



Adrian Gashi, Yamir Haque, Leo Härkönen

## Palautepomppu -sovellus

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Tieto- ja viestintäteknologian tutkinto-ohjelma

Toteutusdokumentti

17.10.2023

# Sisällysluettelo

1	Johdanto	1
1.1	Sovelluksen esittely	1
2	Tuotteen vaatimukset	2
2.1	Tarve	2
2.2	Toimintaidea	2
2.3	Visiokuva	2
2.4	Toiminnallisia vaatimuksia	2
2.5	Laadullisia vaatimuksia	3
2.6	Määritelmät	3
3	Käyttäjäroolit ja käyttötapaukset	4
3.1	Käyttötapauskaavio	4
4	Ohjelmiston tietomalli	5
4.1	ER-kaavio	5
4.2	SQL-kaavio	7
5	Ohjelmiston rakenne	8
5.1	Luokkakaavio	8
5.2	Pakkauskaavio	10
5.3	Sijoittelukaavio	11
6	Ohjelmiston toiminta	12
6.1	Sekvenssikaavio	12
6.2	Aktiviteetikaavio	14
7	Kehitysprosessi ja kehitysvaiheen tekniikat	15
7.1	Projektinhallinta	15
7.2	Versionhallinta	15
7.3	Tietokanta	16
7.4	Testaus	16
7.5	Julkaisu	16
7.6	Dokumentointi	16

8	Yhteenveto	16
8.1	Jatkokehitysideat	17
8.1.1	Käyttäjäprofiilit	17
8.1.2	Anonyymiuden valinta kyselyä luodessa	17
8.1.3	Pelillistäminen	17
8.1.4	Palauteanalyysi	17
8.1.5	Käyttöliittymän parantaminen	18
8.2	Toteuttamiskelvottomaksi todetut kehitysideat	18
8.2.1	Tekstistä puheeksi	18
8.2.2	Ääniviesti	18
8.2.3	Integrointi	18
8.2.4	Viestintäkanavat	18

# 1 Johdanto

Tämä dokumentti kertoo Palautepomppu-sovelluksen kehityshankkeesta. Dokumentin tavoitteena on esitellä sovelluksen tarve, toimintaidea, vaatimukset ja toteutus.

Dokumentti on suunnattu kaikille, jotka ovat kiinnostuneita Palautepomppu-sovelluksesta tai sen kehittämisestä. Se tarjoaa kattavan kuvan sovelluksesta ja sen taustalla olevasta ajatuksesta.

## 1.1 Sovelluksen esittely

Palautepomppu on verkkosovellus, joka mahdollistaa alakoululaisille helpon ja monipuolisen tavan antaa palautetta opettajilleen. Sovelluksen kautta oppilaat voivat välittää tuntemuksiaan ja ajatuksiaan käyttäen erilaisia ilmaisutapoja kuten emojiä, piirtämisen tai kirjoituksen muodossa.

Sovelluksen avulla oppilaat voivat:

- Antaa palautetta oppitunneista, eri aktiviteeteista ja muista kouluun tai oppimiseen liittyvistä asioista.
- Ilmaista itseään luontevalla tavalla.
- Osallistua aktiivisesti opetuksen kehittämiseen.

Sovelluksen tavoitteena on luoda helppokäyttöinen ja hauska työkalu, joka kannustaa oppilaita antamaan palautetta.

## 2 Tuotteen vaatimukset

### 2.1 Tarve

Nykyaikaisessa koulutuksessa on yhä tärkeämpää ymmärtää oppilaiden tarpeet ja kokemukset. Opetus ei ole pelkästään tiedon jakamista, vaan keskeinen osa on myös vuorovaikutus oppilaiden kanssa ja heidän kokemustensa huomioiminen. Palautteiden kerääminen on keskeinen työkalu opettajille ympäri maailmaa.

### 2.2 Toimintaidea

Palautepomppu on verkkosovellus, joka mahdollistaa alakoululaisille helpon ja monipuolisen tavan antaa palautetta opettajilleen esimerkiksi oppitunneista, eri aktiviteeteista ja muista kouluun tai oppimiseen liittyvistä asioista. Sovelluksen kautta oppilaat voivat välittää tuntemuksiaan ja ajatuksiaan käyttäen erilaisia ilmaisutapoja kuten emojiä, piirtämisen tai kirjoituksen muodossa.

### 2.3 Visiokuva

Palautepomppu-sovelluksen visio on luoda helppokäyttöinen ja hauska työkalu, joka kannustaa oppilaita antamaan palautetta. Sovelluksen avulla oppilaat voivat ilmaista itseään luontevalla tavalla ja osallistua aktiivisesti opetuksen kehittämiseen.

### 2.4 Toiminnallisia vaatimuksia

- Sovelluksen tulee mahdollistaa palautteen antamisen erilaisilla tavoilla, kuten emojiä, piirtämisen ja kirjoituksen avulla.
- Sovelluksen tulee tarjota opettajille työkalut palautteiden hallintaan ja analysointiin.

- Sovelluksen tulee olla helppokäyttöinen ja käyttäjäystävällinen.

## 2.5 Laadullisia vaatimuksia

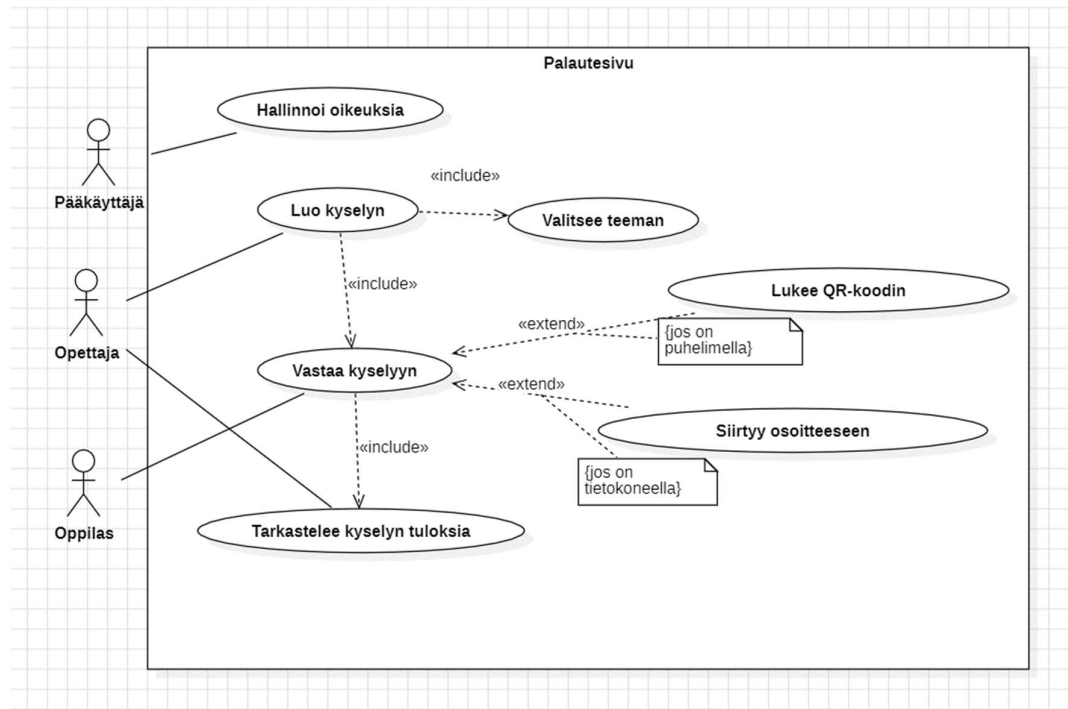
- Sovelluksen tulee olla turvallinen ja tietosuojan mukainen.
- Sovelluksen tulee olla esteetön.

## 2.6 Määritelmät

- Opettaja on henkilö, joka vastaa oppilaiden opettamisesta ja ohjaamisesta.
- Oppilas on henkilö, joka osallistuu kouluopetukseen.
- Alakoulu on oppilaitoksen taso, joka vastaa Suomen perusopetuksen ensimmäisiä neljää luokkaa.

### 3 Käyttäjäroolit ja käyttötapaukset

#### 3.1 Käyttötapauskaavio



Kuva 1: aktiviteettikaavio

Käyttötapauskaavion kuvaus

Käyttäjäroolit ja Käyttötapaukset

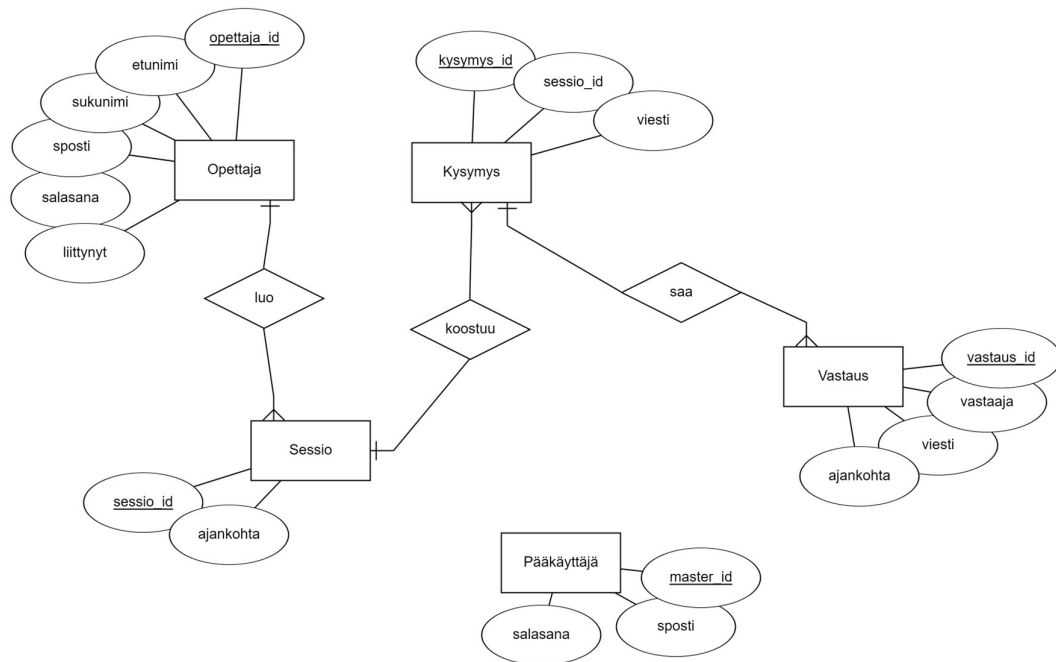
**Pääkäyttäjällä** on ylin oikeus hallita sovelluksen käyttöoikeuksia. Hän voi lisätä, muuttaa ja poistaa muita käyttäjiä (opettajia) sekä määrittää heidän oikeutensa.

**Opettaja** luo kyselyitä, valitsee niille teemat ja tarkastelee kyselyiden tuloksia. Opettajalla on oikeus nähdä vain omiin kyselyihin liittyvät tiedot.

**Oppilas** vastaa opettajan luomiin kyselyihin, joko lukemalla QR-koodin puhelimella tai siirtymällä URL-osoitteeseen tietokoneella.

## 4 Ohjelmiston tietomalli

### 4.1 ER-kaavio



Kuva 2: ER-kaavio

ER-kaavion kuvaus

#### 1. Entiteetti: Opettaja

- Kuvaus: Tämä entiteetti kuvaa koulun opettajaa, joka voi luoda useita sessioita.
- Attribuutit:
  - opettaja\_id (PK): Opettajan yksilöllinen tunniste.
  - etunimi: Opettajan etunimi.
  - sukunimi: Opettajan sukunimi.
  - sposti: Opettajan sähköpostiosoite.



- salasana: Opettajan salasana (salattu).
- liittynyt: Päivämäärä, jolloin opettaja rekisteröityi järjestelmään.

## 2. Entiteetti: Sessio

- Kuvaus: Tämä entiteetti kuvaa tiettyä palautteenantosessiota, jonka opettaja on luonut. Sessio voi koostua useista kysymyksistä.
- Attribuutit:
  - sessio\_id (PK): Session yksilöllinen tunniste.
  - ajankohta: Session luontipäivämäärä.

## 3. Entiteetti: Kysymys

- Kuvaus: Tämä entiteetti kuvaa yksittäistä kysymystä jossain tietyssä sessiossa. Jokaiseen kysymykseen voidaan antaa useita vastauksia.
- Attribuutit:
  - kysymys\_id (PK): Kysymyksen yksilöllinen tunniste.
  - viesti: Kysymyksen sisältö.

## 4. Entiteetti: Vastaus

- Kuvaus: Tämä entiteetti kuvaa yksittäistä vastausta johonkin kysymykseen.
- Attribuutit:
  - vastaus\_id (PK): Vastauksen yksilöllinen tunniste.
  - vastaaja: Vastaajan tunniste, joka ei kuitenkaan ole yksilöllinen, UUID.
  - viesti: Vastauksen sisältö.
  - ajankohta: Aikaleima, jolloin vastaus annettiin.

## 5. Entiteetti: Pääkäyttäjä

- Kuvaus: Järjestelmän pääkäyttäjä, jolla on ylempiä oikeuksia mutta ei yhdisty muihin entiteetteihin.
- Attribuutit:
  - master\_id (PK): Pääkäyttäjän yksilöllinen tunniste.
  - sposti: Pääkäyttäjän sähköpostiosoite.
  - salasana: Pääkäyttäjän salasana (salattu).

### Yhteystyypit (Relationships):

#### 1. Luo (Opettaja - Sessio)

Yhden suhde moneen -yhteys. Yksi opettaja voi luoda useita sessioita, mutta yksi sessio liittyy vain yhteen opettajaan.

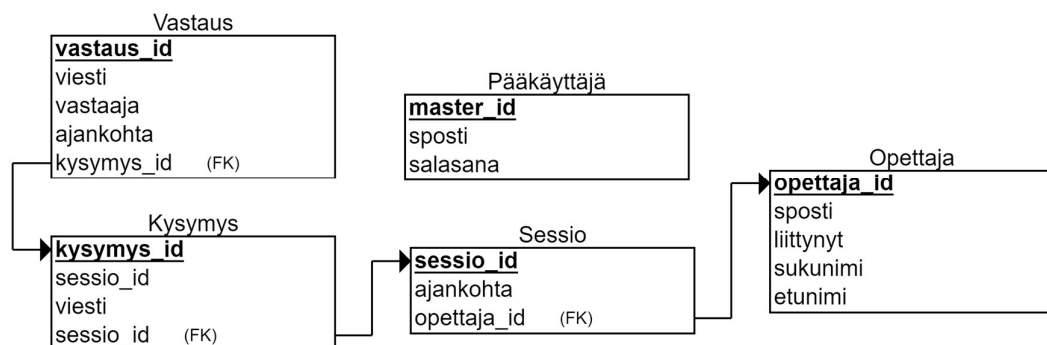
#### 2. Koostuu (Sessio - Kysymys)

Yhden suhde moneen -yhteys. Yksi sessio voi koostua useista kysymyksistä, mutta yksi kysymys kuuluu vain yhteen sessioon.

#### 3. Saa (Kysymys - Vastaus)

Yhden suhde moneen -yhteys. Yksi kysymys voi saada useita vastauksia, mutta yksi vastaus liittyy vain yhteen kysymykseen.

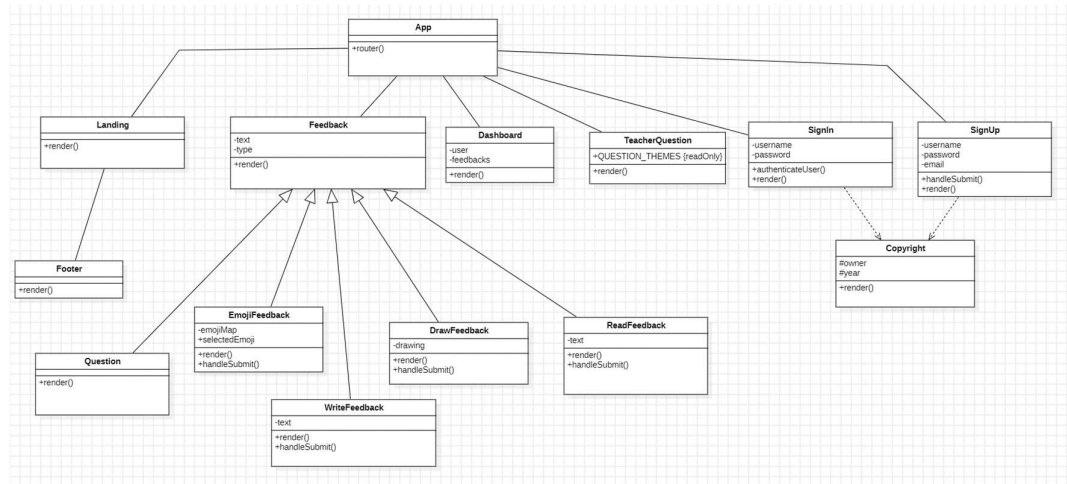
### 4.2 SQL-kaavio



Kuva 3: SQL-kaavio

## 5 Ohjelmiston rakenne

### 5.1 Luokkakaavio



Kuva 4: Luokkakaavio

Luokkakaavion kuvaus

#### 1. Sivut ja komponentit:

- App: Pääkomponentti, joka reitittää eri sivuille URL-osoitteen mukaan.
- Landing: Sovelluksen pääsivu, jota käytetään sovelluksen ominaisuuksien esittelyyn. Tämän sivun kautta pääsee kirjautumissivulle.
- SignIn ja SignUp: Sivut, jotka mahdollistavat opettajan ja pääkäyttäjän kirjautumisen ja rekisteröitymisen sovellukseen.
- Dashboard: Opettajan päänäkymä kirjautumisen jälkeen, josta hän voi tarkastella ja hallinnoida palautteita.
- Feedback: Oppilaalle näytettävä sivu, jonka kautta palaute annetaan.

- EmojiFeedback, DrawFeedback, WriteFeedback, ReadFeedback: Alakomponentteja, jota Feedback-sivu käyttää
- Footer: Landing-sivun alapalkissa näkyvä komponentti.
- TeacherQuestion: Komponentti, joka mahdollistaa opettajille kysymysten luomisen.

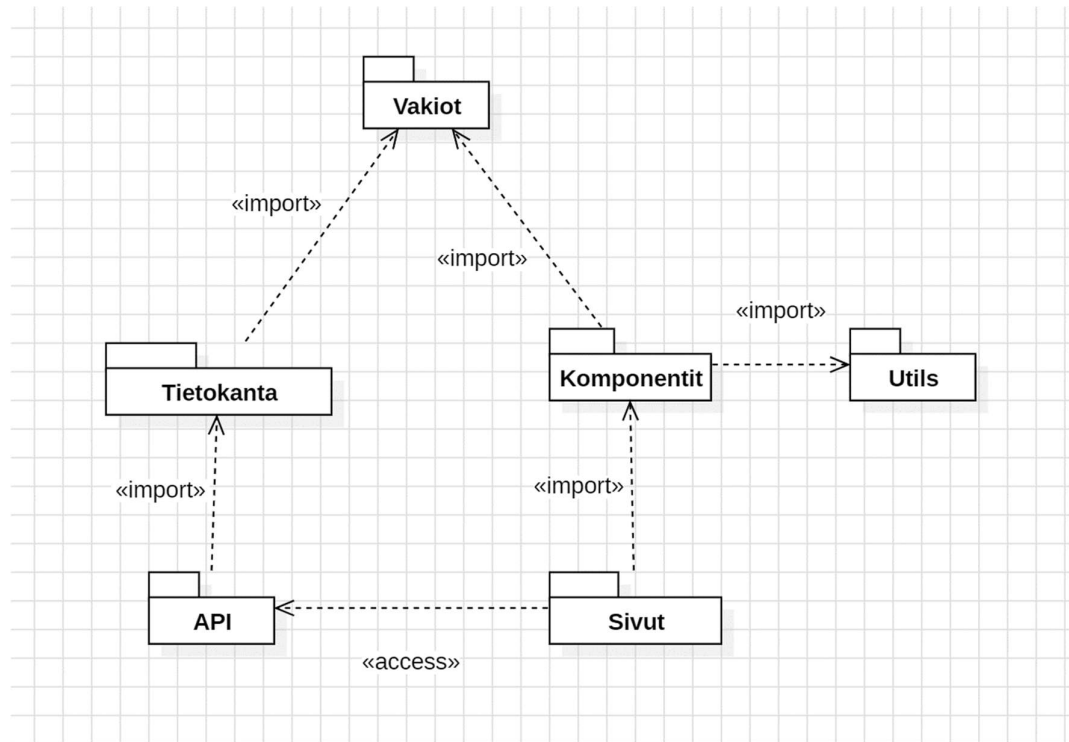
## **2. Suhteet:**

- Feedback komponentti käyttää EmojiFeedback, DrawFeedback, WriteFeedback ja ReadFeedback komponentteja.
- Landing-sivu käyttää Footer-komponenttia.
- SignIn- ja SignUp-sivut käyttää Copyright-komponenttia

## **3. Attribuutit ja Operaatiot:**

- Kaaviossa näkyvät komponenttien metodit esittävät sisältöä visuaalisesti ja toteuttavat lomakkeiden täyttöön liittyviä operaatioita.
- Attribuutit ovat pääosin komponenttien ja sivujen tilojen hallintaan liittyviä yksityiskohtia.

## 5.2 Pakkauskaavio

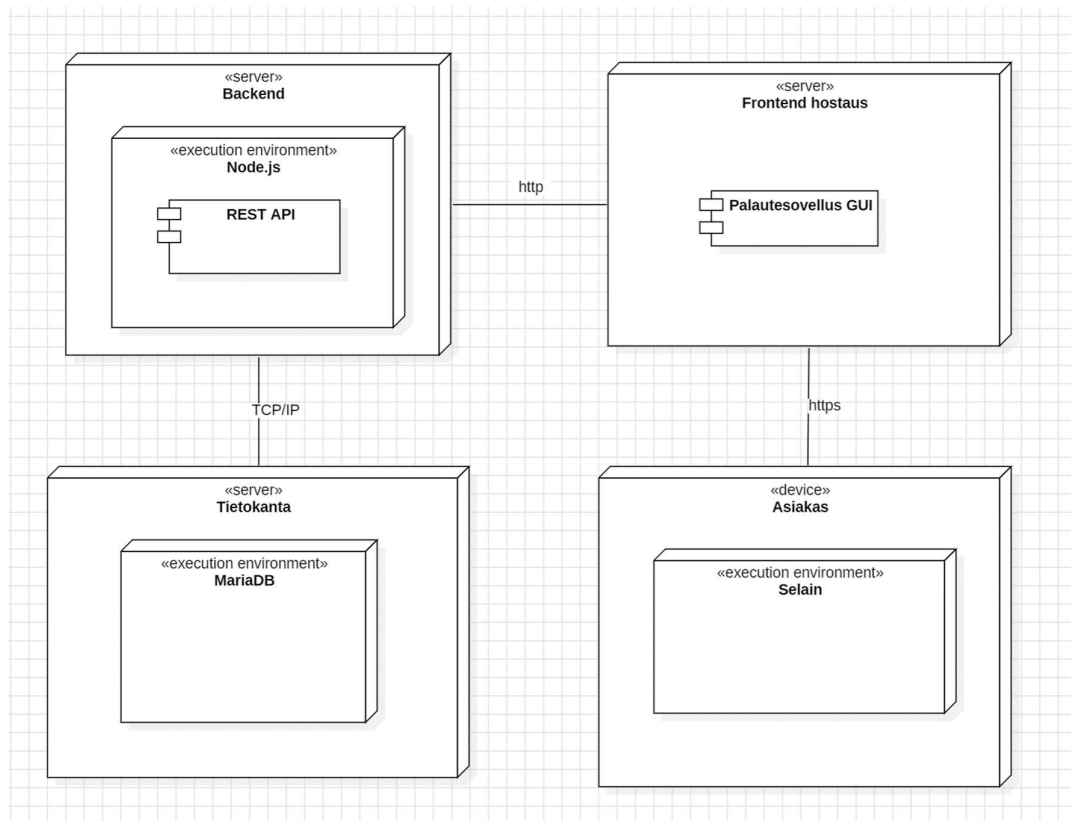


Kuva 5: Pakkauskaavio

Pakkauskaavion kuvaus

- Sivut koostuvat komponenteista ja ne tekevät API-pyyntöjä.
- Komponentit ovat pieniä elementtejä, joista käyttöliittymä koostuu.
- API käyttää tietokanta pakkauksen funktioita tiedon välittämiseen.
- Tietokantapakkaus sisältää funktioita tietokannan käyttämiseen.
- Pakkaukset Utils ja Vakiot sisältävät yleiskäyttöisiä funktioita ja muuttujia, joita voidaan käyttää eri puolella sovellusta.

### 5.3 Sijoittelukaavio



Kuva 6: Sijoittelukaavio

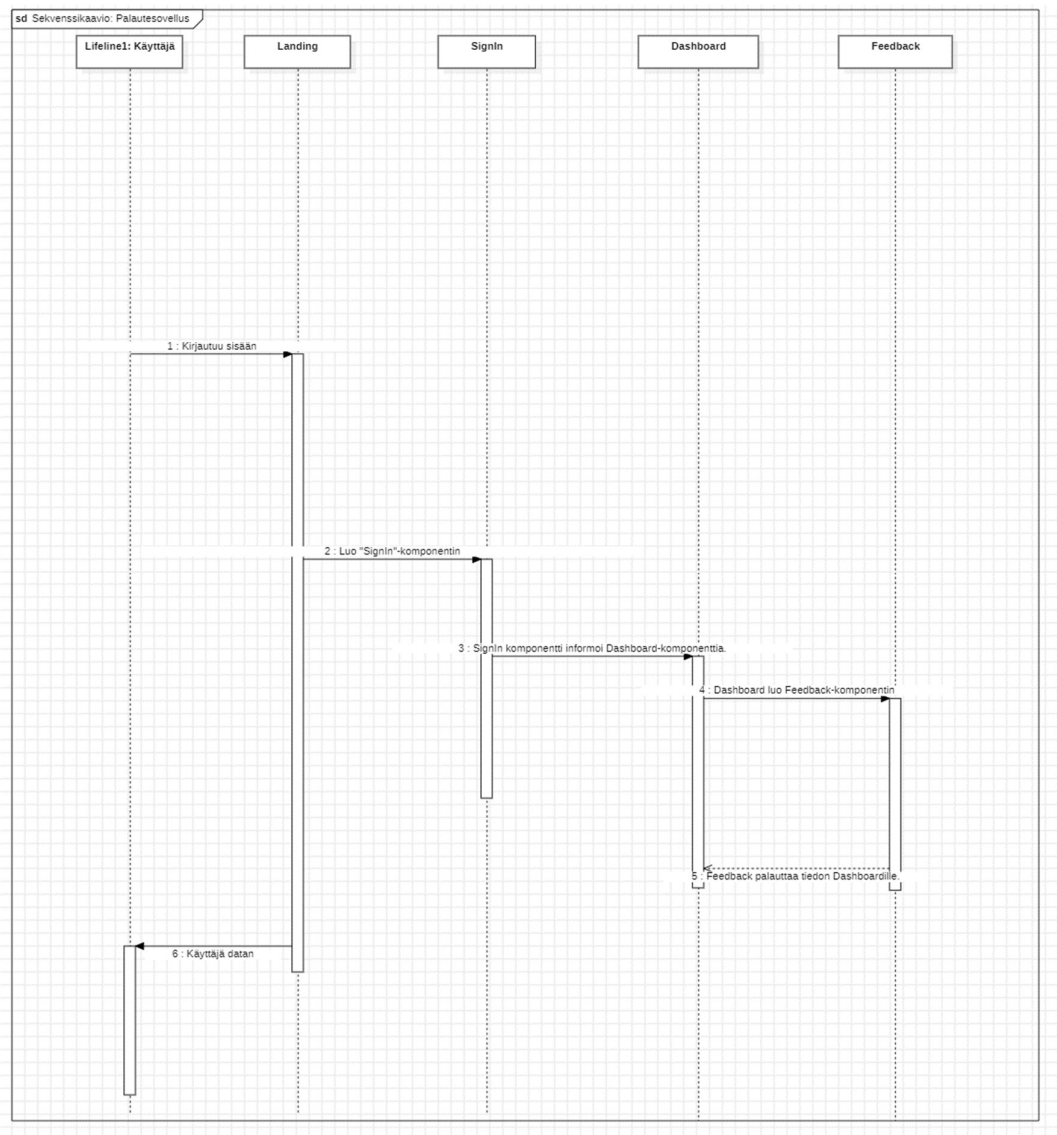
Sijoittelukaavion kuvaus

- **Tietokanta:** MariaDB -relaatiotietokantajärjestelmällä toteutettu tietokantaratkaisu.
- **Backend:** Palvelin, jossa REST API on toteutettu Node.js ympäristössä. API käyttää tietokantaa TCP/IP-yhteydellä.
- **Frontend hostaus:** Isännöi asiakkaalle toimitettavia web-tiedostoja (HTML, CSS, JS, kuvat). Asiakkaalle toimitetut tiedostot kutsuvat palvelinpuolen API:a http-yhteydellä.

- **Asiakas:** Loppukäyttäjä, joka pyytää web-tiedostoja https-yhteydellä.

## 6 Ohjelmiston toiminta

### 6.1 Sekvenssikaavio



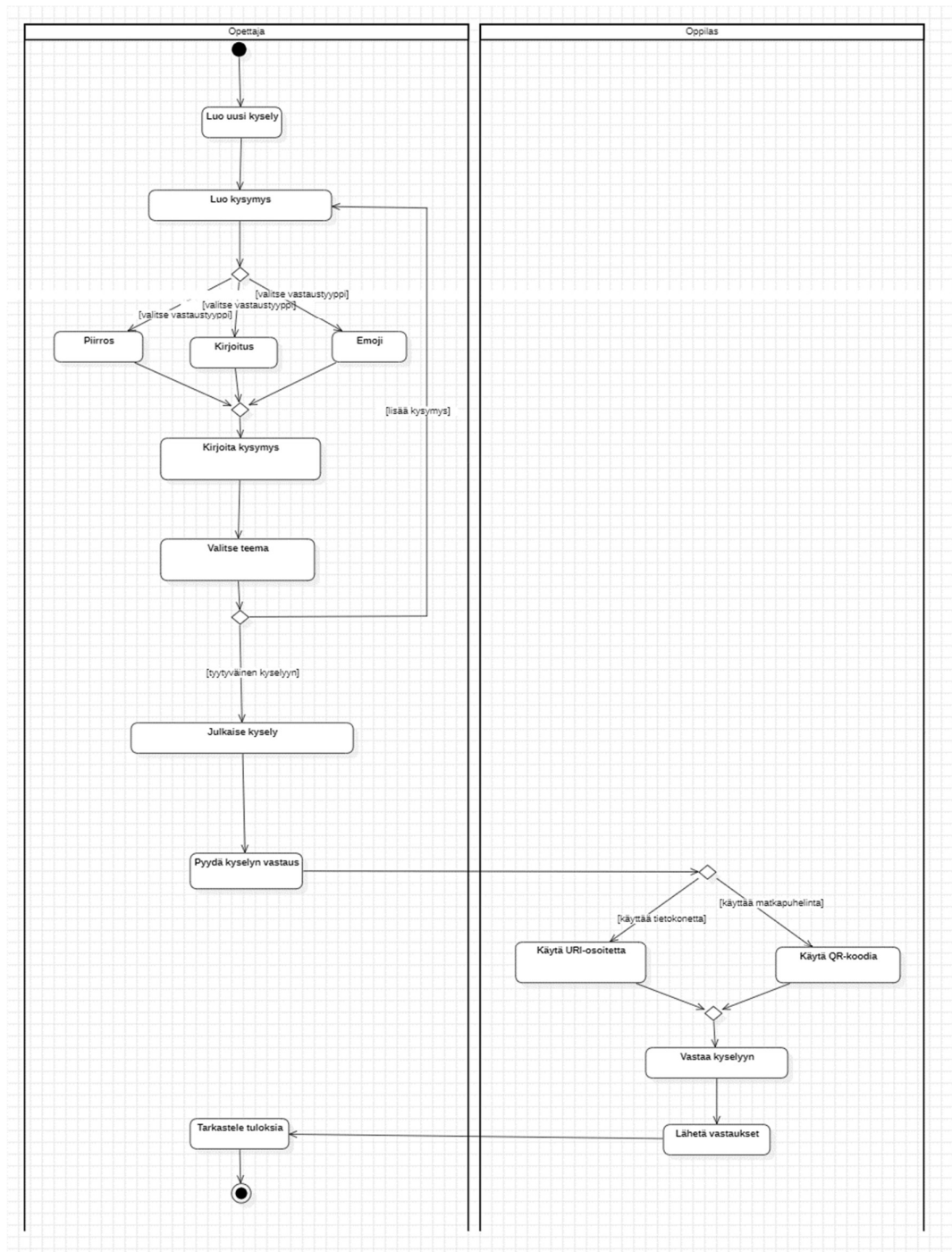
Kuva 7: Sekvenssikaavio

Sekvenssikaavion kuvaus

1. Käyttäjä aloittaa kirjautumisen.
2. SignIn -komponentti hallinnoi kirjautumisprosessia.
  - a. Selaimen tallennustilasta tarkistetaan, onko käyttäjällä jo oikeudet.
  - b. Jos kirjautumistietoja ei ole tallessa, luodaan pyyntö tietokantaan.
  - c. Tietokannan vastaus tallennetaan selaimeen.
  - d. Selaimesta tarkistetaan uudestaan kirjautumisstatus.
  - e. Tieto välitetään SignIn -komponentille.
3. Käyttäjä on kirjautuneena sisään.



## 6.2 Aktiviteettikaavio



Kuva 8: Aktiviteettikaavio

Aktiviteettikaavion kuvaus

1. Opettaja luo kyselyn. Kysely voi sisältää useamman kysymyksen.
2. Opettaja lisää kysymyksen kyselyyn.
3. Opettaja valitsee vastaustyyppin yhdestä seuraavista jolla oppilaat tulevat vastaamaan:
  - a. Piirros,
  - b. Kirjoitus tai
  - c. Emoji.
4. Opettaja kirjoittaa itse kysymyksen.
5. Opettaja valitsee teeman kysymykselle.
6. Opettaja valitsee lisätäänkö kyselyyn kysymyksiä, jos hän lisää kysymyksen palataan kohtaan 2.
7. Opettaja julkaisee kyselyn.
8. Opettaja pyytää oppilaitaan vastaamaan kyselyyn.
9. Riippuen käyttääkö oppilas tietokonetta vai matkapuhelinta, navigoi hän kyselyyn joko käyttämällä:
  - a. URL-osoitetta tietokoneella tai
  - b. QR-koodia matkapuhelimella.
10. Oppilas täyttää kyselyn vastaten kaikkiin kysymyksiin.
11. Oppilas lähettää kyselyn vastattuaan kysymyksiin.
12. Opettaja tarkastelee vastauksia sen jälkeen, kun oppilaat ovat vastanneet kyselyyn.

## **7 Kehitysprosessi ja kehitysvaiheen tekniikat**

### **7.1 Projektinhallinta**

Projekti käyttää Trello-palvelua tehtävien hallinnointiin sekä ketterän kehityksen Scrum-menetelmää.

### **7.2 Versionhallinta**

Käytimme git-versionhallintaa ja GitHub-palvelua koodin isännöintiin.

### 7.3 Tietokanta

Sovelluksemme hyödyntää Metropolian MariaDB tietokantapalvelinta datan tallennusta varten.

### 7.4 Testaus

Yksikkötestien luomisessa käytimme Vitest-testauskehystä. Testit ajetaan jokaisen commitin jälkeen Jenkins-automaatiopalvelimella.

### 7.5 Julkaisu

Sekä React-asiakaspuoli että NodeJS-palvelinpuoli julkaistaan automaattisesti Jenkinsin avulla koulun palvelimille, kun main-haaraan tulee muutoksia GitHubissa. Molempien julkaisussa käytetään Metropolian eCloud-palvelimia.

### 7.6 Dokumentointi

Projekti käyttää Postman-palvelua API-rajapintojen dokumentointiin README.md tiedoston lisäksi.

## 8 Yhteenveto

Palautepomppu on verkkosovellus, joka mahdollistaa alakoululaisille helpon ja monipuolisen tavan antaa palautetta opettajilleen. Sovelluksen avulla oppilaat voivat ilmaista itseään luontevalla tavalla ja osallistua aktiivisesti opetuksen kehittämiseen.

## 8.1 Jatkokehitysideat

### 8.1.1 Käyttäjäprofiilit

Henkilökohtaisten tunnusten tarjoamista oppilaille, joiden avulla oppilaat voisivat monitoroida palautteen antamistaan historiallisesti. Oppilaat saisivat tämän lisäksi valita oman profiilikuvansa esimerkiksi kuudesta ennalta määrätystä eläimestä. Tässä ominaisuudessa tulisi vain varmistaa, että opettaja ei voisi selvittää palautteenantajan henkilöllisyyttä puolueellisuuden välttämiseksi.

### 8.1.2 Anonyymiuden valinta kyselyä luodessa

Opettajille voitaisiin tarjota mahdollisuus päättää kyselyn anonyymisyys luomisvaiheessa. Tämä toiminnallisuus liittyy keskeisesti käyttäjäprofiilit-ominaisuuteen, sillä oppilaat tarvitsevat käyttäjän, jotta heidän henkilöllisyytensä tiedetään ei-anonyymissä kyselyssä.

### 8.1.3 Pelillistäminen

Oppilaiden tyytyväisyyttä sovellukseen voisi lisätä pellillisyyys. Tätä voisi esimerkiksi olla käyttäjäprofiiliin liitetty taso, joka nousee aina kyselyihin vastattaessa.

### 8.1.4 Palauteanalyysi

Opettajien hallinnointipaneelia voisi kehittää tarjoamaan kattavammat työkalut palautteiden analysoimiseksi pidemmällä aikavälillä useamman kyselyn ajalta. Lisäksi esimerkiksi emoji palautenäkymää voisi parantaa näyttämään esimerkiksi ympyrädiagrammi saaduista palautteista numeerisen arvon sijaan.

### 8.1.5 Käyttöliittymän parantaminen

Vaikka olemme tyytyväisiä nykyiseen käyttöliittymään, voisi siitä tehdä entistä paremman lisäämällä animaatioita kysymysten välille sen sijaan että kysymyksen ovat yhdellä sivulla.

## 8.2 Toteuttamiskelvottomaksi todetut kehitysideat

Muutama kehitysidea todettiin huonoksi resurssien puutteen takia.

### 8.2.1 Tekstistä puheeksi

Tekstistä puheeksi -ominaisuus oli alkuperäisessä visiossamme, mutta resurssien puutteen vuoksi päätimme jättää sen toteuttamatta ainakin projektin ensimmäisellä puoliskolla.

### 8.2.2 Ääniviesti

Ääniviestiominaisuus oli alkuperäisessä visiossamme, mutta resurssien puutteen vuoksi päätimme jättää sen toteuttamatta ainakin projektin ensimmäisellä puoliskolla.

### 8.2.3 Integrointi

Mietimme että sovelluksen yhdistäminen koulun muihin palveluihin voisi olla hyödyllistä esimerkiksi paremman käyttäjänhallinnoinnin takia, mutta valitettavasti tämä ei ole mahdollista toteuttaa projektin nykyisellä omistajalla.

### 8.2.4 Viestintäkanavat

Viestintäkanavat olivat myös yksi kehitysidea, mutta emme todennut tämän oleva kehitystyön arvoinen sillä alaikäisten viestintäpalvelun toteuttamisessa on omat juristiset haasteet.