivalikkoon kyseinen valinta.	Voisiko nuolivalikko aueta niin että vaihtoehdot näkyvät esimerkiksi kun nuolivalikon kohdalla painaa jotain kirjainta?
Muuta huomioitavaa:		Kehitysehdotuksia / huomioita:
Yleisnäkymä	Ikkuna ehkä hieman suuri ainakin testattavalla kannettavalla; syötettävää ikkunaa joutuu hieman vierittämään hiirellä.	Olisi ehkä hieman tiivistämisen varaa sivun asettelussa.

TAULUKKO 3. Testitapaus 3: Tarjouslaskija syöttää pyydetyn tarjouksen tuotteiden tiedot järjestelmään.

5.2.4 Testitapaus 4: Tarjouksen tulostaminen

Testitapauksen nimi: Tarjouksen tulostaminen	Testauksen tarkoitus: Testataan voidaanko tulostamista järjestelmästä tehostaa	
Toteutettu testaus:	Testauksen tulos:	Kehitysehdotuksia / huomioita:
Tarjouksen tiedot on syötetty, ollaan Rivit-näytöllä.	Kyseisellä näytöllä tulostaminen ei onnistu; painiketta tulostamiselle ei ole.	Sivulle voisi sijoittaa painikkeen jolla voisi tulostaa kyseisen sivun.
Siirytään askel taaksepäin, ollaan Rivivalinta-ikkunassa.	Kyseisellä näytöllä tulostaminen ei onnistu; painiketta tulostamiselle ei ole.	Voisi olla painike että saataisiin tulostettua tarvittaessa Rivivalinta-listaus.
Siirytään askel taaksepäin, ollaan Otsikko-ikkunassa.	Ikkunassa on linkki jonka kautta tulostaminen onnistuu. Kun linkkiä napautetaan aukeaa uusi ikkuna, joka ilmoittaa että tulostus on käynnissä. Samasta ikkunasta voidaan palata tarjoukseen takaisin.	Omalla koneellasi selaimen tietoturva esti tulostamisen, piti sallia selaimen ylälaidasta että sallitaan tiedoston lataaminen. Pitäisikö sivulla olla mahdollinen vinkki jos tulostaminen ei onnistu.
Aukeaa ikkuna jossa PDF-dokumentti voidaan joko avata tai tallentaa.	Oletetaan että käyttäjä tietää minne PDF-dokumentti tallennetaan. Oletetaan että asiakas osaa liittää tiedoston sähköpostiin. Jos tiedosto avataan, voidaan tarjous tulostaa paperiversiona asiakkaalle.	PDF-dokumenttimuotona on yleinen ja siksi hyvä.

TAULUKKO 4. Testitapaus 4: Tarjouksen tulostaminen.

5.2.5 Testitapaus 5: Opittavuus, tehokkuus, muistettavuus, virheettömyys, miellyttävyys ja hyödyllisyys

ISO on kansainvälinen standardointijärjestö. Tämä taho määrittelee käytettävyyden kokonaisuudeksi, joka kuvaa miten hyvin käyttäjät kykenevät käytössä olevia työvälineitä tiettyjen tehtävien suorittamiseen tietyssä ympäristössä, tiettyjen tavoitteiden saavuttamiseksi. (Kuutti 2003, 15.) ISO 9241-11 -standardin mukaan käytettävyys on mittari, jolla mitataan tuotteen käytön tuottavuutta, tehokkuutta ja miellyttävyyttä. Nämä arvioidaan aina suhteessa käyttäjiin sekä työhön ja käyttöympäristöön ja joihin tuote on tarkoitettu. ISO 9241-11 –standardin mukaiset käytettävyyden termit ovat:

- **Tuottavuus** tarkoittaa sitä, että tehtävät tulevat tehdyksi täydellisesti ja virheettömästi.
- **Tehokkuus** mittaa sitä, paljonko resursseja tuotteen käytössä tarvitaan henkilöinä, rahana ja aikana.
- **Miellyttävyys** kertoo, kuinka miellyttävä tuotetta on käyttäjien mielestä käyttää.

(Sinkkonen 2010.)

Ohjelmiston käytettävyydelle Jakob Nielsen antaa viisi laatukomponenttia: opittavuus, tehokkuus, muistettavuus, virheettömyys ja miellyttävyyys ja lisäksi Nielsen puhuu tuotteen hyödyllisyydestä. (Nielsen 2000, 145.) Seuraavaksi on avattuna Nielsenin viisi laatukomponenttia, sekä hyödyllisyys näkökulma rinnastettuna WebKAPU:n toimintoihin:

- **Opittavuus:** Kuinka helppoa käyttäjien on tehdä sovelluksen avulla perusasiat ensimmäisellä kerralla?
- **Tehokkuus:** Kuinka nopeasti käyttäjät pystyvät tekemään tehtävät, kun sovelluksen käyttö on jo opittu?
- **Muistettavuus:** Kuinka kauan heiltä menee saman tuottavuuden saavuttamiseen uudelleen, kun edellisestä sovelluksen käyttökerrasta on kulunut pitkämpi aika?
- **Virheettömyys:** Kuinka paljon käyttäjät tekevät virheitä, kuinka vakavia ne ovat ja kuinka helppoa niistä on toipua?
- **Miellyttävyyys:** Onko jotain mikä aina tai usein nousee esille epämiellyttävänä sovelluksen käyttämisessä?
- **Hyödyllisyys:** Sopiiko tuote hyvin tarjouslaskenta- ja tilaustenhallintasovellukseksi?

(Sinkkonen 2010.)

Testitapauksen nimi: Opittavuus, tehokkuus, muistettavuus, virheettömyys, miellyttävyyys ja hyödyllisyys	Testauksen tarkoitus: Minimoidaan toistuvat ongelmat, parannetaan opittavuutta, tehokkuutta, muistettavuutta, virheettömyyttä, miellyttävyyttä ja hyödyllisyyttä.	
Toteutettu testaus:	Testauksen tulos:	Kehitysehdotuksia / huomioita:
Opittavuus/käytettävyys: Kuinka helppoa käyttäjien on tehdä sovelluksen avulla perusasiat ensimmäisellä kerralla?	Ohjelmalle tulee olla hyvä, yrityskohmainen ohje. Ohje ohjelman käytöstä, sekä ohjeen tulee kattaa yrityksen standardit eli esimerkiksi missä muodossa mikäkin kenttä halutaan syötettävä. Syötettävien kenttien tekstitys helpposelkoista eli se ohjaa hyvin tarjouksen tekijää. Ongelmia kokemattomalla toimialan osajalla eli tulee tietää paljon teknistä tietoa ikkunoiden ja ovien eri tarvikkeista.	Voisiko kenttien nimistä aueta joissain tapauksissa opastava ja ohjaava ikkuna? Vaikkapa kun klikataan Merkki-sanaa, aukeaisi ohje, mitä tähän kenttään tulee syöttää. Ohje ohjelmaan helposti saataville, esimerkiksi linkiksi vaikkapa joka ikkunan näkymään. Sekä ohjelmakohtainen ohje, kuin yrityksen oma spesifinen ohjeistus. Tai näistä kahdesta ohjeesta tehty koottu, yhdistetty ohjeistus.
Tehokkuus: Kuinka nopeasti käyttäjät pystyvät tekemään tehtävät, kun sovelluksen käyttö on jo opittu?	Tehokkuuteen vaikuttavia tekijöitä ovat: <ul style="list-style-type: none"> ohjelman helpposelkoisuus, ikkuna näkyy aika kerralla liikkuminen kentissä on helppoa kenttiin kirjoitettava teksti hyvin ja yksikertaisesti ohjeistettu linkit aina ikkunan yläreunassa, niiden avulla on helppo liikkua eri ikkunoihin. 	Joitain pikatoimintoja voisi vielä miettiä, esimerkiksi enterpainikkeen käyttö eri ikkunoissa, sekä mahdolliset pikapainikkeet. Nämä eivät kuitenkaan saa hankaloittaa käyttöä eli ikkunoihin ei saa tulla liikaa pikakomentomerkkejä tms.
Muistettavuus: Kuinka kauan heiltä menee saman tuottavuuden saavuttamiseen uudelleen, kun edellisestä sovelluksen käyttökerrasta on kulunut pidempi aika?	Ohjelmisto on sen verran selkeä että jos sitä on aiemmin käyttänyt, on käyttäminen helppoa pidemmänkin tauon jälkeen. Ongelmaksi uskon ennemmin muodostuvan toimialan omista säännöksistä, määräyksistä, toteutustavasta, tuotteiden määrittelyistä, hinnoittelusta jne.	
Virheettömyys: Kuinka paljon käyttäjät tekevät virheitä, kuinka vakavia ne ovat ja kuinka helppoa niistä on toipua?	Virheet eivät tarjouksen tekemisen vaiheessa ole kovin vakavia. Virheet voi huomata mahdollisesti vasta tulosteessa, mutta ne on helppo korjata. Vakavimmat virheet voisi olettaa tulevan siinä vaiheessa kun tuotteet lähtevät tuotantoon toteutettavaksi. Siinä vaiheessa ei sovi olevan virheitä esimerkiksi ikkunan mallissa tai värissä.	Onko mahdollista toteuttaa oikolukua teksteihin? Ohjeistuksen tärkeys ja että se on kaikille helpolla saatavissa. Ei ainoastaan ohjelmiston ohjeistus, vaan myös tehtaantottama ohjeistus.
Miellyttävyyys: Onko jotain mikä aina tai usein nousee esille epämiellyttävänä sovelluksen käyttämisessä?	En löydä ohjelmasta mitään epämiellyttävää.	

<p>Hyödyllisyys: Sopiiko tuote hyvin tarjouslaskenta- ja tilaustenhallintaso- vellukseksi?</p>	<p>WebKAPU soveltuu erittäin hyvin tarjouslaskentaan. Loistava keksintö, sillä sitä voidaan käyttää missä paikassa tahansa ja millä koneella tahansa, jos vain on nettiyhteys ja tunnukset järjestelmään. Ohjelma on helppokäyttöinen ja uskon että sen omaksuminen uudelle henkilölle ei vie kauan aikaa.</p>	
---	--	--

TAULUKKO 5. Testitapaus 5: Opittavuus, tehokkuus, muistettavuus, virheettömyys, miellyttävyys ja hyödyllisyys.

Opinnäytetyössä selvitettiin virheettömyyttä Jukka Raisiota haastatteleamalla:

- Kuinka paljon käyttäjät tekevät virheitä?
- Ovatko virheet vakavia?
- Onko niistä helppo toipua?

Raisio totesi että käyttäjävirheet ovat pääasiassa asiakkaan, myyjän ja tehtaan välisen asioiden ymmärtämisen eroja. Asiakkaalle on saatettu luvata asioita, joita ei olekaan muistettu kirjata tarjoukselle. Asiakas ei aina huomaa tilausvahvistuksesta jos jotain uupuu, vaikka hänellä olisi tilausvahvistuksen tarkistamisen velvollisuus. Ohjelman varsinaisessa toiminnassa tarjouslaskija voi jättää tallettamatta tietoja, ja mahdollisesti kuvitella että on jo tallennuksen tehnyt. Virhe voi olla vakava lähinnä kustannussyistä, sillä jos tuote joudutaan tekemään uudelleen virheellisen tiedon takia, tuottaa se ylimääräisiä kuluja. Kokeneelle ja useimmille käyttäjille ei tapahdu käytännössä virheitä.

(Raisio 2009)

6 JOHTOPÄÄTÖKSET

Galileo Oy:ssä Gapassi-projekti on aloitettu jo vuonna 2006. Siitä eteenpäin kehitystä on voinut eteenpäin lähinnä Jukka Raisio muiden projektien rinnalla. Opin näytetyön kautta projektiin liitettiin vuonna 2008. Tuolloin ohjelmistosta oli beta-versio toteutettu ja ohjelmistolla oli muutama testikäyttäjä. Vuoden mittaan testikäyttäjät lisääntyivät pilottiyrityksessä Alavus Ikkunat Oy:ssä. Tänä päivänä WebKAPU:lla on jo neljä yritystä käyttäjänään. Yritykset ovat Alavus Ikkunat Oy, Lammin Ikkuna, Salpa Ikkunat ja Arcon Engineering. Ohjelmiston käyttäjiä on täten noin 100. Ohjelmiston käyttäminen ja asiakaskunnan laajeneminen on edennyt tavoitteiden mukaisesti, mutta kasvuun pyritään edelleen. Gapassi-projekti on tähän mennessä osoittanut toimivuutensa ja käyttökelpoisuutensa tuotantoympäristöissä.

Yrityksen, esimerkiksi Alavus Ikkunat Oy:n, käyttäessä pientä ohjelmointiyritystä sovelluksen-ohjelmoinnissa, takaa erittäin hyvän asiakaspalvelun. Tällaisessa yhteistyössä asiakas ja ohjelmoija voivat keskustella keskenään suoraan ilman välisiä. Jos asiakkaan ja ohjelmoijan työsuhde on jatkuva eli ilman aikarajoitteita, ei ole painetta saada tuotantokäyttöön täysin valmista ohjelmistoa, vaan kehitystyötä voidaan jatkaa beta-version käyttöönoton jälkeenkin. Tämä on ollut etu myös Alavus Ikkunat Oy:n ja Galileo Oy:n yhteistyössä. Galileo Oy on edelleen jatkanut muun muassa Alavus Ikkunat Oy:n WebKAPU:n jatkokehitystä.

Gapassi-ohjelmiston ylläpito ei vaadi enää pääsovellukseen eli KAPU:uun merkittäviä muutoksia. KAPU-sovellukseen tehdyt korjaukset ovat suoraan käytettävissä WebKAPU:n kehityksessä. Käyttöliittymän yrityskohtainen muokkaus ja kieliversioiden teko helpottui huomattavasti WebKAPU:n selainkäyttöliittymän ansiosta. Vastaavan käyttäjäkohtaisen muokkauksen teko KAPU:ssa olisi ollut huomattavasti vaativampi kuin erillisen käyttöliittymän tarkentaminen.

Tämä ohjelmistoprojekti on kestänyt varsin kauan, jos verrataan isoon ohjelmointitaloon, jossa sovellus toteutetaan asiakkaalle ja otetaan tuotantoon esimerkiksi yhdessä vuodessa. Näissä isojen ohjelmistotalojen projekteissa on kyse monen henkilön työmäärästä ja pohjalla hyödynnetään jo valmiina olevaa materiaalia eli valmista sovellusta tai valmista ohjelmointikoodia. Gapassi-projekti on hieman erilainen luonteeltaan. Se ei ole kertaluonteinen projekti vaan, projekti jota on tarkoitus viedä eteenpäin askel kerrallaan. Eteneminen tapahtuu joko niin, että asiakas pyytää uusia, haluamiaan toimintoja ohjelmistoonsa tai toinen vaihtoehto on se että Galileo Oy kehittää ohjelmistoon uuden hyvän toiminnon ja myy tämän toiminnon asiakkaalleen. Tämä on tietysti hyvä tapa Galileo Oy:lle sillä tuloja eri yrityksiltä on ympäri vuoden. Jos ohjelmistoprojekti olisi kertaluonteinen, ei se tuottaisi tuloa valmistumisen jälkeen. Nykyinen malli takaa myös asiakkaille juuri heille räätälöidyn ohjelmiston mahdollisuuden.

Gapassi-projektissa on aloitettu suunnittelemaan ja toteuttamaan ohjelmistoa alusta alkaen. Toki pohjalla on toiminut jo aikaisemmin toteutettu KAPU-ohjelmisto, jonka perusteella WebKAPU:a on toteutettu. Ohjelmointityötä se ei ole kuitenkaan vähentänyt. Ohjelmistoa on rakennettu KAPU-pääsovelluksen kehityksen ja ylläpidon rinnalla. WebKAPU:lle ei ollut varsinaista yritystä tilaajana vaan kehityskohteen näki Jukka Raisio ja lähti viemään projektia eteenpäin. Koska sovellukselle ei ollut valmiiksi tilaajaa, oli edettävä omin voimin mikä tarkoitti tietenkin sitä että rahoitusta ei tullut mistään ja työtä oli tehtävä niin sanotusti oman työn ohessa. Gapassi-projektille haettiin hankerahoitusta, mutta valitettavasti rahoituksia myönnettiin lähinnä vain laitehankintoja sisältäviin projekteihin, ei niinkään työn toteuttamiseen. Näin ollen rahoitusta ei saatu.

6.1 Gapassi-ohjelmiston jatkokehitys

Galileo Oy on vienyt Gapassi-projektia eteenpäin edelleen. Gapassi on käytössä neljässä yrityksessä ja Galileo Oy toteuttaa heille edelleen tukea ohjelmistoon, sekä kehittää edelleen ohjelmaan asiakkaan vaatimusten ja tarpeiden mukaan unohtamatta toteuttajan itse ideoimia hyviä kehityskohteita.

Tällä hetkellä WebKAPU kattaa tarjousten- ja tilaustenhallinta toiminnot. Jatkossa sovellusta on tarkoitus kehittää siten että se kattaa myös raportoinnin, työsuunnittelun, laskutuksen, osto- ja lähetystoiminnot, sekä esimerkiksi ikkunamallien ja nimikkeiden hallinnan. Tämä tarkoittaa sitä että login-sovelluksen jälkeen käyttäjä pääsee valikkoon, jossa hän voi valita sovelluksen johon haluaa kirjautua. Sovellusvalikoima määrittyy käyttäjälle annettujen oikeuksien mukaisesti.

Jatkossa on tarkoitus kehittää myös ohjelmiston testausta. Rutiinitestauksia on tavoite automatisoida. Tällä hetkellä automatisointia ei ole. Kokemuksen pohjalta testaus pystytään pitkälti automatisoimaan ja virheet saadaan kiinni jo aikaisessa testausvaiheessa.

6.2 Gapassi-projektin tulevaisuus

Tavoitteita on vielä saavutettavana. Galileo Oy:n tulee hankkia uusia asiakkaita ja vallata uusia teollisuuden tuotannonaloja. Galileo Oy:llä on edelleen tavoite laajentua eri teollisuuden aloille, siten että ei rajoituttaisi pelkästään ovi- ja ikkunateollisuuteen. Esimerkki muusta teollisuuden alasta on metalliteollisuus tai vaikka seinäelementtejä valmistava yritys. Mahdolliseen Gapassi-jatkoprojektiin asiakkaaksi soveltuu lähinnä mikä tahansa tuotannon yritys, joka valmistaa kappaletavaraa eli ohjelmisto olisi samoin tavoin toteutettavissa kuin ikkuna- ja ovituotannon alalla. Tavoitteista ei sovi unohtaa jo olemassa olevien, KAPU- ja Gapassi-ohjelmiston jatkuvaa eteenpäin viemistä, ohjelmiston jatkokehitystä. Sitä tulee jatkaa olemassa olevien asiakkaiden kanssa ja samalla uusia asiakkaita hankkien.

LÄHTEET

- Aaltola, J. & Syrjälä, L. 1999. Tiede, toiminta ja vaikuttaminen. Atena.
- Aaltola, J. & Valli, R. 2001. Ikkunoita tutkimusmetodeihin 1. Jyväskylä: PS-kustannus.
- Fowler, M. & Kendal, S. 2002. UML. Addison-Wesley. Jyväskylä: Docendo.
- Haikala, I. & Märijärvi, J. 2002. Ohjelmistotuotanto. Helsinki: Talentum.
- Heikkilä, T. 2002. Tilastollinen tutkimus. Helsinki: Edita Prima Oy.
- Heikkinen, H. & Huttunen, R. & Moilanen, P. 1999. Siinä tutkija missä tekijä. Jyväskylä: Atena.
- Heikkinen, H. UML-kuvauskielten käyttö ohjelmistojen vaatimusmäärittelyissä. [Verkkajulkaisu] [Viitattu 27.2.2010]. Saatavana: <http://www.cs.helsinki.fi/u/taina/sem/gradu/k-2008/hh-seminaariraportti.PDF>
- Jaakonhuhta, H. 2007. Tietotekniikan sanakirja. Readme.
- Järvinen, P. & Järvinen, A. 2004. Tutkimustyön Metodeista. Tampere: Opinpaja.
- Karlsson, Å. & Marttala, A. 2001. Projekti kirja. Onnistuneen projektin toteuttaminen. Helsinki: Kauppakaari.
- Kettunen, S. 2009. Onnistu projektissa. Helsinki: WSOYpro Oy.
- Kettunen, S. Filenius, M. 1998. Elektroninen kaupankäynti. Jyväskylä: Gummerus.
- Koskimies, K. & Mikkonen, T. 2005. Ohjelmisto-arkkitehtuurit. Helsinki: Talentum Media Oy.
- Koskimies, K. 2000. Oliokirja. Helsinki: Kauppakaari.

Kuutti, W. 2003. Käytettävyys, suunnittelu ja arviointi. Helsinki: Talentum.

Lehtimäki, Timo, 2006. Ohjelmistoprojektit käytännössä. Helsinki: Readme.

Likitalo, H. Rissanen, R. 1998. Tutkimusmenetelmät: Menetelmätietoutta tradenomiopiskelijoille: opetusmoniste. Kuopio: Pohjois-Savon ammattikorkeakoulu.

Lukkari, U. 2004. Digitaalisen sisällöntuotantoprojektin hallinta. Helsinki: IT-Press.

Nielsen, J. 2000. WWW suunnittelu. Helsinki: IT-Press.

Paalanen, A. Alavus Ikkunat Oy, haastattelu. Vantaa 13.4.2009.

Raisio, Jukka. 1996. Diplomityö. Tietojärjestelmä puusepänteollisuuden myyntiin ja tuotannonohjaukseen. Tampere: Tampereen yliopisto.

Raisio, J. 2007. Galileo Oy, hankehakemus dokumentit.

Raisio, J. Galileo Oy, atk-päällikön haastattelu. Alavus 21.12.2007.

Raisio, J. Galileo Oy, atk-päällikön haastattelu. Alavus 1.4.2008.

Raisio, J. Galileo Oy, atk-päällikön haastattelu. Alavus 21.3.2009.

Ruuska, K. 2005. Pidä projekti hallinnassa. Tampere: Talentum.

Sinkkonen, I. Käyttöliittymät ja käytettävyys. [Verkkoartikkeli]. [Viitattu 1.3.2010]. Saatavana http://www.adage.fi/julkaisut/arkisto/kayttoliittymat_ja_kaytettavyys.html . Adage.

LIITTEET

LIITE 1 Galileo Oy, Gapassi-hankerahoitushakemus

LIITE 2 Galileo Oy, pelkistetty tekninen toteutusluonnos

LIITE 3 Johanna Vainionpää, WebKAPU-ohjekirja käyttäjille