

Lappeenrannan teknillinen yliopisto

Tietotekniikan osasto

KANDIDAATINTYÖ

Harjoitustöiden vertaisarviointi opetusvälineenä

Kandidaatintyön aihe on hyväksytty 29.10.2008

Työn tarkastajana toimii professori Jari Porras

Lappeenrannassa 17.12.2008

Jukka Tarkkonen

jtarkkon@lut.fi

TIIVISTELMÄ

Lappeenrannan teknillinen yliopisto

Tietotekniikan osasto

Jukka Tarkkonen

Harjoitustöiden vertaisarviointi opetusvälineenä

Kandidaatintyö

2008

39 sivua, 2 kuvaa, 1 taulukko

Ohjaaja: professori Jari Porras

Hakusanat: harjoitustyöt, vertaisarviointi, verkko-opetus, pedagogiikka

Harjoitustöiden vertaisarvioinnissa opiskelijat tarkastavat toistensa töitä ja antavat niistä palautteen annettujen ohjeiden mukaan. Yleensä opiskelijoille annetaan tätä tehtävää tukemaan tietty joukko arviointikriteereitä, joiden osalta työtä ainakin on kommentoitava. Vertaisarviointi voi antaa opiskelijoille monipuolisempaa palautetta kuin ilman sitä olisi mahdollista, mahdollisuuden tehtävien vaihtoehtoiisiin ratkaisuihin tutustumiseen sekä kokemusta ongelmanratkaisun analyttisestä ja kriittisestä tarkastelusta. Lisäksi vertaisarviointi voi tukea opettajien työtä helpottamalla riittävän palautteen antamista opiskelijoille ja antamalla jonkinlaista kuvaa opiskelijoiden näkemyksistä tehtävän aihealueeseen liittyen.

Tämän työn tavoitteena on selvittää, mitä mainituista vertaisarviointiin liitettyistä hyödyistä voidaan todellisuudessa saavuttaa ohjelmointipainotteisilla kursseilla. Ohjelmointikurssin opiskelijat arvioivat kurssiin kuuluvien harjoitustöiden yhteydessä toistensa ratkaisuja yhteen tehtävistä. Opiskelijoille suunnatulla kyselyllä pyritään kartoittamaan, kuinka hyvin vertaisarvioinnille asetetut tavoitteet toteutuvat. Kokemuksia opettajan näkökulmasta selvitetään haastattelujen avulla. Testauksen toteutuksessa käytetään verkkosovellusta, johon opiskelijat tallentavat omat työnsä, josta he lataavat itselleen määrättyt työt arvioitavaksi, johon he tallentavat arvionsa ja jonka kautta he lopulta saavat palautteen omasta työstään.

ABSTRACT

Lappeenranta University of Technology
Department of Information Technology

Jukka Tarkkonen

Peer Review of Practical Assignments as a Teaching Tool

Bachelor's thesis

2008

39 pages, 2 images, 1 table

Supervisor: professor Jari Porras

Keywords: practical assignments, peer review, online teaching, pedagogy

In peer review of practical assignments, students check each other's works and give feedback on them according to the instructions that are given to them. Usually a specific set of evaluation criteria is given to the students, that has to be commented on at the very least. Through peer review, students may be provided with more diverse feedback than would be possible without it, a possibility to consider alternative solutions to assignments, and experience in the analytical and critical assessment of problem solving. In addition, peer review may support the work teachers by making it easier to ensure that sufficient feedback is being given to the students, and by giving some insight into how the students perceive the field of the assignment.

The aim of this work is to find out which of the benefits mentioned above can actually be achieved on courses focused on programming. The students of a programming course will review some of each other's solutions to one of the programming assignments that are a part of the course. By using a survey aimed at the students, an understanding is sought about how well the goals set to the peer review are reached. Experiences from the teacher's point of view are investigated through interviews. The testing will be executed by using a web application, in which the students will upload their works, from which they will download works assigned to them for evaluation, in which they will save their reviews, and through which they will finally receive feedback on their own work.

SISÄLLYSLUETTELO

LYHENTEET	2
1 JOHDANTO	3
1.1 Tausta.....	3
1.2 Tavoitteet ja rajaukset.....	3
1.3 Työn rakenne.....	4
2 VERTAISARVIOINTI	5
2.1 Vertaisarviointi tietotekniikan opetuksessa	6
2.2 Vertaisarviointia tukevat sovellukset	6
2.3 Tutkimusongelma ja ratkaisumenetelmät	7
3 VERTAISARVIOINNIN TESTAUS	8
3.1 Koemenettelyn määrittely.....	8
3.1.1 Network Programming	9
3.1.2 Concurrent and Parallel Computing	9
3.2 Sovelluksen valinta.....	10
3.2.1 MyReview ja OpenConf	10
3.3 MyReview-ohjelmiston käyttö.....	13
3.3.1 Asetukset ja valmistelut	14
3.3.2 Harjoitustyön tallentaminen järjestelmään.....	16
3.3.3 Arvioijien asettaminen harjoitustöille.....	17
3.3.4 Vertaisarviointien tekeminen	17
3.3.5 Arviointipalautteen lähettäminen töiden tekijöille	18
3.4 Testauksen toteutus	19
3.5 Opiskelijakysely	21
3.6 Opettaja-haastattelut	22
3.7 Tulosten yhteenveto ja tulkinta	23
4 JOHTOPÄÄTÖKSET	28
5 YHTEENVETO.....	30
LÄHDELUETTELO.....	31
LIITE 1. MyReview-sovelluksen asetuksia	
LIITE 2. Opiskelijoille tehty kysely	

LYHENTEET

HTML	HyperText Markup Language
PDF	Portable Document Format
PHP	Hypertext Preprocessor
RTF	Rich Text Format
SQL	Structured Query Language
URL	Uniform Resource Locator
XML	Extensible Markup Language

1 JOHDANTO

1.1 Tausta

Vertaisarviointia käytetään toisinaan eräänä opetuksen menetelmänä. Opiskelijat tutustuvat toistensa tehtävien tai harjoitustöiden ratkaisuihin, ja antavat näistä toisilleen palautetta opettajan asettamien kriteerien mukaisesti. Vertaisarvioinnin tarkoituksena on muun muassa tarjota opiskelijoille uusia näkökulmia tehtävien ratkaisuihin toisistaan eroavien lähestymistapojen muodossa.

Tällainen vertaisarviointi on mahdollista toteuttaa käyttäen apuna Internetissä toimivaa eräänlaista yhteisöpalvelua, jonka kautta opiskelijat voivat ladata arvioitavat ratkaisut itselleen tarkastettaviksi ja palauttaa arvioinnin tulokset. Tällaista verkkototeutusta Lappeenrannan teknillisellä yliopistolla ei vielä toistaiseksi ole varsinaisessa käytössä kyseistä tarkoitusta varten, mutta vastaavasti toimivia järjestelmiä on kuitenkin olemassa.

1.2 Tavoitteet ja rajaukset

Jotta vertaisarviointia ja sen tekemistä helpottavia ohjelmistoja kannattaisi kehittää ja käyttää, siitä tulisi olla hyötyä ihannetapauksessa kaikille osapuolille, niin arvioijille, arvioitaville, kuin tehtävän ohjaajallekin, eli opettajalle. Tässä työssä testataan verkkopalvelun avulla tehtävää vertaisarviointia Tietotekniikan osaston kurssilla siihen kuuluvien harjoitustöiden yhteydessä. Tavoitteena on selvittää, onko vertaisarvioinnista hyötyä ohjelmointipainotteisilla kursseilla.

Arviointia kokeillaan verkossa toimivan ohjelmiston avulla, mutta työssä keskitytään nimenomaan vertaisarviointiin ja sen mahdollisesti tuottamiin tuloksiin. Käytettävän ohjelmiston tekninen toteutus ei kuulu tähän työhön, vaan apuna käytetään jo olemassa olevaa järjestelmää. Lisäksi harjoitustyö, jonka kohdalla arviointia työn aikana sovelletaan, on ohjelmointitehtävä. Arvioinnin kohteena on täten ohjelmakoodi. Vertaisarviointia voidaan toki soveltaa hyvin moneen erilaiseen aineistotyyppiin, mutta esimerkiksi mahdolliset erot arvioinnin soveltuvuudessa niiden välillä eivät kuulu tämän tutkimuksen piiriin.

1.3 Työn rakenne

Työ koostuu viidestä luvusta, joista seuraavat ovat 2 Vertaisarviointi, 3 Vertaisarvioinnin testaus, 4 Pohdinta ja 5 Yhteenveto. Luvussa 2 käydään hieman läpi vertaisarvioinnin taustaa, sekä selvitetään vertaisarvioinnin soveltuvuutta tietotekniikan opetukseen ja opiskeluun. Lopuksi myös käsitellään vertaisarvioinnin toteuttamista verkkosovelluksen avulla ja esitellään tutkimusongelma sekä käytettävät ratkaisumenetelmät.

Luvussa 3 käydään läpi työn käytännön vaiheita. Luvussa esitellään kuinka työ käytännössä toteutettiin, minkälaisia menettelytapoja käytettiin, kuinka niihin päädyttiin ja lopuksi käsitellään työn tuloksia.

Luvussa 4 pohditaan tulosten merkitystä sekä tulevaisuuden näkymiä ja mahdollisuuksia aiheeseen liittyen. Luvussa 5 esitetään yhteenveto työssä käsitellyistä asioista siinä valossa, mitä todellisuudessa toteutui.

2 VERTAISARVIOINTI

Vertaisarvioinnin hyödyntämisestä opetuksessa ja opiskelussa löytyy useita mainintoja pedagogiikkaa käsittelevästä kirjallisuudesta. Vertaisarvioinnin ansiosta opiskelijalle avautuu mahdollisuus suhteuttaa näkemyksiään omasta oppimisestaan toisilta saatuun palautteeseen (Kiviniemi 2000). Lisäksi toistensa töitä arvioidessaan opiskelijat pääsevät tutustumaan vaihtoehtoihin ratkaisutapoihin, mikäli tehtävän ratkaisuun on useita erilaisia mahdollisuuksia. Esimerkiksi ohjelmointitehtävän ollessa kyseessä erilaisia ratkaisuja voi olla hyvinkin paljon. Tällaisessa tilanteessa vertaisarviointi voi auttaa opiskelijoita ymmärtämään ratkaisuvaihtoehtojen moninaisuuden sen käsityksen sijaan, että ainoa oikea ratkaisu olisi nimenomaan opettajan mallivastaus (Gehring & al. 2005). Koodin tapauksessa entistä parempien ratkaisujen löytymisellä voi olla merkittävä vaikutus esimerkiksi ohjelman tehokkuuteen.

Oppimisen kannalta vertaisarviointi voi olla avuksi perehdyttämällä opiskelijoita käsiteltävään aiheeseen ja tehtävään laajemmin yhtenä kokonaisuutena. Tutustumalla opiskelijoiden tekemiin arvioihin opettaja voi puolestaan saada jonkinlaisen kuvan siitä, mitä tekijöitä opiskelijat pitävät ratkaisuisa oleellisina, minkälaiseksi he käsittävät hyvän ratkaisun ja sitä kautta myös miten opiskelijat hahmottavat tehtävän kokonaisuuden.

Muina etuina vertaisarvioinnissa pidetään esimerkiksi sen yhteisöllisyyttä edistävää luonnetta sekä sitä, että opiskelijat voivat saada sen kautta säännöllistä palautetta työskentelystään esimerkiksi verkko-opiskelussa, tilanteessa, jossa työskentely on pääasiassa omatoimista, oppimistehtäviä on paljon ja opiskelijoiden määrä on suuri (Kiviniemi 2000). Laajan ja yksityiskohtaisen palautteen antaminen suurelle määrälle opiskelijoita vie opettajalta paljon aikaa, eikä ole sen takia aina riittävän tehokasta oppimistuloksia ajatellen (Clark & al. 2005). Opiskelijoiden tehdessä vertaisarviointia toisilleen osa palautteen antamiseen tarvittavasta työmäärästä saadaan hajautettua, eikä se kuormita yhtä henkilöä liikaa suurenkaan opiskelijamäärän tapauksessa.

Marttunen (1997) kertoo argumentoivaan ja perustelevaan dialogiin sekä opiskelijoiden väittelyyn perustuvien oppimistilanteiden vähyyden olevan ongelma Suomen yliopistoissa. Hänen mukaansa kynnys kritiikin esittämiseen ja opettajien sekä toisten opiskelijoiden kyseenalaistamiseen on todettu korkeaksi esimerkiksi seminaariopiskelun ja siihen liittyvien opponenttijärjestelyjen kohdalla. Myös tutkittaessa opiskelijoiden välistä vuorovaikutusta sähköpostiympäristössä

toteutetulla kasvatustieteen opintojaksolla oli tullut esiin, että erimielisyyksiä vältellään. Tämän hän kertoo johtuvan siitä, että erimielisyyksien ilmaiseminen saatetaan kokea epäkohteliaansa ja toisten loukkaamista halutaan välttää. Tällaisilla seikoilla voi olla omat vaikutuksensa myös vertaisarviointeihin. Tyypillinen menettely, jolla tämänkaltaisia ongelmia yleensä pyritään välttämään, on arviointien pitäminen nimettöminä (Kali & Ronen 2005). Toisaalta vertaisarviointi on myös sellainen käytäntö, jonka avulla on mahdollista oppia sekä antamaan että ottamaan vastaan rakentavaa kritiikkiä (Anewalt 2005).

2.1 Vertaisarviointi tietotekniikan opetuksessa

Vertaisarviointi on huomioitu laajalti sovelluskelpoisena myös nimenomaan tietotekniikan alalla, jossa mahdollisia arvioinnin kohteita ovat esimerkiksi projektiryhmän tuottama suunnitteludokumentaatio ja koodi sekä erilaiset kirjoitustehtävät (Gehringer & al. 2005). Lisäksi muun muassa koodin vertaisarviointia on käsitelty myös ryhmätyötehtävänä. Tällä tavoin on huomioitu se, että ohjelmistoteollisuus edellyttää alalle valmistuvilta opiskelijoilta hyviä ryhmätyötaitoja. Vertaisarvioinnin soveltaminen ohjelmakoodiin opettaa myös lukemaan koodia, joka on koodin kirjoittamisen ohella tärkeä, mutta hyvin erilainen taito (Trytten 2005).

Toisten tuottaman koodin ja dokumentaation läheinen tarkastelu arviointia varten voi myös auttaa ymmärtämään selkeän ohjelmointi- ja dokumentointityylin tärkeyden (Papalaskari 2003). Sen kautta on mahdollista nähdä esimerkkejä tilanteista, joissa selkeydessä olisi parantamisen varaa. Lisäksi koska oman tuotoksen ymmärtäminen on itselleen luonnollisesti helpompaa kuin muille, muilta saatu palaute auttaa tunnistamaan mahdollisia tyylin heikkouksia, joille itse on usein sokea.

2.2 Vertaisarviointia tukevat sovellukset

Tietoverkkojen ja verkkosovellusten opetukseen tuomia uusia mahdollisuuksia on alettu tutkia ja käsitellä kirjallisuudessa niiden yleistymisen myötä. Erilaisia verkko-oppimisympäristöjä on alettu kehittää ja kokeilla eri aloilla, ja tällaisten ympäristöjen mahdollistama yhteisöllinen vuorovaikutus ja palautteenantomahdollisuus on huomioitu. Esimerkiksi taideteollisen korkeakoulun ja Helsingin yliopiston psykologian laitoksen yhteisöllisen suunnittelun opetukseen kehittämässä Future Learning Environment -ympäristössä on oma osionsa, jossa erilaisia ideoita, luonnoksia ja tuotoksia

voidaan kommentoida, arvioida ja jopa muokata esimerkiksi muiden ryhmään kuuluvien suunnittelijoiden tai opiskelijayhteisön toimesta (Seitamaa-Hakkarainen & Hakkarainen 2000). Myös vertaisarvioinnin mahdollistavia tai sitä tukevia verkkosovelluksia on toteutettu ja kokeiltu käytännössä. Esimerkiksi Washington State University on ottanut vertaisarvioinnin osaksi verkkokurssiensa ohjelmistorakennetta (Kiviniemi 2000). Näillä verkkokursseilla ohjelmisto on tukenut opettajan työtä tarjoamalla määrällistä ja laadullista tietoa vertaisarvioiden luonteesta helpottaakseen opiskelijamäärältään suurten kurssien hallintaa. Lisäksi muissa yhdysvaltalaisissa yliopistoissa on kokeiltu sähköisten vertaisarviointijärjestelmien soveltuvuutta esimerkiksi ohjelmistotuotannon (Wolfe 2004) ja usean muun tietotekniikan osa-alueen kuten ohjelmoinnin (Gehring 2001) opetuksessa. Näissä kokeiluissa on hyvin yleistä, että saadut tulokset ovat olleet positiivisia.

2.3 Tutkimusongelma ja ratkaisumenetelmät

Tutkimusongelmana tässä työssä on selvittää, onko vertaisarvioinnista hyötyä. Jotta vertaisarviointien tekeminen ja sitä tukevien ohjelmistojen kehittäminen ja käyttäminen olisi perusteltua, vertaisarvioinnista on oltava hyötyä oppimistavoitteiden saavuttamisessa. Parhaassa tapauksessa kaikki osapuolet voivat hyötyä: opiskelijat sekä arvioijan että palautteensaajan asemassa ja myös luennoitsija tai tehtävän tarkastaja.

Koska vertaisarvioinnin käyttö Lappeenrannan teknillisellä yliopistolla on tähän asti ollut rajallista, näistä mahdollisista hyödyistä ja niiden toteutumisesta käytännössä ei ole ennestään kattavaa kokemusta. Tässä työssä testataan vertaisarviointia verkkosovelluksen avulla tehtynä. Työ toteutetaan yhteistyössä ohjelmointipainotteisen kurssin kanssa, ja kokeellisen vertaisarvioinnin kohteena on koodi, jota opiskelijat tuottavat kurssin ohjelmointitehtäviä varten. Näin ollen tutkimuksen kohteena on erityisesti vertaisarvioinnin soveltuvuus ja sen tuottamat hyödyt ohjelmoinnin opetuksessa. Testauksessa mukana oleva kurssi on Concurrent and Parallel Computing, jonka luennoi Jari Porras.

3 VERTAISARVIOINNIN TESTAUS

Työn toteutus koostuu tietyistä selkeästi oman työvaiheensa muodostavista osista. Ensimmäinen vaihe on vertaisarvioinnin testaukseen osallistuvan kurssin luennoitsijan kanssa tehtävä kartoitus arvioinneista, jotka työssä toteutetaan. Tämän jälkeen on valittava verkossa toimiva sovellus, jonka avulla arvioinnit tehdään. Sen asennukseen tarvitaan käyttöoikeudet sopivaan yliopiston verkkopalvelimeen, joten sen tekee henkilökunta. Seuraava vaihe on valitun järjestelmän käyttöön perehtyminen, jota seuraa järjestelmän valmistelu käyttökuntoon ja harjoitustöiden vertaisarviointiin sopivaksi sekä varsinaisen testauksen toteutus. Tämän jälkeen tehdään kysely, jossa opiskelijoita pyydetään vastaamaan vertaisarviointia koskeviin kysymyksiin. Kyselyllä pyritään kartoittamaan arvioinnin hyötyjä opiskelijoiden näkökulmasta. Lisäksi käsitellään myös opettajan puoli asiasta haastattelujen avulla. Näiden vaiheiden kautta kootun aineiston perusteella voidaan lopulta suorittaa analyysiä ja verrata työn tuloksia asetettuihin tavoitteisiin.

3.1 Koemenettelyn määrittely

Aluksi vertaisarviointia suunniteltiin testattavaksi kahdella kurssilla. Concurrent and Parallel Computing -kurssin lisäksi mukana oli tarkoitus olla Jouni Ikosen luennoima Network Programming. Jälkimmäisen kurssin kohdalla testaus kuitenkin jäi toteutumatta, koska vertaisarvioinnin tekninen toteutus ei vastannut kurssin vaatimuksia. Kummankin kurssin luennoitsijoiden kanssa kuitenkin määriteltiin, mitä vaatimuksia heillä oli tehtävälle arvioinnille, eli minkälaisen arvioinnin kurssien opiskelijat tulisivat suorittamaan. Arviointien suunnitteluun kuului myös kuinka paljon aikaa arvioiden tekemiseen tulisi olemaan sekä päivämäärät, jolloin kunkin arvioinnin olisi oltava valmis. Aikataulut oli lopulta jätettävä tässä vaiheessa vielä melko avoimiksi, sillä niihin vaikutti ratkaisevasti kurssien harjoitustöiden aikataulut, joita ei ollut vielä lyöty lukkoon. Aikataulu muuttui työn kuluessa myös arviointiin käytettävän sovelluksen valinnan, asennuksen ja muiden valmistelevien vaiheiden kohdalla. Se oli kuitenkin tiedossa, että arviointien tulisi olla valmiit ennen kyseessä olevan kurssin tenttiä.

Määrittelyllä oli oma merkityksensä sovelluksen valintaan, jonka avulla vertaisarviointia tultaisiin kokeilemaan. Sen kautta tuli selvitettyksi tietyt järjestelmältä vaadittavat piirteet ja ominaisuudet, jotka olisi löydettävä, jotta suunnitellunkaltaisen arvioinnin tekeminen olisi mahdollista. Siten saatiin myös kriteerit käytettävän järjestelmän valinnalle. Tämän jälkeen oli selvitettävä mikä

saatavilla olevista vaihtoehdoista soveltuisi tarkoitukseen parhaiten, ja miten määritelty vertaisarviointi voitaisiin käytännössä toteuttaa valitulla järjestelmällä. Järjestelmän valintaan vaikuttavia toimintoja olivat esimerkiksi se, miten työt tuotaisiin järjestelmään opiskelijoiden saataville ja arvioitavaksi, miten työt jaettaisiin arvioiville opiskelijoille, sekä miten opiskelijat arvioinnin suorittaisivat ja palauttaisivat.

3.1.1 Network Programming

Kurssin Network Programming kohdalla alustavassa suunnitelmassa päätettiin, että opiskelijoiden ratkaisut ohjelmointitehtäviin tallennettaisiin järjestelmään kooditiedostopaketteina, jotka arvioijat voivat ladata itselleen käsiteltäviksi. Opiskelija arvioisi noin kolmen muun opiskelijan työt annettujen perusteiden mukaisesti. Arvioinnissa opiskelijan olisi annettava töille pisteet, ja pisteytys perusteltaisiin sanallisilla kommentteilla. Opiskelijat saisivat ensin palautetta työstään toistensa tekemien vertaisarviointien välityksellä, ja tämän jälkeen olisi mahdollisuus asettaa arvioinnissa parhaita pisteitä saaneita ratkaisuja kaikkien opiskelijoiden nähtäväksi. Pisteytyksellä voitaisiin tällä tavalla muun muassa osoittaa selkeimmät koodiratkaisut, kun selkeys olisi eräänä pisteytyskriteerinä. Lopulta toteutus jäi kuitenkin tämän kurssin osalta tekemättä.

3.1.2 Concurrent and Parallel Computing

Concurrent and Parallel Computing -kurssilla oli tarkoitus olla neljä ohjelmointitehtävää, joista vertaisarviointia suunniteltiin sovellettavaksi kahdessa, ja toisten kahden kohdalla se jätettäisiin tekemättä. Tarkoituksena oli pyrkiä selvittämään mahdollista eroa oppimistuloksissa tilanteiden välillä, joista toisessa vertaisarviointi oli ollut käytössä ja toisessa ei. Lopulta vertaisarviointi oli käytössä yhden tehtävän kohdalla, mutta kumpikin tilanne toteutui myös näin. Arvioinnissa ratkaisut tallennettiin järjestelmään lähdekoodipaketteina siten, että arvioitaessa koodit oli mahdollista kääntää ja suorittaa. Koodiratkaisut tuotettiin ryhmitöinä, ja kukin ryhmä arvioi yhden muun ryhmän työn. Harjoitustyöryhmiä kurssilla oli yksitoista ja opiskelijoita yhteensä kaksikymmentä. Harjoitusten tarkastajan tehtäväksi suunniteltiin valinta kullekin ryhmälle arvioitavaksi sopivasta ratkaisusta sillä perusteella, että esiin tulisi keskenään erilaisia lähestymistapoja ja saavutettaisiin laajempaa vertailupohjaa myös suhteessa ryhmän omaan ratkaisuun. Opiskelijoiden tuli antaa palautetta pääosin sanallisesti perustellen. Lisäksi arvioinnissa oli muutamia sellaisia harjoitustyön ratkaisun ominaisuuksia käsitteleviä kysymyksiä, joihin oli valittavissa valmiista vastausvaihtoehdoista kuinka paljon kyseinen ominaisuus kuvasi ratkaisua.

3.2 Sovelluksen valinta

Työn alussa, suunnitteluvaiheessa, harkittiin ensimmäisenä mahdollisuutta arviointien toteuttamiseen pääosin Philippe Rigaux'n kehittämällä konferenssipapereiden arviointiin tarkoitettulla MyReview-nimisellä sovelluksella (Rigaux 2008). Koska vertaisarvioinnin testaus oli suunniteltu tehtäväksi kahdella kurssilla, eikä MyReview tarjoa mahdollisuutta useamman kohteen, tässä tapauksessa kurssin, erilliseen käsittelyyn samanaikaisesti, päätettiin kuitenkin tehdä työssä myös kartoitus tarjolla olevista vertaisarviointiin mahdollisesti paremmin soveltuvista ohjelmistoista. Näistä valittaisiin lopulta se, joka soveltuisi työn toteutukseen kaikkein parhaiten.

Sovellusten etsintä ei lopulta tuottanut tulokseksi kovinkaan monia täysin varteenotettavia vaihtoehtoja. Käytön ja käyttöönoton kannalta lupaavimmilta vaikuttaneet sovellukset eivät olleet tämän työn kannalta tarkoituksenmukaisia. Suuri osa tarkastelluista järjestelmistä oli MyReview'n tapaan konferenssikäyttöön suunniteltuja, mutta sillä tavalla jäykemmän rakenteen omaavia, ettei niiden soveltaminen harjoitustöiden vertaisarviointiin olisi ollut kovin helppoa, jos edes mahdollistakaan. Osasta puuttui tämän työn kannalta toivottavia ominaisuuksia kuten tiedostojen tallentaminen järjestelmään ladattavaksi, sanallisten arvioiden tekeminen ilman suuria muutoksia ohjelmistoon tai tiettyjen arvioijien määrittäminen tietyille töille. Lisäksi monet esille tulleet ohjelmistot olivat maksullisia, eivätkä olisi sen takia soveltuneet käytettäväksi tässä työssä. Osa niistä oli myös vanhentuneita, eikä enää saatavilla.

3.2.1 MyReview ja OpenConf

Lopulta tarkempaan tarkasteluun arviointijärjestelmistä valikoitui kaksi ilmaiseksi saatavilla olevaa ohjelmistoa: MyReview ja sen lisäksi Zakon Groupin OpenConf (Zakon Group 2008). Kumpikin on toteutettu pääosin PHP-kielellä ja MySQL-tietokannalla. Kummassakin järjestelmässä ohjelmaan rekisteröidyt henkilöt tai henkilöryhmät voivat ladata tiedoston muiden arvioitavaksi, ja ylläpitäjä voi määrittää arvioijat kullekin tiedostolle. Arvioijat voivat tämän jälkeen ladata itselleen määrätyn tiedoston tarkasteltavaksi, ja antaa palautteen täyttämällä arviointilomakkeen järjestelmässä.

Kumpikin ohjelmisto mahdollistaa jossain määrin esimerkiksi lomakkeiden muotoilemisen, asetusten säätämisen ja lisäksi arvioijien automaattisen tai manuaalisen asettamisen arviointikohteille. Molemmissa ohjelmistoissa on myös valittavissa hyväksyttävät tiedostotyytit muutamasta valmiista vaihtoehdosta, ja oletuksena hyväksyttävät muodot ovat Adobe PDF tai

Microsoft Word -dokumentti ja MyReview:ssa lisäksi muun muassa Zip-paketti. Esimerkiksi OpenConf tarjoaa ylläpitäjälle mahdollisuuden määrittää asetussivujen kautta hyväksyttäväksi formaateiksi näiden lisäksi joitain seuraavista vaihtoehdoista: Corel WordPerfect, HTML, PostScript, RTF, Text (muotoilematon tekstitiedosto) ja XML.

Vertailtaessa järjestelmiä niiden Internet-sivujen perusteella jäi vielä OpenConfin osalta epäselväksi, mikä olisi helpoin tapa tallentaa kyseiseen järjestelmään kooditiedostopaketteja. Dokumentaatiosta tai Internet-demosta ei selvinnyt, voiko hyväksyttäväksi tiedostomuodoksi asettaa minkä tahansa, vai olisiko sitä varten kenties tehtävä muutoksia itse ohjelmiston koodiin. MyReview'n dokumentaatiosta sen sijaan selvisi, että oletuksena tarjottujen tiedostotyyppien listaa voi muuttaa "Constant.php" -nimisestä tiedostosta löytyvää taulukkoa nimeltä "FILE_TYPES" muokkaamalla.

Arvioiden saattaminen työn tekijän nähtäville on MyReview:ssa toteutettu sähköpostia käyttäen. Siinä vaiheessa, kun kaikki arvioinnit on tehty, on mahdollista lähettää järjestelmään kuuluvan toiminnon kautta kullekin tiedoston järjestelmään tallentaneelle viesti, joka kertoo arviointivaiheen tuloksista ja sisältää kaikki tiedostoa koskevat arviot. Viestin sisällön kohdalla käytetään etukäteen valmisteltavaa mallipohjaa, joka mahdollistaa viestien lähettämisen kaikille arvioitavia töitä palauttaneille automaattisesti yhdellä kertaa. OpenConf sisältää ylläpitäjän sivulta löytyvän sähköpostitoiminnon, jolla voi myös lähettää automaattisesti erilaisia ilmoitusviestejä määritellyille vastaanottajaryhmille. Tämän toiminnon avulla kuitenkin lähetetään ainoastaan ilmoitus arviointivaiheen päättymisestä, jonka jälkeen töiden tekijät voivat nähdä arvioinnit kirjautumalla järjestelmään ja avaamalla omasta näkymästään löytyvän arviointien tarkasteluun tarkoitettun sivun. Muita sovellusten välisiä eroja on koottuna taulukossa 1.

Ominaisuus	MyReview	OpenConf Community Edition
Ohjelmointikieli	PHP	PHP
Tietokanta	MySQL	MySQL
Tiedostojen tallentaminen järjestelmään	Kyllä, valittavissa olevat oletusformaatit: Adobe PDF, Microsoft Word, PostScript ja Zip	Kyllä, valittavissa olevat oletusformaatit: Adobe PDF, Microsoft Word, Corel WordPerfect, HTML, PostScript, RTF, Text ja XML
Muiden hyväksyttävien tiedostomuotojen lisääminen	Muokkaamalla taulukkoa FILE_TYPES tiedostossa Constant.php	Ei dokumentoitu järjestelmän kotisivuilla
Arvioijien asettaminen tiedostoille	Manuaalinen tai automaattinen	Manuaalinen tai automaattinen
Arviointi järjestelmässä olevan lomakkeen avulla	Kyllä, muokattavissa oleva arviointilomake	Kyllä
Sähköposti-ilmoitukset	Kyllä	Kyllä
Arviointien tuottaman palautteen toimittaminen työn tekijälle	Muokattavaan malliin pohjautuvilla automatisoitavilla sähköpostiviesteillä	Tekijöiden nähtävissä omalla sivullaan heidän kirjaututtuaan järjestelmään.
Käyttöliittymän muokattavuus	Muokattava, perustuu PHP Templateihin eli malleihin. Useimmat järjestelmässä esiintyvät tekstit ovat muokattavissa XML-dokumentin kautta.	Jotkin lomakkeet/tekstit ovat osittain muokattavissa ylläpitäjän käyttöliittymän kautta. Jotkin osiot kuten esimerkiksi konferenssiohjelmat ja konferenssipapereiden yhteenvedot ovat piilotettavissa.
Tuki usealle samanaikaiselle konferenssille/tapahtumalle	Ei	Ei

Taulukko 1. MyReview'n ja OpenConfin olennaisimmat ominaisuudet.

Kaikkein selkeimmin järjestelmien välisistä eroista tuli esille käyttöliittymien erilaisuus. MyReview vaikutti vertailussa yksinkertaisemmalta, selkeämmältä ja helpommalta omaksua. On kuitenkin todettava, että MyReview'n demosivustoon Internetissä ei pystynyt kunnolla tutustumaan, sillä ohjeessa mainittu tunnus-salasana-pari "myreview@lri.fr - 08b271" tuotti vain seuraavan virheilmoituksen: "No user myreview@lri.fr !". Toisaalta MyReview-sivustolta löytyi laaja dokumentaatio kyseisestä järjestelmästä, joskin se oli joiltain osin ilmeisesti vanhentunut tai muuten puutteellinen. Esimerkiksi ohjelmiston tiedostojen nimet eivät olleet täysin yhtäpitävät dokumentaation kanssa. OpenConf-sivustolla sen sijaan dokumentaatio oli hyvin suppea. OpenConfin demosivuston avulla järjestelmään voi sen sijaan tutustua kaikkien eri käyttäjien näkökulmista. Kuitenkaan esimerkiksi mitään muutoksia asetuksiin ei voinut tallentaa, eikä täten kokeilla niiden todellisia vaikutuksia.

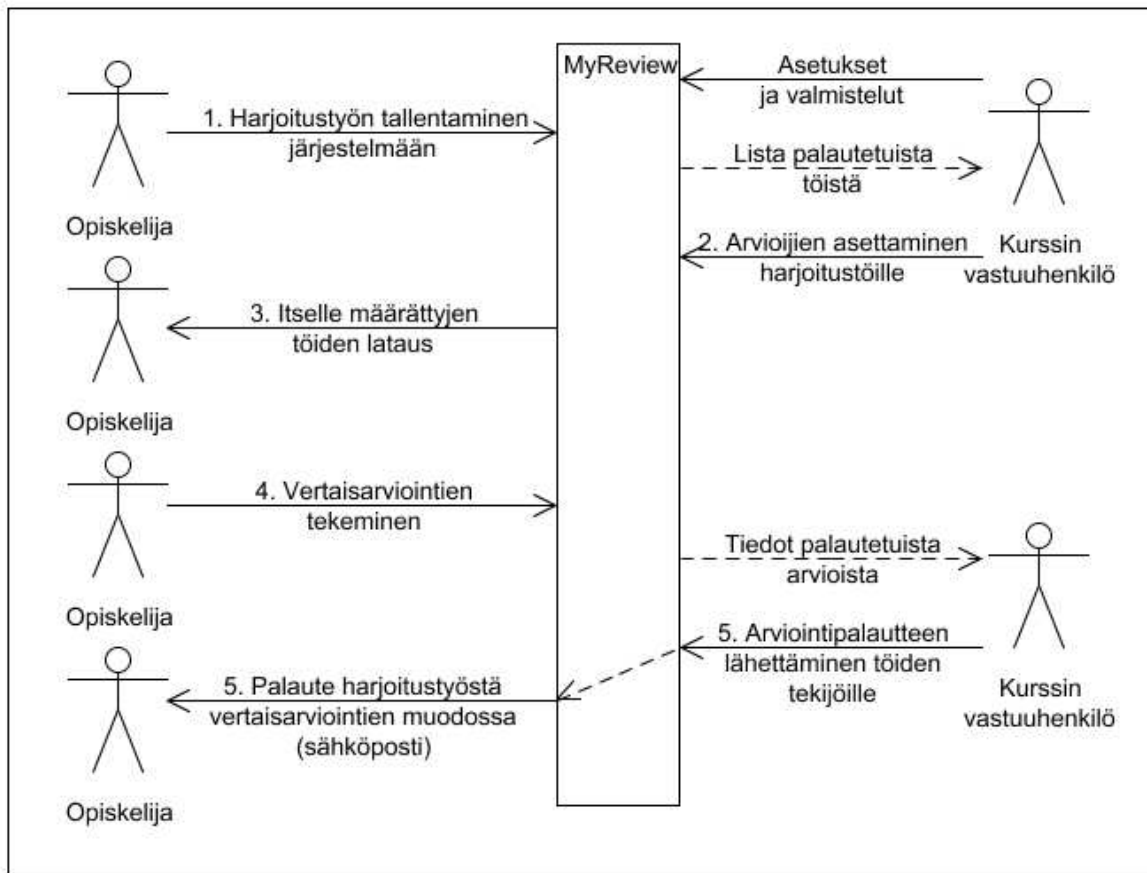
MyReview oli ensimmäinen sovellus, jota harkittiin käytettäväksi tämän työn toteuttamisessa. Tarve muiden vaihtoehtojen kartoittamiseen ja tutkintaan syntyi siitä huomiosta, että MyReview

vaatii uuden asennuksen jokaiselle tapahtumalle, jossa arviointeja tehdään, eli konferenssille tai tässä tapauksessa kurssille. Varteenotettavin vaihtoehto MyReview:lle, eli OpenConf, ei vaikuttanut olevan tässä suhteessa yhtään erilainen sen perusteella, mitä oli mahdollista sanoa asentamatta järjestelmää ja kokeilematta sitä käytännössä. Tapahtuman, eli konferenssin, kurssin tai vastaavan, tiedot voi kenties poistaa ja muuttaa sopivaksi seuraavaa tapahtumaa varten. Mitään varsinaista ominaisuutta, joka viittaisi mahdollisuuteen hallita useampaa tapahtumaa yhtä aikaa, ei kuitenkaan löytynyt.

Näillä perusteilla vertailussa päädyttiin siihen lopputulokseen, että ohjelmointitehtävien vertaisarvioinnin voi todennäköisesti toteuttaa suurin piirtein samalla tavalla kummallakin järjestelmällä. MyReview oli kuitenkin oletettavasti käytännöllisempi vaihtoehto selkeämmän käyttöliittymänsä sekä varsinkin sen seikan takia, että sitä oli käytetty yliopistolla aikaisemminkin. Näin ollen asennus ja käyttöönotto sekä mahdollisten ongelmatilanteiden selvittäminen kävisi helpommin, sillä käytettävissä olisi ainakin jonkinlaista tietoa ja kokemusta järjestelmän käytöstä. Merkittävin seikka, joka puhui OpenConfin valintaa vastaan, oli puolestaan työn toteuttamisen tiukka aikataulu, josta johtuen viimeistelemättömältä vaikuttavalla käyttöliittymällä varustetun uuden järjestelmän käyttöönotto vaikutti selkeästi riskialttiimmalta vaihtoehdolta.

3.3 MyReview-ohjelmiston käyttö

MyReview'n käyttäjän on mahdollista toimia erilaisessa rooleissa. Käyttäjä voi olla työn järjestelmään arvioitavaksi tallentavan henkilön lisäksi joko järjestelmän ylläpitäjä, konferenssin tai muun tapahtuman, esimerkiksi tässä tapauksessa kurssin, vastuhenkilö, eli sen ylläpitäjä arviointijärjestelmän osalta, tai arvioija. Näiden roolien erilaiset yhdistelmät ovat myös mahdollisia. Esimerkiksi vertaisarvioinnin testauksessa arvioijat ja töiden arvioitavaksi laittajat ovat keskenään samoja henkilöitä, ja vastaavasti järjestelmän ja tapahtumien ylläpitäjät ovat myös samoja. Kuvassa 1 on esitetty MyReview'n avulla toteutettavan vertaisarviointiprosessin päävaiheet.



Kuva 1. Vertaisarviointiprosessi MyReview'n avulla.

3.3.1 Asetukset ja valmistelut

Ennen vertaisarvioinnin toteutusta MyReview-sovelluksen asetukset on säädettävä harjoitustöiden arviointia varten sopiviksi. Yleisimmän tason muokattavia yksityiskohtia ohjelmistossa ovat esimerkiksi arviointia käyttävän konferenssin – tai tässä tapauksessa kurssin – nimi, ylläpitäjän sähköpostiosoite ja muun muassa käytettävä päivämäärien ilmaisumuoto. Konfiguroitavana on runsaasti myös varsinaiseen arviointien toteutukseen liittyviä yksityiskohtia kuten käytettävät tiedostomuodot, arvioijien määrä yhtä työtä kohti ja päivämäärät, jolloin prosessin eri vaiheiden tulee olla valmiit. Muun muassa nämä edellä mainitut asetukset on määritettävä järjestelmän ylläpitäjän käyttöliittymän kautta. Konfiguroitavia parametreja, sekä huomioita ja ehdotuksia niiden arvoihin liittyen on esitelty lisää liitteessä 1. Näiden lisäksi ylläpitäjän käyttöliittymän kautta on mahdollista muokata käytettävät lomakkeet asianmukaisiksi ja esimerkiksi asettaa arviointilomakkeelle valmiita kysymyksiä vastausvaihtoehtoinen. Arviointisivulle on hyvä lisätä arvioijien nähtäväksi myös arviointikriteerit, joiden perusteella töitä tulisi kommentoida.

MyReview tarjoaa mahdollisuuden luokitella tallennettavia konferenssipapereita aihealueiden ("Research Topics") mukaan. Tällä toiminnolla ei välttämättä ole kuitenkaan suurta käyttöä harjoitustöiden vertaisarvioinnissa, mikäli käsitellään vain yhtä kurssia kerrallaan. Mikäli yhtä järjestelmän asennusta käytettäisiin yhtä aikaa esimerkiksi kahdella kurssilla, aihealueita voitaisiin ehkä käyttää töiden luokitteluun eri kurssien piiriin kuuluviksi. Tämä voisi kuitenkin aiheuttaa komplikaatioita järjestelmän käyttöön kurssien ylläpitäjien näkökulmasta, mikäli useamman kurssin työt ja arvoinnit olisivat käytännössä sekaisin keskenään. Sovelluksessa on kuitenkin myös hakutoiminto, jolla voi valita näytettäväksi esimerkiksi vain tiettyyn aiheeseen liittyvät työt. Aihealueita voi muokata järjestelmän ylläpitäjän käyttöliittymän kautta.

Järjestelmästä löytyy myös toiminto, joka mahdollistaa kokonaisarvosanan laskemisen arvioitavalle työlle sen mukaan, minkälaisia arvosanoja arvioija on erilaisten kriteerien perusteella sille antanut. Nämä kriteerit sekä niiden painotus kokonaisarvosana laskettaessa ovat ylläpitäjän muokattavissa. Oletuksena arvosanat lasketaan osatekijöiden alkuperäisyys ("originality"), laatu ("quality"), olennaisuus ("relevance"), esitys ("presentation") ja suositeltu arvosana ("recommendation") perusteella. Ohjelmakoodia arvioitaessa sanalliset kommentit ovat tärkein palautteen muoto, mutta mikäli myös arvosanat halutaan arviointeihin mukaan, on syytä huomata, että oletuksena kokonaisarvosanan paino on kokonaan suositellulla arvosanalla, eli ne ovat automaattisesti keskenään samat. Lisäksi ohjelmiston toiminnan kannalta on tärkeää, ettei kriteereitä muuteta enää sen jälkeen, kun arviointeja on alettu tallentaa.

Edellä mainittujen asetusten ja valmistelujen lisäksi järjestelmän ylläpitäjän tehtävänä on määrittää järjestelmän käyttäjät, eli esimerkiksi lisätä järjestelmään tarvittavat tunnukset arvioijia varten. Ryhmätyön ollessa kyseessä kätevinä lienee määrittää ryhmästä vastuuhenkilö, jonka sähköpostiosoite toimii tunnukseksi. Ylläpitäjän on myös syytä huolehtia, että järjestelmä on konfiguroitu hyväksymään tallennettavat työt halutussa tiedostoformaattissa. Mikäli haluttua tiedostomuotoa ei löydy listattuna valmiiden vaihtoehtojen joukossa, puuttuvan tiedostotyypin voi lisätä "Constant.php" -nimisestä tiedostosta löytyvää taulukkoa nimeltä "FILE_TYPES" muokkaamalla.

1. **Administrator menu > Configure the system:** Järjestelmän asetusten määrittäminen.
2. **Administrator menu > Program committee:** Tunnusten luominen arvioijia varten. Täytettävät tiedot: opiskelijan (tai ryhmän vastuuhenkilön) sähköpostiosoite, nimet, yhteys ("affiliation", esimerkiksi ryhmän nimi/numero) ja käyttäjärooli (valittava vain "Reviewer"). Valitsemalla tunnuksen kohdalta sarakkeesta "Instructions" linkin "send" voi lähettää luodun tunnus-salasana-pari kyseiselle arvioijalle.
3. **Administrator menu > Criteria:** Arviointilomakkeen tarjoamien alavetovalikolla valittavien arviointikriteerien muokkaus sopiviksi tai poistaminen. (Oletuksena vastausvaihtoehdot näihin ovat väliltä vahva hyväksyntä ["strong accept"] - vahva hylkäys ["strong reject"].)
4. **Administrator menu > Paper questions:** Harjoitustyöntallennuslomakkeella kysyttävien lisätietojen muokkaus tai poistaminen. Mahdollisiin kysymyksiin määritetään valmiit vastausvaihtoehdot, sekä näiden järjestys numeroin.
5. **Administrator menu > Review questions:** arviointilomakkeen sisältämien valmiiden kysymysten muokkaus. Mahdollisiin kysymyksiin määritetään valmiit vastausvaihtoehdot, sekä näiden järjestys numeroin.
6. **Administrator menu > Templates edition:** Järjestelmän käyttämien tekstien mallipohjien muokkaus. Vähintään töiden tekijöille lähetettävän vertaisarvioinnit sisältävän palautesähköpostin (esimerkiksi "MailAccept") teksti kannattaa muuttaa vastaamaan tarkoitustaan. Lisäksi "TxtInfoReview" -malliin on mahdollista lisätä ohjeita opiskelijoille arvioinnin tekemistä varten, esimerkiksi mihin arvioinnissa tulisi kiinnittää huomiota. Kyseinen teksti esitetään arviointikäyttöliittymässä arviointilomakkeen yläpuolella. Mallien loppuun kannattaa jättää kaksi tyhjää riviä, jotta niiden ja niiden käyttöympäristön sisällön väliin jää tilaa.

MyReview-ohjelmiston käytön vaatimat askeleet.

3.3.2 Harjoitustyön tallentaminen järjestelmään

Seuraava vaihe on tiedostojen tallentaminen. Asetusten ja muiden valmistelujen tekemisen jälkeen opiskelijoille on annettava tehtäväksi harjoitustyötiedostojen tallentaminen järjestelmään arviointia varten. Opiskelijoille on annettava ohjeet tämän tekemiseen sekä linkki tiedostontallennussivulle, jolta löytyvällä lomakkeella tallentaminen tapahtuu. Lomakkeen kaikki kentät on täytettävä, joten sen olisi syytä olla tarvittaessa siten muokattu, ettei se sisällä ylimääräisiä kohtia. Kun opiskelija lähettää lomakkeen tarvittavilla tiedoilla kuten tekijöiden nimillä ja harjoitustyötiedostolla täydennettynä, hän saa työlle tunnusteen ja salasanan. Näiden avulla työtä ja sitä koskevia tietoja on mahdollista muuttaa niin kauan kunnes ylläpitäjä päättää tämän vaiheen ja sulkee tiedostojen palautuksen. Kun käytetään nimetöntä arviointia ("Blind review"), MyReview ei näytä töiden tekijöiden nimiä arvioijille. On kuitenkin huomioitava, että nimettömyyden säilyttämiseksi

myöskään tallennettavista tiedostoista, tai muista kuin lomakkeiden varsinaisista nimikentistä ei tule löytyä tekijöiden nimiä.

7. **Authors > Submit an abstract:** Tiedostontallennuslomake, johon on täytettävä kohdat: työn nimi, opiskelijan (vastuuhenkilön) sähköpostiosoite, työn tekijöiden nimet, mahdollisten muiden kysymysten vastaukset, "tiivistelmä" sekä harjoitustyötiedosto.
8. **Authors > Upload a paper:** Tarvittaessa työn tallentanut opiskelija voi muuttaa tiedostoaan tai sen tietoja kirjautumalla tallennuksen yhteydessä saamallaan "paper id" -tunnuksella ja salasanalla kunnes ylläpitäjä asettaa tiedostojen tallennuksen pois käytöstä.

MyReview-ohjelmiston käytön vaatimat askeleet.

3.3.3 Arvioijien asettaminen harjoitustöille

Tapahtuman ylläpitäjän tehtävänä on määrittää kuka tai mikä ryhmä arvioi minkäkin työn. Tämän voi tehdä kahdella tavalla: joko työ kerrallaan tai kaikki työt sisältävästä taulukkonäkymästä. Sovellus mahdollistaa tämän tehtävän suorittamisen myös automaattisesti. Kun halutaan antaa opiskelijoille mahdollisimman laaja ja monipuolinen näkemys erilaisista töistä sekä tarjota vaihtoehtoisia ratkaisumalleja opiskelijoiden tarkasteltavaksi, opettajan on kuitenkin parempi valikoida arvioitavat työt arvioijille itse. Tällä tavalla voidaan pyrkiä varmistamaan, ettei saman opiskelijan käsiteltävänä olisi ainoastaan hyvin samankaltaisia töitä.

9. **PC chair menu (Submission phase) > List of submitted papers:** Kunkin työn kohdalla on lista mahdollisista toiminnoista ("Actions"), joista valitsemalla "Assign" -linkin pääsee valitsemaan kyseiselle työlle arvioijat alasettovalikkojen avulla.

Tai

- PC chair menu (Submission phase) > Manual assignment – check and modify:** Tallennetut työt ja järjestelmään luodut arvioijat ovat taulukossa, jossa jokaisen työ-arvioija-parin kohdalta voidaan määrätä arviointi tehtäväksi ("Y") tai poistaa määräys ("N").

MyReview-ohjelmiston käytön vaatimat askeleet.

3.3.4 Vertaisarviointien tekeminen

Kun harjoitustöille on asetettu arvioijat, ja tarvittavat tunnukset sekä linkki oikealle sivulle on toimitettu vertaisarvioinnin tekeville opiskelijoille, voidaan aloittaa arviointivaihe. Tällöin opiskelijat voivat kirjautua järjestelmään, nähdä heille arvioitavaksi määrättyt työt, ladata tiedostot itselleen ja tallentaa niistä arvion siihen tarkoitettulla lomakkeella. Arvioijien käyttöliittymä

mahdollistaa harjoitustyötiedoston lataamisen järjestelmästä, arviointilomakkeen täyttämisen ja päivittämisen kunnes arviointivaihe päättyy, ja mikäli kyseinen ominaisuus on käytössä, myös muiden arvioijien palauttamien arvioiden tarkastelemisen arvioijien välisen vuorovaikutuksen vaiheessa ("discussion phase").

10. **Reviewers:** Kirjaututtuaan tälle sivulle ylläpitäjältä saamallaan tunnuksilla opiskelija näkee listan harjoitustöistä, joita hänen tulee arvioida. Kunkin työn kohdalla on toimintovalikko ("Actions"), jonka "download" -linkistä kyseisen työn voi ladata itselleen tarkasteltavaksi, ja Submit/Update my review -linkistä voi siirtyä täyttämään arviointilomaketta. Lisäksi arvionsa voi avata vain katselua varten tai tulostaa.
11. Arviointilomakkeella opiskelijan on valittava vastauksensa ylläpitäjän määrittämiin mahdollisiin arviointikriteereihin ja kysymyksiin valmiista vaihtoehdoista sekä annettava kirjalliset kommenttinsa yhteenveto ("Summary") ja yksityiskohdat ("Details") -kenttiin. Myös ylläpidolle on mahdollista antaa palautetta ("Comments for Program Committee").

MyReview-ohjelmiston käytön vaatimat askeleet.

3.3.5 Arviointipalautteen lähettäminen töiden tekijöille

Lopuksi, kun kaikki tiettyä työtä käsittelevät arviot on palautettu, ylläpitäjä voi määrittää työlle tilan ("Status of papers"). Tämän voi tehdä yksi kerrallaan tai asettamalla kaikille töille kerralla saman tilan. Tilan on oltava asetettuna, kaikkien arviointien on oltava palautettuna, ja viestipohjien on oltava valmiina ja sopiviksi muotoiltuina, jotta töiden tekijöille voidaan lähettää palauteviestit, joissa on mukana kaikki tekijän omaa työtä koskevat arviot. Ylläpitäjän käyttöliittymästä löytyvä "Close the selection phase" -linkki tarkistaa näiden ehtojen toteutumisen. Palauteviestien lähettämisen voi tehdä esimerkiksi muokkaamalla viestipohjan "MailAccept" tarkoitukseen sopivaksi ylläpitäjän "Templates edition" -sivulta, ja asettamalla töiden tilaksi hyväksytty ("Accept"). Tämän jälkeen palautteet voi lähettää yksi kerrallaan valitsemalla töiden tilat listaavalta sivulta ilmoituslinkin, joka johtaa edellä mainitulla viestipohjalla alustettuun sähköpostilomakkeeseen. Tässä vaiheessa viestiä on vielä mahdollista muokata ennen sen lähettämistä. Toinen vaihtoehto on lähettää kaikki palautteet kerralla ylläpitäjän käyttöliittymän pääsivulta löytyvän linkin kautta. Tässä tapauksessa ennen varsinaista lähetystä näkyviin tulee lista viestipohjista, joita palauteviesteissä tullaan käyttämään.

12. **PC chair menu (selection phase) > Status of papers:** Kunkin työn kohdalla on valittavana jokin etukäteen asetetuista vaihtoehdoista työn tilaksi. Oletuksena tilat ovat "Accept" ja "Reject". Valitsemalla tästä töille esimerkiksi vaihtoehto "Accept" valitaan samalla harjoitustöiden tekijöille lähetettävän palautesähköpostin käyttämäksi tekstipohjaksi "MailAccept".
13. **PC chair menu (selection phase) > Status of papers:** Töiden kohdalta löytyy "Show infos" -linkki, josta ylläpitäjä näkee tallennuksen yhteydessä annettujen tietojen lisäksi työstä palautetut arviot.
14. Samalta "Status of papers" -sivulta löytyy töiden kohdalta myös "Notify authors" -linkki, josta aukeavalla automaattisesti esitäytetyllä lomakkeella kyseisen työn tekijöille voi lähettää palautteen kaikkine arvioineen. Mikäli kaikki arviot on palautettu, töiden tilat asetettu ja viestipohjat muokattu sopiviksi, kaikki palautteet voi lähettää myös kerralla "PC chair menu" -valikon lopusta löytyvällä "Send Paper status to each author" -toiminnolla. Lähetettävä viesti ("Paper status") valitaan alavetovalikosta, jonka jälkeen Check template -nappia painamalla päästään lomakkeelle, jolla palautteet lähetetään. Lomakkeelle ei kirjoiteta varsinaista lähetettävää viestiä, vaan se kertoo, minkä pohjien mukaisia tekstejä viesteissä tullaan käyttämään

MyReview-ohjelmiston käytön vaatimat askeleet.

3.4 Testauksen toteutus

Käytetyn sovelluksen asennus jäi melko myöhäiseen vaiheeseen työn aikataulua monestakin syystä johtuen. Eri järjestelmävaihtoehtojen vertailuun päätettiin käyttää aikaa vielä sen jälkeen, kun asennus ensimmäisen kerran oli suunniteltu jo aloitettavaksi. Lisäksi yliopiston henkilöstö ei ollut aina tavoitettavissa muun muassa sairastumisen vuoksi, ja tietoa asennuksen valmistumisesta, tunnuksia ja salasanoja jouduttiin odottamaan jonkin aikaa.

Käytännön toteutuksen ensimmäisenä vaiheena käytettävän ohjelmiston asennuksen jälkeen oli sen laittaminen käyttöön. MyReview'n käyttö oli helppo oppia, sillä sovelluksen käyttöliittymä on suhteellisen intuitiivinen, ja suurimman osan tarvittavista muutoksista asetuksiin voi tehdä suoraan sen kautta. Ennen varsinaisen vertaisarvioinnin aloittamista prosessi käytiin läpi kaikkine asetuksineen kokeilutunnuksia, -tiedostoja ja -arviointeja käyttäen, jotta nähtäisiin, toimiiko kaikki kuten on tarkoitus. Kaikki tällä tavalla testattu toiminnallisuus toimi ongelmitta, mutta testatessa järjestelmään väliaikaisesti tallennetut tiedot jouduttiin poistamaan suoraan ohjelmiston käyttämästä tietokannasta, sillä niiden poistaminen käyttöliittymän toimintojen kautta on estetty turvallisuussyistä. Samaan tapaan myös järjestelmän muun muassa sähköpostiviesteihinsä

käyttämien tekstipohjien kirjoitusoikeuden muuttaminen jouduttiin tekemään ulkopuolelta käsin, sillä oletusarvoisesti oikeutta pohjien muokkaamiseen ei ollut, eikä käyttöliittymästä löydy toimintoa sen muuttamista varten.

Valmisteltaessa varsinaista vertaisarviointia testattiin vielä, toimiiko tunnusten ja salasanojen lähettäminen arvioijille järjestelmän sähköpostitoiminnolla, joka löytyy samasta näkymästä, jossa tunnukset luodaan. Tässä vaiheessa vaikutti siltä, ettei lähetetty kokeiluviesti saavu perille. Järjestelmässä itsessään ei kuitenkaan ollut vikaa, vaan ongelma oli testitapauksessa sen ulkopuolella. Varsinaiset viestit saatiin lähetettyä arvioijille, ja ainoaksi hankaluudeksi jäi epäkäytännöllisyys toiminnon luonteessa, sillä viestit oli lähetettävä yksi kerrallaan automaattisen joukkoviestityksen sijaan.

Koska kahta kurssia ei ole mahdollista käsitellä kätevästi samalla asennuksella, laitettiin tarvittavat asetukset ensin valmiiksi Concurrent and Parallel Computing -kurssin harjoitustöitä varten, kun MyReview oli ensimmäisen kerran saatu asennettua. Sen jälkeen asennettiin toinen kappale ohjelmistoa kurssia Network Programming varten. Asennuksen valmistuttua kyseisen kurssin harjoitustyöt oli jo tehty ja palautettu, mutta suuremman ongelman muodosti kuitenkin MyReview'n soveltuvuus kurssin arviointitarpeisiin. MyReview'n toiminnallisuus ja olemus kokonaisuudessaan ei vastannut vaatimuksia, jotka sen olisi tullut täyttää sopiakseen käytettäväksi kurssin harjoitustöiden vertaisarvioinnissa. Sovellusta olisi pitänyt muokata konferenssipapereiden arvioinnista nimenomaan harjoitustöiden suuntaan. Esimerkiksi kaikki käyttöliittymästä löytyvät viittaukset konferenssipapereihin ja niiden tiivistelmiin olisi pitänyt poistaa tai muokata kyseiset kohdat siitä täysin erilaisiksi.

Kyseinen muutostyö olisi suurella todennäköisyydellä edellyttänyt käyttöoikeuksia suoraan ohjelmiston asennukseen kuuluviin tiedostoihin, mitä työn toteutuksessa ei ollut käytettävissä, sekä vienyt enemmän aikaa kuin siihen oli mahdollista käyttää. Seuraava työvaihe oli opiskelijakyselyn valmistelemine ja toteuttamine, joka täytyi saada ajoissa valmiiksi, jotta tulokset saataisiin kerättyä ennen kurssin ja lukukauden päättymistä. Näin ollen, koska sovellus ei valmiiksi ollut kelpoisessa kunnossa Network Programming -kurssia varten, eikä tarvittavien parannuksien tekemiseen ollut riittävästi aikaa, vertaisarvioinnin testaus rajoittui lopulta yhteen kurssiin, ja toinen jouduttiin jättämään pois.

Järjestelmänvalmisteluvaiheen jälkeen opiskelijat saivat vertaisarvioinnin tehtäväkseen. Tämä vaihe ei itsessään tuottanut muita suuria ongelmia, mutta hankaluudet aikataulun kanssa olivat todella merkittäviä. Kaikki arvioinnit eivät olleet palautettuina alkuperäisen suunnitelman mukaiseen päivämäärätakarajaan mennessä, eikä niitä saatu palautuksen jälkeen riittävän nopeasti vastaanottajilleen, joten myös seuraavan vaiheen, eli opiskelijakyselyn, tulosten valmistuminen viivästy edelleen. Opiskelijoiden oli saatava palaute ennen kuin he voisivat vastata arviointia ja sen kautta saatua palautetta koskeviin kysymyksiin.

3.5 Opiskelijakysely

Seuraavassa työvaiheessa opiskelijoille tehtiin kysely, jonka avulla pyrittiin selvittämään vertaisarvioinnin hyötyjä. Kysely toteutettiin Webropol-kyselysovelluksen avulla. Kysymykset tallennettiin siihen valmiiksi ennen vertaisarviointivaiheen päättymistä, jotta opiskelijat voisivat vastata kyselyyn välittömästi, kun he olivat saaneet palautteen toisiltaan. Tämä oli työn aikataulun loppuvaiheessa välttämätöntä, sillä tulokset pyrittiin saamaan kootuiksi kurssin aikana.

Kysymysten suunnittelussa keskityttiin erityisesti niihin asioihin, joita etukäteen arvioitiin mahdollisiksi saavuttaa arvioinnin avulla. Oppimisen ja vertaisarvioinnin kannattavuuden kannalta paras mahdollinen lopputulos olisi se, että vertaisarviointista olisi hyötyä kaikille osallistuville osapuolille. Kysymykset yritettiin laatia ja muotoilla siten, että vastausten perusteella olisi mahdollista arvioida, kuinka hyvin tämä tavoite saavutettiin. Kysymyskohtaisesti otettiin huomioon myös muoto, jossa vastaukset tuli antaa. Tarkan muotoilun tarkoituksena oli varmistaa, että tuloksena saatavaa aineistoa olisi helppo tulkita ja että sen perusteella olisi mahdollista tehdä päätelmiä, joilla on todellista merkitystä.

Pääosa kysymyksistä jakautui opiskelijoiden kokemuksiin ensinnäkin vertaispalautteen saajan roolissa ja toiseksi vertaispalautteen antajan, eli toisten harjoitustöiden arvioijan roolissa. Muutama kysymys käsitteli vertaisarviointia myös kokonaisuutena, ja lisäksi opiskelijoilla oli mahdollisuus antaa lopuksi vapaata palautetta. Kyselyssä esitetyt kysymykset löytyvät liitteestä 2.

Vastauksia kyselyyn alettiin saada vasta tarkoitetun aikarajan jälkeen, ja näin ollen kyselyn tuloksia päästiin käsittelemään työssä suunnitellun aikataulun kannalta myöhässä. Kaikilta opiskelijoilta ei

ehditty saamaan vastauksia ennen ajan loppumista. Kurssilla oli kaksikymmentä opiskelijaa ja yksitoista heistä muodostettua harjoitustyöryhmää, joilta työhön saatiin ajoissa mukaan kuusi vastausta. Ongelma aiheutui tämän vaiheen sijoittumisesta aivan testauksessa mukana olleen kurssin loppuun sekä siitä, että opiskelijat eivät saaneet vertaispalautteita omista töistään vielä silloin kuin olisi jo pitänyt. Näin ollen he eivät voineet aloittaa kyselyyn vastaamista riittävän aikaisin.

3.6 Opettaja-haastattelut

Kyselyn tuloksia odotellessa tehtiin kurssin opettajien haastattelut. Kurssin luennoitsijana haastattelussa oli professori Jari Porras. Kokonaisuutena vertaisarvioinnista muodostui hänelle positiivinen kuva. Opettajan jokaiselle työlle antamaa yksilöllistä palautetta vertaisarvioinnilla ei voitaisi korvata, mutta sen avulla voidaan saavuttaa lisäarvoa opiskelijoiden muuten saamaan palautteeseen. Luennoitsijan näkemyksen mukaan vertaispalaute tuo useiden lähteiden myötä keskenään erilaisia näkökulmia töiden arviointiin opettajan kenties paljolti tekniseen puoleen keskittyneen rinnalle. Lisäksi opiskelijoiden välillä tapahtuvalla vuorovaikutuksella on positiivinen vaikutus yhteisöllisenä toimintana.

Harjoitustyön ohjaajana kurssilla toimineella Jani Peusaarella oli samansuuntaisia ajatuksia. Vertaispalaute voisi olla hyödyllistä opettajan antaman palautteen lisäksi, mutta hänkään ei mielellään laskisi opiskelijoiden palautteensaantia kokonaan sen varaan. Opiskelijoiden tekemissä arvioissa oli hänen mukaansa yksittäisiä tapauksia, joissa jokin oli mennyt väärin. Kokonaisarvostelut vaikuttivat kuitenkin hänen mielestään menneen kuten oli pitänytkin, ja pisteytykset olivat olleet oikeassa järjestyksessä.

Vertaisarvioinnin heikkoutena nousi luennoitsijan haastattelussa esille yhdenmukaisuuden puute arvioiden välillä. Kun eri töitä arvioivat eri henkilöt, arvioiden välinen vertailukelpoisuus on heikko. Vaikka opiskelijoiden tekemät arviot olisivatkin keskimäärin hyvin tehtyjä, heillä on keskenään erilaiset lähtökohdat arvioinnin tekemiseen, erilaisia näkemyksiä aiheesta, eivätkä kaikki heistä panosta arviointiin aivan yhtä paljon. Arvioijien itsensä välisten erojen lisäksi heillä on arviointia tehdessään vertailukohtana omat keskenään erilaiset ratkaisunsa kyseessä olevaan

tehtävään. Niiden erot arvioitaviin töihin sekä arvioijien töiden väliset erot voivat myös vaikuttaa arvioiden vertailtavuuteen.

Tässä työssä toteutetussa vertaisarvioinnissa opiskelijat olivat kummankin haastatellun opettajan mielestä arvioineet pääosin kuten oli tarkoituskin niitä kohtia, joihin heitä oli ohjeistettu kiinnittämään huomiota, ja arvosteluille löytyi sanallisia perusteluita. Samoin esimerkiksi ongelmaksi havaittuja kohtia oli tuotu esille, ja kommentoitu sanallisesti. Arvioiden välillä ei esiintynyt kovin suurta vaihtelua tässä suhteessa.

Vertaisarvioinnin toteutuksessa aiheutti jonkin verran hankaluuksia se, että käytetty ohjelmisto on suunniteltu toisenlaista ja hieman monimutkaisempaa tarkoitusta varten. Kohdatuista ongelmista kuitenkin opittiin työn aikana, ja niitä osataan ensimmäisen kokeilukerran jälkeen välttää. Luennoitsijan mukaan erityisesti opiskelijamäärältään suurilla kursseilla tämänkaltaisesta järjestelmästä voisi olla apua vertaisarviointien hallintaan, ja on mahdollista, että toteutusta tullaan tulevaisuudessa kokeilemaan sellaisella kurssilla.

3.7 Tulosten yhteenveto ja tulkinta

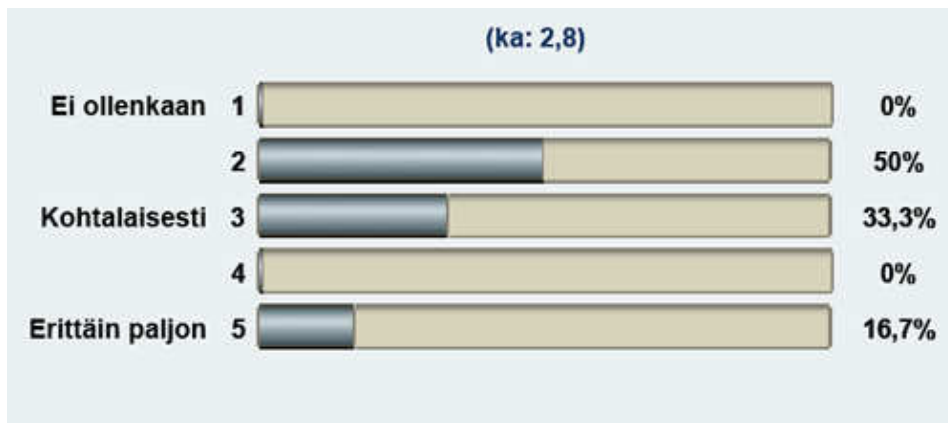
Edellä esitettyjen opettajahaastatteluissa käsiteltyjen kokemusten lisäksi tämän työn aikana ei ehditty saada käsittelyyn kovin suurta määrää uutta aineistoa vertaisarvioinnin hyödyllisyyden analysointia varten. Aiheesta aikaisemmin tehdyissä tutkimuksissa tosin on saatu pääosin sen suuntaisia tuloksia, että vertaisarvioinnilla voidaan saavuttaa etuja oppimisessa. Opiskelijoiden mielipiteissä vertaisarviointia kohtaan on kuitenkin havaittavissa jonkin verran vaihtelua. Useissa tutkimuksissa on myös saatu viitteitä siitä, että opiskelijat pystyvät arvioimaan toistensa töitä laadukkaammin kuin itse uskovat. Niiltä osin kuin tämän työn aikana kokoon saatuja tuloksia voidaan verrata muihin tutkimuksiin, näyttö on näiden seikkojen osalta samansuuntaista.

Kyselyyn vastanneista opiskelijoista noin 67 %, eli suurin osa, kertoi saavansa vain hyvin harvoilla kursseilla palautetta ratkaisuihinsa ohjelmointitehtäviin. Yleensä kerrotaan vain se, onko tehtävä hyväksytty vai hylätty. Yhdessä vastauksessa kerrottiin hyvien ja huonojen puolien mainitsemisesta saadussa palautteessa, mutta kukaan ei ollut yleensä saanut palautetta suoraan lähdekoodista.

Pääosin opiskelijat tunsivatkin saavansa palautetta liian vähän. Kaikki vastanneet pitivät vertaisarvioinnista saatua palautetta hyvänä lisänä muulle, pääosin opettajalta saadulle, palautteelle.

Kysyttäessä määrällisiä kysymyksiä vertaisarvioiden laadusta ja hyödyllisyydestä vastaukset jakautuivat erittäin tasaisesti. Vastaukset kysymyksiin kuten ”kuinka kattavaa palaute oli?” ja ”kuinka yksityiskohtaista palaute oli?” eivät osoittaneet merkittävää taipumusta yksimielisyyteen opiskelijoiden keskuudessa. Tämä ilmiö voi selittyä paljolti eri töiden sekä eri arvioijien tuottaman palautteen vaihtelevalla laadulla ja myös osittain opiskelijoiden välisillä eroilla. Näiden seikkojen merkitys voisi korostua tai tulla paremmin esille suuremmalla otoksella, mutta pienen otoskoon tapauksessa ei kannata tehdä liian suuria johtopäätöksiä. Vastaukset kysymyksiin siitä, kuinka paljon vertaisarvioinnista saatu palaute auttaa oppimista muun palautteen puuttuessa, kallistuivat hieman siihen suuntaan, että se auttaa enemmän huomattavasti kuin vain vähän. Samaan tapaan toisten kirjoittaman koodin tulkitseminen oli ollut hieman useammin helppoa kuin vaikeaa.

Kysyttäessä opiskelijoiden mielipidettä siitä, kuinka paljon vertaisarviointi kokonaisuudessaan auttoi tehtävän opettavaisuutta verrattuna tehtävään ilman vertaisarviointia, saatiin tulosten keskiarvoksi 2,8 asteikolla 1–5, joka tarkoitti kohtalaista määrää väliltä ”ei ollenkaan” – ”erittäin paljon”. Kuvassa 2 on nähtävissä vastausten jakautuminen asteikolle. Opiskelijoista 50 % piti arvioijana toimimista ja toisten ratkaisujen tarkastelemista hyödyllisempänä kuin palautteen saamista, ja noin 33 % oli sitä mieltä, että kumpikin rooli oli yhtä hyödyllinen. Vastaukset muihin kysymyksiin antoivat lisää viitteitä siitä, että keskimäärin opiskelijat pitivät arvioimista parempana vaiheena näistä kahdesta. Kahdessa vastauksessa myös kerrottiin, että toisten kirjoittamaa koodia tarkastellessa oli löytynyt jokin omasta poikkeava ratkaisutapa, jonka avulla olisi mahdollista parantaa arvioijien omaa ratkaisua.



Kuva 2. Kuinka paljon vertaisarviointi opiskelijoiden mielestä lisäsi tehtävän opettavaisuutta verrattuna tehtäviin, joissa vertaisarviointia ei tehty.

Kenties mielenkiintoisimpia tuloksia olivat vastaukset kysymyksiin, joista opiskelijat olivat erittäin yksimielisiä. Sen lisäksi, että opiskelijat pitivät vertaispalautetta hyödyllisenä lisänä muulle palautteelle, kaikki heistä olisivat myös olleet kiinnostuneita saamaan palautteissa enemmän korjausehdotuksia suoraan omaan ratkaisuunsa. Kaikki olivat myös sitä mieltä, että oppivat paremmin, kun näkevät useampia ratkaisuvaihtoehtoja samaan tehtävään. Opiskelijat myös luottivat toistensa harjoitustyöratkaisujen tarkastelemisen hyödyllisyyteen siinä määrin, että useat heistä pitivät sitä opettavaisempana kuin luennoilla ja harjoituksissa esitettyjä koodiesimerkkejä. Jokainen vastannut opiskelija myös uskoi esimerkiksi vertaisarvioinnin aikana saatavan harjoituksen ja kokemuksen opettavan ymmärtämään toisten kirjoittamaa koodia paremmin.

Tässä tutkimuksessa kukaan opiskelijoista ei pitänyt vertaisarviointia täysin hyödyttömänä. Selkeästi suurin osa oli myös kiinnostunut saamaan entistä rakentavampaa ja laadukkaampaa palautetta, ja vain pieni osa oli tyytyväinen siihen palautteen määrään, jonka yleensä saa. Aikaisemmissa tutkimuksissa todetusta opiskelijoiden suhtautumisesta vertaisarviointiin voidaan mainita esimerkkinä muun muassa Sao Paulon yliopistolla kokeiltu WebCoM-nimisen (Web Course Manager) verkkokurssiohjelmiston avulla toteutettu vertaisarviointijärjestely (Silva & Moreira 2003). Siinä arvioinnit olivat avoimia, ja opiskelijaryhmät esittelivät harjoitustyönsä arviointivaiheen jälkeen opettajalle sekä toisille opiskelijoille. Samalla he saivat tilaisuuden puolustaa omaa ratkaisuaan opponoivan ryhmän kritiikiltä. Menetelmää oli kokeiltu käyttöjärjestelmä- ja olio-ohjelmointikursseilla. Mukana olleista opiskelijoista ylemmällä vuosikurssilla 71 % oli pitänyt vertaisarvioinnista ja alemmalla vuosikurssilla 81 %. Myös tämän

työn opiskelijakyselyssä oli yhdessä vastauksessa mainittu eräänä käytännössä toteutuneena palautteen muotona opettajan ja opiskelijoiden käymä keskustelu opiskelijoiden kurssilla esittämien koodiratkaisujen pohjalta, mutta sen oli kerrottu olevan liian lyhyt ja tarvitsevan opiskelijoiden mielestä enemmän aikaa.

Kyselyssä oli havaittavissa, että toisten ratkaisujen arvioiminen oli opiskelijoiden vastausten perusteella hieman parempi vaihe vertaisarvioinnissa kuin palautteen saaminen. Opiskelijat suhtautuvatkin erittäin varauksellisesti toisiltaan saamaansa palautteeseen. Tästä on myös aikaisemmin saatu samankaltaisia tuloksia. Esimerkiksi Sitthiworachart ja Joy (2004) kertovat Warwickin yliopistolla UNIX-ohjelmointitehtävän yhteydessä toteutetusta verkkopohjaisesta vertaisarviointikokeilusta, jossa oli ollut mukana 215 opiskelijaa. Kokeilussa opettajan antaman arvosanan lisäksi osa opiskelijan arvosanasta oli muodostunut toisten opiskelijoiden antamista arvosanoista, ja myös opiskelijan itse tekemästä arvioinnista oli annettu arvosana. Opiskelijoista 51 % oli pitänyt vertaisilta saamiaan arvioita hyödyllisinä. Loput, eli lähes puolet, eivät pitäneet vertaisarvioita hyödyllisinä, ja he perustelivat tätä muun muassa sillä, etteivät pitäneet itseään eivätkä itsensä kanssa samalla tasolla olevia opiskelijoita pätevinä arvostelijoina. Kuitenkin arvioinnin laadusta annetut arvosanat olivat hyviä, ja tutkimuksen mukaan 95 % opiskelijoista oli ymmärtänyt annetut arviointikriteerit ja -ohjeet.

Opiskelijoiden tekemien arviointien laadusta esimerkiksi verrattuna kokeneen asiantuntijan suorittamaan arvosteluun ei tässä työssä tuotettu juuri näyttöä. Esimerkiksi Wolfe (2004) kertoo kuitenkin opiskelijoiden potentiaalista laadukkaankin palautteen tuottamiseen. Verkkototeutukseen perustuvaa vertaisarviointia oli testattu suuressa mittakaavassa California State University:n tietotekniikan sekä useiden muiden aineiden kuten matematiikan, markkinoinnin ja psykologian kursseilla. Tässä tapauksessa esimerkiksi erään ohjelmistotuotantokurssin kohdalla kurssin jokaisen 34 opiskelijan ratkaisut useisiin tehtäviin olivat kaikkien muiden arvioitavissa, ja tällä tavalla oli saatu aikaan yhteensä yli 5000:n arvion tietokanta. Vertaisarvioinnissa annetuista eri tehtävien pisteytyksistä oli laskettu oppilaskohtaisesti keskiarvot. Opiskelijoiden tällä tavalla lasketuille arvosanoille muodostui jakauma, jossa esiintyneet pienetkin erot opiskelijoiden välillä olivat vastanneet opettajan omaa arvostelua merkillepantavan hyvin.

Kun samaa menettelyä oli käytetty erään matematiikan kurssin yhteydessä, opiskelijoilta kysyttiin jälkeinpäin heidän kokemuksiaan vertaisarviointiprosessista (Wolfe 2003). Arviointiin osallistuneista neljästätoista opiskelijasta 8, eli hieman yli puolet, oli kokenut oppineensa vertaisarvioinnin avulla paljon. Kolmentoista mielestä se oli auttanut heitä ymmärtämään useamman mahdollisen ratkaisuvaihtoehdon olemassaolon. Kovin monet eivät pitäneet varsinaista toisilta opiskelijoilta vastaanottamaansa palautetta erityisen hyödyllisenä, mutta reilusti suurin osa piti sitä kuitenkin kohtuullisen oikeudenmukaisena ja paikkansapitävänä. Tulokset olivat siis olleet jokseenkin samantapaisia kuin tämän työn opiskelijakyselyssä. Opiskelijoiden kokonaiskuva vertaisarvioinnista on enemmän positiivinen kuin negatiivinen. Toisilta opiskelijoilta saatavaa palautetta ei pidetä kovin arvokkaana kuten opettajalta saatavaa pidetään, mutta opiskelijat tuntevat kuitenkin saavansa hyötyä vertaisarvioinnista, ja varsinkin toisten opiskelijoiden ratkaisujen tutkimisessa on heidän vastaustensa perusteella etuja.

Kun opiskelijat kyselyn lopussa kommentoivat omin sanoin työssä toteutettua vertaisarviointia, suurin osa huomioista keskittyi käytetyn järjestelmän ongelmiin ja hankaluuksiin sen käytössä. Palautteet, eli toisten opiskelijoiden tekemät arviot, olisi pitänyt saada kätevämmiin ja nopeammin perille. Varsinaista vertaisarviointia koskeneet kommentit olivat kohtalaisen optimistisia vertaisarvioinnin hyötyjen suhteen, ja myös kehitysehdotuksia oli mietitty esimerkiksi arvioitavien töiden ja arvioijien toisilleen määrittämiseen taitotason perusteella.

4 JOHTOPÄÄTÖKSET

Tämän työn aikana tehdystä tutkimuksesta saatujen tulosten merkitystä rajoittaa hieman se, että otoksen koko oli lopulta melko pieni. Muutamien opiskelijoille esitettyjen kysymysten kohdalla oli joka tapauksessa selvästi havaittavissa vastausten painottumista tiettyyn suuntaan, joten jonkinlaisia päätelmiä voidaan tehdä. Yleensä ottaen opiskelijat kokivat vertaisarvioinnin kohtalaisen hyödyllisenä oppimiselleen. He eivät tavallisesti saa mielestään riittävästi palautetta ohjelmointitehtävistä, mutta uskovat vertaisarvioinnista saatavan palautteen olevan hyvästä muun palautteen lisäksi. Opiskelijat pitivät toistensa ratkaisujen arvioimista hyödyllisempänä kuin toisiltaan saamaansa palautetta, joskin on otettava huomioon, että palautteen saamisessa perille oli toteutuksen aikana ongelmia, joka voi myös vaikuttaa tähän tulokseen.

Kun verrataan saatuja tuloksia alussa esitettyihin mahdollisiin hyötyihin, joita vertaisarvioinnilla voitaisiin saavuttaa, voidaan todeta, että niiden seikkojen kohdalla, joissa opiskelijoiden kyselyvastaukset olivat keskenään eniten yhdenmukaisia, oletetut hyödyt ovat opiskelijoiden näkemysten perusteella toteutuneet hyvin. He kokevat saadun lisäpalautteen hyödyllisenä, he voivat oppia toistensa ratkaisuja tarkastelemalla ja osa heistä olikin löytänyt arvioimastaan työstä ainakin jonkin omastaan poikkeavan ratkaisutavan, jonka avulla uskoisivat pystyvänsä parantamaan omaa ratkaisuaan. Opiskelijat olivat myös usein mieltäneet arvioimansa koodin kaipaavan parannuksia selkeyteen erityisesti rakenteen osalta, ja toisaalta he ovat sitä mieltä, että saamalla tämänkaltaista harjoitusta toisten kirjoittaman koodin tulkitsemisesta he oppisivat siinä paremmaksi. Opettajan näkökulmasta tarkasteltuna vertaisarviointia voidaan käyttää apuna pyrittäessä varmistamaan riittävää palautteensaantia opiskelijoille, kun opettajan resurssit ovat liian rajalliset, mutta kokonaan opettajan antamaa palautetta sillä ei voida korvata.

Tulevaisuuden kannalta tulokset vaikuttavat varovaisen rohkaisevilta. Jatkossa vertaisarviointia voidaan kehittää varsinkin käytännön prosessin osalta, ja vastineeksi voidaan olettaa saatavan oppimisen kannalta positiivisia vaikutuksia. Myös aiheen mahdollinen jatkotutkimus vaikuttaa perustellulta, sillä tässä vaiheessa vertaisarvioinnissa tuntuisi olevan potentiaalisia hyötyjä, mutta kaikkia tekijöitä varsinkaan käytännön toteutuksen ja teknisen puolen osalta tässä työssä ei tutkimusongelman rajauksen mukaisesti edes pyritty kattamaan. Toteutus on jatkossa suunniteltava hyvissä ajoin ja huolellisesti niin, että se toimii varmasti kaikilta osin ja että aikataulussa pystytään pysymään.

Otos jäi työssä pieneksi opiskelijakyselyn vastausten vajaan määrän lisäksi myös osin sen takia, että vaikeudet aikataulujen sovittamisessa yhteen työn ja ohjelmointikurssien välillä aiheuttivat lopulta toisen kurssin jäämisen kokonaan pois vertaisarviointikokeilusta. Tulevaisuudessa olisikin kannattavaa tehdä vastaavanlaista tutkimusta myös siten, että mukana olisi suurempi määrä opiskelijoita ja mahdollisesti useampia kursseja kuten tässäkin työssä alun perin oli tarkoitus. Suuremmalla opiskelijamäärällä voitaisiin mahdollisesti saada enemmän vastauksia kyselyihin ja selvempiä tuloksia toteutetusta kokeilusta. Suuremmalla opiskelijamäärällä voitaisiin myös saada paremmin esille sitä, millainen vaikutus verkkosovelluksen käyttämisellä on vertaisarvioinnin toteuttamiseen. Esimerkiksi arvioitavien töiden määräämisen tietyille arvioijille, arvioiden palautusten seurannan ja muiden vastaavien hallintatehtävien tuki sovelluksen osalta voisi saada suuremman merkityksen, kun käsiteltävänä olisi enemmän töitä, arvioita ja opiskelijoita.

Tässä työssä selvitettiin lähinnä vertaisarvioinnin hyötyjä. Taustatutkimuksen ja arvioinnin toteutuksesta saatujen kokemusten kautta tuli esiin myös tiettyjä tunnistettavissa olevia vahvuuksia ja heikkouksia. Vaikka työ keskittyikin juuri näihin seikkoihin, ja saman aiheen jatkotutkimuksellekin jäi vielä paljon sijaa, sivuttiin työssä myös erästä uutta mahdollisen jatkotutkimuksen kohdetta. Varsinkin sitten, kun on tarpeeksi hyvin saatu selvitettyä minkälainen vertaisarviointi on käytännössä kannattavaa ja tiedetään minkälaisen vaatimusten täyttämistä toteutus edellyttää, vertaisarvioinnin toteutusta siihen tarkoitettun ohjelmiston avulla voitaisiin tutkia ja selvittää sen mahdollisesti mukanaan tuomia hyötyjä.

Tietotekniikka-alan julkaisuista löytyi useita mainintoja tämänkaltaisten opiskelu- ja opetuskäyttöön suunniteltujen järjestelmien kokeiluista, ja niistä oli saatu lupaavia tuloksia. Nyt toteutettuun vertaisarvioinnin testaukseen ei kuitenkaan saatu käyttöön varsinaisesti tähän tarkoitettua sovellusta, vaan käytetyn sovelluksen alkuperäinen käyttötarkoitus oli konferenssijulkaisujen arviointi. Tämä aiheutti selkeästi hankaluuksia. Esimerkiksi vertaisarvioinnin tuottaman palautteen saaminen perille vastaanottajilleen tulisi voida varmistaa helposti ja joustavasti. Ongelmia voitaisiin jatkossa välttää kehittämällä ohjelmistoa nimenomaan harjoitustöiden arviointia varten joko olemassa olevan järjestelmän pohjalta tai kokonaan puhtaalta pöydältä. Ensimmäinen vaihtoehto voisi toimia kokeiluvaiheessa, mutta pitkällä aikavälillä lienee viisaampaa panostaa tekniikkaan, jossa on alusta asti otettu huomioon käyttötarkoitus. Oli tekninen toteutus lopulta minkälainen hyvänsä, on sen toimivuus syytä tutkia huolellisesti ennen käytännön aikataulun asettamista.

5 YHTEENVETO

Vertaisarviointia on yleistymisensä myötä käytetty opetustarkoituksessa jo erittäin monilla aloilla, eikä tietotekniikka ole poikkeus. Tietotekniikan opetuksessa on tunnistettavissa lukuisia opiskelijoiden tuottamia aineistoja, joihin vertaisarviointia on mahdollista soveltaa ja on jossain määrin jo sovellettu. Lappeenrannan teknillisellä yliopistolla tämä on kuitenkin ollut suhteellisen vähäistä. Tässä työssä selvitettiin vertaisarvioinnin hyödyllisyyttä kokeilemalla sitä käytännössä eräällä yliopiston ohjelmointikurssilla. Kurssin Concurrent and Parallel Computing opiskelijat arvioivat toistensa kurssin harjoitustyötä varten tekemiä ohjelmakoodiratkaisuja.

Vertaisarvioinnin toteuttamiseen käytettiin verkkosovellusta, jonka kautta arvioitavat harjoitustyötiedostot ja arviot välitettiin, ja jolla arvioitavat työt määrättiin arvioijille. Kun opiskelijoiden tehtäväksi tulevasta vertaisarvioinnista oli laadittu alustava suunnitelma, selvitettiin sellaisten ohjelmistojen tarjontaa, jolla toteutus kannattaisi tehdä, ja valittiin löytyneistä vaihtoehdoista MyReview-niminen konferenssijulkaisujen arviointia varten kehitetty järjestelmä.

Tavoitteena vertaisarvioinnissa oli saavuttaa hyötyjä sekä opiskelijoiden että opettajan kannalta – opiskelijoiden osalta sekä arvioijan että palautteen saajan roolissa. Tämän tavoitteen toteutumista mitattiin arvioinnin päättymisen jälkeen opiskelijoille suunnatulla kyselyllä, jossa oli huomioitu kumpikin opiskelijan rooli, ja opettajien haastatteluilla, joissa käsiteltiin heidän kokemuksiaan toteutetusta vertaisarvioinnista.

Tutkimuksesta saatujen tulosten perusteella vaikuttaa siltä, että tavoitteena olleita hyötyjä voidaan todella saavuttaa vertaisarvioinnin avulla myös käytännössä ohjelmointitehtävien yhteydessä. Selkeämmän ja luotettavamman kuvan muodostamiseksi asiasta tämänlaiseen kokeiluun tarvittaisiin kuitenkin mukaan opiskelijamäärältään suurempi kurssi tai suositeltavimmin useampia sellaisia. Lisäksi huomattiin, että myös vertaisarvioinnin toteutus on aiheellinen valinta kohteeksi mahdolliselle lisäselvitykselle ja jatkokehitykselle.

LÄHDELUETTELO

Anewalt, K.: Using Peer Review as a Vehicle for Communication Skill Development and Active Learning. *Journal of Computing Sciences in Colleges*, 21(2): 148—155, 2005.

Clark, N., Davies, P., Skeers, R.: Self and Peer Assessment in Software Engineering Projects. *Proceedings of the 7th Australasian conference on Computing education – Volume 42*, Australian Computer Society Inc., 2005, 91—100.

Gehring, Edward F.: Electronic PeerReview and Peer Grading in Computer-Science Courses. *Proceedings of the thirty-second SIGCSE technical symposium on Computer Science Education*, ACM, 2001, 139—143.

Gehring, Edward F., Chinn, Donald D., Pérez-Quiñones, Manuel A., Ardis, Mark A.: Using Peer Review in Teaching Computing. *Proceedings of the 36th SIGCSE technical symposium on Computer science education*, ACM, 2005, 321—322.

Kali, Y., Ronen, M.: Design Principles for Online Peer-Evaluation: Fostering Objectivity. *Proceedings of the 2005 conference on Computer support for collaborative learning: the next 10 years!*, International Society of the Learning Sciences, 2005, 247—251.

Kiviniemi, K.: *Johdatus verkkopedagogiikkaan*. Keski-Pohjanmaan ammattikorkeakoulu, Kokkola, 2000.

Marttunen, M.: Vuorovaikutus sähköpostiopiskelussa yliopisto-opinnoissa. *Verkkopedagogiikka*, Edita, Helsinki, 1997, 112—127.

Papalaskari, M.-A.: Peer Evaluation in an Algorithms Course. *Proceedings of the 8th annual conference on Innovation and technology in computer science education*, ACM, 2003, 274—274.

Rigaux, P. *MyReview: Open-Source Web-Based Paper Submission and Review*. WWW-sivu, URL: <http://myreview.intelligence.eu/> (6.11.2008).

Seitamaa-Hakkarainen, P., Hakkarainen, K.: Verkostopohjainen oppimisympäristö yhteisöllisen suunnittelun tukena. *Opettajatiedon kipinöitä – Kirjoituksia pedagogiikasta*, Joensuun yliopistopaino, Joensuu, 2000, 87—101.

Silva, E., Moreira, D.: WebCoM: a Tool to Use Peer Review to Improve Student Interaction. *Journal on Educational Resources in computing*, 3(1): Article No. 3, 2003.

Sitthiworachart, J., Joy, M.: Effective Peer Assessment for Learning Computer Programming. *ACM SIGCSE Bulletin*, 36(3): 122—126, 2004.

Trytten, Deborah A., A Design for Team Peer Code Review. *ACM SIGCSE Bulletin*, 37(1): 455—459, 2005.

Wolfe, William J.: Student Peer Reviews in an Upper-Division Mathematics Class. *Exchanges, the online journal of teaching and learning in the CSU*, 3.9.2003, WWW-sivu, URL: http://www.calstate.edu/ITL/exchanges/classroom/1156_Wolfe.html (15.12.2008).

Wolfe, William J.: Online Student Peer Reviews. *Proceedings of the 5th conference on Information technology education*, ACM, 2004, 33—37.

Zakon Group. *OpenConf Conference Management Peer-Review Software & Hosting Service*. WWW-sivu, URL: <http://www.openconf.com/> (6.11.2008)

LIITE 1. MyReview-sovelluksen asetuksia

Conference acronym	Lyhenne tapahtuman nimestä. Käytetään esimerkiksi joissain automaattisista viestipohjista. Kurssin nimi.
Conference name	Tapahtuman, tässä tapauksessa kurssin nimi.
URL of the conference	Sovelluksen URL.
Conference mail.	Tapahtuman yleinen sähköpostiosoite, joka näkyy järjestelmästä lähetetyissä viesteissä lähettäjänä.
Chair mail	Tapahtuman ylläpitäjän sähköpostiosoite. Osoite, johon voidaan lähettää automaattisia kopioita viesteistä.
Password generator	Merkkijono, jonka pohjalta salasanat generoidaan.
Uploaded papers directory	Kansio järjestelmään tallennettaville tiedostoille.
Files format	Hyväksyttävien tiedostoformaattien valinta järjestelmään tallennettaville tiedostoille. Constant.php-nimisestä tiedostosta löytyvää taulukkoa nimeltä FILE_TYPES on lisäksi muokattava niin, että taulukosta löytyy vaihtoehtona sopiva formaatti kooditiedostopakettile.
Two phases submission?	Tiivistelmän tallentaminen ennen varsinaisen tiedoston tallentamista järjestelmään. Kun järjestelmään tallennetaan harjoitustyötiedostoja konferenssipapereiden sijasta, tämän on kätevämpi olla pois käytöstä. Tällöin tiedosto tallennetaan tiivistelmälomakkeella.
Extended submission form?	Laajan tai suppean tallennuslomakkeen valinta.
Multi-topics?	Useiden aihealueiden tallennettavalle paperille mahdollistava valinta. Ei tarpeellinen harjoitustyötiedostoja tallennettaessa.
Is abstract submission open?	Tiivistelmäntallennusvaihe käynnissä tai loppunut.
Is submission open?	Varsinaisten tiedostojen tallennus käynnissä tai loppunut.
Blind review	Työn tekijöiden nimien piilottaminen arvioijien näkyvistä. Mahdollisimman rehellisten arvioiden saavuttamiseksi tämän on syytä olla käytössä.
Discussion mode	Työtä koskevien arvioiden näkyvyys kaikille arvioijille tai vain kyseisen työn arvioineille. Oletuksena kokonaan pois päältä.
Ballot mode	Kaikkien konferenssipapereiden tai vain arvioijan aiheeseen liittyvien näkyvyys arvioijille, näiltä suosikkeja ja ristiriitoja kyseltäessä. Tämä ei ole kovin olennainen harjoitustöiden vertaisarvioinnissa.
Nb reviewers per paper	Arvioijien määrä yhtä työtä kohti.
Mail on abstract submission	Lähetetäänkö tapahtuman ylläpitäjälle kopio vastaanottoilmoitusviestistä tiivistelmän tallennuksen yhteydessä.
Mail on paper upload	Lähetetäänkö tapahtuman ylläpitäjälle kopio vastaanottoilmoitusviestistä varsinaisen tiedoston tallennuksen yhteydessä.
Mail on review submission	Lähetetäänkö tapahtuman ylläpitäjälle kopio arviosta, kun sellainen tallennetaan järjestelmään.
Submission deadline	Tiedostojen tallentamisen takaraja. Tämä on näkyvillä tiedoston tallentamiseen liittyvissä yhteyksissä, mutta ei automaattisesti estä tallentamista niitä kyseisen päivämäärän jälkeen.
Review deadline	Arvioiden tallentamisen takaraja. Tämä on näkyvillä arvioinnin tekemiseen liittyvissä yhteyksissä, mutta ei automaattisesti estä tekemästä niitä kyseisen päivämäärän jälkeen.
Camera-ready deadline	Arviointivaiheen jälkeisen korjatun ja lopullisen konferenssipaperin palautuksen takaraja. Harjoitustöiden vertaisarvioinnissa tätä ei välttämättä tarvita.
Date format	Päivämäärän formaatti.

LIITE 2. Opiskelijoille tehty kysely

CT30A7001 CAPC Peer review 2008

The significance of getting feedback on assignments

1) On how many of all programming courses do you get feedback on your solutions to programming assignments?

0-10% 10-20% 20-30% 30-40% 40-50% 50-60% 60-70% 70-80% 80-90% 90-100%

2) In a few words, describe how much feedback on your solutions you usually get in cases where feedback is given.

3) Who gives you feedback, the teacher, your friends...?

4) Do you think that you get enough feedback?

Yes No

5) Do you think that the feedback provided by peer review is a good addition to other feedback you get?

Yes No

6) If yes, how much?

1 very little 2 3 moderately 4 5 helped a lot

7) Is the feedback provided by peer review helpful to you, if you don't get enough feedback otherwise?

Yes No

8) How comprehensive was the feedback you received, compared to how comprehensive you think it should have been in order to be useful?

1 Not comprehensive at all 2 3 Moderately comprehensive 4 5 Very comprehensive

9) How detailed were the comments on things that were dealt with in the feedback that you received?

1 Not detailed at all 2 3 Moderately detailed 4 5 Very detailed

10) Did you receive any suggestions for improvement to your solution?

Yes No

11) Would you have liked to receive more suggestions for improvement?

Yes No

The Significance of Seeing Solutions by Other Students

12) Do you learn better if you see more than one solution to a problem?

Yes No

13) How educational do you think it is to see other students' solutions, compared to seeing code examples that are presented at lessons and exercises?

1 A lot less educational 2 3 Equally Educational 4 5 A lot more educational

14) Did you find out at least one new way to solve some problem while you were reviewing the work of other students?

Yes No

15) Would you be able to make your own solution more efficient by using some part of a solution that was used in a work that you reviewed, and that was different from your own solution?

Yes No

16) How easy was it to understand the code written by other students?

1 Very difficult 2 3 Not difficult nor easy 4 5 Very easy

17) Would the code you reviewed need improvements in the clarity of its structure or commenting?

Structure Commenting Both Neither

18) Did you comment on lack of clarity when you reviewed code that was written by other students?

Yes No

19) Comparing your own code to the code that you reviewed, would your own code need improvements in clarity?

Yes No

20) Did you receive comments on lack of clarity in your own code?

Yes No

21) Do you think that you would learn to understand code written by others better, if you got more practice, for example, by reviewing other students' solutions to programming assignments?

Yes No

Peer Review in General

22) How much did peer review add to the educational value of the assignment, compared to the assignments where peer review was not used?

1 Not at all 2 3 A moderate amount 4 5 Very much

23) How much did your learning benefit from the feedback you received from other students during the peer review process?

1 Not at all 2 3 A moderate amount 4 5 Very much

24) How much did your learning benefit from examining other students' solutions during the peer review process?

1 Not at all 2 3 A moderate amount 4 5 Very much

25) Which of the two things mentioned above was more useful?

The feedback Student solutions Both were equally useful

The Implementation of the Peer Review

26) Was the peer review process easy enough?

Yes No

27) Name two of the most serious problems you had, in case there were such problems.

Feedback on the Peer Review Process in Your Own Words

28) Please give some feedback in your own words. For example, what should peer review be like in order for it to be beneficial to your learning? You can also explain or comment on your answers to the questions above.

Jukka (B.Sc. worker), Jari and Harri thank you for your answers