

Käki-projekti

Huttunen Sami
Lamminmäki Tatu
Lappi Juha
Pelkkikangas Eija

Vaatimusmäärittely
18.11.2003
Versio 2.0

Jyväskylän yliopisto
Tietotekniikan laitos

Tietoja dokumentista

Tekijät

Huttunen Sami	(samih@cc.jyu.fi)
Lamminmäki Tatu	(tplammin@cc.jyu.fi)
Lappi Juha	(jujolapp@cc.jyu.fi)
Pelkkikangas Eija	(ekpelkki@cc.jyu.fi)

Yhteystiedot

Agora, huone C223.3, puh.nro. 014-260 49 65. Projektioorganisaation sähköpostilista on kaki@korppi.jyu.fi, jonka arkisto on nähtävillä osoitteessa <http://korppi.it.jyu.fi/list-archive/kaki/>. Kotisivu löytyy osoitteesta <http://kotka.it.jyu.fi/kaki/>.

Työn nimi

Käki-projekti, vaatimusmäärittely

Työ

Vaatimusmäärittely tietotekniikan sovellusprojektiin

Tiivistelmä

Käki-projekti toteuttaa WWW-pohjaisen henkilökohtaisen opiskelusuunnitelman suunnittelu- ja seurantasovelluksen olemassaolevaan Korppi-järjestelmään. Vaatimusmäärittelyssä kuvataan sovelluksen toiminnalliset ja tekniset vaatimukset sekä käyttäjryhmien toimintotapaukset.

Avainsanat

Henkilökohtainen opiskelusuunnitelma, opiskelujen suunnittelu, opinto-ohjaus, urasuunnittelu, WWW-sovellus, Korppi-järjestelmä, joustavuus, käytettävyys.

Versiohistoria

Versio	Päiväys	Tehnyt	Muutokset
0.1	28.9.2003	Tatu Lamminmäki	Ensimmäinen versio
0.2	6.10.2003	Tatu Lamminmäki, Sami Huttunen	Vaatimusten täsmentäminen
1.0	8.10.2003	Tatu Lamminmäki, Sami Huttunen	Tarkennukset projektipalaveria varten
1.1	14.10.2003	Sami Huttunen	Santasen muutoksia versioon 1.0 lukujen 1, 2, 3 ja 6 osalta. Luvut 4, 5 ja 7 ovat keskeneräisiä.
1.5	28.10.03	Sami Huttunen	Kaikki luvut ovat valmiita. Dokumentin sisällön muokkaaminen täsmällisemmäksi.
1.6	2.11.03	Sami Huttunen	Santasen muutokset versioon 1.5.
2.0	18.11.03	Sami Huttunen	Pieniä korjauksia hyväksytyyn versioon 1.6.

Sisällysluettelo

1. Johdanto	1
2. Termejä	2
2.1. Aihealueeseen liittyvät termit	2
2.2. Käytettyihin tekniikoihin liittyvät termit	2
3. HOPS-sovelluksen taustaa ja tavoitteita.....	4
3.1. Tutkintovaatimukset ja opintokokonaisuudet	4
3.2. HOPSin laatimisprosessi tällä hetkellä	4
3.3. HOPS-sovelluksen yleiset tavoitteet.....	4
4. Sovelluksen toiminnalliset vaatimukset	5
4.1. Sihteerin toiminnalliset vaatimukset.....	5
4.2. Opintojen ohjaajan toiminnalliset vaatimukset.....	5
4.3. Hyväksyjän toiminnalliset vaatimukset	5
4.4. Opiskelijan toiminnalliset vaatimukset.....	5
4.5. Opettajan toiminnalliset vaatimukset.....	6
4.6. Vieraan toiminnalliset vaatimukset.....	6
4.7. Jatkokehityksessä huomioitavia asioita	6
5. Sovelluksen tekniset määrittelyt ja vaatimukset.....	7
5.1. Toteutustekniikka ja palvelinratkaisut	7
5.2. Tuetut WWW-selaimet	7
5.3. Muita teknisiä vaatimuksia	7
6. Käyttäjryhmien toimintotapaukset.....	8
6.1. Toimintokaavio	9
6.2. Opintokokonaisuuksien suunnitteleminen ja koostaminen kursseista	10
6.3. Tutkintovaatimusten koostaminen opintokokonaisuuksista	10
6.4. Yhteenvetoraportin tekeminen opiskelijoiden valinnoista.....	11
6.5. Opintokokonaisuuden muokkaus, perusteleminen ja tallentaminen.....	11
6.6. Poikkeavan opintokokonaisuuden hyväksyminen	12
6.7. Tutkintovaatimuksen vaatimuksen valitsemien HOPSiin.....	12
6.8. Opintokokonaisuuksien lisääminen HOPSiin.....	13
6.9. Opintokokonaisuuksien poistaminen HOPSista	13
6.10. Opintokokonaisuudessa olevien vaihtoehtojen ja valinnaisten kurssien valinta.....	13
6.11. Opintokokonaisuuksien ja kurssien aikataulutus	14
6.12. Tutkintovaatimuksen, opintokokonaisuuksien ja kurssien perustelu.....	14
6.13. Opiskelusuunnitelman raportointi.....	15
6.14. Tutkintovaatimusten ja opintokokonaisuuksien tarkastelu.....	15
7. Käki-ryhmän toteuttama sovelluksen osa.....	16
8. Yhteenvedo	17
Lähteet.....	18

1. Johdanto

Käki-projekti on Jyväskylän yliopiston tietotekniikan laitoksella toteutettava Sovellusprojekti. Kyseinen opiskelijaprojekti suunnittelee ja toteuttaa Jyväskylän yliopistolle henkilökohtaisten opiskelusuunnitelmien laadinta-, seuranta- ja raportointisovelluksen Korppi-järjestelmään. Sovelluksen osia ovat opiskelijan opintojen suunnittelu, opintojen seuranta ja raportointi.

Dokumentissa kuvataan sovelluksen toiminnalliset ja tekniset vaatimukset sekä käyttäjäryhmien toimintotapaukset. Vaatimusmäärittely esittää sovelluksesta selkeän kokonaiskuvan, jonka sovelluksen tulevat käyttäjät voivat helposti ymmärtää. Vaatimusmäärittely vastaa kysymykseen, ”**mitä toimintoja ja tietoja sovelluksen tulee sisältää**”.

Projektin taustoja, aikataulua, tehtäviä ja muita läpivientiin liittyviä asioita on kuvattu erillisessä projektisuunnitelmassa. Sovelluksen sisäinen rakenne ja tekniset (ohjelmointi) ratkaisut esitellään sovellussuunnitelmassa. Projektin osapuolten roolit, velvollisuudet ja oikeudet on esitelty projektisopimuksessa.

Luvussa 2 esitellään aiheeseen liittyviä termejä. Luvussa 3 kuvataan sovelluksen taustaa ja tavoitteita. Luvussa 4 määritellään kehitettävän sovelluksen toiminnalliset vaatimukset sekä luvussa 5 tekniset määrittelyt ja vaatimukset. Luvussa 6 esitellään käyttäjäryhmien toimintotapaukset. Luvussa 7 kuvataan Käki-ryhmän toteuttama sovelluksen osa.

2. Termejä

Luvussa kuvataan projektin aihealueeseen liittyvää, dokumentissa käytettyä termistöä.

2.1. Aihealueeseen liittyvät termit

Käki-projektin aihealueeseen ja Korppi-järjestelmään liittyviä termejä ovat seuraavat:

HOPS	eli henkilökohtainen opiskelusuunnitelma (tai opintosuunnitelma) on opiskelijan laatima ja opinto-ohjaajan hyväksymä suunnitelma opintojen suorittamisesta.
JOO	eli joustava opinto-oikeus on yliopistojen välinen sopimus yliopistorajat ylittävästä opiskelusta.
JOPS	on opiskelijan laatima suunnitelma jatko-opintojen suorittamisesta.
Korppi	on Jyväskylän yliopiston tietotekniikan laitoksella kehitetty opintotietojärjestelmä.
Korvaavuus	on tapaus, jossa opiskelija korvaa esimerkiksi aikaisemmilla opinnoillaan tutkintoon kuuluvia opintoja.
Kotka	on Korppi-järjestelmän tietokanta ja henkilötietojen hallintaosio.
Opintokokonaisuus	on yhdestä tai useammasta kurssista muodostettu kokonaisuus. Kurssit voivat olla pakollisia, valinnaisia tai vaihtoehtoisia. Opintokokonaisuus voi olla tasoltaan esimerkiksi approbatur -, cum laude approbatur -, laudatur -tasoinen.
Oppimispäiväkirja	on opiskelijan kirjoittama muistio oppimistaan asioista, niiden omaksumisesta ja oppimisen analysoinnista.
Portofolio	(tai näytesalkku) on kokoelma esimerkiksi opiskelujen tai työn aikana tuotetuista materiaaleista. Portfolioon liittyy oman oppimisprosessin kuvaus sekä itsereflektio, jossa opiskelija pohtii omaa oppimistaan sekä jatkosuunnitelmiaan.
Tutkintovaatimus	koostuu opintokokonaisuuksista.

2.2. Käytettyihin tekniikoihin liittyvät termit

Käytettäviin työkaluihin ja tekniikoihin liittyviä termejä ovat seuraavat:

Apache	on avoimen lähdekoodin WWW-palvelinohjelmisto, joka palauttaa asiakkaan HTTP-protokollalla pyytämän HTML-sivun.
CSS	eli Cascading Style Sheets on HTML-sivujen ulkoasun määrittelyyn käytetty kieli.
CVS	(Concurrent Version System) on ohjelmistokehityksessä käytettävä versionhallintajärjestelmä.
HTML	(Hyper Text Markup Language) on merkkäuskieli tekstin sisällön ja rakenteen esittämiseen.
HTTP	on mm. Internetin WWW-palvelussa käytössä oleva tiedonsiirtoprotokolla.
Java	on Sunin kehittämä laitteistoriippumaton olio-ohjelmointikieli.

JDBC	eli Java Database Connectivity määrittelee Java-pohjaisen ohjelmointirajapinnan SQL relaatiotietokantoihin
JSP	eli JavaServer Pages on servlettien laajennus, joka liittyy dynaamisten WWW-sivujen tuottamiseen. JSP sivut koostuvat HTML:stä ja JSP-komponenteista, jotka voivat sisältää mm. puhdasta Java-koodia. Palvelimen Tomcat ohjelmisto kääntää JSP sivut servleteiksi ennen esittämistä.
Palvelin	on WWW-sovellusten tapauksessa ohjelmisto, joka palvelee asiakkaana toimivien selainten pyyntöjä.
PostgreSQL	on eräs ilmainen relaatiotietokannan hallintajärjestelmä.
Selain	on ohjelma, joka käyttäjän koneella tulkaa HTML-kieliset sivut kuvaruudulla esitettävään muotoon.
Servletti	on Javalla ohjelmoitu ohjelmakomponentti, joka sijoitetaan WWW-palvelimeen. Servlet tuottaa vastaussivun asiakkaan pyyntöön..
SQL	eli Structured Query Language on kieli, jolla relaatiotietokannassa olevaa dataa voi määritellä, hakea ja muokata.
Tietokanta	on kokoelma yhteen liittyvää dataa, joka on sijoitettu tietokantatauluihin.
Tomcat	on ohjelmisto, jota käytetään servlet- ja JSP-moottorina mm. Apache palvelimella. Välittää asiakaspyynnön servletille ja toimittaa sen tuottaman vastauksen takaisin pyytäjälle.
WWW	(World Wide Web) on maailmanlaajuinen tietoverkko.

3. HOPS-sovelluksen taustaa ja tavoitteita

Tässä luvussa kuvataan tutkintovaatimuksia ja opintokokonaisuuksia, HOPSien laatimisprosessia sekä HOPS-sovelluksen yleisiä tavoitteita.

Projektin yleistä taustaa on kuvattu tarkemmin projektisuunnitelman luvussa 3.

3.1. Tutkintovaatimukset ja opintokokonaisuudet

HOPS-sovelluksen lähtökohtana on, että opiskelija tähtää tutkintonsa suorittamiseen pääaineensa tutkintovaatimuksen mukaisesti. Tutkintovaatimuksen hän voi valita ajalta, jolloin hän on ollut opiskelijana Jyväskylän yliopistossa. Tutkintovaatimukset koostuvat opintokokonaisuuksista (ei siis pelkästään yksittäisistä kursseista), jotta opiskelija saa riittävän laajan käsityksen pääaineeseen ja suuntautumisvaihtoehtoon liittyen. Vastaavasti opintokokonaisuuden ideana on antaa opiskelijalle riittävä käsitys aineen osakokonaisuuksista tason määrätessä tarkastelun syvyyden.

Opiskelijan edelliset opinnot ja työkokemus huomioidaan HOPSissa korvaavuuksien kautta. Korvaavuudet voivat liittyä opintokokonaisuuksiin tai tutkintovaatimuksiin.

3.2. HOPSin laatimisprosessi tällä hetkellä

HOPSien laatiminen Jyväskylän yliopistossa on nykyisin hyvin laitoskohtaista. HOPS-sovelluksella pyritään yhdenmukaistamaan HOPSien laatimisprosessia. Seuraavassa on kaksi toisistaan poikkeavaa esimerkkiä HOPSien laatimisesta. Ensimmäinen esimerkki kuvaa tämänhetkistä HOPSien laatimisprosessia tietotekniikan laitoksella. Toinen esimerkki kuvaa HOPSien laatimista aikuiskouluttajan pedagogissa opinnoissa.

Tietotekniikan laitoksella laaditaan HOPSeja vain muuntokoulutettaville ja vaihto-opiskelijoille, joilla on paljon aiempia opintosuorituksia muista korkeakouluista. HOPSin laatimisprosessi alkaa opiskelun alussa opiskelijan ja opinto-ohjaajan tai HOPSeista vastaavan henkilön palaverilla. Siinä käydään läpi opiskelijan työ- ja opiskeluhistoria sekä sovitaan mahdollisista korvaavuuksista ja tehdään tämän pohjalta opiskelijalle HOPS. HOPSin allekirjoittavat laitoksen varajohtaja ja opiskelija, jonka jälkeen se on virallinen sopimus laitoksen ja opiskelijan välillä.

Tänä vuonna aikuiskouluttajan pedagogisia opintoja opiskelee 85 opiskelijaa ja kaikki tekivät HOPSin. HOPSien rakenne on väljä, ne ovat yksilöllisiä ja ehdottomasti henkilökohtaisia. HOPSissa yhdistetään työ ja opiskelu. Tärkeintä HOPSissa on omien pedagogisten taitojen reflektiivinen tarkastelu. HOPSien tarkoituksena on saada opiskelijat perehtymään omiin pedagogisiin kehitystarpeisiinsa ja vastaamaan kysymykseen: ”Miten haluan opettajana kehittyä tämän vuoden aikana?”. HOPSin laatiminen alkaa kesällä, kun opiskelija saa hyväksymiskirjeen opintoihin. HOPSia viimeistellään syksyllä ja se hyväksytään lokakuun puolessa välissä.

3.3. HOPS-sovelluksen yleiset tavoitteet

Sovellus tarjoaa opiskelijalle välineen, jonka avulla opiskelija pystyy laatimaan ja ylläpitämään henkilökohtaista opiskelusuunnitelmaansa. WWW-ympäristöön toteutettuna sovellus on käytettävissä ajasta ja paikasta riippumattomasti.

Opintojen ohjaajalle opiskelijan laatima opiskelusuunnitelma toimii pohjana ohjauskeskustelulle ja siinä sovittujen asioiden kirjaamiselle. Sovellus tarjoaa ohjaajalle oman näkymän, jossa esimerkiksi opiskelijan suunnitelma voidaan muokata.

Laitokset pystyvät käyttämään järjestelmän tietoja pohjana opetuksen suunnittelussa, ajoituksessa ja resursoinnissa. Järjestelmästä esimerkiksi saadaan selville, milloin kullekin kurssille olisi kysyntää.

4. Sovelluksen toiminnalliset vaatimukset

Luvussa esitellään sovelluksen sisältämät **kuusi** käyttäjäryhmää ja niiden toiminnalliset vaatimukset. Käyttäjäryhmien toiminnot kuvataan tarkemmin luvussa 6.

4.1. Sihteerin toiminnalliset vaatimukset

Sihteeri käyttää sovellusta opintokokonaisuuksien suunnittelemiseen ja koostamiseen kurseista. Hän voi määritellä opintokokonaisuudessa olevien kurssien pakollisuuden, vaihtoehtoisuuden ja valinnaisuuden.

Sihteeri koostaa myös tutkintovaatimukset opintokokonaisuuksista. Tutkintovaatimus voi sisältää kurssien tapaan pakollisia, vaihtoehtoisia ja valinnaisia opintokokonaisuuksia.

Sihteeri voi luoda raportteja opiskelijoiden opintokokonaisuuksien ja kurssien valinnoista sekä opintojen etenemisestä. Laitos voi hyödyntää raportteja opetuksen suunnittelussa.

Sihteeri voi tutustua tutkintovaatimukseen ja opintokokonaisuuksiin kirjautumatta Korppi-järjestelmään.

4.2. Opintojen ohjaajan toiminnalliset vaatimukset

Opintojen ohjaajalla on oppilaan käyttöoikeuksien lisäksi oikeus käyttää sovellusta henkilökohtaisten opintokokonaisuuksien ja tutkintovaatimusten muokkaamiseen, kommentoimiseen ja tallentamiseen. Muokkaaminen tapahtuu yleensä palaverissa opiskelijan kanssa, jossa keskustellaan tarvittavista muutoksista esimerkiksi korvaavuuksien osalta. Muutetun opintokokonaisuuden hyväksyy hyväksyjä.

Opintojen ohjaaja voi tutustua tutkintovaatimukseen ja opintokokonaisuuksiin kirjautumatta Korppi-järjestelmään.

4.3. Hyväksyjän toiminnalliset vaatimukset

Hyväksyjä voi hyväksyä tai hylätä opintojen ohjaajan tekemän henkilökohtaisen opintokokonaisuuden tai tutkintovaatimuksen hallitsemansa pääaineen tai suuntautumisvaihtoehdon osalta. Hyväksyjä voi myös liittää perusteluja päätökselleen.

Hyväksyjä voi tutustua tutkintovaatimukseen ja opintokokonaisuuksiin kirjautumatta Korppi-järjestelmään.

4.4. Opiskelijan toiminnalliset vaatimukset

Opiskelijat käyttävät sovellusta opintojensa suunnittelemiseen, perustelemiseen ja raportointiin.

Opiskelija voi valita HOPSiinsa tutkintovaatimuksen ajalta, jolloin hän on ollut opiskelijana Jyväskylän yliopistossa. Tutkintovaatimuksesta tulee pohja hänen HOPSilleen. HOPSiin hän liittää opintokokonaisuuksia tai kursseja tutkintovaatimuksen asettamissa rajoissa. Tutkintovaatimuksen ja opintokokonaisuuden valinnassa ei aseteta rajoja esim. pääaineen tai opiskeluoikeuden mukaan, koska kyseessä on suunnitelma.

Opiskelija voi perustella vapaamuotoisesti valitsemaansa tutkintovaatimusta, opintokokonaisuuksia tai kursseja. Opintojen suunnittelussa hän voi aikatauluttaa opintokokonaisuuksia ja kursseja. Opiskelija voi raportoida tekemänsä suunnitelman ja opiskeluhistorian.

Opiskelija voi tutustua tutkintovaatimuksiin ja opintokokonaisuuksiin kirjautumatta Korppi-järjestelmään.

4.5. Opettajan toiminnalliset vaatimukset

Opettaja voi opiskelijan luvalla tutustua hänen opintosuunnitelmiinsa.

Lisäksi opettajalla on vieraan toiminnalliset vaatimukset.

4.6. Vieraan toiminnalliset vaatimukset

Vieras voi tutustua tutkintovaatimuksiin ja opintokokonaisuuksiin kirjautumatta Korppi-järjestelmään.

4.7. Jatkokehityksessä huomioitavia asioita

Tässä luvussa on kerätty listaan asioita, joita tulee pohtia jatkokehityksen yhteydessä.

- Käyttäjryhmien oikeuksien pohtiminen ja tarkentaminen. Kenellä on oikeudet mihinkin toimintoon? Esimerkiksi onko sihteerillä oikeudet muokata opiskelijan HOPSia, jne.
- HOPSien tallentamisen ja arkistoinnin organisointi. Mikä HOPSin versio tallennetaan ja mikä ei? Voiko opiskelijalla olla useita HOPSeja?
- Aktiivisen ja hyväksytyt HOPSin esittäminen. Jos opiskelijalla on useampia HOPSeja niin mitkä niistä ovat aktiivisia?
- Korvaavuuksien hallinnan tarkentaminen, eli kuka hyväksyy ja mitä hyväksyy.
- Suoritettujen opintojen esittäminen HOPSissa.
- Raportoinnin tarkentaminen, eli mitä kukakin voi raportoida.
- Urasuunnittelun liittäminen HOPSiin.
- Kyselyiden liittäminen HOPSiin.
- Jyväskylän yliopiston ulkopuolella suoritettavien kurssien hallinta.
- Kansainväliseen vaihtoon liittyvän kyselyn liittäminen HOPSiin.

5. Sovelluksen tekniset määrittelyt ja vaatimukset

Luvussa esitellään tiiviisti sovelluksen liittyvät tekniset määrittelyt ja vaatimukset.

5.1. Toteutustekniikka ja palvelinratkaisut

Sovellus on dynaaminen WWW-sovellus, jota käytetään WWW-selaimilla. Dynaamisuus toteutetaan Javan JSP-tekniikalla, joka perustuu Javan servletteihin ja HTML:ään. JSP-sivuihin voidaan siis sijoittaa sekä HTML- että Java-koodia.

Käyttäjä tekee WWW-selaimen avulla pyyntöjä, joihin servletti tuottaa vastauksen WWW-palvelimessa. Kommunikointi tapahtuu HTTP-protokollan avulla.

Sovelluksessa tullaan käyttämään JSP-tekniikkaa ja servlettimoottorina TomCatiä, joka asennetaan Apache-nimiseen HTTP-palvelinohjelmistoon. Tietokannan hallinnassa käytetään PostgreSQL-ohjelmistoa.

5.2. Tuetut WWW-selaimet

Sovellusta käytetään WWW-selaimilla ja sovelluksen tulee toimia ainakin seuraavilla selaimilla:

- Firebird 0.7 (Windows ja Linux),
- Internet Explorer 6.0 (Windows),
- Lynx 2.8.4 (Windows ja Linux)
- Netscape 7.1 (Windows),
- Mozilla 1.4 (Windows ja Linux) sekä
- Opera 7.10 ja 6.05 (Windows ja Linux)

5.3. Muita teknisiä vaatimuksia

Sovellus liitetään osaksi KORPPI-järjestelmää, joten sovelluksen tulee täyttää KORPPI-järjestelmän yleiset tekniset vaatimukset.

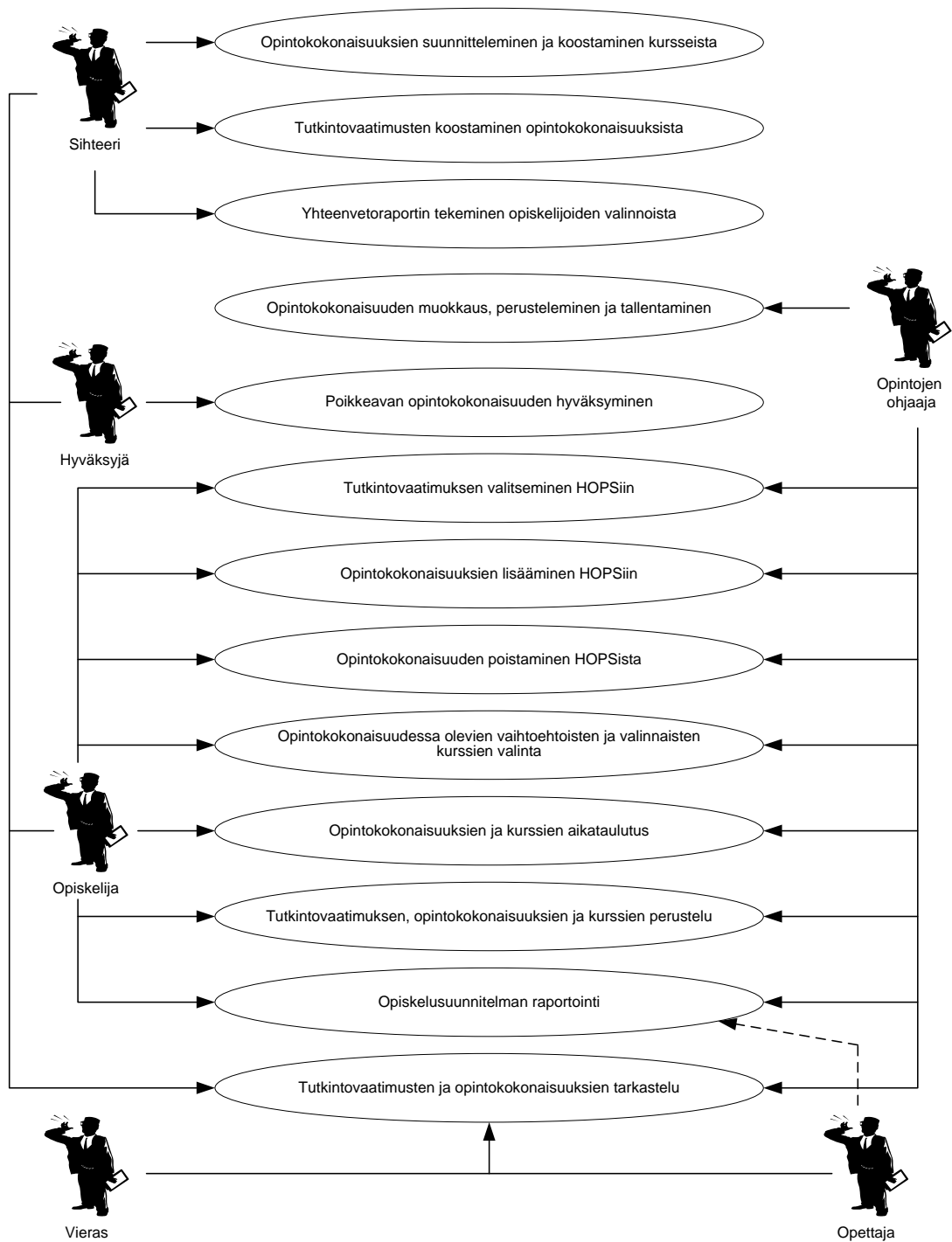
Sovelluksen sivujen latausajan ei tule ylittää viittä (5) sekuntia.

6. Käyttäjäryhmien toimintotapaukset

Luvussa kuvataan luvussa 4 määritellyt kuusi käyttäjäryhmää ja niiden toiminnot toimintakaaviona (kuva 1). Lisäksi määritellään jokaiseen toimintoon yhteenveto, toimijat, esiehdot, kuvaukset, poikkeukset ja jälkiehdot.

6.1. Toimintokaavio

Käyttäjärühmien toiminnot ja käyttäjien väliset suhteet kuvataan kuvassa 1.



Kuva 1: HOPS-sovelluksen käyttäjärühmien toimintokaavio.

6.2. Opintokokonaisuuksien suunnitleminen ja koostaminen kursseista

Yhteenveto: Toimija muodostaa opintokokonaisuuden Korppi-järjestelmän kurssitietojen pohjalta.

Toimija: Sihteeri.

Esiehto: Toimija on kirjautunut järjestelmään. Kurssit ovat olemassa.

Kuvaus: Toimija luo opintokokonaisuuden olemassaolevien tutkintovaatimusten perusteella. Aineeseen liittyvä opintokokonaisuus voisi olla tasoltaan esimerkiksi approbatur, cum laude approbatur, laudatur tai jokin muu vastaava kokonaisuus. Tässä määritellään opintokokonaisuuteen kuuluvat kurssit, jotka voivat olla pakollisia, vaihtoehtoisia tai valinnaisia. Toimija voi tallentaa välillä suunnittelemansa opintokokonaisuuden ja jatkaa sitä myöhemmin.

Poikkeukset: (1) Opintokokonaisuus on jo olemassa, jolloin hyväksymällä varmistuksen, opintokokonaisuus korvaa aiemmin määritellyn kokonaisuuden.

(2) Kurssia ei löydy, jolloin tallennetaan keskeneräinen opintokokonaisuus, lisätään kantaan ko. kurssi kurssikirjanpitosovelluksella ja jatketaan keskeneräisestä.

Jälkiehdot: Valmis opintokokonaisuus on toimijan käytössä tutkintovaatimuksia koostettaessa.

6.3. Tutkintovaatimusten koostaminen opintokokonaisuuksista

Yhteenveto: Toimija muodostaa tutkintovaatimukset olemassaolevista opintokokonaisuuksista.

Toimija: Sihteeri.

Esiehto: Toimija on kirjautunut järjestelmään. Tarvittavat opintokokonaisuudet on jo olemassa.

Kuvaus: Toimija koostaa opintokokonaisuuksista tutkintovaatimukset. Tutkintovaatimukseen määritellään pakolliset, vaihtoehtoiset ja valinnaiset opintokokonaisuudet. Toimija voi tallentaa välillä suunnittelemansa tutkintovaatimuksen ja jatkaa sitä myöhemmin.

Poikkeukset: (1) Saman niminen tutkintovaatimus on jo olemassa, jolloin hyväksymällä varmistuksen, uudet tutkintovaatimukset korvaavat aiemmin määritellyt tutkintovaatimukset.

(2) Tarvittavaa opintokokonaisuutta ei löydy, jolloin tallennetaan keskeneräinen tutkintovaatimus, lisätään opintokokonaisuus ja jatketaan keskeneräisestä.

Jälkiehdot: Valmis tutkintovaatimus on käyttäjien käytössä.

6.4. Yhteenvetoraportin tekeminen opiskelijoiden valinnoista

Yhteenveto: Toimija tarkastelee opintokokonaisuuksien ja kurssien aikataulua ja mahdollisesti raportoi tarkastelun tulokset.

Toimija: Sihteeri

Esiehto: Toimija on kirjautunut järjestelmään. Kurseja ja niistä koostettuja opintokokonaisuuksia on olemassa. Opiskelijat ovat valinneet opintokokonaisuuksia tutkintosuunnitelmaansa.

Kuvaus: Toimija tarkastelee, miten opiskelijat ovat valinneet opintokokonaisuuksia ja niiden sisältämiä kurseja tietynä ajanjaksona. Toimija ei pääse henkilötasolle, vaan näkee yleiskuvan jakaumista. Toimija voi halutessaan muodostaa raportit.

Poikkeukset: (1) Opintokokonaisuuksia ei ole, jolloin annetaan virheilmoitus.

(2) Opiskelijat eivät ole valinneet opintokokonaisuuksia ja siten raportoitavaa ei ole, jolloin annetaan virheilmoitus.

Jälkiehdot: Toimijalla on raportti opiskelijoiden opintojen suunnittelusta.

6.5. Opintokokonaisuuden muokkaus, perusteleminen ja tallentaminen

Yhteenveto: Toimija muuttaa opiskelijan opintokokonaisuutta lisäämällä, poistamalla, siirtämällä tai muokkaamalla siinä olevia kurseja. Toimija lisää perusteluja muutoksilleen. Toimija tallentaa tekemänsä opintokokonaisuuden.

Toimija: Opintojen ohjaaja

Esiehto: Toimija on kirjautunut järjestelmään. Kurseja ja niistä koostettuja opintokokonaisuuksia on olemassa. Toimija näkee opiskelijan tekemän suunnitelman.

Kuvaus: Toimija voi muokata henkilökohtaisen opintokokonaisuuden opiskelijalle tarjolla olevista virallisista tai jonkun muun opiskelijan opintokokonaisuuksista. Toimija voi poistaa, lisätä, siirtää opintokokonaisuuteen kuuluvia kurseja. Toimija voi perustella tekemänsä muutokset vapaamuotoisesti. Opintokokonaisuuteen tulee merkintä siitä, että sitä on muokattu ja kuka sitä on muokannut. Toimija tallentaa opintokokonaisuuden, jonka jälkeen opiskelija ei voi muokata tai poistaa sitä. Opiskelijan opintokokonaisuus on lukittuna, kunnes hyväksyjä on hyväksynyt sen.

Poikkeukset: (1) Opintokokonaisuuksia ei ole, jolloin annetaan virheilmoitus.

(2) Toimija ei näe opiskelijan tekemää suunnitelmaa, jolloin annetaan virheilmoitus.

(3) Tallentaminen epäonnistui, jolloin annetaan virheilmoitus.

Jälkiehdot: Toimija saa tehtyä ja tallennettua opiskelijalle henkilökohtaisen opintokokonaisuuden, jolloin se siirtyy hyväksyttäväksi.

6.6. Poikkeavan opintokokonaisuuden hyväksyminen

Yhteenveto: Toimija voi hyväksyä tai hylätä poikkeavan opintokokonaisuuden.

Toimija: Hyväksyjä

Esiehto: Toimija on kirjautunut järjestelmään. Poikkeava opintokokonaisuus on olemassa.

Kuvaus: Toimija voi tarkastella poikkeavaa opintokokonaisuutta. Hän voi hyväksyä tai hylätä sen. Hylkäyksen ja hyväksymisen yhteydessä toimija voi perustella päätöksensä.

Poikkeukset: Poikkeavaa opintokokonaisuutta ei ole, jolloin annetaan virheilmoitus.

Jälkiehdot: (1) Opintokokonaisuuteen tulee merkintä siitä, että onko se hyväksytty vai hylätty ja kuka sen on tehnyt.

(2) Hyväksytty opintokokonaisuus ilmestyy opiskelijalle.

(3) Hylätty opintokokonaisuus palautuu opintojen ohjaajalle.

6.7. Tutkintovaatimuksen vaatimuksen valitsemien HOPSiin

Yhteenveto: Toimija valitsee jonkin tarjolla olevan tutkintovaatimuksen.

Toimija: Opiskelija, opintojen ohjaaja

Esiehto: Toimija on kirjautunut järjestelmään. Valittavia tutkintovaatimuksia on olemassa.

Kuvaus: Toimija valitsee valittavissa olevan tutkintovaatimuksen HOPSiinsa pohjaksi.

Poikkeukset: (1) Tutkintovaatimuksia ei ole, jolloin annetaan virheilmoitus ja ilmoitetaan sihteerille asiasta.

(2) Jo olemassa olevaa aiemmin valittua tutkintovaatimusta ei voi uudestaan valita, jolloin annetaan virheilmoitus.

(3) Opiskelijalla voi olla useampikin kuin yksi HOPS.

Jälkiehdot: Toimija on lisännyt HOPSiinsa tutkintovaatimuksen pohjaksi.

6.8. Opintokokonaisuuksien lisääminen HOPSiin

Yhteenveto: Toimija valitsee jonkin valittavissa olevan opintokokonaisuuden ja lisää sen HOPSiinsa.

Toimija: Opiskelija, opintojen ohjaaja

Esiehto: Toimija on kirjautunut järjestelmään. Toimija on valinnut tutkintovaatimuksen HOPSiinsa. Tutkintovaatimus sisältää lisättäviä opintokokonaisuuksia.

Kuvaus: Toimija valitsee valittavissa olevan opintokokonaisuuden, määrää sen aloitusajan ja lisää sen HOPSiinsa.

Poikkeukset: (1) Tutkintovaatimusta ei ole, jolloin annetaan virheilmoitus.

(2) Tutkintovaatimuksessa ei ole opintokokonaisuuksia, jolloin annetaan virheilmoitus ja ilmoitetaan sihteerille asiasta.

(3) Jo olemassa olevaa aiemmin valittua opintokokonaisuutta ei voi uudestaan valita, jolloin annetaan virheilmoitus.

Jälkiehdot: Opiskelija on lisännyt HOPSiin opintokokonaisuuden.

6.9. Opintokokonaisuuksien poistaminen HOPSista

Yhteenveto: Toimija valitsee HOPSista opintokokonaisuuden ja poistaa sen.

Toimija: Opiskelija, opintojen ohjaaja

Esiehto: Toimija on kirjautunut järjestelmään. Poistettavia opintokokonaisuuksia on olemassa tutkintovaatimuksessa.

Kuvaus: Toimija poistaa HOPSista lisäämänsä opintokokonaisuuden.

Poikkeukset: (1) Opiskelija ei voi poistaa opintojen ohjaajan tallentamaa henkilökohtaista opintokokonaisuutta.

(2) Opintojen ohjaaja voi poistaa tallentamansa henkilökohtaisen opintokokonaisuuden.

Jälkiehdot: Opintokokonaisuus on poistettu opiskelijan HOPSista.

6.10. Opintokokonaisuudessa olevien vaihtoehtoisten ja valinnaisten kurssien valinta.

Yhteenveto: Toimija valitsee opintokokonaisuuteen haluamansa vaihtoehtoiset ja valinnaiset kurssit.

Toimija: Opiskelija, opintojen ohjaaja

Esiehto: Toimija on kirjautunut järjestelmään. Toimija on valinnut opintokokonaisuuden. Opintokokonaisuus sisältää kursseja.

Kuvaus: Toimija valitsee opintokokonaisuudessa olevia vaihtoehtoisia ja valinnaisia kursseja.

Poikkeukset: Opintokokonaisuudessa ei ole vaihtoehtoisia tai valinnaisia kursseja, jolloin annetaan virheilmoitus.

Jälkiehdot: Toimija on saanut täydennettyä opintokokonaisuuden vaihtoehtoisten ja valinnaisten kurssien osalta.

6.11. Opintokokonaisuuksien ja kurssien aikataulut

Yhteenveto: Toimija aikatauluttaa HOPSissa olevat opintokokonaisuudet tai niissä olevia kursseja.

Toimija: Opiskelija, opintojen ohjaaja

Esiehto: Toimija on kirjautunut järjestelmään. Toimija on valinnut aikataulutettavan opintokokonaisuuden tai kurssin.

Kuvaus: Toimija aikatauluttaa opintokokonaisuudet tai opintokokonaisuudessa olevat kurssit antaen haluamansa suunnitellun suoritusajankohdan. Ajankohta annetaan lukukauden (kevät, kesä tai syksy ja vuosi) tarkkuudella.

Poikkeukset: (1) Opintokokonaisuuksia ei ole valittu, jolloin annetaan virheilmoitus.
(2) Opintokokonaisuus ei sisällä valittuja kursseja, jolloin annetaan virheilmoitus.

Jälkiehdot: Opintokokonaisuudet ja niissä olevia kursseja on aikataulutettu.

6.12. Tutkintovaatimuksen, opintokokonaisuuksien ja kurssien perustelu

Yhteenveto: Toimija perustelee HOPSiin valitsemansa tutkintovaatimuksen ja siinä olevat opintokokonaisuudet ja kurssit.

Toimija: Opiskelija, opintojen ohjaaja

Esiehto: Toimija on kirjautunut järjestelmään. Toimija on valinnut olemassa olevan tutkintovaatimuksen, siihen liitetyn opintokokonaisuuden tai opintokokonaisuuteen sisältyvän kurssin.

Kuvaus: Toimija perustelee vapaamuotoisesti HOPSiin valitun tutkintovaatimuksen, opintokokonaisuuden tai kurssin.

Poikkeukset: Perusteltavaa tutkintovaatimusta, opintokokonaisuutta tai kurssia ei ole, jolloin annetaan virheilmoitus.

Jälkiehdot: Toimija on saanut perusteltua HOPSiinsa valitsemia osasia.

6.13. Opiskelusuunnitelman raportointi

Yhteenveto: Toimija voi raportoida opiskelusuunnitelman tai sen osia.

Toimija: Opiskelija, opintojen ohjaaja, opettaja

Esiehto: Toimija on kirjautunut järjestelmään. Jos toimijana on opettaja tulee hänellä olla opiskelijan lupa opintojen tarkastelemiseen. Vähintään opintokokonaisuuksia on aikataulutettu.

Kuvaus: Toimija tarkastelee kaikkia opintokokonaisuuksia tai kursseja tietyllä aikavälillä, joista hän laatii raportin, jossa näkyy selkeästi sekä opiskeluhistoria että -suunnitelma. Kurssien ajankohdat saadaan opintokokonaisuuksista kerättyjen kurssitietojen pohjalta.

Poikkeukset: (1) Ei opintosuunnitelmaa, jolloin annetaan virheilmoitus.

(2) Ei aikataulutettuja opintoja, jolloin annetaan virheilmoitus.

(3) Opettajalla ei ole oppilaan antamia oikeuksia kurssien tarkastelemiseen, jolloin annetaan virheilmoitus.

Jälkiehdot: Toimijalla on raportoitu dokumentti opiskelusuunnitelmasta.

6.14. Tutkintovaatimusten ja opintokokonaisuuksien tarkastelu

Yhteenveto: Toimija tarkastelee kannassa olevia tutkintovaatimuksia ja niihin kuuluvia opintokokonaisuuksia.

Toimija: Sihteeri, opintojen ohjaaja, hyväksyjä, opiskelija, opettaja ja vieras

Esiehto: Tutkintovaatimuksia ja opintokokonaisuuksia on olemassa.

Kuvaus: Toimija tarkastelee tutkintovaatimuksia ja niihin kuuluvia opintokokonaisuuksia kirjautumatta järjestelmään.

Poikkeukset: Tutkintovaatimuksia tai opintokokonaisuuksia ei löydy, jolloin annetaan virheilmoitus.

Jälkiehdot: Toimija on saanut tietoa eri tutkintovaatimukseen kuuluvista opintokokonaisuuksista ja niihin kuuluvista kursseista.

7. Käki-ryhmän toteuttama sovelluksen osa

Käki-projektin aikana toteutetaan sovelluksesta osa. Toteutettava osa tulee sisältämään sihteerin vaatimuksista opintokokonaisuuden ja tutkintovaatimuksen koostamisen sekä vieraan toiminnalliset vaatimukset.

8. Yhteenveto

Tämän dokumentin kirjoittaminen aloitettiin käyttäjäryhmien ja niiden sisältämien toimintojen määrittelemisestä. Tätä kautta saatiin hyvä kuva koko sovelluksen vaatimuksista. Dokumentissa kuvataan koko sovelluksen toiminnalliset ja tekniset vaatimukset. Käki-projekti toteuttaa sovelluksesta luvussa 7 määritellyn osan, jota muut voivat myöhemmin jatkokehittää.

Lähteet

- [1] Hillebrand Minna, Silván Markus, Vanhanen Antti ja Ylitalo Marko, ”Koppelo-projektin projektikansio”, Jyväskylän yliopisto, tietotekniikan laitos, 2002.
- [2] Hilpinen Toni, Koivuniemi Marko, Mäkinen Jussi, Nurminen Miika, ”Kiuru-projektin projektikansio”, Jyväskylän yliopisto, tietotekniikan laitos 2002.
- [3] Hyvämäki Kekke ”Hops-kartoituksen analysointia”, Jyväskylän yliopisto, Virtuaaliyliopistohanke, 2003.
- [4] Jukka Korpela, ”Kirjoita asiaa. Arkisen asiakirjoittamisen opas”, saatavilla html-muodossa <URL: <http://www.cs.tut.fi/~jkorpela/kirj/>>, 30.12.2002.
- [5] Jukka Korpela, ”Ohjeita ja tietoja suomen kielestä”, saatavilla html-muodossa <URL: <http://www.cs.tut.fi/~jkorpela/kielikello/>>, 1995.
- [6] Korppi-kehittäjät, ”Korppi-järjestelmän ominaisuudet ja kehitys”, saatavilla html-muodossa <URL: <https://korppi.it.jyu.fi/kotka/portal/tietoja.html>>, Jyväskylän yliopisto, tietotekniikan laitos, 21.8.2003.
- [7] Lappalainen Teemu, Valkonen Anu, Vieru Mikko ja Viklund Markus, ”Uraani-projektin projektikansio”, Jyväskylän yliopisto, tietotekniikan laitos 2002.
- [8] Lesonen Minna, Pekkanen Hannu, Tawast Tuukka ja Uuksulainen Heikki, ”Korppi-projektin projektikansio”, Jyväskylän yliopisto, tietotekniikan laitos, 2001.
- [9] Lintunen Sampsa, Pöyhönen Tiina, Vähä-Ruka Teemu ja Ylönen Timo, ”Kottarainen-projektin projektikansio”, Jyväskylän yliopisto, tietotekniikan laitos, 2003.
- [10] Santanen Jukka-Pekka, ”Tietotekniikan Sovellusprojektien ohje”, Jyväskylän yliopisto, tietotekniikan laitos, 2002.
- [11] Santanen Jukka-Pekka, ”Opinnäytteiden kirjoittaminen, lyhyt oppimäärä”, Jyväskylän yliopisto, tietotekniikan laitos, 2000.