

eMo
eAssari Moodle-ympäristössä
Vaatusmäärittely

Helsinki 15.12.06
Ohjelmistotuotantoprojekti
HELSINGIN YLIOPISTO
Tietojenkäsittelytieteenlaitos

Kurssi

581260 Ohjelmistotuotantoprojekti (9+1op/6ov)

Projektiryhmä

Järviniitty Juho

Karkulahti Ossi

Katainen Riku

Tverin Teemu

Vainio Ville-Pekka

Asiakas

Laine Harri

Vastuhenkilö

Taina Juha

Ohjaaja

Halko Antti

Kotisivu

<http://www.cs.helsinki.fi/group/emo/>

Versiohistoria

<i>Versio</i>	<i>Päiväys</i>	<i>Tehdyt muutokset</i>
0.1	25.9.2006	Rakenne
0.2	26.9.2006	Johdanto, vaatimuksia
0.4	27.9.2006	Sidosryhmät, täydennyksiä, käyttötapaukset ym.
0.5	2.10.2006	Täydennystä ja korjailua
0.7	3.10.2006	Täydennyksiä ja korjailua
0.8	3.10.2006	Tehtävätyypin lisäys, viilausta
1.0	9.10.2006	Puutteiden korjaus
1.0.1	15.12.2006	Viimeistely

Sisällys

1 Johdanto.....	1
2 Sanasto.....	1
2.1 Question engine terminologia.....	2
3 Sidosryhmät.....	3
4 Ohjelmiston arkkitehtuuri.....	4
4.1 Yleisrakenne.....	4
4.1.1 Tehtävän määrittelijä (TEMAE).....	5
4.1.2 Tehtävän analysoija (ANALIS).....	5
4.1.3 Tehtävän esittäjä (TEES).....	5
4.1.4 Tietokanta.....	5
5 Käyttötapaukset.....	5
5.1 Opettajan käyttötapaukset.....	5
5.1.1 Tehtävän lisääminen (UCT1).....	5
5.1.2 Tehtävän muokkaaminen (UCT2).....	6
5.1.3 Tehtävän poistaminen (UCT3).....	6
5.1.4 Tehtävien selailu (UCT4).....	7
5.2 Laatijan käyttötapaukset.....	7
5.2.1 Uuden tehtävätyypin lisääminen (UCT5).....	7
5.2.2 Tehtävätyypin muokkaaminen.....	8
5.2.3 Tehtävätyypin poistaminen.....	8
5.3 Opiskelijan käyttötapaukset.....	9
5.3.1 Tehtäviin tutustuminen (UCS 1).....	9
5.3.2 Tehtävään vastaaminen (UCS 2).....	9
6 Ohjelmiston vaatimukset.....	10
6.1 Järjestelmävaatimukset.....	10
6.1.1 Pisteytys (J1).....	10
6.1.2 Arvotut tehtävän muuttujat (J2).....	11
6.1.3 Palautteen antaminen (J3).....	11
6.1.4 Tehtävien automaattinen tarkastus (J4).....	11
6.1.5 Javascript-tuki (J5).....	11
6.1.6 Avustustoiminnot (J6).....	11
6.1.7 Kieliriippumattomuus (J7).....	11
6.1.8 Yksi tietokantarakenne, joka toimii kaikilla tehtävätyypeillä (J8).....	11
6.1.9 Geneerinen tehtävätyyppi (J9).....	12
6.1.10 Tehtävätyyppikohtainen tyylimäärittely (J10).....	12
6.1.11 Vastauksen arvostelun tallennus (J11).....	12
6.1.12 Arvottavien tehtävämuuttujien tallennus (J12).....	12
6.1.13 Tehtävän näyttäminen (J13).....	12
6.2 Käyttöympäristövaatimukset.....	12
6.3 Käyttäjävaatimukset.....	12
6.3.1 Uuden tehtävätyypin lisäys (K1).....	13
6.3.2 Uuden tehtävätyypin luominen (K2).....	13
6.3.3 Selkeä käyttöliittymä lisäykselle (K3).....	13
6.3.4 Hyvin kommentoitu koodi (K4).....	13
6.3.5 Selkeät rajapinnat (K5).....	13
6.3.6 Eri käyttäjäryhmät (K6).....	14
6.3.7 Tehtävätyypin erikoistaminen (K7).....	14
6.3.8 Käyttöohje (K8).....	14
6.3.9 Tehtävään vastaaminen (K9).....	14
6.3.10 Omien tulosten selaaminen (K10).....	14
6.4 Kurssilla toteutettava tehtävätyyppi.....	14
7 Järjestelmän elinkaari.....	15

7.1 Käyttöönotto.....	15
7.2 Muutoksiin varautuminen.....	15
7.2.1 Siirtäminen uuteen ympäristöön.....	15
7.2.2 Tietokantamuutokset.....	15
7.2.3 Selainmuutokset.....	15
7.3 Ylläpito.....	16
7.4 Järjestelmän poistaminen käytöstä.....	16

1 Johdanto

Ohjelmistotuotantoprojektin ryhmä eMo toteuttaa asiakkaalleen Tietojenkäsittelytieteen laitokselle eAssari-järjestelmän Moodle-ympäristöön. Tarkoituksena on luoda järjestelmä, joka tekee uusien eAssari-tehtävätyyppien lisäämisen Moodleen helpoksi. Ohjelmisto laaditaan kurssien vetäjille ja opiskelijoille.

Alkukartoituksena selvitettiin, miten Moodleen oli mahdollista lisätä uusia tehtävätyyppejä ja se todettiin turhankin hankalaksi. Kyseessä on liitännäistyypinen ratkaisu, jossa asennukseen liittyy Moodlen omien tiedostojen muokkaamista ja muita säätelyitä. Moodleen tarvitaan siis selkeä tehtävätyyppien lisäysosio ja tämä taas vaatii yhtenevien koodirunkojen ja saman yläluokan käyttöä, jotka laaditaan tämän projektin yhteydessä. Tässä dokumentissa käsitellään vaatimuksia, jotka on määritelty yhdessä asiakkaan kanssa.

2 Sanasto

Tässä selvitetään muutamia käsitteitä, joita tarvitsee tämän dokumentin ymmärtämisessä ja hieman terminologiaa, jotta käytetyt sanat vastaisivat tarkoitettuja.

Moodle

Moodle on ilmainen, avoimen lähdekoodin sähköinen oppimisolusta, joka soveltuu erilaisille käyttäjäryhmille monenlaisiin käyttötarkoituksiin. Moodle tarjoaa työvälineitä mm. vuorovaikutukseen, sisällöntuottamiseen ja materiaalin jakamiseen. Samankaltaisia oppimisolustoja ovat mm. WebCt, BlackBoard ja FirstClass.

eAssari

eAssari on tietokantapohjainen ympäristö ohjelmallisesti tarkastettavien harjoitus- ja koetehtävien suorittamiseen, missä jokainen tehtävätyyppi käyttää yhteistä tietokantarakennetta. Monikielisyys ja helppo laajennettavuus ovatkin olleet päätavoitteita järjestelmää suunniteltaessa. Ohjelmiston avulla opiskelijalle voidaan selainkäyttöliittymässä esittää tietokannassa määriteltyjä tehtäviä. Opiskelija antaa vastauksen tehtävään käyttöliittymän kautta. Vastaus tarkistetaan normaalisti heti ja opiskelijalle annetaan palautetta tehdyn tarkistuksen perusteella.

eMo-järjestelmä

Ohjelmistotuotantoprojektin ryhmä eMo toteuttaa eAssari-järjestelmän Moodle-ympäristöön ja tätä kutsutaan eMo-järjestelmäksi.

PHP

PHP (PHP Hypertext Preprocessor) on ohjelmointikieli, jota käytetään erityisesti palvelinympäristöissä luotaessa dynaamisia web-sivustoja. PHP-koodi tulkitaan ohjelman suorituksen aikana kääntämisen sijaan.

2.1 Question engine terminologia

Moodlen käyttämä Question engine on kysymysmoottori, joka mahdollistaa valmiita kysymystyyppejä käyttäen kysymyssarjojen luonnin, esittämisen ja arvostelun, sekä mahdollistaa tehtävätyyppeille itännäiset. Sen yhteydessä tulee olla tarkkana tiettyjen termien kanssa, sillä ne eroavat arkikielen vastaavista ilmauksista.

Kysymys (question)

Kysymys on määritelmien (kysymyksen nimi, kysymysteksti, mahdolliset vastaukset, arvostelusäännöt, palaute jne.) joukko, mistä syntyy uudelleenkäytettävä arviointimenetelmä. Kysymyksiä voi olla eri tyyppiä, kuten monivalinta, aukkotäydennys tai numeerinen tehtävä.

Vastaus (answer, response)

Moodle-ympäristössä vastaus tarkoittaa nimenomaan kysymyksen luoja (opettajan) tarkoittamaa vastausta tai niiden joukkoa. Englanninkielinen termi ”answer” tarkoittaa juuri näitä kysymyksen tekijän määrittelemiä vastauksia ja sana ”response” kysymykseen vastaavien henkilöiden vastauksia. Lisäksi kaikki vastaukset eivät ole välttämättä oikeita, esimerkiksi monivalinnassa kaikki vaihtoehdot ovat vastauksia.

Yritys (attempt)

Yritys määritellään nimenomaan tehtävän ratkaisuyritykseksi. Riippuen tehtävätyypistä oppilaalla voi olla useita yrityksiä tehtävää kohti. Yritys päättyy kun oppilas painaa ennalta määriteltyä nappia yrityssivulla. Question engine jokaisen moduulin tulisi tallettaa sen omat yritykset tietokantatauluihinsa.

Istunto (session)

Jokaista yritystä kohden aloitetaan uusi istunto jokaiselle kysymykselle. Vastausistunto ei kestä kauemmin kuin yksi yritys ja jokaista kysymystä kohden voi olla vain yksi istunto yrityksen sisällä.

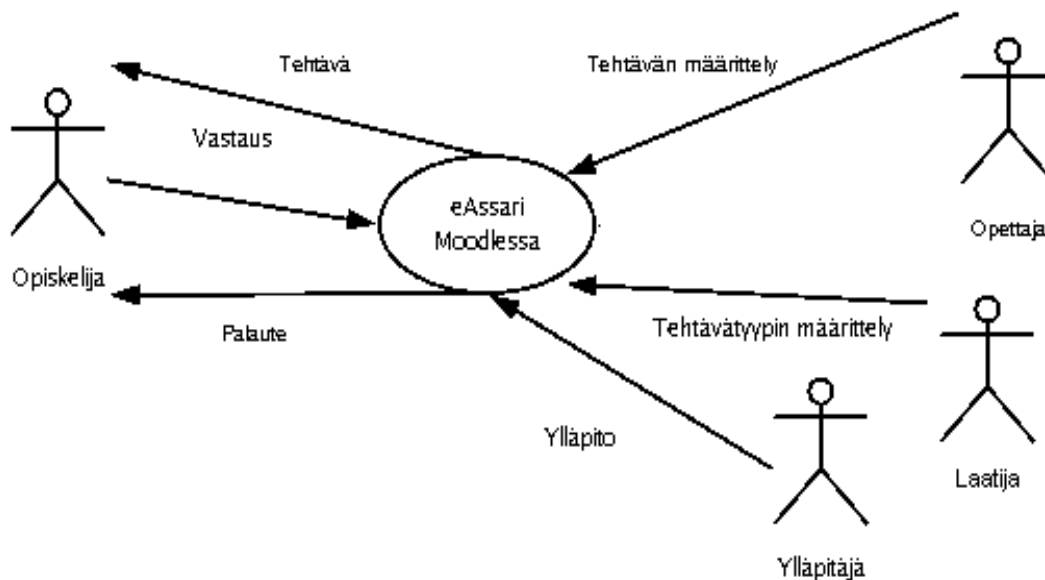
3 Sidosryhmät

Opiskelija: Opiskelija on henkilö, joka käyttää järjestelmää harjoitus- tai koetehtävien suorittamiseen. Järjestelmä arvostelee vastaukset ja tallentaa arvostelun. Opiskelija saa palautetta vastauksestaan. Mikäli tehtävässä on käytetty arvottavia muuttujia, järjestelmä tallentaa tiedon siitä ja näyttää samat osat, jos opiskelija yrittää samaa tehtävää uudelleen.

Opettaja: Opettaja on henkilö, joka käyttää järjestelmää harjoitustehtävien luomiseen. Hän voi tehdä uusia tehtäviä olemassa olevien tehtävätyyppien perusteella.

Laatija: Laatija on opettaja tai muu toimihenkilö, joka luo järjestelmään uusia tehtävätyyppejä PHP-kielellä.

Ylläpitäjä: Ylläpitäjä on henkilö, joka ylläpitää Moodlea ja sen kanssa toimivaa eMo-järjestelmää. Tässä projektissa ei toteuteta ylläpitoon liittyviä komponentteja.



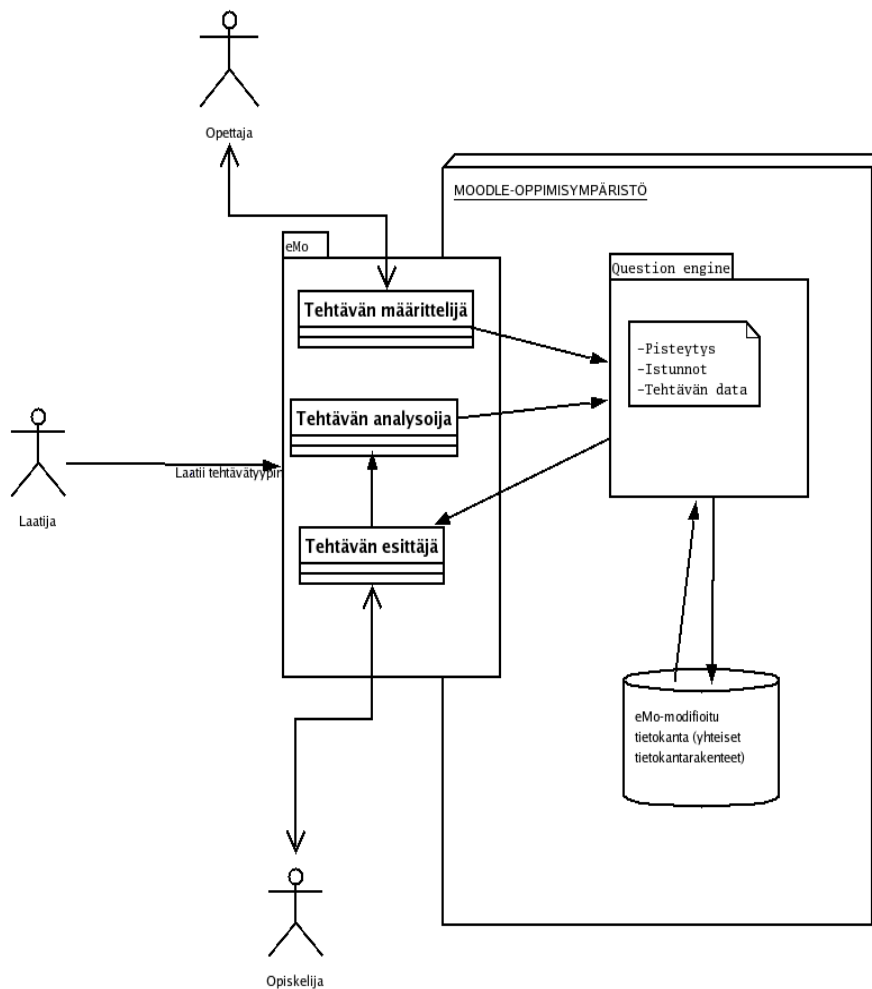
Kuva 1: Sidosryhmäkaavio

4 Ohjelmiston arkkitehtuuri

Tässä luvussa kerrotaan järjestelmän komponenteista, niiden toiminnasta ja yhteyksistä. Yleisesti eMo-ohjelmisto on lisäys Moodlen oppimisympäristöön ja se kommunikoi Moodlen valmiin kysymysmoottorin eli Question enginein kanssa. Pääkomponentit eMo-järjestelmässä ovat tehtävän määrittelijä, analysoija ja esittäjä. Nämä komponentit pitävät sisällään rutiinit, jotka luovat varsinaisen toiminnallisuuden ja kommunikoivat Moodlen kanssa Question enginein välityksellä.

Ohjelmistoon liittyy myös joukko koodirunkoja, joiden avulla laatija luo oman tehtävätyypinsä. Nämä koodirungot käyttävät eMo:n pääkomponenttien tarjoamia palveluita.

4.1 Yleisrakenne



Kuva 2: Ohjelmiston yleisrakenne

Kuva 2 esittää ohjelmiston yleisrakenteen, jossa on esillä pääkomponentit ja niiden väliset suhteet.

4.1.1 Tehtävän määrittelijä (TEMAE)

Tehtävän määrittelijä on eMo-järjestelmän komponentti, joka sisältää funktiot uuden tehtävän lisäämiseen. Komponentti tallentaa tehtävän tiedot tietokantaan joko suoraan tai välillisesti Question enginein kautta.

4.1.2 Tehtävän analysoija (ANALIS)

Tehtävän analysoija sisältää funktiot tallennettujen vastausten hakemiseen. Komponentti mahdollistaa ratkaisuyrityksen ja vastausten vertaamisen tai tehtävätyyppiin liittyvän tarkastusmetodin suorittamisen ja tämän pohjalta arvostelun. Komponentti sisältää myös funktiot arvostelun kirjaamiseen tietokantaan.

4.1.3 Tehtävän esittäjä (TEES)

Tehtävän esittäjä sisältää funktiot, jotka mahdollistavat tehtävien sisällön tarjoamisen Moodlen tehtävännäyttöympäristölle ja tyyli tiedostojen hyväksikäytön tehtävätyyppikohtaisesti.

4.1.4 Tietokanta

Järjestelmää asennettaessa luodaan automaattisesti Moodlen tietokantaan eMo-järjestelmän vaatima tietokantarakenne. Tähän tietokantarakenteeseen kuuluu tauluja, joihin talletetaan uusien tehtävätyyppien tietoja. Tietokannan taulut suunnitellaan siten, että kaikki tulevat tehtävätyypit voivat käyttää tietokantaa tekemättä sen rakenteeseen muutoksia.

5 Käyttötapaukset

Käyttötapaukset ovat yksi tapa järjestelmän toiminnallisten vaatimusten selvittämiseen. Jokainen käyttötapaus käsittelee yhden tai useamman skenaarion, joka kertoo kuinka järjestelmä toimii käyttäjien kanssa jonkin tavoitteen saavuttamiseksi.

5.1 Opettajan käyttötapaukset

Opettajaksi kutsutaan henkilöä, joka pitää kurssia Moodlessa ja haluaa tehdä kurssilleen uusia tehtäviä sekä muokata jo olemassa olevia versioita.

5.1.1 Tehtävän lisääminen (UCT1)

Yleiskuvaus: Opettaja lisää tehtävän järjestelmään.

Ennakkoehdot: Opettajan täytyy olla kirjautuneena Moodleen riittävin käyttöoikeuksin.

Tapahtumien normaali kulku: Opettaja käynnistää tehtävän lisäämisen. Hän määrittelee tehtävänannon, oikean vastauksen, opiskelijalle annettavan pistemäärän ja palautteen ja tallentaa tehtävän järjestelmään.

Poikkeustilanteet: Jokin edellä mainituista osista jää määrittelemättä, jolloin näytetään virheilmoitus ja pyydetään palaamaan takaisin.

Yhteys Moodle-järjestelmään katkeaa, jolloin näytetään virheilmoitus ja pyydetään yrittämään myöhemmin uudestaan.

5.1.2 Tehtävän muokkaaminen (UCT2)

Yleiskuvaus: Opettaja muokkaa olemassa olevaa tehtävää.

Ennakkoehdot: Opettajan täytyy olla kirjautuneena Moodleen riittävin käyttöoikeuksin ja muokattavan tehtävän on oltava olemassa.

Tapahtumien normaali kulku: Opettaja valitsee muokattavan tehtävän ja muokkaa jotain tehtävän osaa, kuten tehtävänantoa, oikeaa vastausta, pistemäärää tai palautetta. Jos joku on jo vastannut tehtävään, opettajalta kysytään poistetaanko olemassa olevat vastaukset vai jätetäänkö ne talteen. Näin siksi, että jos tehtävästä on korjattu vain vähäinen kirjoitusvirhe, se ei vaikuta annettuihin arvosanoihin. Lopuksi opettaja tallentaa muokatun tehtävän järjestelmään.

Poikkeustilanteet: Jokin tarvittava tehtävän määrittelyn osuus puuttuu. Tällöin näytetään virheilmoitus ja pyydetään korjaamaan virhe.

Yhteys Moodle-järjestelmään katkeaa, kun opettaja on tallentamassa muokkauksia sinne. Tällöin näytetään virheilmoitus ja pyydetään yrittämään myöhemmin uudestaan.

5.1.3 Tehtävän poistaminen (UCT3)

Yleiskuvaus: Opettaja poistaa olemassa olevan tehtävän järjestelmästä.

Ennakkoehdot: Opettajan täytyy olla kirjautuneena Moodleen riittävin käyttöoikeuksin ja poistettavaksi tarkoitetun tehtävän on oltava olemassa.

Tapahtumien normaali kulku: Opettaja valitsee poistettavan tehtävän ja hyväksyy poiston. Jos joku

on jo vastannut tehtävään, opettajaa varoitetaan siitä että jos tehtävän poistaa niin myös suoritukset poistetaan. Mikäli opettaja hyväksyy tämän, tehtävä poistetaan järjestelmästä. Muuten palataan edelliseen tilanteeseen.

Poikkeustilanteet: Yhteys Moodle-järjestelmään katkeaa, kun opettaja on hyväksymässä tehtävän poistoa. Tällöin näytetään virheilmoitus ja pyydetään yrittämään myöhemmin uudestaan.

5.1.4 Tehtävien selailu (UCT4)

Yleiskuvaus: Opettaja katselee järjestelmään tallennettuja tehtäviä.

Ennakkoehdot: Opettajan täytyy olla kirjautuneena Moodleen riittävin käyttöoikeuksin.

Tapahtumien normaali kulku: Opettaja valitsee tehtävälustasta tehtävän, jonka tiedot järjestelmä näyttää ruudulla.

Poikkeustilanteet: Yhteys Moodle-järjestelmään katkeaa. Tällöin näytetään virheilmoitus ja pyydetään yrittämään myöhemmin uudestaan.

5.2 Laatijan käyttötapaukset

Laatija on henkilö, joka laatii järjestelmään uusia tehtävätyyppejä PHP-kielellä. Laatija voi olla opettaja tai muu toimihenkilö.

5.2.1 Uuden tehtävätyypin lisääminen (UCT5)

Yleiskuvaus: Laatija lisää järjestelmään uuden tehtävätyypin, joka on ohjelmoitu PHP-kielellä eMo-järjestelmän asettamien vaatimusten mukaisesti.

Ennakkoehdot: Laatijan täytyy olla kirjautuneena Moodleen riittävin käyttöoikeuksin. Laatijalla täytyy olla uuden tehtävätyypin toteuttavaa PHP-koodia ja kuvaus tietokantaan laitettavasta datasta. Koodin ei tarvitse olla täydellistä, vaan myös keskeneräistä koodia voidaan testata asentamalla se järjestelmään. Datan kuvauksen tulee sisältää datan nimi ja tietotyyppi.

Tapahtumien normaali kulku: Laatija valitsee lisäämistoiminnon ja antaa järjestelmälle tiedostot, jotka sisältävät tehtävätyypin PHP-koodia ja tietokantamäärittelyn. Uusi tyyppi lisätään järjestelmään ja sen perusteella voi lisätä uusia kysymyksiä.

Poikkeustilanteet: Yhteys Moodle-järjestelmään katkeaa. Tällöin näytetään virheilmoitus ja

pyydetään yrittämään myöhemmin uudestaan. Laatijan syöttämä PHP-koodi on virheellistä tai ei toteuta sille asetettuja vaatimuksia. Laatijalle näytetään virheilmoitus, jossa muistutetaan järjestelmään syötettävän koodin vaatimuksista.

5.2.2 Tehtävätyypin muokkaaminen

Yleiskuvaus: Laatija muokkaa järjestelmässä olevaa tehtävätyyppiä.

Ennakkoehdot: Laatijan täytyy olla kirjautuneena Moodleen riittävin käyttöoikeuksin.

Tehtävätyypin koodia tai tietokannan määrittelyä ei voi muokata, jos järjestelmässä on olemassa sen perusteella tehtyjä tehtäviä.

Tapahtumien normaali kulku: Laatija valitsee muokattavan tehtävätyypin, tekee muutokset ja tallentaa ne järjestelmään. Vain tehtävätyypin nimeä ja vastaavia kenttiä voi muuttaa, jos tehtävätyypin perusteella on tehty olemassa olevia tehtäviä. PHP-koodia ja tietokannan määrittelyä voi muuttaa vain, jos tällaisia tehtäviä ei ole.

Poikkeustilanteet: Jos laatija yrittää muuttaa PHP-koodia tai tietokannan määrittelyä silloin kun se ei ole sallittua, näytetään virheilmoitus.

Yhteys Moodle-järjestelmään katkeaa. Tällöin näytetään virheilmoitus ja pyydetään yrittämään myöhemmin uudestaan.

5.2.3 Tehtävätyypin poistaminen

Yleiskuvaus: Laatija poistaa järjestelmässä olevan tehtävätyypin.

Ennakkoehdot: Laatijan täytyy olla kirjautuneena Moodleen riittävin käyttöoikeuksin.

Tehtävätyypin voi poistaa vain, jos se ei ole käytössä millään käynnissä olevalla kurssilla.

Tapahtumien normaali kulku: Laatija valitsee poistettavan tehtävätyypin. Tehtävätyyppi poistetaan järjestelmästä.

Poikkeustilanteet: Poistettavaksi valittu tehtävätyyppi on käytössä jollain käynnissä olevalla kurssilla. Tällöin laatijalle näytetään virheilmoitus eikä poistoa sallita.

Yhteys Moodle-järjestelmään katkeaa. Tällöin näytetään virheilmoitus ja pyydetään yrittämään myöhemmin uudestaan.

5.3 Opiskelijan käyttötapaukset

Opiskelijaksi kutsutaan henkilöä, joka on mukana opettajan Moodlessa pitämällä kurssilla ja haluaa vastata siellä tehtäviin sekä tutustua niihin.

5.3.1 Tehtäviin tutustuminen (UCS 1)

Yleiskuvaus: Opiskelija tutustuu järjestelmään tallennettuihin tehtäviin.

Ennakkoehdot: Opettajan on täytynyt lisätä tehtäviä järjestelmään ja antaa opiskelijoille oikeus katsella niitä. Opiskelijan tulee olla kirjautuneena järjestelmään ja kyseiselle kurssille.

Tapahtumien normaali kulku: Opiskelija valitsee tehtävälisteristä tehtävän, jonka tiedot (kysymyksen ja vastausmahdollisuudet) järjestelmä näyttää ruudulla. Arvotut muuttujat arvottaa tässä vaiheessa.

5.3.2 Tehtävään vastaaminen (UCS 2)

Yleiskuvaus: Opiskelija vastaa tehtävään ja saa siitä arvostelun.

Ennakkoehdot: Opiskelijan on oltava kirjautuneena Moodleen ja tehtävään liittyvälle kurssille.

Tapahtumien normaali kulku: Opiskelija valitsee tehtäviin vastaamisen toiminnon. Järjestelmä näyttää yhden tehtävän kerrallaan, antaa mahdollisesti pistemäärän opiskelijalle ja kertoo, oliko vastaus oikein vai väärin. Opiskelijan tekemät tehtävät ja niistä saadut pisteet tallennetaan. Mikäli kyseessä on ollut tehtävä, johon kuuluu arvottavia osia, arvotut osat tallennetaan myös, jotta mahdollisella uudella yrityskerralla opiskelijalle näytetään täysin sama versio kysymyksestä.

6 Ohjelmiston vaatimukset

Tämä luku käsittelee ohjelmiston järjestelmä- ja käyttäjävaatimuksia. Näiden pohjalta laaditaan ohjelmiston toiminnot. Kullakin vaatimuksella on tyyppi, joka ilmaisee sen toiminnallisuuden tai ei-toiminnallisuuden.

Vaatimusten prioriteettiasteikko on yhdestä kolmeen, missä:

- 1 = välttämätön vaatimus, joka toteutetaan tämän projektin yhteydessä
- 2 = vaatimus, jonka pyrimme toteuttamaan annetussa aikarajassa, mutta ei ole kriittinen osa koko ohjelmistoa
- 3 = vaatimus, jonka toteuttaminen on lähinnä kosmeettinen lisä tai jokin ylimääräinen toiminto, joka helpottaa ohjelmiston käyttöä

6.1 Järjestelmävaatimukset

<i>id</i>	<i>Vaatus</i>	<i>Tyyppi</i>	<i>Prioriteetti</i>
J1	pisteytys	toiminnallinen	1
J2	arvotut tehtävän muuttujat	toiminnallinen	1
J3	palautteen antaminen	toiminnallinen	1
J4	tehtävien automaattinen tarkistus	toiminnallinen	1
J5	Javascript-tuki	ei-toiminnallinen	1
J6	avustustoiminnot	toiminnallinen	1
J7	kieliriippumattomuus	ei-toiminnallinen	1
J8	yksi tietokantarakenne	ei-toiminnallinen	1
J9	geneerinen tehtävätyyppi	ei-toiminnallinen	1
J10	tehtävätyyppikohtainen tyylimääritys	ei-toiminnallinen	3
J11	vastauksen arvostelun tallennus	toiminnallinen	1
J12	arvottavien tehtävämuuttujien tallennus	toiminnallinen	1
J13	tehtävän näyttäminen	toiminnallinen	1

6.1.1 Pisteytys (J1)

Opiskelijat saavat pisteitä ratkaistujen tehtävien oikeellisuuden mukaisesti. Tehtävätyypistä riippuen tehtävällä voi olla kynnyisarvo, jonka mukaan tehtävä on joko hyväksytty tai hylätty.

6.1.2 Arvotut tehtävän muuttajat (J2)

Järjestelmän on mahdollistettava eri tehtäviin muuttujia, joiden arvo vaihtelee oppilaskohtaisesti. Esimerkiksi laskutehtävässä X-muuttujalla olisi eri arvo eri opiskelijoilla. Opiskelija voi halutessaan tehdä saman tehtävän samoilla arvoilla (vaatimus J12) tai pyytää uudelleenarvontaa, joka on estettävissä tehtäväkohtaisesti.

6.1.3 Palautteen antaminen (J3)

Järjestelmän on mahdollistettava palautteen antaminen ratkaisuyrityksen yhteydessä. Palaute voi olla ratkaisuyrityksien syötteisiin mukautettua.

6.1.4 Tehtävien automaattinen tarkastus (J4)

Järjestelmä tarkastaa yritetyn tehtävän automaattisesti. Järjestelmä vertaa tehtävän luomisessa annettuja tai tehtävän itse arpomia vastauksia oppilaan syötteisiin ja antaa palautetta vastauksen oikeellisuudesta. Palautteen tyyppi riippuu tehtävätyypistä. Esim. SQL-trainer -tyyppisessä tehtävässä järjestelmä voi antaa palautteena SQL-kyselyn tuloksen.

6.1.5 Javascript-tuki (J5)

Järjestelmän on mahdollistettava Javascript-ohjelmien ajaminen. Javascript-koodia voidaan käyttää esim. lomakkeiden tarkastamiseen ennen lähettämistä tai vaikkapa graafisemman kysymyksen esittämiseksi.

6.1.6 Avustustoiminnot (J6)

Järjestelmän on tarjottava ohjeet tehtävätyyppien luomiseen, sekä mahdollisuus luoda ohjeistus näiden uusien tehtävätyyppien käyttämiseen. Järjestelmän on tuettava myös näiden ohjeiden näyttämistä.

6.1.7 Kieliriippumattomuus (J7)

Opettaja voi valita kielen, jolla tehtävät määritellään ja oppilas kielen, jolla tehtävät tehdään. Jos valittua kieltä ei ole saatavilla, tehtävät näytetään oletuskielellä.

6.1.8 Yksi tietokantarakenne, joka toimii kaikilla tehtävätyypeillä (J8)

Tieto- ja tietokantarakenteet suunnitellaan sellaisiksi, että kaikki tehtävätyypit voidaan toteuttaa muuttamatta tietokannan rakenteita. Tämä vähentää tehtävätyypin ohjelmoijalta työmäärää ja

yhdenmukaistaa eMo-järjestelmän talletusjärjestelmää.

6.1.9 Geneerinen tehtävätyyppi (J9)

Järjestelmä tarjoaa tehtävätyypeille yläluokan, josta voi erikoistaa omia tyyppejä. Yläluokassa määritellään rajapinnat, joiden puitteissa erikoistukset toteutetaan. Yläluokka kommunikoi Moodleen rakennettujen moduuleitten kanssa, joten eMo-järjestelmän yläluokka on rajapintana Moodlen ja eMo:n tehtävätyyppien välillä.

6.1.10 Tehtävätyyppiokohtainen tyylimäärittely (J10)

Jokaiseen tehtävätyyppiin voi liittää oman tyylimäärittelyn CSS-tiedostolla.

6.1.11 Vastauksen arvostelun tallennus (J11)

Järjestelmä tallentaa opiskelijan saamat arvostelut tehtävistä Moodlen tietokantaan. Opettajilla on oikeudet tutkia opiskelijoiden saamia tuloksia.

6.1.12 Arvottavien tehtävämuuttujien tallennus (J12)

Järjestelmä tallentaa tehtävissä käytetyt arvotut muuttujat tietokantaan, jotta opiskelija voi myöhemmin tehdä saman tehtävän samoilla arvoilla.

6.1.13 Tehtävän näyttäminen (J13)

Tehtävät näytetään opiskelijalle Moodlen omalla graafisella rungolla sekä vaatimuksessa J10 mainitulla tavalla.

6.2 Käyttöympäristövaatimukset

eMo-järjestelmä ohjelmoidaan Moodle 1.7-ympäristöön PHP 4.3.0-kielellä ja tietokannanhallintajärjestelmänä toimii MySQL 4.1.12. Järjestelmä käyttää SQL-standardia. Sen tulee toimia Tietojenkäsittelytieteen laitoksen ATK-ympäristössä käytettävillä Mozilla Firefox - ja Internet Explorer -selaimilla.

6.3 Käyttäjävaatimukset

Tämän ohjelmiston käyttäjiä ovat opettajat, jotka lisäävät Moodle-ympäristöön omia tehtäviä, laatijat, jotka lisäävät uusia tehtävätyyppejä ja oppilaat, jotka tekevät opettajien tekemiä tehtäviä. Käyttäjävaatimukset koskevat ainoastaan heitä.

<i>id</i>	<i>Vaatus</i>	<i>Tyyppi</i>	<i>Prioriteetti</i>
K1	uuden tehtävätyypin lisäys	toiminnallinen	1
K2	uuden tehtävätyypin luominen	toiminnallinen	1
K3	selkeä käyttöliittymä lisäykselle	ei toiminnallinen	1
K4	hyvin kommentoitu koodi	ei toiminnallinen	1
K5	selkeät rajapinnat	ei toiminnallinen	1
K6	eri käyttäjäryhmät	toiminnallinen	1
K7	tehtävätyypin erikoistaminen	toiminnallinen	1
K8	käyttöohje	ei toiminnallinen	1
K9	tehtävään vastaaminen	toiminnallinen	1
K10	omien tulosten selaaminen	toiminnallinen	1

6.3.1 Uuden tehtävätyypin lisäys (K1)

Jotta Moodleen olisi mahdollista luoda helppo ja automatisoitu tehtävätyyppien lisäysosio, kaikille tehtävätyypeille laaditaan yläluokka ja samat koodirungot, joilla voi toteuttaa esimerkiksi SQL-kyselytehtäviä.

6.3.2 Uuden tehtävätyypin luominen (K2)

Uudet tehtävätyypit laaditaan eMo-ryhmän luoman yläluokan ja siellä määriteltyjen rajapintojen ja tietokantarakenteiden pohjalta. Tarjoamme myös valmiit koodirungot, joiden sisälle oma tehtävätyyppi kirjoitetaan.

6.3.3 Selkeä käyttöliittymä lisäykselle (K3)

Tehtävätyyppien lisäykseen luodaan käyttöliittymä, jolla voi etsiä oman tehtävätyypinsä hakemistopuusta ja lisätä sen osaksi eMo-järjestelmää. Käyttöliittymä noudattaa Moodlen valmista ulkoasua.

6.3.4 Hyvin kommentoitu koodi (K4)

Koodirunkojen tulee olla erittäin hyvin kommentoituja, jotta niiden sisälle on helppo alkaa ohjelmoida omaa tehtävätyyppiään. Kommentit ovat englanninkielisiä.

6.3.5 Selkeät rajapinnat (K5)

Rajapintojen tulee olla selkeillä nimillä laadittuja ja niiden toiminnot helposti ymmärrettäviä. Rajapintojen toiminta on kommentoitu lähdekoodissa englanniksi.

6.3.6 Eri käyttäjärühmät (K6)

Järjestelmässä tulee olla omat tunnukset ylläpidolle, tehtävätyypin laatijalle, tehtävän laatijalle ja tehtävien tekijälle eli opiskelijalle. eMo-järjestelmä käyttää Moodlen valmiita käyttäjärühmiä. Laatijoilla on erityisoikeus lisätä tehtävätyyppejä.

6.3.7 Tehtävätyypin erikoistaminen (K7)

Tehtävätyyppejä voi erikoistaa eMo-ryhmän laatimasta yläluokasta. Ohjelmiston tulee olla tarpeeksi geneerinen, jotta siitä pystyy erikoistamaan monia erilaisia tehtävätyyppejä.

6.3.8 Käyttöohje (K8)

Ohjelmistoon tulee täsmällinen käyttöohje tehtävätyyppien laatijoille. Ohjeet koskevat tehtävätyyppien ohjelmointia ja Moodleen lisäämistä. Koodirunkojen sisällä on käyttöohjeita Moodlen eri komponenttien ja eMo-ryhmän luomien rajapintojen käytöstä.

6.3.9 Tehtävään vastaaminen (K9)

Tehtävätyyppeihin määritellään eri vastausmetodeja, jotka toimivat html-lomakkeiden puitteissa. Eri metodeita ovat esimerkiksi numeerinen vastaus, säännöllinen lauseke ja oikein/väärin -vastaus. Tehtävissä voi olla useampia vastauskenttiä.

6.3.10 Omien tulosten selaaminen (K10)

Opiskelija voi kirjautuneena selata tekemiensä tehtävien tuloksia Moodlen rekisteröitymisestä nykyhetkeen asti.

6.4 Kurssilla toteutettava tehtävätyyppi

eMo-järjestelmään laaditaan yksi tehtävätyyppi valmiiksi, jossa harjoitellaan säännöllisiä lausekkeita. Opiskelijan tehtävänä on antaa säännöllinen lauseke, joka tunnistaa tehtäväkuvauksessa määritellyn rakenteen (esimerkiksi URL:n tai sähköpostiosoitteen). Vastauksen tarkistusta varten on määritelty joukko kelpaavia merkkijonoja, jotka lauseke pystyy tunnistamaan ja joukko epäkelpoja, joita lauseke ei saa tunnistaa. Palautteena annetaan virheellisesti tunnistetut merkkijonot, ja pistemääränä oikein tunnistettujen prosenttiosuus.

7 Järjestelmän elinkaari

Järjestelmän tai ohjelmiston elinkaarella tarkoitetaan aikaa, joka kuuluu järjestelmän kehityksen aloittamisesta sen poistamiseen käytöstä.

7.1 Käyttöönotto

Järjestelmä otetaan käyttöön Tietojenkäsittelytieteen laitoksen ympäristössä ohjelmistotuotantoprojektin lopussa.

7.2 Muutoksiin varautuminen

Järjestelmään kohdistuu monia muospaineita eri suunnilta järjestelmän elinkaaren aikana. Tämän takia on hyvä varautua näihin paineisiin jo järjestelmää määriteltäessä.

7.2.1 Siirtäminen uuteen ympäristöön

Toistaiseksi järjestelmää ei ole tarvetta siirtää toisiin ympäristöihin, joten tätä mahdollisuutta ei ole juurikaan huomioitu ohjelmistoa toteutettaessa. Ohjelmisto suunnitellaan käytettäväksi ainoastaan laitoksen Moodle-ympäristössä. Kuitenkin voidaan olettaa, että ohjelmisto tulee toimimaan kohtuullisella muokkaamisella myös muiden organisaatioiden Moodle-ympäristöissä johtuen siitä, että Moodle on itsessään samanlainen ympäristö kaikkialla. Moodlen päivittyminen saattaa aiheuttaa muospaineita myös ohjelmistoon. Tällä hetkellä laitoksella on käytössä Moodlen versio 1.6, mutta tuleva 1.7 julkaistaan lähiaikoina ja toteutammekin ohjelmiston tälle versiolle. Ohjelmisto pyritään tuottamaan mahdollisuuksien rajoissa siten, että sen siirtäminen muihin Moodle-versioihin olisi mahdollisimman kivutonta.

7.2.2 Tietokantamuutokset

Tietokantamuutokset vaikuttavat ohjelmiston tietokantaoperaatioihin. Tietokantaoperaatioita joudutaan päivittämään ja muuttamaan, jos tietokantarakenteita muutetaan.

7.2.3 Selainmuutokset

Tämänhetkisten vaatimusten mukaan järjestelmän on toimittava laitoksen koneilla ja selaimilla. Selaimien tulevaisuudessa julkaistavat versiot saattavat vaatia järjestelmän muuttamista joiltakin osin.

7.3 Ylläpito

Ohjelmiston ylläpito rajataan ohjelmistotuotantoprojektin ulkopuolelle. Tämän takia ohjelmiston ylläpidosta ei ole tehty ohjelmistotuotantoprojektin ylittäviä suunnitelmia. Ohjelmistosta pyritään kuitenkin kehittämään sellainen, että sitä voidaan halutessa jatkokehittää tulevaisuuden ohjelmistotuotantoprojekteissa. Tämä edellyttää selkeää toteutustapaa sekä kattavaa dokumentaatiota.

7.4 Järjestelmän poistaminen käytöstä

Järjestelmän poistaminen sen elinkaaren umpeuduttua on yksinkertaista. Ohjelmiston tiedostot poistetaan Moodle-palvelimelta ja tietokanta tietokantapalvelimelta. Moodleen tehdään tarvittavat muutokset, jotta käyttäjät eivät harhau du toimimattomille alueille. Tämän jälkeen järjestelmä on poistettu käytöstä.