Tietojenkäsittelytiede **33** Joulukuu 2011 sivut 5–14
Toimittaja: Hannu-Matti Järvinen

Rubyric — yksilöllistä sanallista palautetta massaopetuksessa

Tapio Auvinen
Aalto-yliopisto
Tietotekniikan laitos
tapio.auvinen@aalto.fi

Tiivistelmä

Jotta opiskelijat oppisivat harjoitustöistä mahdollisimman paljon, heidän täytyy saada niistä yksityiskohtaista palautetta. Pelkkä arvosana ei kerro opiskelijalle, mikä työssä oli hyvää tai miten sitä olisi voinut parantaa. Suurilla kursseilla rajalliset opetusresurssit vaikeuttavat yksilöllisen palautteen antamista. Arviointi voidaan jakaa usean tuntiassistentin vastuulle, mutta tällöin arvioinnin yhtenäisyyden varmistaminen vaikeutuu. Rubriikit ovat arviointityökalu, jossa määritellään harjoitustyön arviointikriteerit sekä palautefraaseja eri tasoisille töille. Arviointiprosessi nopeutuu, kun tyypillisistä virheistä voidaan antaa palautetta uudelleenkäytettävien fraasien avulla. Tarkasti määritellyt arviointikriteerit myös yhtenäistävät arviointia eri arvioijien välillä. Tässä artikkelissa luodaan katsaus rubriikkeja käsittelevään kirjallisuuteen ja esitellään Rubyric-arviointityökalu, joka mahdollistaa rubriikkien luomisen ja harjoitustöiden arvostelemisen verkossa.

1 Johdanto

On tärkeää, että opiskelijat saavat harjoitustöistään rakentavaa palautetta. Pelkkä numeerinen arvosana ei kerro opiskelijalle, mikä vastauksessa oli hyvää, ja mitä voisi vielä parantaa. Jos arvosana on hyvä, opiskelija voi pitää tarpeettomana paneutua aihepiiriin enempää, ja jos arvosana on huono, se voi vahvistaa opiskelijan uskomusta, että hän ei pysty oppimaan aihetta [9]. Yksilöllisen ohjaavan palautteen tuottaminen voi kuitenkin olla suurilla kursseilla liian työlästä käytettävissä olevilla opetusresursseilla. Automaattista arviointia voidaan käyttää joidenkin harjoitustehtävätyyppien, esimerkiksi pienten ohjelmointitehtävien, tarkastamiseen [4]. Se ei kuitenkaan sovellu muun muassa esseevastauksiin tai laajoihin harjoitustöihin, joissa on jätetty tilaa luoville ratkaisuille. Suurilla kursseilla arvioijien työmäärää voidaan vähentää palkkaamalla useita tuntiassistentteja. Tällöin haasteeksi muodostuu arvioinnin yhtenäisyyden ja tasapuolisuuden varmistaminen.

Rubriikit ovat arvostelusapluunoita, jotka sisältävät arvostelukriteerit sekä onnistumista kuvaavia lauseita [5]. Niillä voidaan nopeuttaa arviointiprosessia ja parantaa arvioinnin yhtenäisyyttä. Rubriikit kuvataan usein kaksiulotteisina matriiseina, joissa rivit kuvaavat arvostelukriteereitä, sarakkeet arvosanatasoja ja solut sisältävät palautefraaseja. Taulukossa 1 on esimerkki perinteisestä kaksiulot-

OHJE KIRJAPAINOLLE: B5-arkin vasen- ja yläreuna kohdistetaan A4-arkin vasempaan ja yläreunaan. Näin pitäisi marginaaliksi tulla taitteen puolella noin 33 mm ja muualla noin 22 mm.

Taulukko 1: Lyhyt esimerkkirubriikki ohjelmointiprojektin arvosteluun

	Heikko	Kelvollinen	Hyvä
Ohjelmointityyli	Et ole noudattanut tyyliohjeita.	Koodi on melko siistiä, mutta paikoin poikkeaa tyyliohjeista.	Koodi on kauttaaltaan siistiä ja helppolukuista.
Testit	Projektissa ei ole käytetty yksikkötestejä lainkaan.	Projektissa on käytetty jonkin verran yksikkötestejä.	Yksikkötestit ovat hyvin kattavat.
Arkkitehtuuri	Ohjelma kannattaisi jakaa selkeämmin itsenäisiin osiin.	Luokkien väillä on jonkin verran tarpeettomia riippuvuuksia.	Ohjelman rakenne on hyvin modulaarinen.

teisesta rubriikista. Palautteen antaminen on rubriikilla vaivatonta, koska arvostelijan tarvitsee vain ympyröidä oikeat solut, ja opiskelija saa melko yksityiskohtaista tietoa eri osa-alueiden hyvyydestä.

Tässä artikkelissä esitellään Aalto-yliopistossa tehty rubriikkeihin perustuva arviointityökalu Rubyric ja perustellaan rubriikkien käyttökelpoisuutta harjoitustöiden arvioinnissa. Kohdassa 2 luodaan lyhyt katsaus arviointia ja palautteen antamista koskevaan kirjallisuuteen. Kohdassa 3 käsitellään rubriikkien käyttöä opetuksessa. Kohdassa 4 esitellään lyhyesti rubriikkipohjaisia arvostelutyökaluja ja kohdassa 5 esitellään Rubyric-arvostelutyökalu. Lopuksi kohdassa 6 kootaan yhteenveto.

2 Arvioinnin muotoja

Arviointi voidaan jakaa summatiiviseen eli kokoavaan arviointiin ja formatiiviseen eli ohjaavaan arviointiin. Termejä ehdotti ensimmäisenä Scriven [17], jonka mukaan formatiivinen arviointi on sellaista, jossa arvioinnin tulokset vaikuttavat opetukseen, ts. prosessissa on takaisinkytkentä. Summatiivisessa arvioinnissa taas tavoitteena on arvioida opiskelijan nykyistä osaamisen tasoa esimerkiksi oppimistavoitteisiin tai muihin opiskelijoihin nähden pyrkimättä enää parantamaan sitä. Jos esimerkiksi opettaja havaitsee pistokokeen avulla, että suuri osa opiskelijoista on ymmärtänyt jonkin asian väärin ja käyttää aikaa väärinkäsitysten korjaamiseen, arviointi on formatiivista, kun taas kurssin lopussa oleva tentti on selvästi summatiivista arviointia. Nykyisin formatiivisella arvioinnilla tarkoitetaan usein arviointia, jonka tarkoituksena on ohjata opiskelijaa kohti parempia oppimistuloksia eikä vain arvioida nykyistä osaamisen tasoa, ts. opiskelija saa työstään palautetta kurssin aikana [19]. Black ja Wiliam [10] ovat luoneet laajan katsauksen formatiivista arviointia käsittelevään kirjallisuuteen, ja tulokset osoittavat, että formatiivinen arviointi parantaa oppimistuloksia johdonmukaisesti eri harjoitustyypeissä, oppiaineissa ja koulutustasoilla. Tästä huolimatta korkeakouluopetuksessa on yhä tavallista painottaa summatiivista arviointia [12]. Yorken mukaan keskeinen syy tähän on opetusresurssien rajallisuus: ohjaavan palautteen antaminen vaatii työtä ja vie aikaa [19].

Korkeakouluopetuksessa sekä summatiivisella että formatiivisella arvioinnilla on paikkansa. Opiskelijoille täytyy yleensä antaa kurssin lopuksi numeerinen arvosana todistusta varten. Toisaalta kurssien varsinaisena tarkoituksena on lisätä opiskelijoiden tietomäärää ja osaamista. Rubriikit tukevat molempia arvioinnin muotoja. Summatiivista arviointia varten ne tarjoavat selkeät arviointikriteerit, joiden avulla töitä voidaan arvioida mahdollisim-

man objektiivisesti. Formatiivista arviointia varten taas rubriikit tarjoavat vaivattoman tavan tuottaa yksityiskohtaista palautetta, joka ohjaa opiskelijaa harjoittelemaan lisää tarvittavia osa-alueita.

Summatiivinen ja formatiivinen arviointi eivät sulje toisiaan pois, vaan niitä voidaan käyttää samanaikaisesti. Esimerkiksi Aalto-yliopiston Tietorakenteet ja algoritmit -kurssilla on käytetty projektityön ohjaamisessa mallia, jossa opiskelijat palauttavat harjoitustyöstä ensin kaksi versiota, joista kurssihenkilökunta antaa palautetta, ja lopuksi kolmas versio arvostellaan [8]. Kahdessa ensimmäisessä palautuksessa palautteen antoon käytetään rubriikkia, joka sisältää vihjeitä työn parantamiseen, mutta ei anna lainkaan arvosanaa. Viimeinen palautus taas arvioidaan rubriikilla, jonka sisältö painottuu arvosanan perusteluun.

3 Rubriikit

Rubriikit voivat tukea arvosteluprosessia yhtenäistämällä arviointia ja nopeuttamalla sitä. Kun kurssilla on useita arvioijia, on vaikea varmistaa, että eri arvioijat painottavat töissä eri arvostelukriteerejä samalla tavalla [11]. Arvioinnin yhtenäisyys voi myös olla ongelma, vaikka arvioijia olisi ainoastaan yksi. Kontrastivaikutuksen takia keskitason työ voi vaikuttaa huonolta, jos se arvioidaan muutaman erityisen hyvän työn jälkeen, ja vastaavasti hyvältä, jos se arvioidaan muutaman huonon työn jälkeen. Myös arvioijan mieliala tai väsymys voi vaikuttaa arvosanoihin.

Ahoniemi ja muut ovat tutkineet arvioinnin yhtenäisyyttä käytettäessä tietokoneavusteisia rubriikkeja [2]. Tutkimuksessa perusohjelmointikurssin projektityö arvioitiin yhtenä lukuvuotena paperirubriikilla ja toisena lukuvuotena tietokoneavusteisella ALOHA-rubriikkityökalulla. Tehtävänanto, rubriikin sisältö ja arvosteluasteikko pidettiin molemmilla kursseilla samana. Opiskelijoiden palautukset jaettiin satunnaisesti yhdeksälle tuntiassistentille, jotka olivat korkeamman vuosikurssin perusopiskelijoita, eli melko kokemattomia arvioijia. Paperirubriikkeja käytettäessä assistenttien arvosanajakaumissa oli tilastollisesti merkitseviä eroja, eli arvioinnin objektiivisuus oli kyseenalainen. Tietokoneavusteisia rubriikkeja käytettäessä arvosanajakaumien välillä ei ollut tilastollisesti merkitseviä eroja. Kirjoittajat esittävät, että parempi objetiivisuus tietokoneavusteisia rubriikkeja käytettäessä johtuu siitä, että työkalu pakotti arvioijat käymään arvostelukriteerit yksitellen läpi ennen lopullisen arvosanan antamista. Paperirubriikkeja käytettäessä arvostelija saattoi muodostaa työn arvosanasta mielipiteen jo ennen kuin kävi kriteerejä tarkemmin läpi. On huomattava, että assistentit käyttivät paperirubriikkeja ainoastaan arvosteluohjeena, mutta eivät tehneet niihin merkintöjä tai palauttaneet niitä opiskelijoille. Luultavasti vastaava parannus arvostelun objektiivisuudessa saataisiin aikaiseksi myös paperirubriikeilla, jos arvioijien olisi pakko ympyröidä rubriikista valitut solut ja muodostaa lopullinen arvosana näiden valintojen perusteella. Tutkimuksen perusteella vaikuttaa siltä, että tarkasti määritellyt arviointiohjeet eivät takaa arvostelun yhtenäisyyttä, jos arvioijan ei tarvitse erikseen perustella arvosanaa kriteerien avulla. Tietokoneavusteisen rubriikkityökalun etuna on, että ohjelmistolla voidaan pakottaa arvioijan käymään kriteerit kohta kohdalta läpi ennen kuin lopullisen arvosanan antaminen on mahdollista.

Anglin ja muut vertailivat harjoitustöiden arviointiin kuluvaa aikaa käytettäessä perinteistä punakynää, paperirubriikkeja, tietokoneella ilman apuvälineitä kirjoitettua palautetta ja tietokoneavusteisia ru-

briikkeja [6]. Tutkimuksessa 73 opiskelijaa liiketoiminnan kurssilla palautti kolme esseetä, ja palautukset jaettiin satunnaisesti neljään eri menetelmillä arvioitavaan ryhmään. Yksi arvostelija arvosteli kaikki työt, ja kulunut aika mitattiin. Tietokoneavusteiseen rubriikkiin verrattuna punakynällä kirjoitettu palaute kulutti kaksi kertaa enemmän aikaa, paperirubriikit kolme kertaa enemmän, ja tietokoneella ilman apuvälineitä kirjoitettu palaute 3.5 kertaa enemmän aikaa. Arviointiin kuluneeseen aikaan laskettiin mukaan rubriikin valmisteluun ja tulostukseen kulunut aika sekä lopuksi arvosanojen tietokoneelle syöttämiseen kulunut aika. Kirjoittajat mittasivat myös kyselyllä opiskelijoiden tyytyväisyyttä palautteen laatuun. Tuloksissa ei havaittu tilastollisesti merkitseviä eroja siinä, kuinka hyödyllisenä opiskelijat pitivät eri menetelmillä tuotettuja palautteita. Tietokoneavusteiset rubriikit tuottivat siis likimain yhtä hyvää palautetta nopeammin. Ei ole yllättävää, että palautteen tuottaminen on rubriikeilla nopeampaa kuin jos palaute kirjoitetaan kokonaan manuaalisesti. Jos samaa rubriikkia on mahdollista käyttää uudelleen seuraavan vuoden kurssilla, etu vielä korostuu.

Ahoniemi ja Karavirta [1] tutkivat, millä tavalla arvioijat käyttävät rubriikkeja ALOHA-arviointityökalussa. ALOHAssa rubriikkia käytetään palautteen rungon koostamiseen, mutta palautetekstiä pystyy muokkaamaan käsin ja siihen voi lisätä ylimääräisiä kommentteja asioista, joita rubriikki ei kata. Tutkimuksessa yhdeksän tuntiassistenttia arvioi kukin 16-18 laajaa ohjelmointiprojektia, ja valmiista palautteksteistä mitattiin rubriikista poimittujen ja käsin kirjoitettujen palautefraasien osuus. Keskimäärin 50% palautteista koostui valmiista palautefraaseista ja 50% oli käsin kirjoitettua yksilöllistä palautetta. Assistenttien välillä oli melko suuria eroja. Osa kirjoitti jopa 70% palautetekstistä itse, kun taas osalla 70% palautteesta koostui valmiista fraaseista. Kukaan ei tyytynyt käyttämään valmiita fraaseja sellaisenaan. Tulosten perusteella rubriikkien käyttö ei näytä johtavan täysin yksilöimättömään palautteeseen, joka koostuisi ainoastaan kierrätetyistä fraaseista.

Rubriikeista voi olla apua myös opetuksen kehittämisessä. Popham [15] huomauttaa, että kun opettaja joutuu erittelemään yksityiskohtaisesti, mitä hyvältä työltä odotetaan, hän voi samalla pohtia, opetetaanko kurssilla todella kaikki vaaditut taidot. Kun rubriikki on tehty, sillä voidaan myös vaivattomasti viestittää arviointiperusteet opiskelijoille.

4 Rubriikkityökaluja

ALOHA on verkossa toimiva arviointityökalu, joka on kehitetty Tampereen teknillisessä yliopistossa [3]. ALOHAlla opiskelijoiden harjoitustyöt voidaan jakaa arvioitavaksi usealle arvioijalle. Arvioijat voivat lukea palautuksia verkon kautta, ja tuottaa palautetta sekä arvostella töitä rubriikkien avulla. Arvioijat valitsevat opettajan laatimasta rubriikista palautelauseita palautteen pohjaksi, mutta voivat muokata fraaseja käsin ja lisätä ylimääräisiä kommentteja asioista, joita rubriikki ei kata. Palautteet lähetetään opiskelijoille sähköpostitse. ALOHAssa ei kuitenkaan ole työkaluja rubriikin rakentamiseen, vaan se on syötettävä järjestelmään XML-tiedostona. Rubriikkia ei myöskään ole mahdollista muokata sen jälkeen, kun arvostelu on alkanut, ja lisäksi arvosana-asteikkojen määritteleminen vaatii ohjelmointia.

Agar [18] yhdistää samaan järjestelmään ohjelmointitehtävien automaattisen arvioinnin ja manuaalisen palautteen annotoimalla, eli tekemällä merkintöjä pa-

lautukseen. Opettaja luo järjestelmällä rubriikin, jossa määritellään arvostelukriteerit, mutta ei palautefraaseja. Ohjelmointitehtävissä arvostelukriteerit voivat liittyä automaattisiin testeihin, jolloin kriteerin täyttyminen määräytyy siihen liittyvien läpäistyjen testien määrästä. Automaattisen arvioinnnin jälkeen arvioija voi kirjoittaa manuaalisesti palautetta opiskelijalle. Jokaisen kommentin voi liittää arviointikriteeriin, jolloin arviointi vastaa perinteistä punakynällä tapahtuvaa tarkastamista, jossa arvioija merkitsee kohdat, joista opiskelija saa tai menettää pisteitä. Kommentit tallennetaan, jolloin toistuvia palautteita ei tarvitse kirjoittaa aina uudelleen. Palautefraaseja ei kuitenkaan laadita ennalta arvostelukriteerien yhteyteen samalla tavalla kuin perinteisissä paperirubriikeissa. Agar ei ole verkkotyökalu, vaan se on erikseen asennettava arvioijan tietokoneelle. Tästä voi olla haittaa kursseilla, jolla arvioijina on paljon vuosittain vaihtuvia tuntiassistentteja. Opiskelijat kuitenkin palauttavat harjoitustyönsä verkon kautta ja saavat palautteet sähköpostitse.

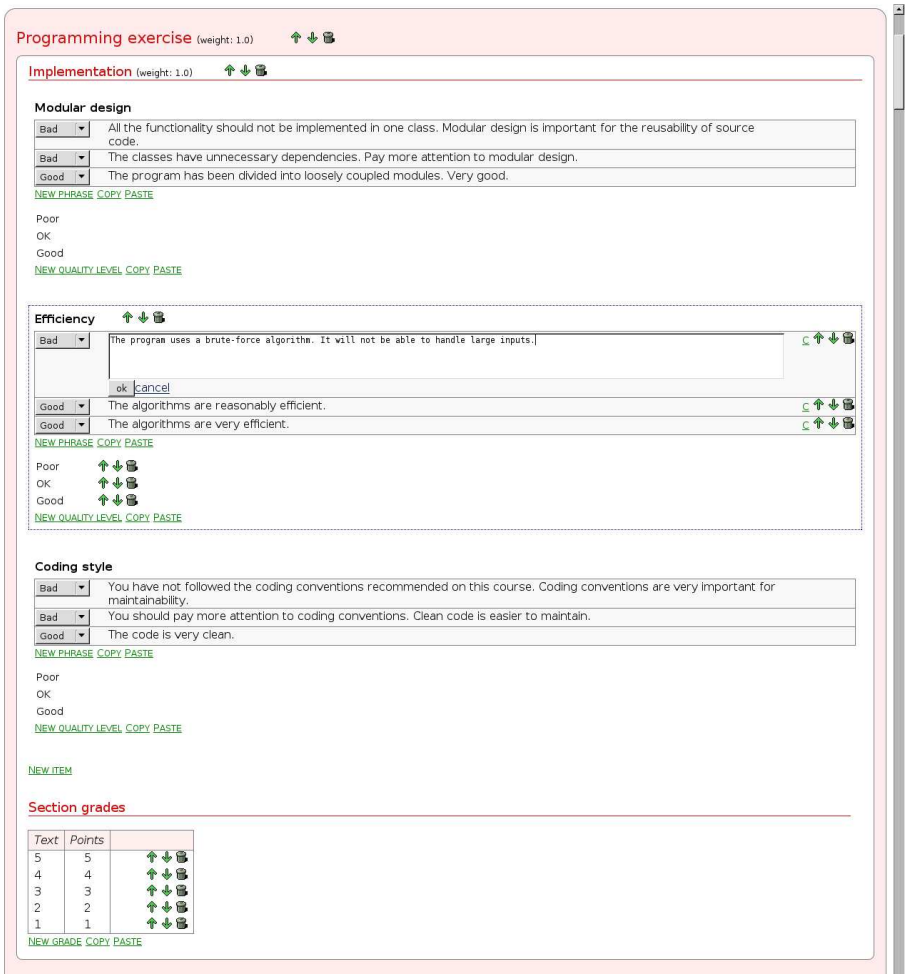
Edmison ja muut [13] ovat luoneet rubriikkipohjaista arviointia varten lisäosan Moodle-kurssinhallintajärjestelmään[1]. Työkalulla on mahdollista luoda rubriikkeja ja lisätä palautetta opiskelijoiden palautuksiin annotoimalla. Tällöin opiskelija ei vastaanota palautetta erillisenä tekstinä, vaan merkintöinä alkuperäisessä työssä. Kirjoittajat perustelevat annotoinnin etua sillä, että jos palauteteksti on erillään alkuperäisestä palautuksesta, se jää helposti liian yleisluontoiseksi. Kun palaute on liitetty työn yksityiskohtiin, opiskelija näkee konkreettisesti, mitkä osat työssä vaativat vielä huomiota. Työkalussa rubriikkien fraaseja ei voi muokata kutakin yksilöllistä palautusta varten, vaan niitä käytetään sellaisenaan. Rubriikkia ei siis käytetä niinkään palautteen pohjana, vaan arvosanan perusteluun. Lisäkommentteja on mahdollista kirjoittaa manuaalisesti, mutta usein toistuvia palautteita ei voi tallettaa uudelleenkäytettäväksi. Järjestelmää ei voi käyttää Moodlesta erillään, joten se ei sovi hyvin kursseille, joilla on käytössä jokin muu kurssinhallintajärjestelmä.

Rubriikkeja käytetään myös joissakin vertaisarviointiin suunnatuissa järjestelmissä arviointiohjeina. Ne sopivat vertaisarviointiin hyvin antamalla kokemattomalle arvioijalle yksityiskohtaiset arviointikriteerit, ja auttavat näin parantamaan arvioinnin yhtenäisyyttä. Aropä [14] on verkkotyökalu, johon opiskelijat palauttavat harjoitustyönsä ja arvioivat toistensa töitä rubriikkien avulla. Rubriikin fraaseja ei ole mahdollista muokata jokaista yksittäistä työtä varten. Järjestelmä painottaa siis arvioinnin yhtenäisyyttä palautteen yksilöllisyyden kustannuksella, joka lienee vertaisarvioinnin tapauksessa hyvin perusteltua. Powell ja muut [16] ovat luoneet Moodle-kurssinhallintajärjestelmään lisäosan vertaisarviointia varten. Myös siinä rubriikkia käytetään arviointisapluunana eikä palautteen pohjana. Palautuksiin on mahdollista kirjoittaa merkintöjä ja palautetta manuaalisesti, mutta usein toistuvien fraasien kierrättämiseen ei ole työkaluja.

5 Rubyric

Rubyric on rubriikkeihin perustuva arviointityökalu, joka on toteutettu Teknillisessä korkeakoulussa diplomityönä vuonna 2008 [7]. Järjestelmää on käytetty sittemmin Aalto-yliopistossa yli 20 kurssilla. Järjestelmä ottaa vastaan opiskelijoiden palautukset, ne voidaan jakaa arvioitavaksi assitenteille, ja arvioijat voivat lu-

[1] http://moodle.org/



Kuva 1: Rubriikin laatimisnäkymä

kea töitä verkon kautta ja antaa palautetta rubriikkien avulla. Järjestelmä sisältää lisäksi työkalun rubriikkien laatimiseen. Rubyric on kehitetty ALOHA-arviointityökalun [3] pohjalta sen seuraajaksi.

Rubyric ei ole kurssinhallintajärjestelmä, vaan tehtävänannnot jaetaan opiskelijoille kurssin kotisivulla tai jossakin kurssinhallintajärjestelmässä. Tehtävänannon yhteydessä opiskelijoille jaetaan linkki, jonka kautta tehtävä palautetaan Rubyric-järjestelmään. Jos kyseessä on ryhmätyö, opiskelijaa pyydetään ensiksi syöttämään ryhmän jäsenten opiskelijanumerot ja sähköpostiosoitteet. Ryhmän luonnin jälkeen opiskelija syöttää palautettavan tiedoston, joka voi olla esimerkiksi pdf-dokumentti tai ohjelmointiprojektin sisältävä zip-arkisto. Palaute lähetetään opiskelijoille myöhemmin sähköpostitse. Järjestelmän suunnittelussa on pyritty pitämään oppimiskynnys ja käyttäjän kuormitus opiskelijan näkökulmasta mahdollisimman matalana hyödyntämällä opiskelijoiden muutenkin käyttämiä välineitä kuten sähköpostia.

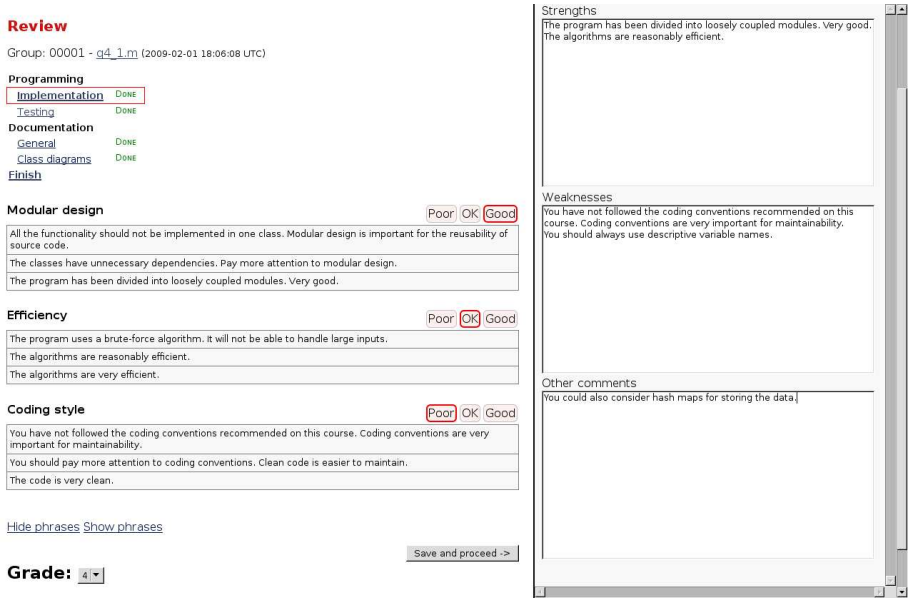
Opettaja voi luoda rubriikin selainkäyttöisellä työkalulla (kuva 1), mutta sitä on mahdollista käsitellä myös erillisenä XML-tiedostona. Rubriikkia ei kuvata perinteisenä kaksiulotteisena matriisina, vaan arvostelukriteerit esitetään otsikkoina ja kuhunkin kriteeriin voi liittyä vaihteleva määrä palautefraaseja. Tällöin järjestelmä tukee hyvin formatiivista eli ohjaavaa palautteenantoa, kun kuhunkin arvostelukriteeriin liittyviä tyypillisiä virheitä varten voidaan koota valmiita palautteita. Fraasien ei tarvitse muodostaa arvosteluasteikkoa, vaan ne voivat olla vaikkapa kokonaisia tekstikappaleita, joissa selvennetään opiskelijan väärin ymmärtämää oppisisältöä. Fraasit voidaan myös jättää osittain arvioijan täytettäviksi. Esimerkiksi ohjelmointiprojektissa voisi olla palautefraasi *Harkitse luokan [...] jakamista kahdeksi*. Tällöin opiskelija saa yksilöllistä palautetta työn yksityiskohdista, vaikka palaute perustuu suurilta osin kierrätettäviin fraaseihin.

Arvostelun yhtenäisyyden varmistamiseksi jokaiseen arvostelukriteeriin voidaan liittää myös arvosteluasteikko, jolta arvioijan on valittava täsmälleen yksi vaihtoehto. Järjestelmä edellyttää, että jokaisen kriteerin arvosana on valittu ennen kuin arvostelun voi päättää. Näin voidaan varmistaa, että jokainen arvioija käy kaikki arvostelukriteerit yksitellen läpi.

Rubriikki on mahdollista jakaa alasivuihin, joilla on eri painokertoimet lopullisen arvosanan laskennassa. Esimerkiksi ohjelmointiprojektin arvioinnnin voisi jakaa toteutukseen ja projektiraporttiin, jotka arvioija arvioi erikseen, mutta joiden painotetusta keskiarvosta muodostuu harjoitustyön lopullinen arvosana.

Kun opiskelijoiden palautukset on vastaanotettu, ne voidaan helposti jakaa arvioitavaksi usean arvioijan kesken. Arviojat pääsevät verkon kautta lukemaan itselleen määrättyjä töitä ja antamaan palautetta opettajan luoman rubriikin avulla. Arvostelunäkymä esitetään kuvassa 2. Arvostelukriteerit näkyvät vasemmalla otsikkoina, ja palautefraasit ovat otsikon alla listana. Fraasin klikkaaminen kopioi sen johonkin oikealla näkyvistä tekstialueista, joissa palautetekstiä saa muokata vapaasti. Valmiita fraaseja voi näin muokata paremmin vastaamaan kutakin yksilöllistä työtä, ja palautteeseen on mahdollista kirjoittaa lisäkommentteja aiheista, joita rubriikki ei kata.

Palauteteksti jaetaan kolmeen tekstialueeseen, jotka opettaja voi otsikoida esimerkiksi *hyvää*, *kehitettävää* ja *muita kommentteja*. Tällöin lopullisessa palautepostissa on mahdollista ryhmitellä samaan kategoriaan kuuluvat kommentit jokaisel-



Kuva 2: Arvostelunäkymä

ta alasivuilta yhteen.

Arvostelukriteerin vieressä näkyy arvosteluskaala, jolta arvioijan on valittava yksi arvosana jokaiselle kriteerille ennen kuin arvostelun voi päättää. Tällä varmistetaan, että arvioijat käyvät kaikki kriteerit läpi. Sivun lopussa arvioija valitsee kokonaisuudelle arvosanan. Yksittäisiin kriteereihin liittyvät arvosanat ohjaavat arvioijaa, mutta lopullisen arvosanan valinnassa arvioija saa käyttää harkintaa. Arvioija voi esimerkiksi katsoa, että erityiset ansiot jossakin osa-alueessa oikeuttavat hyvään arvosanaan, vaikka jostakin yksittäisestä arvostelukriteeristä olisi tullut huono arvosana.

Kun kaikki alasivut on arvioitu, palautetekstit kaikilta alasivuilta kootaan yhteen, ja tekstiä on vielä mahdollista muokata kokonaisuutena. Valmis palaute arkistoituu järjestelmään. Kun kaikki työt on arvioitu, palautteet lähetetään sähköpostitse opiskelijoille. Arvosanat on mahdollista siirtää taulukkolaskentaohjelmaan, jossa Rubyricillä arvostellun harjoitustyön tulokset voidaan yhdistää vaikkapa tenttitulosten kanssa. Järjestelmä tuottaa myös kuvaajat eri arvioijien arvosanajakaumista, jotta mahdolliset erot eri arvioijien arvosanajakaumissa on helppo havaita.

5.1 Tekninen toteutus

Rubyric on verkkotyökalu ja se on toteutettu Ruby on Rails[2]-ohjelmistokehyksellä. Verkkotyökalun etuna on, että käyttäjien ei tarvitse asentaa erityisiä ohjelmistoja omalle tietokoneelleen. Tästä on etua erityisesti kursseilla, joilla on arvioijina useita tuntiassistentteja, jotka vaihtuvat vuosittain. Järjestelmä hoitaa töiden vastaanottamisen, jakamisen arvioijille ja palautteiden jakelun takaisin opiskelijoille vaivattomasti. Sen myös pitää palautukset,

[2] http://rubyonrails.org/

palautteet ja arvosanat keskitetysti tallessa sen sijaan, että ne olisivat hajallaan useiden ihmisten sähköpostilaatikoissa.

Rubyric on avoimen lähdekoodin ohjelmisto, ja sen voi ladata osoitteesta *http://rubyric.cs.hut.fi/*.

5.2 Tulevia kehityssuuntia

Rubyric rajoittuu nykymuodossaan tekstimuotoiseen palautteeseen, joka lähetetään opiskelijalle sähköpostiviestinä. Kuten Edmison ja muut [13] toteavat, palaute olisi hyödyllistä näyttää alkuperäiseen palautukseen liitettynä, jotta siinä olisi helppo viitata työn yksityiskohtiin. Rubyriciin onkin suunnitteilla annotointiominaisuus.

Rubyric on suunniteltu opetushenkilökunnan apuvälineeksi, jonka pitäisi kuormittaa opiskelijoita mahdollisimman vähän. Ihanteellisesti opiskelijoiden ei tarvitsisi käyttää järjestelmää edes harjoitustöiden palauttamiseen, vaan he voisivat palauttaa työt ja vastaanottaa palautteet jossakin kurssinhallintajärjestelmässä, jota käytetään myös kurssin oppimateriaalien jakamiseen. Rubyricin sovittaminen toimimaan yhdessä muiden oppimisalustojen kanssa onkin tuleva kehittämiskohde. Ihanteellisessa tapauksessa opettaja voisi koota kurssilleen eri osa-alueita parhaiten tukevia työkaluja, jotka toimisivat saumattomasti yhteen. Yksi työkalu voisi esimerkiksi hoitaa ohjelmointitehtävien automaattisen tarkastamisen, toinen plagioinnin tunnistamisen ja kolmas manuaalisen palautteen antamisen. Tämä edellyttää järjestelmien välisten rajapintojen ja yhteentoimivuuden kehittämistä nykyisestä.

Rubriikkien käytöstä olisi myös mahdollista kerätä hyödyllistä tietoa opetuksen kehittämistä varten. Esimerkiksi tilastot useimmin käytetyistä palautefraaseista kertovat opiskelijoiden yleisimmistä virheistä, joita voisi olla syytä käsitellä enemmän myöhemmillä luennoilla. Rubyric kerää jo tilastoja joistakin muuttujista, mutta tietojen saattaminen opettajien käytöön esimerkiksi visualisointien avulla vaatii lisätutkimusta.

6 Yhteenveto

Harjoitustöistä kannattaa aina antaa muutakin palautetta kuin pelkkä arvosana, mikäli tavoitteena on opiskelijan osaamisen kehittäminen eikä vain mittaaminen. Eräs hyvä käytäntö on projektityön jakaminen useaan välipalautukseen, joista annetaan ohjaavaa palautetta. Palautetta ei kannata jättää antamatta siksi että se on työlästä, sillä tietotekniset apuvälineet mahdollistavat laajankin sanallisen palautteen tuottamisen vähällä vaivalla. Palautesapluunan valmistelu vaatii edelleen jonkin verran työtä, mutta työ maksaa itsensä takaisin suurilla kursseilla tai jos kurssi järjestetään samassa muodossa useampana lukuvuotena.

Rubriikit ovat käyttökelpoisia arvioinnin apuvälineitä. Ne nopeuttavat arviointia, sillä osia palautteesta voidaan rakentaa uudelleenkäytettävistä fraaseista. Ne myös yhtenäistävät arviointia sekä yhden että usean arvioijan tapauksessa, kun arvioijat joutuvat antamaan arvosanan tarkasti määriteltyjen arviointikriteerien perusteella.

Tietokoneavusteiset rubriikit tarjoavat useita etuja paperirubriikkeihin verrattuna. Verkkotyökalun avulla opiskelijoiden palautukset ja arvioijien palautteet voidaan vaivattomasti jakaa oikeille vastaanottajille suurillakin kursseilla, joilla on satoja opiskelijoita ja useita arvioijia. Lisäksi palautukset, palautteet ja arvosanat pysyvät keskitetysti tallessa. Tietokoneen avulla on myös mahdollista rakentaa palautetta uudelleen käytettävistä osista, mutta silti säilyttää palautteen vapaa muokattavuus kutakin yksilöllistä työtä varten.

Viitteet

1. T. Ahoniemi ja V. Karavirta. Analyzing the use of a rubric-based grading tool. Teoksessa *ACM SIGCSE Bulletin*, nide 41, ss. 333–337. ACM, 2009.
2. T. Ahoniemi, E. Lahtinen ja T. Reinikainen. Improving pedagogical feedback and objective grading. Teoksessa *ACM SIGCSE Bulletin*, nide 40, ss. 72–76. ACM, 2008.
3. T. Ahoniemi ja T. Reinikainen. Aloha-a grading tool for semi-automatic assessment of mass programming courses. Teoksessa *Proceedings of the 6th Baltic Sea conference on Computing education research: Koli Calling 2006*, ss. 139–140. ACM, 2006.
4. K. Ala-Mutka. A survey of automated assessment approaches for programming assignments. *Computer Science Education*, 15(2):83–102, 2005.
5. H. Andrade. Using Rubrics to Promote Thinking and Learning. *Educational Leadership*, 57(5):13–19, 2000.
6. L. Anglin, K. Anglin, P. Schumann ja J. Kaliski. Improving the Efficiency and Effectiveness of Grading Through the Use of Computer-Assisted Grading Rubrics. *Decision Sciences Journal of Innovative Education*, 6(1):51–73, 2008.
7. T. Auvinen. Rubyric - A rubrics-based online assessment tool for effortless authoring of personalized feedback. Diplomityö, Teknillinen korkeakoulu, 2009.
8. T. Auvinen, L. Hakulinen ja A. Korhonen. Tackling the challenges of a large course with blended learning. *Blended Learning in Finland*, ss. 126–137, 2010.
9. P. Black ja C. Harrison. Feedback in questioning and marking: The science teacher. *School science review*, 82(301):55–61, 2001.
10. P. Black ja D. Wiliam. Assessment and classroom learning. *Assessment in education*, 5(1):7–74, 1998.
11. M. Daniels, A. Berglund, A. Pears ja S. Fincher. Five myths of assessment. Teoksessa *Proceedings of the sixth conference on Australasian computing education-Volume 30*, ss. 57–61. Australian Computer Society, Inc., 2004.
12. F. Dochy ja L. McDowell. Assessment as a tool for learning. *Studies in Educational Evaluation*, 23(4):279–298, 1997.
13. Edmison, Edwards ja A. Pérez-Quiñones. Using a Rubric-based assessment system to improve feedback and student performance in course-management systems. Teoksessa *Proceedings of 2010 ASEE Southeast Section Conference*, 2010.
14. J. Hamer, C. Kell ja F. Spence. Peer assessment using aropä. Teoksessa *ACE '07: Proceedings of the ninth Australasian conference on Computing education*, ss. 43–54, Darlinghurst, Australia, Australia, 2007. Australian Computer Society, Inc.
15. W. Popham. What's wrong-and what's right-with rubrics. *Educational Leadership*, 55:72–75, 1997.
16. A. Powell, S. Turner, M. Tungare, M. Pérez-Quiñones ja S. Edwards. An online teacher peer review system. Teoksessa *Proceedings of Society for Information Technology and Teacher Education International Conference*, ss. 126–133, 2006.
17. M. Scriven. The Methodology of Evaluation (AERA Monograph Series on Curriculum Evaluation, No. 1), 1967.
18. T. Winters ja T. Payne. Computer Aided Grading with Agar. Teoksessa *FECS*, ss. 245–251, 2006.
19. M. Yorke. Formative assessment in higher education: Moves towards theory and the enhancement of pedagogic practice. *Higher Education*, 45(4):477–501, 2003.