
Timppa-sovellusprojekti, sovellusraportti

Matias Berg
Bek Eljurkaev
Minna Lehtomäki
Juhani Sihvonen
Hannu Viinikainen

Versio 1.0.0
Julkinen

Jyväskylän yliopisto
Tietotekniikan laitos
Jyväskylä

Tietoa dokumentista

Tekijät:

- Matias Berg (MB), matias.j.k.berg@student.jyu.fi
- Bek Eljurkaev (BE), kaeljurk@student.jyu.fi
- Minna Lehtomäki (ML), minna.j.lehtomaki@student.jyu.fi
- Juhani Sihvonen (JS), juhani.j.p.sihvonen@student.jyu.fi
- Hannu Viinikainen (HV), hannu.j.viinikainen@student.jyu.fi

Dokumentin nimi: Timppa-sovellusprojekti, sovellusraportti

Tiivistelmä: Timppa-projekti kehitti Jyväskylän yliopiston tietotekniikan laitoksella kehitettyyn TIM-oppimisympäristöön WWW-sovelluksen tukemaan välitöntä vuorovaikutusta massaluennoilla. Kehitetyllä sovelluksella luennoitsijat voivat luoda luentoja, laatia kysymyksiä, kysyä luennolle liittyneiltä opiskelijoilta kysymyksiä luennon aikana, sekä esittää kysymysten vastausjakauksia. Oppilaat voivat liittyä luennoille, sekä vastata luentokysymyksiin, kysyä kysymyksiä ja kommentoida luentoa luentoseinän avulla. Sovellusraportissa kuvataan sovelluksen käyttöliittymää ja kokonaisrakennetta, tavoitteiden toteutumista ja jatkokehitysideoita.

Avainsanat: Jatkokehitys, järjestelmätestauksen tulokset, kokonaisrakenne, käyttöliittymä, tavoitteiden toteutuminen, tietokannan taulurakenne.

Muutoshistoria:

Versio	Päivämäärä	Muutokset	Tekijät
0.0.1	30.4.2015	Luotiin dokumenttipohja pääotsikoineen.	ML
0.0.2	15.5.2015	Tarkennettiin otsikointia ja lisättiin kuvauksia lukujen alkuihin.	ML
0.0.3	28.5.2015	Aloitettiin Käyttöliittymän kuvaaminen.	ML
0.1.0	7.6.2015	Ensimmäinen versio raportista lähetettiin projektiorganisaatiolle.	JS, MB
0.1.1	12.6.2015	Tehtiin muutoksia palautteen perusteella.	MB
0.2.0	25.8.2015	Laajennettiin käyttöliittymien kuvausta koskevia lukuja.	MB
0.3.0	26.8.2015	Laajennettiin käyttöliittymien kuvausta, sovelluksen rakennetta ja tietokannan rakennetta koskevia lukuja.	HV, MB
0.3.1	31.8.2015	Tietokannan rakennetta koskevaa lukua korjattiin palautteen perusteella.	MB
0.3.4	2.9.2015	Sovelluksen rakennetta koskevaa lukua korjattiin palautteen perusteella.	HV
0.3.5	3.9.2015	Käyttöliittymien kuvausta koskevia lukuja korjattiin palautteen perusteella ja tehtiin muita muutoksia.	MB
0.4.0	4.9.2015	Termejä ja tavoitteiden toteutumislukuja muokattiin.	BE, MB
0.4.5	9.9.2015	Muokattiin lukuja 1-4 palautteen perusteella.	MB
0.9.0	10.9.2015	Muokattiin loppuja lukuja palautteen perusteella.	MB
1.0.0	14.9.2015	Muokattiin dokumenttia palautteen perusteella.	MB

Tietoa projektista

Timppa-projekti kehitti Jyväskylän yliopiston tietotekniikan laitoksella kehitettyyn TIM-oppimisympäristöön sovelluksen tukemaan välitöntä vuorovaikutusta massaluennoilla.

Tekijät:

- Matias Berg, matias.j.k.berg@student.jyu.fi
- Bek Eljurkaev, kaeljurk@student.jyu.fi
- Minna Lehtomäki, minna.j.lehtomaki@student.jyu.fi
- Juhani Sihvonen, juhani.j.p.sihvonen@student.jyu.fi
- Hannu Viinikainen, hannu.j.viinikainen@student.jyu.fi

Tilaaja:

- Vesa Lappalainen, vesa.t.lappalainen@jyu.fi
- Antti-Jussi Lakanen, antti-jussi.lakanen@jyu.fi

Ohjaajat:

- Mika Lehtinen, mika.k.lehtinen@student.jyu.fi
- Jukka-Pekka Santanen, santanen@mit.jyu.fi

Yhteystiedot:

- Sähköpostilistat timppa@korppi.jyu.fi ja timppa_opetus@korppi.jyu.fi
- Sähköpostiarkistot: <https://korppi.jyu.fi/kotka/servlet/list-archive/timppa> ja https://korppi.jyu.fi/kotka/servlet/list-archive/timppa_opetus

1. Johdanto

Timppa-projekti kehitti Sovellusprojekti-kurssilla keväällä 2015 Jyväskylän yliopiston tietotekniikan laitoksella kehitettyyn TIM-oppimisympäristöön sovelluksen tukemaan välitöntä vuorovaikutusta massaluennoilla. TIM-järjestelmän ohella aikaisemmin kyselyominaisuuksia varten tarvittiin erillinen inSitu-järjestelmä. Kahden erillisen ohjelmiston käyttö, ylläpito ja kehittäminen on todettu hankalaksi ja työlääksi.

Timppa-projektiryhmä lisäsi TIM-järjestelmään kyselyominaisuuden, jolla pystyy luomaan monipuolisempia kysymyksiä kuin vanhassa inSitu-järjestelmässä. Projektiryhmä lisäsi TIM-järjestelmään myös luentoseinän, johon oli olemassa erillinen sivusto ennestään.

Sovellusraportin laatimisessa on hyödynnetty Liikkuva-projektin sovellusraporttia [8] sekä Timppa-projektin projektiraporttia [4], projektisuunnitelmaa [5], vaatimusmäärittelyä [6] sekä järjestelmätestausraportteja [1] ja [2]. Timppa-projektin laatimiin dokumentteihin kuuluu myös järjestelmätestaussuunnitelma [3].

Sovellusraportti muodostuu kahdeksasta luvusta. Luvussa 2 kuvataan sovelluksen kannalta olennaisia termejä. Luvussa 3 kuvataan sovelluksen käyttöliittymän valikkoja, yleisnäkymää ja luentoja koskevia toimintoja. Luvussa 4 kuvataan käyttöliittymän kysymyksiin liittyviä toimintoja. Luvussa 5 kuvataan sovelluksen kokonaisrakennetta ja rajapintoja. Luvussa 6 kuvataan sovelluksen tietokannan rakennetta. Luvussa 7 kuvataan tavoitteiden toteutumista ja listataan jatkokehitettäviä ominaisuuksia. Raportin päättää yhteenvedon sisältävä luku.

2.Termit

Luvussa kuvataan projektissa käytettäviä aihealueen ja kehitetyn järjestelmän termejä.

2.1 Aihealueen termejä

Projektin aihealueen termejä ovat seuraavat:

- **Kysymys** on luennon aikana esitetty monivalinta- tai tekstikenttäkysymys, johon opiskelijat vastaavat.
- **Käyttöliittymä** on sovelluksen osa, jonka avulla käyttäjä käyttää sovellusta.
- **Luento** on tietyn ajanjakson kestävä opetustapahtuma, jonka aikana luentoseinälle voi kirjoittaa ja opettaja voi kysyä kysymyksiä luennoille liittyneiltä oppilailta.
- **Luentoseinä** on opettajan ja opiskelijoiden välinen luennoilla käytettävä pikaviestin.
- **Opiskelija** on luentomateriaalia lukeva ja luennoille osallistuva henkilö.
- **Opettaja** on luentomateriaalin dokumentin omistaja ja luennon pitäjä.
- **TIM** on Jyväskylän yliopiston tietotekniikan laitoksella kehitetty välitöntä vuorovaikutusta tukeva oppimisympäristö, jonka nimi tulee sanoista "The Interactive Material".

2.2 Ohjelmistoja ja teknisiä termejä

Dokumentissa käytettyjä teknisiä termejä ovat seuraavat:

- **AngularJS** on JavaScript-sovelluskehys.
- **Chart.JS** on JavaScript-kirjasto, jonka avulla voi piirtää diagrammeja.
- **HTTP-pyyntö** on asiakkaan palvelimelle lähettämä pyyntö.
- **jQuery** on JavaScript-kirjasto, joka yksinkertaistaa JavaScriptin perustoiminnallisuuksien käyttämistä.
- **JSON** on tiedostomuoto jota käytetään tiedonsiirtoon.
- **Kaksisuuntainen datakytkentä** on prosessi, jossa yhteys käyttöliittymän ja palvelimen välillä toimii molempiin suuntiin ja muutokset heijastuvat automaattisesti mallin ja näkymän välillä.
- **Long-polling** on eräs HTTP-pyyntön alatyyppejä, jossa asiakas tekee HTTP-pyyntöä palvelimelle ja jää odottamaan vastausta. Palvelin ei välttämättä vastaa heti, vaan vastaa vasta tiettyjen ehtojen täytyessä.
- **Ohjain** toimii välittäjänä käyttäjän ja tietokannan välisessä kommunikaatiossa.

3. Käyttöliittymän valikot ja luennot

Luvussa esitellään sovelluksen käyttöliittymän sivu- ja ylävalikoiden toiminnallisuutta, luentojen luomista sekä luentoseinän toiminnallisuutta.

3.1. Opettajan käyttöliittymän näkymät

Kuvassa 3.1 esitetään järjestelmän käyttöliittymän näkymät, kun luento on käynnissä.



Kuva 3.1: TIMin käyttöliittymän näkymät.

Luennon ollessa käynnissä sovellus koostuu kirjaimilla kuvaan 3.1 merkityistä seuraavista näkymistä:

- **A** sivuvalikko,
- **B** ylävalikko,
- **C** luentoseinä ja
- **D** luentomateriaali.

3.2. Sivuvalikko

Kuvan 3.2 sivuvalikossa näkyvät kaikki mahdolliset sivuvalikkoon tulevat ikonit, mutta sovelluksen käytössä niitä kaikkia ei välttämättä aina ole samaan aikaan näkyvissä.



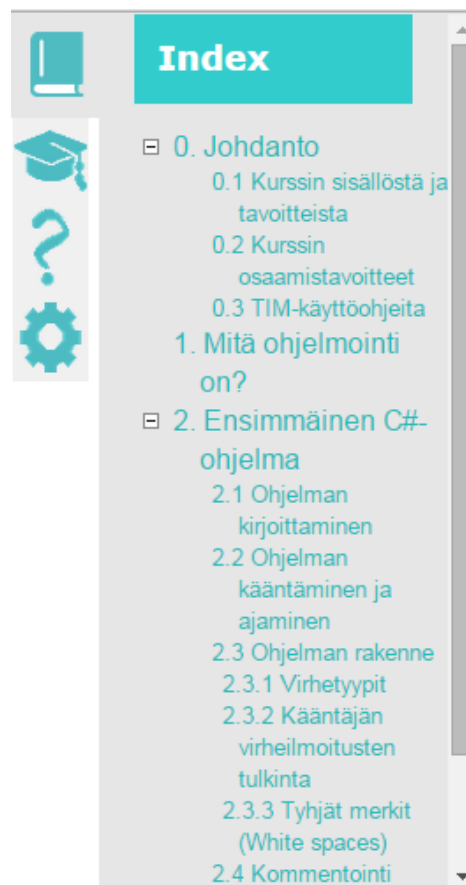
Kuva 3.2: Sivuvalikko kaikilla mahdollisilla ikoneilla.

Ikonien merkitykset ovat seuraavat:

- **A** dokumentin sisällysluettelo,
- **B** dokumentin luennot,
- **C** dokumentin kysymykset,
- **D** luennolla olevat käyttäjät ja
- **E** asetukset.

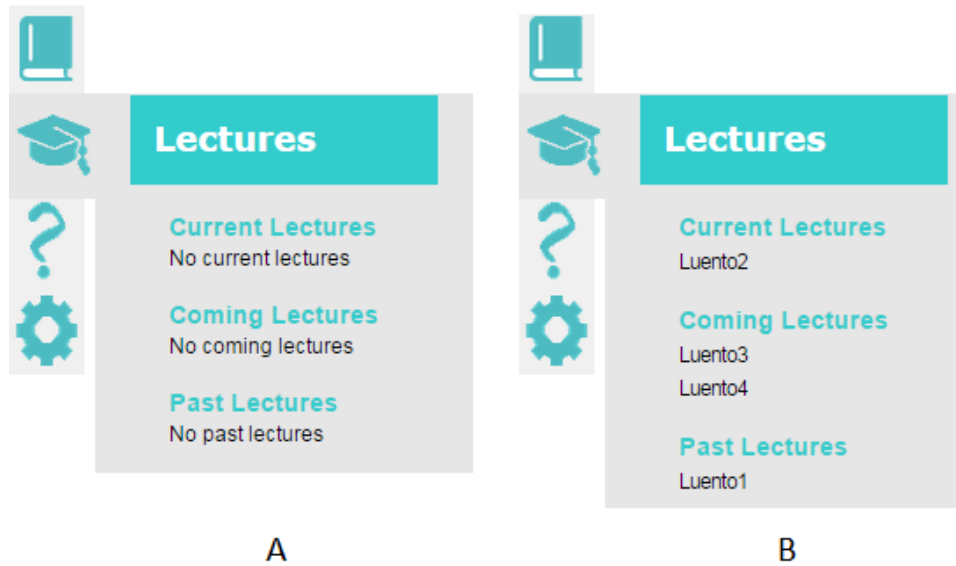
Kutakin ikonia klikkaamalla avautuu kuvissa 3.3-3.7 esitetyt sivuvalikot.

Kuvan 3.3 dokumentin sisällysluettelossa esitetään kaikki kolmen ylimmän tason lukuotsikot.



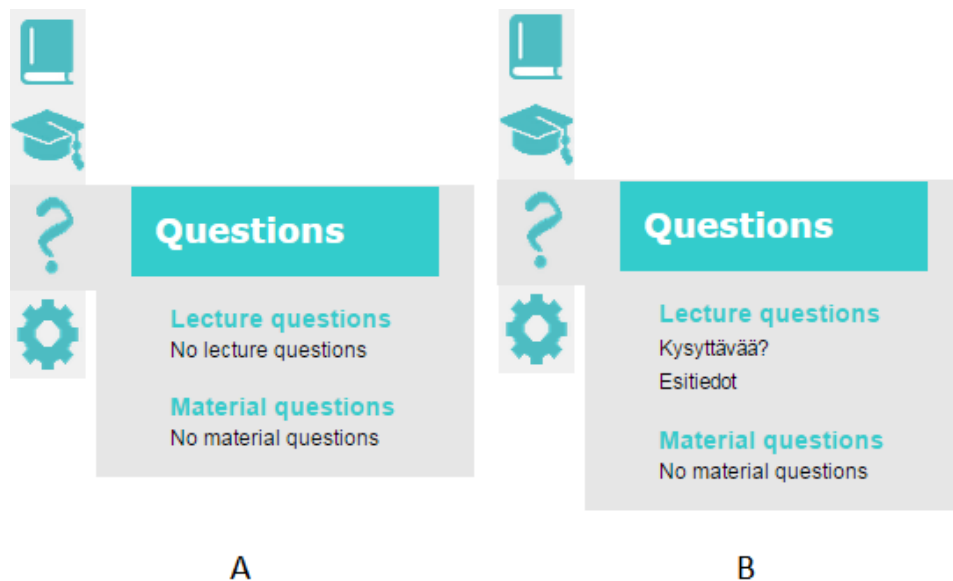
Kuva 3.3: Dokumentin sisällysluettelo.

Kuvan 3.4 sivuvalikossa esitetään dokumentin luennot. Vaihtoehdossa A dokumentissa ei ole yhtään luentoa, sekä vaihtoehdossa B dokumentissa on jo pidetty luento, käynnissä oleva luento ja tulevia luentoja.



Kuva 3.4: Dokumentin luentovalikko.

Kuvan 3.5 sivuvalikossa esitetään dokumentin kysymykset. Vaihtoehdossa A dokumentissa ei ole yhtään kysymystä ja vaihtoehdossa B dokumentissa on kaksi kysymystä.



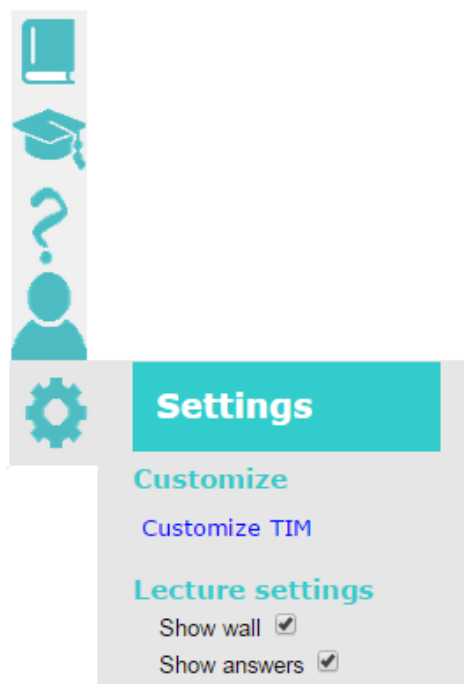
Kuva 3.5: Dokumentin kysymysvalikko.

Kuvan 3.6 käyttäjävalikossa luennolle on liittyneenä kaksi käyttäjää, joista toisella on luennoitsijan oikeudet ja toisella oppilaan oikeudet. Nimen perässä oleva aika tarkoittaa kellonaikaa, jolloin viimeksi yhteys käyttäjään saatiin.



Kuva 3.6:Luennolla olevien käyttäjien valikko.

Kuvan 3.7 asetusvalikossa luento on käynnissä, sillä luentoasetukset näkyvät. Tässä valikossa voi vaikuttaa TIMin ulkoasuun ja muokata luennon aikana näytettäviä näkymiä.



Kuva 3.7:Asetusvalikko.

3.3. Ylävalikko

Kuvan 3.8 ylävalikko näyttää tilanteen, jossa dokumentissa ei ole tulevia tai käynnissä olevia luentoja. Uuden luennon luominen aloitetaan klikkaamalla +-painiketta.

Now: No running lectures

Next: No upcoming lectures

Logged in as: Lehtomäki Minna Johanna (mijosiro)
Logout

Kuva 3.8: Ylävalikko dokumentissa, jossa ei ole luentoja.

Kuvan 3.9 ylävalikko näyttää tilanteen, jossa dokumentissa on jo luentoja.

Now: Luento2

Later: Luento3 2015-07-1

Logged in as: Berg Matias Juho Kalevi (majukabe)
Logout

A B C D B G

Kuva 3.9: Ylävalikko dokumentissa, jossa luento ei ole käynnissä.

Kuvan 3.9 ylävalikkoon merkittyjen kohtien merkitykset ovat seuraavat:

- **A** liity valitulle luennolle,
- **B** muokkaa valittua luentoja,
- **C** luo uusi luento,
- **D** käynnistä valittu luento,
- **E** käynnissä olevien luentojen valikko,
- **F** tulevien luentojen valikko ja
- **G** uloskirjautumispainike.

Kuvan 3.10 ylävalikko näyttää tilanteen, jossa dokumentissa luento nimeltään *ITKS100_1* on käynnissä. Valikossa näkyy luennon aloitus- ja päättymisaika. Painikkeiden merkitykset ovat seuraavat:

- **A** poistu luennolta,
- **B** pysäytä luento (näkyvää vain luennon luojalle) ja
- **C** muokkaa luentoja (näkyvää vain luennon luojalle).

Currently joined in: **ITKS100_1**

Started: 2015-08-25 08:48
Ends: 2015-08-25 11:03

Logged in as: Berg Matias Juho Kalevi (majukabe)
Logout

A B C

Kuva 3.10: Ylävalikko dokumentissa, jossa luento on käynnissä.

3.4. Luennon luominen

Kuvan 3.11 luennon luomisikkunassa luennolle on annettu nimi ja salasana, joka vaaditaan opiskelijoilta luennolle pääsyä varten. Lisäksi luennolle on määrätty aloitusaika, kesto ja opiskelijoilla on mahdollisuus liittyä luennolle 15 minuuttia ennen luennon alkua.

Create Lecture

Lecture name:

Lecture access code:
(Optional)

Starting time:
Date: Time: :

Allow students to join 15 minutes prior to the start of the lecture.

Ending time:
 Use date:
Date: Time: :

Duration:
Hours: Min:

Kuva 3.11: Luennon luomisikkuna.

Kuvan 3.12 päivämääräkenttä ilmestyy, kun aloituspäivämäärää tai lopetuspäivämäärää halutaan muokata.

Create Lecture

Lecture name:

Lecture access code:
 (Optional)

Starting time:
 Date: Time: :

All U D H

Ending time:
 U D H

July 2015

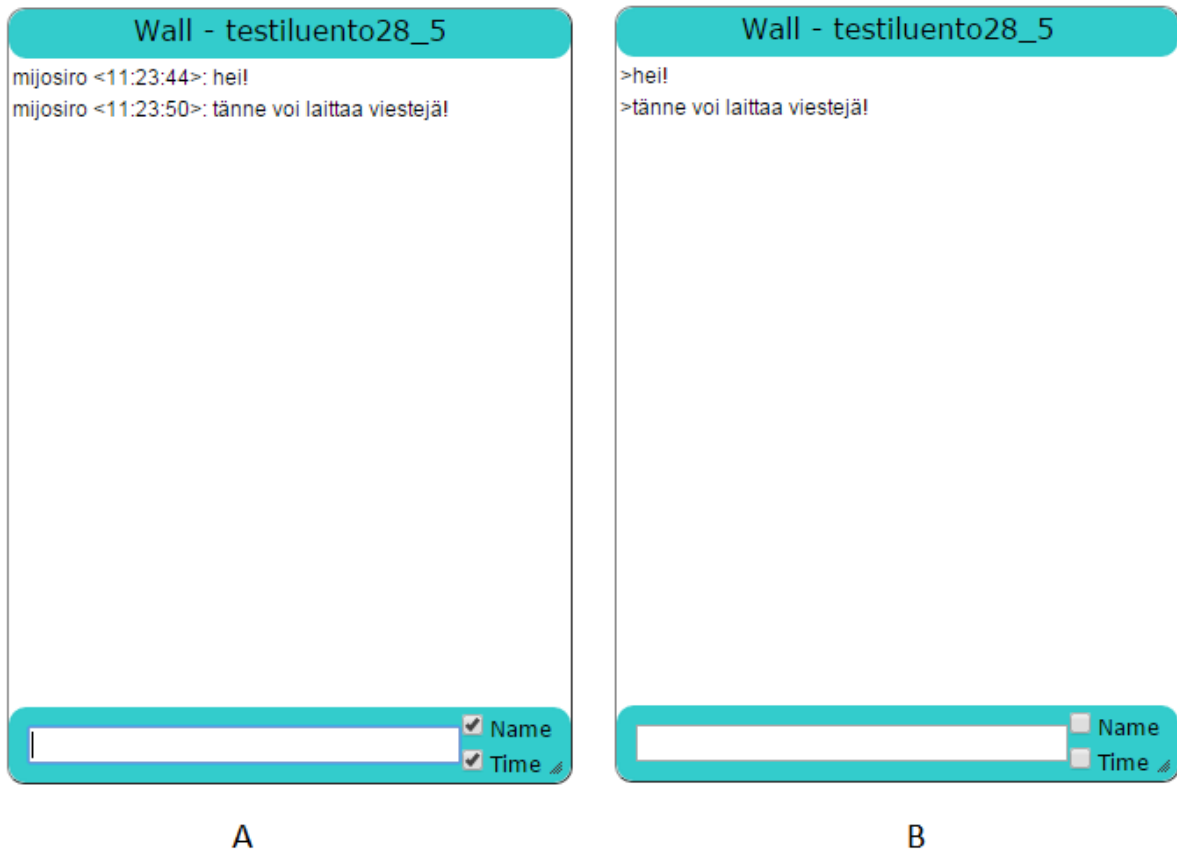
Su	Mo	Tu	We	Th	Fr	Sa
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

Cancel

Kuva 3.12: Päivämääräkentästä avautuva kalenteri.

3.5. Luentoseinä

Kuvassa 3.13 näkyy luentoseinä avattuna. Kohdassa A aika ja lähettäjän käyttäjätunnus näkyvät, sekä kohdassa B molemmat ovat piilotettuna. Lähettääkseen viestin luentoseinälle se tulee kirjoittaa seinän alareunassa olevaan tekstilaatikkoon ja on painettava Enter-näppäintä.



Kuva 3.13: Luentoseinä avattuna.

3.6. Luennon päättäminen ja menneiden luentojen tarkastelu

Luennon päättäminen onnistuu painamalla B-painiketta kuvassa 3.10 olevassa valikossa. Luennon tietoja voi tarkastella painamalla sivuvalikosta (kuvassa 3.4) luennon nimeä, jolloin avautuu kuvassa 3.14 näkyvä sivu.

Lecture info: OHJ2_3

Lecture info

Start time:

2015-08-25 13:35

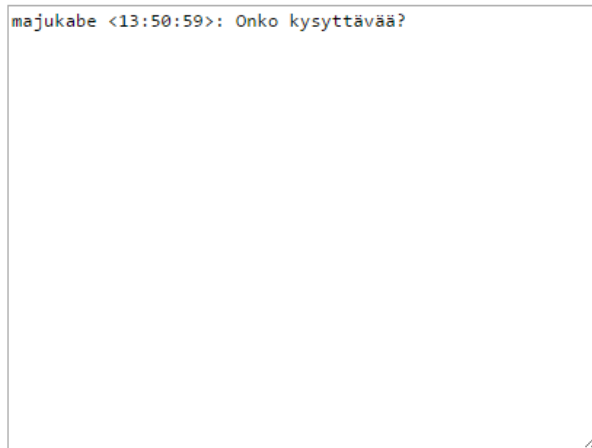
End time:

2015-08-25 13:51

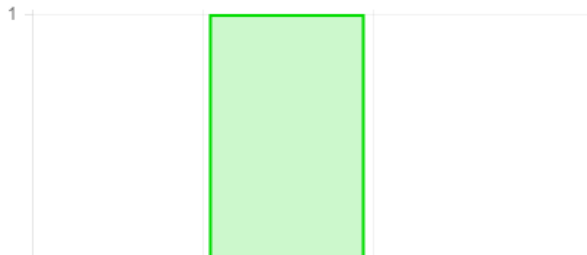
Questions asked:

- Esitiedot

Lecture wall



Find answers

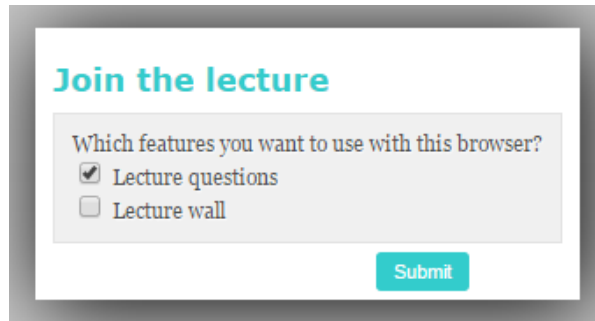
UserName **Esitiedot**

Kuva 3.14: Luennon vastausten ja luentoseinän yhteenveto.

3.7. Oppilaan käyttöliittymä

Oppilaalle sivuvalikko on muuten samanlainen kuin opettajallakin, mutta siinä ei näy kysymyskohtaa (katso kuva 3.2 C), eikä myöskään luennolla olijoita (katso kuva 3.2 D). Oppilas ei voi muokata tai luoda luentoja, joten ylävalikosta häneltä puuttuvat luennon luomis- ja muokkauspainikkeet.

Kuvan 3.15 ikkuna tulee näkyviin, kun oppilas painaa **luennolle liittymispainiketta** (katso kuva 3.9 A). Ikkunassa kysytään, haluaako oppilas luentokysymykset ja luentoseinän näkyviin, koska oppilas saattaa lukea samaa materiaalia kahdella eri laitteella ja haluaa kysymykset vain toiseen.



Kuva 3.15: Luennolle liittymisikkuna.

Luennolta poistuminen onnistuu vastaavasti kuin opettajalla ylävalikosta painamalla painiketta A kuvassa 3.10.

Menneiden luentojen tarkastelu onnistuu kuten opettajallakin kuvassa 3.14.

4. Kysymysten käyttöliittymä

Luvussa esitellään sovelluksen käyttöliittymän toiminnallisuuksista kysymyksen luonti sekä kysymyksen kysymistä ja siihen vastaamista.

4.1. Kysymyksen luominen

Kuvan 4.1 valikko ilmestyy käyttöliittymään, kun opettajan oikeudet dokumenttiin omaava käyttäjä klikkaa haluttua tekstikappaletta kaksi kertaa.



Kuva 4.1: Kappalevalikko.

Jos kuvan 4.1 valikosta valitsee kysymyksen luomisen, ilmestyy kuvan 4.2 mukainen kysymyksen luomisikkuna. Lomake koostuu kysymyksen otsikosta, itse kysymyksestä, kysymystyypistä ja kysymyksen keustosta.

Kuva 4.2: Kysymyksen luonti-ikkuna.

4.2. Kysymyksen vastausvaihtoehdot

Kuvassa 4.3 esitetään kysymyksen luontilomakkeeseen ilmestyvät vastausvaihtoehdot, jos kysymystyypiksi valitaan monivalintakysymys vastaustyyppinä radiopainike. Vastausvaihtoehtojen oikealla puolella oleviin tekstikenttiin kirjoitetaan kyseisestä vastauksesta saatavat pisteet, jos ne halutaan pisteyttää. Uuden vastausvaihtoehdon voi lisätä painamalla +-painiketta.

Question type: Multiple choice (radio button) ▼

<input type="checkbox"/>	Java	<input type="radio"/>	0
<input type="checkbox"/>	Python	<input type="radio"/>	0
<input type="checkbox"/>	C#	<input type="radio"/>	0
<input type="checkbox"/>			

Kuva 4.3: Monivalintakysymyksen (radiopainike) vastausvaihtoehdot.

Kuvassa 4.4 esitetään kysymyksen luontilomakkeen vastausvaihtoehdot, kun kysymystyyppiä valitaan monivalintakysymys vastaustyyppinä valintaruudut.

Question type: Multiple choice (checkbox) ▼

<input type="checkbox"/>	Java	<input type="checkbox"/>	0
<input type="checkbox"/>	Python	<input type="checkbox"/>	0
<input type="checkbox"/>	C#	<input type="checkbox"/>	0
<input type="checkbox"/>			

Kuva 4.4: Monivalintakysymyksen (valintaruutu) vastausvaihtoehdot.

Kuvassa 4.5 esitetään kysymyksen luontilomakkeen vastausvaihtoehdot, kun kysymystyyppiä valitaan True-False-väittämät.

Question type: True/False

	True	False
<input type="checkbox"/> + <input type="checkbox"/> - Ohjelmassa on neljä lausetta	<input type="checkbox"/> <input type="text" value="2"/>	<input type="checkbox"/> <input type="text" value="0"/>
<input type="checkbox"/> + <input type="checkbox"/> - Luokan nimen saa valita itse	<input type="checkbox"/> <input type="text" value="0"/>	<input type="checkbox"/> <input type="text" value="2"/>
<input type="checkbox"/> + <input type="checkbox"/> - Ohjelmassa on yksi pääohjelma ja kaksi aliohjelmaa	<input type="checkbox"/> <input type="text" value="3"/>	<input type="checkbox"/> <input type="text" value="-1"/>
+ (add button)		

Kuva 4.5: True-False-väittämien vastausvaihtoehdot.

Kuvassa 4.6 esitetään kysymyksen luontilomakkeen vastausvaihtoehdot, kun kysymystyyppiä valitaan matriisikysymys ja vastaustyyppiä valitaan tekstikentät.

Question type: Many rows and columns

Answer type: Text area

	+ - + -		
		<input type="checkbox"/> + <input type="checkbox"/> - Kielet joista haluaisin oppia lisää.	<input type="checkbox"/> + <input type="checkbox"/> - Kielet joita en haluaisi opiskella.
<input type="checkbox"/> + <input type="checkbox"/> - Kielet joita osaan.		<input type="text"/>	<input type="text"/>
<input type="checkbox"/> + <input type="checkbox"/> - Kielet joita en osaa.		<input type="text"/>	<input type="text"/>
+ (add button)			

Kuva 4.6: Matriisikysymys vastaustyyppinä tekstikentät.

Kuvassa 4.7 esitetään kysymyksen luontilomakkeen vastausvaihtoehdot, kun kysymystyyppiä valitaan matriisikysymys ja vastaustyyppiä valitaan radiopainikkeet.

Question type: Many rows and columns

Answer type: Radio Button horizontal

	+	-	+	-	+	-	
	Samaa mieltä.	En osaa sanoa.	Eri mieltä.	+			
+	Java on sopiva kieli tämän ongelman	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	3	0	-3
-	Python on sopiva kieli tämän ongelman	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	2	0	-2
+	+						

Kuva 4.7: Matriisikysymys vastaustyyppinä radiopainikkeet.

Kuvassa 4.8 esitetään kysymyksen luontilomakkeen vastausvaihtoehdot, kun kysymystyyppiä valitaan matriisikysymys ja vastaustyyppiä valitaan valintaruudut.

Question type: Many rows and columns ▾

Answer type: Checkbox ▾

	Osaan	Minulla on työkokemusta	En osaa	
				+
C#	pts	pts	pts	
Java	pts	pts	pts	
Python	pts	pts	pts	
				+

Kuva 4.8: Matriisikysymys vastaustyyppinä valintaruudut.

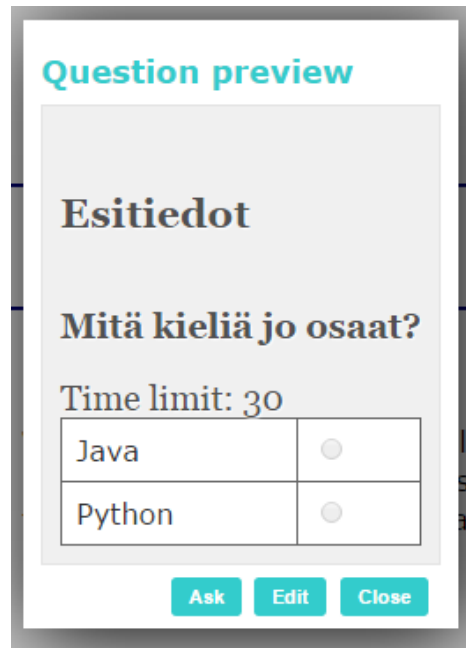
4.3. Kysymyksen kysyminen

Kysymyksen kysyminen aloitetaan klikkaamalla kysymysmerkkikuvaketta kappaleen alla (katso kuva 4.9). Viemällä kursorin kysymysmerkin päälle kysymyksen otsikko tulee näkyviin.

Vaikka kurssi onk
minkä ohjelmointil
toki tehdä myös r
?

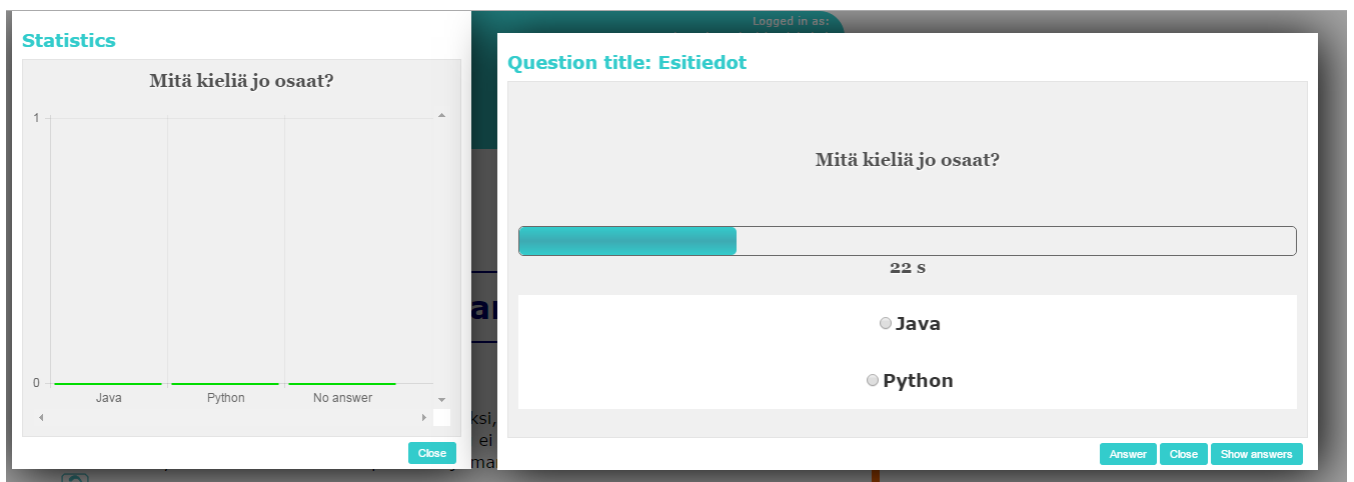
Kuva 4.9: Kysymyksen kysymiskuvake kappaleen alla.

Klikattuasi kysymysmerkkikuvaketta ilmestyy kuvan 4.10 kysymyksen esikatseluikkuna, jossa voit muokata kysymystä, kysyä kysymyksen tai sulkea esikatseluikkunan.



Kuva 4.10: Kysymyksen esikatseluikkuna.

Kuvassa 4.11 näkyy opettajan käyttöliittymän tilanne, kun kysymys on kysytty ja vastausaika on käynnissä.



Kuva 4.11: Kysymyksen käynnissäolon ikkuna.

Vastausajan umpeuduttua ilmestyy kuvassa 4.12 näkyvä teksti. Vastausaika ei ole siltikään loppunut, vaan se loppuu vasta, kun opettaja painaa vastaus- tai suljepainiketta.

Question title: Esitiedot

Mitä kieliä jo osaat?

Time's up

Java

Python

Answer Close Show answers

Kuva 4.12: Kysymysikkuna, kun vastausaika on kulunut umpeen.

4.4. Kysymyksen muokkaaminen ja poistaminen

Kysymyksen muokkaaminen onnistuu painamalla kuvan 4.10 kysymyksen esikatseluikkunassa olevaa muokkauspainiketta. Kuvassa 4.13 näkyy kysymyksen muokkausikkuna, joka vastaa kuvissa 4.2-4.8 esitettyjä kysymyksen luomisikkunoita.

Create question

Question title:

Question:

Question type:

<input type="checkbox"/>	<input type="text" value="Java"/>	<input type="checkbox"/>	<input type="text" value="pts"/>
<input type="checkbox"/>	<input type="text" value="Python"/>	<input type="checkbox"/>	<input type="text" value="pts"/>
<input type="text" value="+"/>			

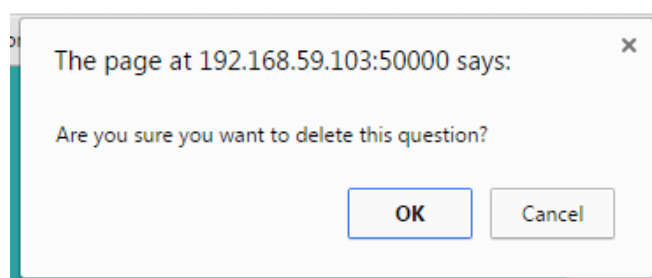
Set question timing:

Starting time: Date: Time: :

Duration: h m s

Kuva 4.13: Kysymyksen muokkausikkuna.

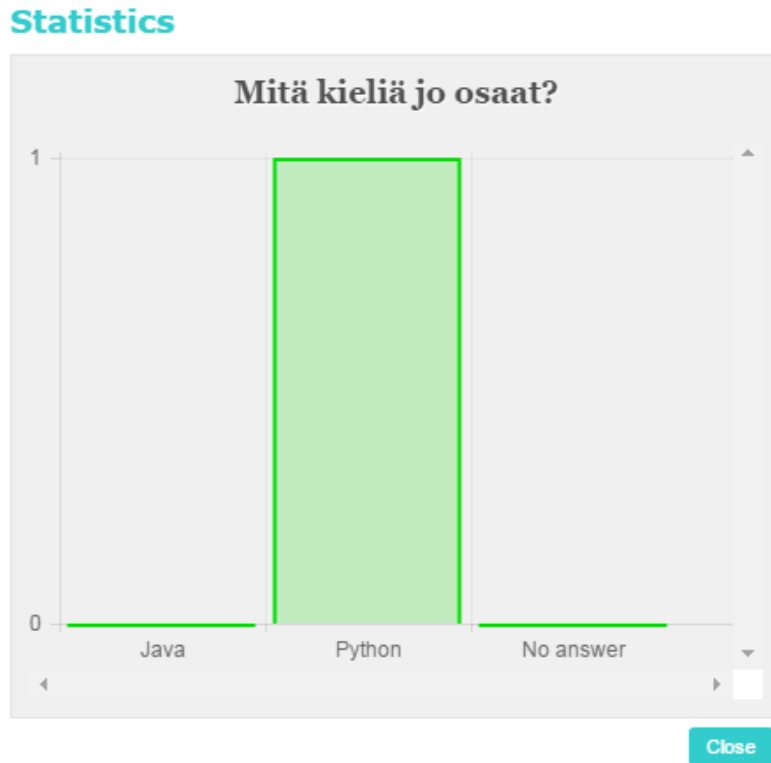
Kuvan 4.13 poistopainikkeesta avataan kuvan 4.14 poiston varmistusikkuna. Ikkuna saattaa näyttää erilaiselta eri selaimissa.



Kuva 4.14: Kysymyksen poistamisen varmistusikkuna.

4.5. Tulosten esitys

Kuvan 4.15 kysymyksen vastausjakaumaikkunan opettaja saa näkyviin kysymyksen aikana ja kysymisen jälkeen. Kysymisen aikana sen saa näkyviin (katso kuva 4.11) *Show answers* -painikkeella. Kysymisen jälkeen tilastoja voi tarkastella luennon yhteenvetosivulta (katso kuva 3.14).



Kuva 4.15: Kysymyksen vastausjakauma.

4.6. Oppilaan käyttöliittymä

Kysymysten osalta oppilaan käyttöliittymä painottuu lähinnä kysymyksiin vastaamiseen. Kuvassa 4.9 näkyvät kysymyskuvakkeet eivät näy opiskelijoille, kun he lukevat luentomateriaalia.

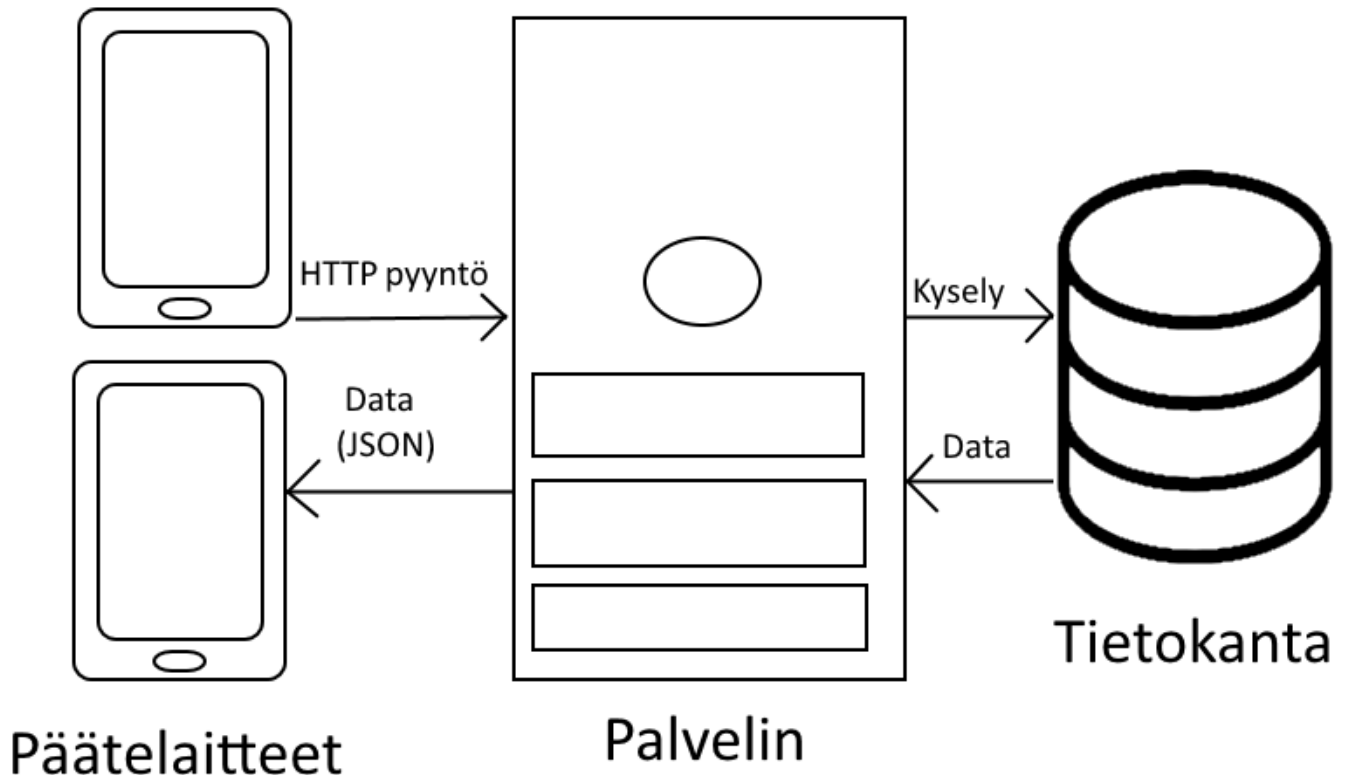
Oppilaalla kysymykseen vastaaminen onnistuu samalla ikkunalla kuin opettajallakin (katso kuva 4.11), mutta siinä ei näy tilastoja, eikä vastausten näyttämispainiketta.

5. Sovelluksen kokonaisrakenne ja rajapinnat

Luvussa kuvataan sovelluksen kokonaisrakennetta sekä käytettyjä ulkopuolisia komponentteja. Luvussa käydään myös lävitse kaksi esimerkkiä käyttöliittymän sekä palvelimen välisestä kommunikaatiosta.

5.1 Sovelluksen kokonaisrakenne

TIM-järjestelmä jakautuu kuvan 5.1 mukaisesti päätelaitteen asiakassovellukseen, sekä palvelinlaitteen palvelinsovellukseen ja tietokantaan.



Kuva 5.1: TIM -järjestelmän kokonaisrakenne.

Päätelaitteilla käytetään WWW-selaimessa pyörivää sovellusta, joka toimii TIMin käyttöliittymänä. Käyttöliittymän tarjoamat tiedot ja toiminnot poikkeavat opettajan ja opiskelijan käyttäjärooleilla.

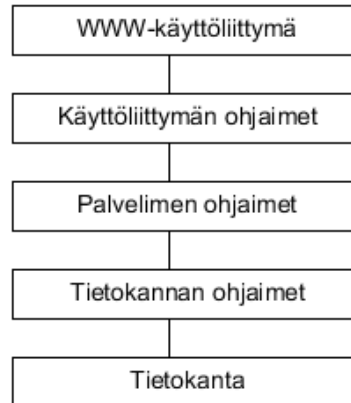
Palvelinsovellus suorittaa toiminnot sekä tekee pyyntöjä tietokantaan ja vastaanottaa dataa tietokannalta. Palvelin ottaa vastaan HTTP-pyyntöjä asiakassovelluksilta ja vastaa niihin lähettämällä JSON-muodossa olevaa dataa päätelaitteille.

Tietokanta sisältää järjestelmän varastoimat tiedot. Tarkemmat tiedot löytyvät luvusta 6.

Yleinen käyttötapaus alkaa siitä, että käyttäjä navigoi WWW-selaimella käyttämään sovellusta. Tämän jälkeen hän suorittaa käyttöliittymässä jonkun toiminnon, jolloin selain lähettää palvelimelle toimintoon liittyvät tiedot HTTP-pyyntöä. Palvelin tarkastaa, että pyyntö on hyväksyttävä sekä tarvittaessa suorittaa kyselyn tietokantaan. Toteutettuaan tämän, palvelin vastaa käyttäjällä lähettämällä vastauksen JSON-muodossa. Käyttäjän selaimessa pyörivä sovellus käsittelee saamansa vastauksen ja esittää sen käyttäjälle.

5.2 Olennaisimmat luokkarakenteet

TIM-järjestelmän olennaisimmat luokkarakenteet ovat esitelty kuvassa 5.2.



Kuva 5.2: TIM-järjestelmän olennaisimmat luokkarakenteet.

Kuvan 5.2 luokkarakenteet ovat seuraavat:

- **WWW-käyttöliittymä** esittää haluttuja tietoja käyttäjälle sekä mahdollistaa käyttäjällä syötteiden antamisen. WWW-käyttöliittymään kuuluvat `html`- ja `css`-päätteiset tiedostot.
- **Käyttöliittymän ohjaimet** tarkistavat käyttäjän syötteet sekä kommunikoivat palvelimen kanssa HTML-pyyntöjen välityksellä. Käyttöliittymän ohjaimet koostuvat `js`-päätteisistä tiedostotoista.
- **Palvelimen ohjaimet** varmistavat, että käyttäjällä on oikeus toteuttaa HTML-pyyntöjen laukaisema toiminto sekä kutsuvat tietokannan ohjaimissa olevia metodeja. Palvelimen ohjaimet sijaitsevat `tim.py`-tiedostossa.
- **Tietokannan ohjaimet** suorittavat vaadittavat pyynnöt tietokantaan. Tietokannan ohjaimet sijaitsevat `py`-päätteisissä tiedostoissa.

Timppa-projektissa kehitettyjen olennaisimpien luokkien kuvaukset ja niiden yhteydet ovat seuraavat:

- **tim.py**-tiedosto sisältää metodit, joita kutsumalla voidaan esim. lisätä kysymys. Itse metodi tosin kutsuu oikeaa tietokannan ohjainta, mutta `tim.py` toimii välittäjänä käyttöliittymän ja tietokannan välissä.
- **static/css/**-kansio pitää sisällään kaikki järjestelmän tyylitiedostot.
- **static/templates/**-kansio sisältää HTML-sivuja, joita kutsutaan erinäisistä ohjaimista.
- **templates/**-kansio sisältää lisää HTML-sivuja, joita kutsutaan erinäisistä ohjaimista.
- **timdb/**-kansio sisältää tietokannan ohjaimet, joiden avulla tehdään pyyntöjä ja vastaanotetaan vastauksia tietokannalta.

Static/scripts/ -kansiossa ovat seuraavat luokat ja kansiot:

- **controllers/**-kansio sisältää käyttöliittymän ohjaimet JavaScript-tiedostoina. Kansiossa sijaitsevat kysymyksiin vastaamisen, luennon tietojen näyttämisen, kysymyksen tietojen näyttämisen ja kysymyksen tilastojen näyttämisen ohjaimet.
- **directives/**-kansio sisältää erinäisiä tiedostoja, joissa määritellään kysymyksen vastaamisikkunan, pop-up-ikkunoiden ja tilastojen luonti.
- **createLectureCtrl.js**-tiedostoa käytetään luennon luomiseen ja luennon

luomislomakkeeseen syötettyjen tietojen tarkastamiseen.

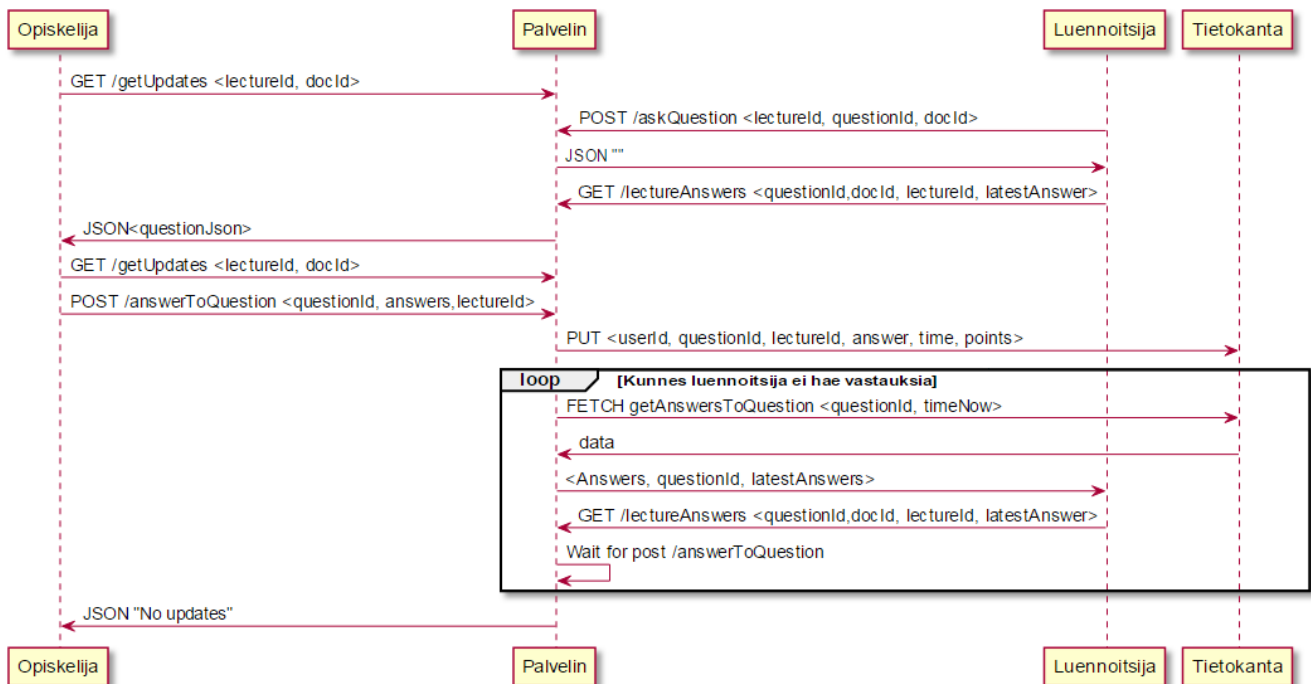
- **lectureController.js**-tiedostoa käytetään luennon ollessa käynnissä. Se sisältää mm. metodit, jolla luentoseinän sisältöä pystyy säätämään (näkykö nimi ja kellonaika).
- **sidebarMenuCtrl.js**- ja **smallMenuCtrl.js** -tiedostoja käytetään sivuvalikon luomiseen sekä niillä voidaan lisätä ja poistaa ikoneita sivuvalikosta.
- **view_html.js**-tiedosto on käyttöliittymä ohjainten ns. pääluokka.

Kaikki yllämainitut tiedostot ja kansiot sijaitsevat **timApp**-kansiossa.

5.3 Kysymysten kysymisen long-poll

Oletuksena kuvan 5.3 sekvenssikaaviossa on se, että luento on käynnissä ja sekä luennoitsija että opiskelija ovat luennolla. Luennoitsijalla on myös käytössä vastausten hakeminen. Kaaviota on myös yksinkertaistettu jättämällä pois viestien hakeminen, joka on osa HTTP-kutsua `getUpdates`.

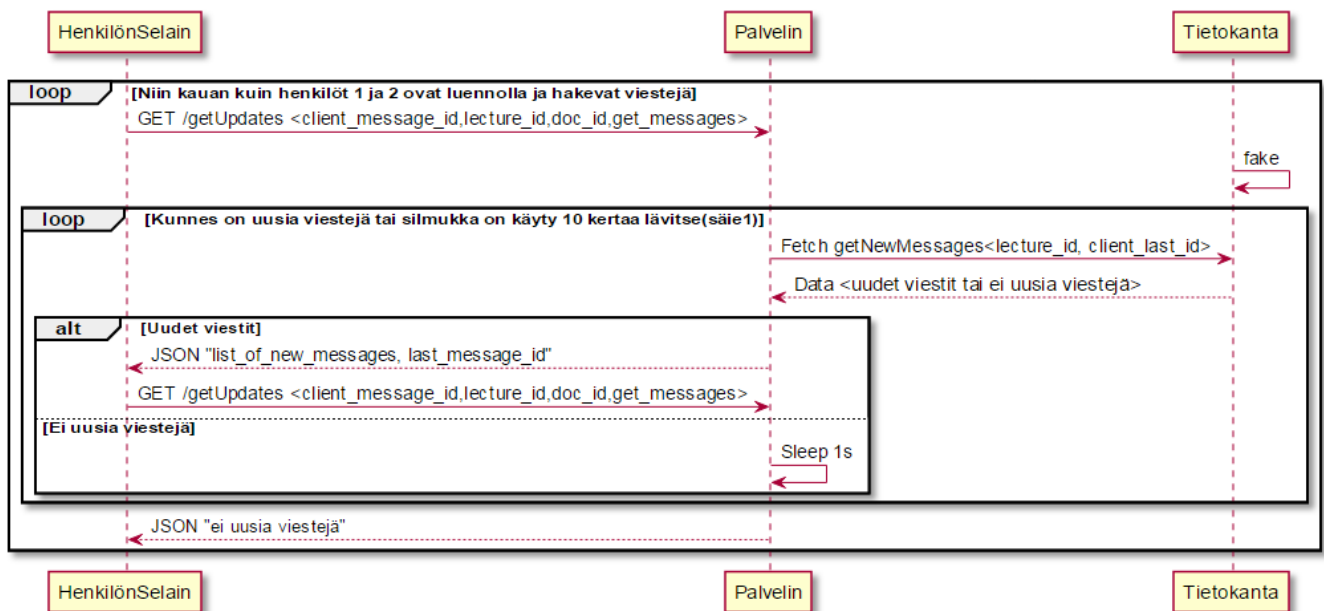
Kysymysten kysymisen long-poll alkaa siitä, että luennoitsija kysyy kysymyksen selaimessaan. Heti kysymisen jälkeen opettajan käyttöliittymä lähettää uuden pyynnön, jonka avulla hän hakee opiskelijoiden vastaukset selaimelleen. Tämän jälkeen palvelin hakee kysymyksen tietokannasta, sekä lähettää sen opiskelijoille. Opiskelijat saavat kysymyksen `getUpdates`-metodiin vastauksena. Tämän jälkeen he vastaavat kysymykseensä selaimellaan, joka lähettää vastauksen palvelimelle. Palvelin tallentaa vastauksen tietokantaan ja lähettää sen opettajalle. Opettajan selain on silmukassa, joka pyytää uudet vastaukset aina entiset saatuaan. Tämä silmukka loppuu sitten, kun opettaja ei enää hae vastauksia.



Kuva 5.3: Kysymyksen kysymisen long-poll sekvenssikaaviona.

5.4. Luentoseinän viestien hakemisen long-poll

Kuvassa 5.4 on selitetty, kuinka luentoseinän viestien hakeminen toimii. Alkutilanteessa molemmat käyttäjät ovat samalla luennolla, ja heillä on seinäviestien hakeminen asetettuna päälle. Molemmat henkilöt lähettävät `getUpdates`-metodin palvelimelle. Jos joku käyttäjistä lähettää seinälle viestin, huomaa palvelin tämän ja lähettää kaikki uudet viestit molemmille käyttäjille. Vaihtoehtoisesti, jos kukaan ei lähetä viestiä kymmenen sekunnin aikavälillä, molempien `getUpdates`-pyynnöt saavat vastaukseksi tiedon siitä, että ei ole tullut uusia viestejä. Tämän jälkeen käyttäjien selaimet lähettävät uudet `getUpdates`-pyynnöt, mikäli henkilöt ovat vielä luennolla.



Kuva 5.4: Luentoseinän viestien hakemisen long-poll sekvenssikaaviona.

5.5 Ulkopuoliset komponentit

Tietojärjestelmä hyödyntää seuraavia kirjastoja ja sovelluskehyskiä:

- **AngularJS** on JavaScript-sovelluskehys, jonka tärkein ominaisuus sovelluksen kannalta on kaksisuuntainen datakytkentä [9].
- **Chart.js** on JavaScript-kirjasto, jonka avulla voi piirtää diagrammeja [10].
- **jQuery** on JavaScript-kirjasto, joka yksinkertaistaa JavaScriptin perustoiminnallisuuden käyttämistä [11]. Sovelluksessa tätä hyödynnetään mm. päivämäärän valinnassa luentoa luodessa.

6. Tietokannan rakenne

Luvussa kuvataan jo ennen projektia sovelluksessa olleita tauluja pintapuolisesti. Projektiryhmän luomat tietokannan taulut ja kentät kuvataan tarkemmin.

6.1 Muuttumattomina säilyneet taulut

Timppa-projektissa ei muokattu seuraavia TIM-järjestelmän tauluja:

- **Answer**-tauluun tallennetaan luentomateriaalin kysymysten oppilaiden vastaukset.
- **AnswerTag**-taulu sisältää luentomateriaalin kysymysten vastausten selitykset.
- **Block**-taulu sisältää viittaukset dokumentteihin.
- **BlockEditAccess**-taulu sisältää tiedot käyttäjistä, joilla on muokkausoikeudet tiettyihin dokumenttiin.
- **BlockRelation**-taulu sisältää kansiorakenteen ja tiedon siitä, mihin kansioon kukin dokumentti kuuluu.
- **BlockViewAccess**-taulu sisältää tiedot käyttäjistä, joilla on katseluoikeudet tiettyihin dokumenttiin.
- **NewUser**-taulua käytetään uuden käyttäjän luomiseen.
- **ParMappings**-taulua käytetään kappaleiden sijoitteluun.
- **ReadParagraphs**-taulu liittää käyttäjäryhmät ja luetut kappaleet.
- **User**-taulu sisältää sovelluksen käyttäjät.
- **UserAnswer**-taulu liittää käyttäjät luentomateriaalin vastauksiin.
- **UserGroup**-taulu sisältää sovelluksen käyttäjäryhmät.
- **UserGroupMember**-taulu liittää käyttäjät ja käyttäjäryhmät.
- **UserNotes**-tauluun tallennetaan käyttäjien tekemät huomiot ja kommentit.

6.2 Projektissa lisätyt taulut

Lecture-taulu sisältää luennot sisältäen seuraavat kentät:

- **lecture_id** luennon yksilöivä tunniste,
- **lecture_code** luennon nimi,
- **doc_id** yksilöi dokumentin, johon luento liittyy,
- **lecturer** yksilöi luennoitsijan, joka on luonut luennon,

- **start_time** aloitusaika,
- **end_time** lopetusaika ja
- **password** salasana.

LectureAnswer-taulu sisältää luennoilla esitettyihin kysymyksiin tulleet vastaukset sisältäen seuraavat kentät:

- **answer_id** vastauksen yksilöivä tunniste,
- **user_id** vastaajan tunniste,
- **question_id** yksilöi kysymyksen, johon vastaus liittyy,
- **lecture_id** yksilöi luennon, jolla kysymys on esitetty,
- **answer** vastaus tekstinä,
- **answered_on** vastauksen saapumishetki palvelimelle ja
- **points** pisteet vastauksesta.

LectureUsers-taulu liittää käyttäjät luentoihin sisältäen seuraavat kentät:

- **lecture_id** luennon yksilöivä tunniste ja
- **user_id** käyttäjän yksilöivä tunniste.

Message-taulu sisältää kaikki sovelluksessa lähetetyt luentoseinäviestit sisältäen seuraavat kentät:

- **msg_id** viestin yksilöivä tunniste,
- **lecture_id** luennon yksilöivä tunniste,
- **user_id** lähettäjän yksilöivä tunniste,
- **message** viesti ja
- **timestamp** aikaleima.

Question-taulu sisältää kaikki luodut kysymykset sisältäen seuraavat kentät:

- **question_id** kysymyksen yksilöivä tunniste,
- **doc_id** dokumentin yksilöivä tunniste,
- **par_index** yksilöi kappaleen, johon kysymys on liitetty,
- **question_title** kysymyksen otsikko,
- **answer** kysymyksen vastaus ja
- **questionJson** kysymyksen tiedot JSON-muodossa.

7. Tavoitteiden toteutuminen

Luvussa kuvataan vaatimusten toteutumista sekä heikkoja ja puutteellisia toteutusratkaisuja. Sovelluksen vaatimukset ja niiden toteutuminen on kuvattu vaatimusmäärittelyssä [6].

7.1 Vaatimusten toteutuminen

Vaatimusten toteutuminen vaatimuskohtaisesti on kuvattu vaatimusmäärittelyssä [6]. Pakollisista 24 vaatimuksesta toteutui kokonaan 21. Tärkeistä 19 vaatimuksesta toteutui 14. Mahdollisista 6 vaatimuksesta toteutui yksi vaatimus. Ainoa ideatason vaatimus ei toteutunut. Vaatimusmäärittelyyn ei kirjattu prioriteetilla *Ei toteuteta* yhtään vaatimusta. Kaikki pakolliset toteutumattomat vaatimukset liittyivät kysymyksen luontiin.

7.2 Suoritetut testaukset ja niiden tulokset

Toukokuussa Matias Berg laati järjestelmätestaus suunnitelman [3] ja suoritti sitä noudattaen järjestelmätestaukset Google Chromella ja Internet Explorerilla. Testausten tulokset on raportoitu järjestelmätestausraporteissa [1] ja [2]. Google Chromella suoritetussa järjestelmätestauksessa [1] 80 testitapauksesta 63 havaittiin virheitä, kun taas Internet Explorerilla suoritetussa testauksessa [2] 83 testitapauksesta 45 havaittiin virheitä ja neljää ei voitu suorittaa.

Projektin aikana sovelluksen eri versioita koekäyttivät vastaava ohjaaja Jukka-Pekka Santanen sekä tilaajan edustajat Antti-Jussi Lakanen ja Vesa Lappalainen. Heidän palautteensa on saatavilla projektin sähköpostiarkistosta [7]. Jatkokehityksessä testaus tulee suorittaa useammalla selaimella ja päätelaitteella.

7.3 Heikot ja puutteelliset toteutusratkaisut

Projektin kehittämissä toiminnoissa TIM-järjestelmään on todettu olevan seuraavat heikot ja puutteelliset toteutusratkaisut:

- Kysymyksen luontilomakkeen tekstikenttien erikoismerkit hajottavat JSONin. Ne korvattiin HTML-tageilla, mikä saattaa aiheuttaa tietoturvariskejä.
- Kysymyksen kysyminen ja päättäminen ennen kuin aika on kulunut umpeen ja sen jälkeen toisen kysymyksen kysyminen aiheuttaa ongelmia kellon pysähtymisessä nollan kohdalla.
- Luentoseinän koko ja tyyli hajoavat, kun luentoseinän kokoa muokkaa tai seinän laittaa piiloon.
- Ensimmäisen siirron yhteydessä pop-up-ikkunat pomppaavat ikkunan oikeaan laitaan.
- Käytettävyyteen ei ryhmä ehtinyt paneutua kunnolla, joten se on otettu huomioon puutteellisesti.
- Lähdekoodissa on todennäköisesti myös useita refaktoroitavia kohtia.

7.4 Sovelluksen kehitysideat

Projektin kuluessa esitettiin seuraavat kehitysideat:

- Sivuvälikkoon pitäisi lisätä tulevien luentojen kohdalle linkki muokkauslomakkeeseen luennon yhteenvetosivun sijaan.
- Viestien ja kysymysten hakemiseen tulisi toteuttaa long-pollin lisäksi vaihtoehtoina normaali poll ja manuaalinen haku.
- Kysymysten poistamisen tulisi poistaa vain kysymysmerkki dokumentista, jos kysymykseen on vastauksia.
- Luentoseinä tulisi vaihtaa tekstialueesta johonkin muuhun muotoon, jotta omat viestit voisi tummentaa.
- ShowChart-direktiiviä tulisi muokata siten, että se loisi oman kanvaksensa.
- Tekstialuevastaukset tulisi näyttää tilastojen sijaan. Nykyinen toteutus kirjoittaa kanvakselle, eikä toimi hyvin useiden vastausten kanssa. Useiden vastausten lukeminen ja anyloisinti on vaikeaa nykyisellä tyyllillä.
- Yhden tekstilaatikon vastaus pitäisi lisätä kysymystyypiksi.

8. Yhteenveto

Timppa-projekti toteutti Sovellusprojekti-kurssilla keväällä ja kesällä 2015 Jyväskylän yliopiston tietotekniikan laitokselle muutoksia TIM-järjestelmään. Projektissa lisättiin TIM-järjestelmään opettajille mahdollisuus luoda luentoja, laatia ja kysyä kysymyksiä, sekä tarkastella vastausten jakaumia. Oppilas voi muutoksien ansiosta vastata kysymyksiin sekä kysyä ja keskustella luennon sisällöstä luentoseinällä.

Toteutut muutokset toimivat pääosin asetettujen vaatimusten mukaan, mutta joitain asioita jäi toteuttamatta. Toteutumattomista vaatimuksista suurin osa koski kysymyksen luontilomaketta. Järjestelmätestausta ei ehditty tekemään kattavasti projektin lopussa, joten jatkokehittäjien tulee testata järjestelmää perusteellisemmin. Projektissa kehitettyjä TIM-järjestelmään tehtyjä muutoksia tullaan jatkokehittämään ja lisäämään julkiseen TIM-järjestelmään jatkokehityksen jälkeen.

Lähteet

[1]	Matias Berg, Bek Eljurkaev, Minna Lehtomäki, Juhani Sihvonen ja Hannu Viinikainen, "Timppa-sovellusprojekti, Järjestelmätestausraportti #1", Jyväskylän yliopisto, tietotekniikan laitos, 2015.
[2]	Matias Berg, Bek Eljurkaev, Minna Lehtomäki, Juhani Sihvonen ja Hannu Viinikainen, "Timppa-sovellusprojekti, Järjestelmätestausraportti #2", Jyväskylän yliopisto, tietotekniikan laitos, 2015.
[3]	Matias Berg, Bek Eljurkaev, Minna Lehtomäki, Juhani Sihvonen ja Hannu Viinikainen, "Timppa-sovellusprojekti, Järjestelmätestaus suunnitelma", Jyväskylän yliopisto, tietotekniikan laitos, 2015.
[4]	Matias Berg, Bek Eljurkaev, Minna Lehtomäki, Juhani Sihvonen ja Hannu Viinikainen, "Timppa-sovellusprojekti, Projektiraportti", Jyväskylän yliopisto, tietotekniikan laitos, 2015.
[5]	Matias Berg, Bek Eljurkaev, Minna Lehtomäki, Juhani Sihvonen ja Hannu Viinikainen, "Timppa-sovellusprojekti, Projektisuunnitelma", Jyväskylän yliopisto, tietotekniikan laitos, 2015.
[6]	Matias Berg, Bek Eljurkaev, Minna Lehtomäki, Juhani Sihvonen ja Hannu Viinikainen, "Timppa-sovellusprojekti, Vaatimusmäärittely", Jyväskylän yliopisto, tietotekniikan laitos, 2015.
[7]	Timppa-projektiorganisaation sähköpostiarkisto, saatavilla HTML-muodossa URL: https://korppi.jyu.fi/kotka/servlet/list-archive/timppa/ , viitattu 3.9.2015.
[8]	Joel Kivelä, Erkki Koskenkorva, Mika Lehtinen, Oskari Leppäaho ja Petri Partanen, "Liikkuva-sovellusprojekti, Sovellusraportti", saatavilla PDF-muodossa URL: http://sovellusprojektit.it.jyu.fi/liikkuva/dokumentit/sovellusraportti/liikkuva_sovellusraportti_1.0.0.pdf , Jyväskylän yliopisto, tietotekniikan laitos, 9.6.2014.
[9]	Google, "What is Angular", saatavilla HTML-muodossa URL: https://docs.angularjs.org/guide/introduction , viitattu 26.8.2015.
[10]	Nick Downie, "Chart.js Documentation", saatavilla HTML-muodossa URL: http://www.chartjs.org/docs/ , viitattu 26.8.2015.
[11]	jQuery Foundation, "What is jQuery", saatavilla HTML-muodossa URL: https://jquery.com/ , viitattu 26.8.2015.