

1. Toiminta

- Kun ohjelma käynnistyy, ohjelmaan avautuu ikkuna, jossa kysytään käyttäjän etunimi, sukunimi, opiskelijanumero ja tutkinto, kun kaikki nämä kentät ovat täytetty, voidaan painaa Submit -nappia (Ohjelma ei tallenna opiskelijan tietoja mihinkään). Tämän jälkeen pääsee toiselle välisivulle, missä näkyy valitun tutkinnon rakenne. Ohjelman toiminta perustuu Sisun rajapintaan. Kurssit näkyvät puurakenteessa. Tutkinto-ohjelman valintanäkymän lisäksi ei ole muuta. Eli opintoja ei voi seurata.
- Ohjelma on toteutettu, niin, että se täyttää minimi toteutuksen ominaisuudet. ainoa ero minimi toteutukseen on se, että käyttöliittymä on toteutettu omatoimisesti.

Opiskelijan tiedot -näkyvä:

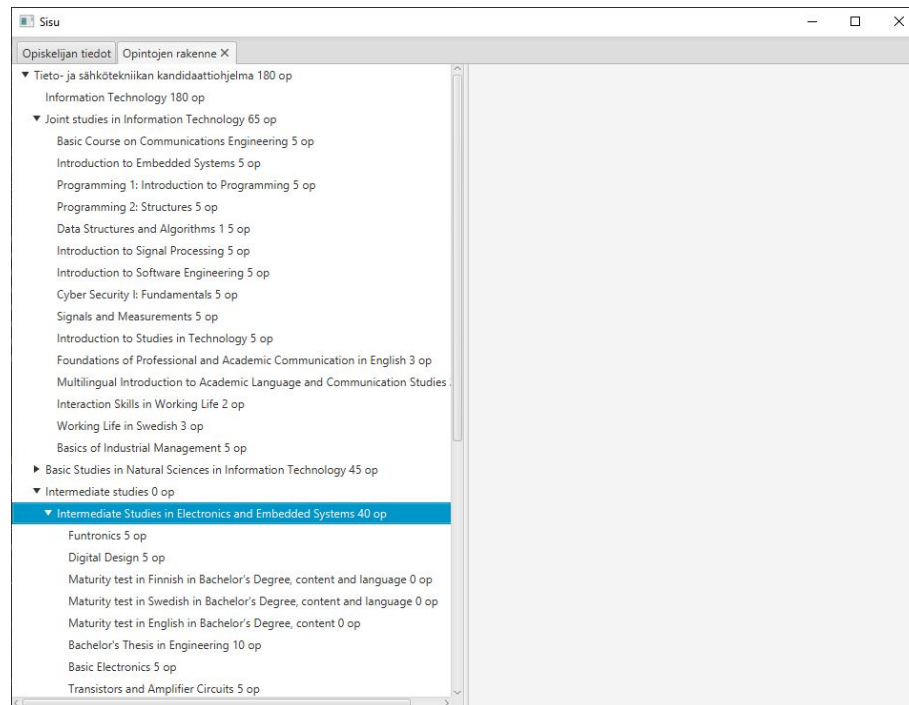
The screenshot shows a web application window titled 'Sisu'. It has two tabs: 'Opiskelijan tiedot' (selected) and 'Opintojen rakenne'. The 'Opiskelijan tiedot' tab contains a form with the following fields:

- Nimi:
- Sukunimi:
- Opiskelijanumero:
- Tutkinto:

Below the form is a 'Submit' button. To the right of the form, the following information is displayed:

- Nimi: Petteri Peruna
- Opiskelijanumero: H123123
- Tutkinto: Tieto- ja sähkötekniikan kandidaattiohjelma

Opintojen rakenne -näkyvä:



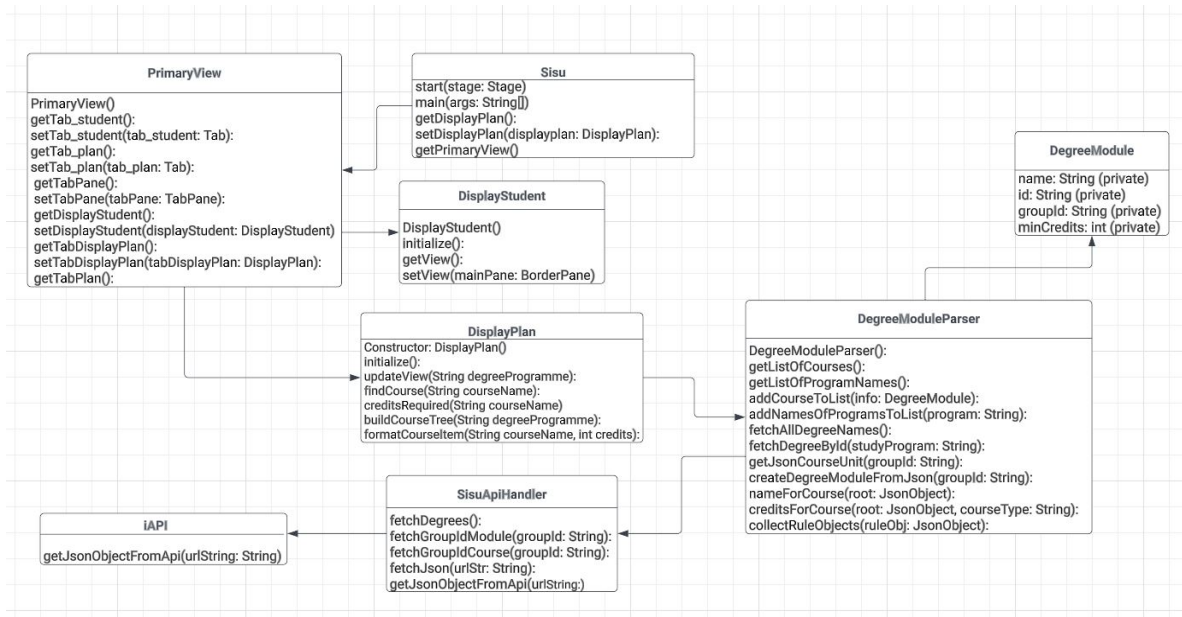
2. Toteutus

- Olen itse toteuttanut ohjelman. Sillä olen tämän ryhmän ainoa jäsen.

3. Ohjelman ongelmat

- Kun käyttäjä valitsee tutkinnon ohjelmassa, kaikki kurssit eivät välttämättä näy puurakenteessa. Lisäksi joissain tutkinnoissa voi olla useita samanlaisia kursseja, jotka näkyvät puurakenteessa erikseen. On myös huomioitava, että tutkinnon valitsemisen jälkeen ohjelman täytyy ladata tietoja Sisu-rajapinnasta, mikä voi viedä jonkin aikaa.

4. UML -kaavio:



5. Luokat ja niiden toiminnallisuus

- **DegreeModule**

- abstrakti luokka, joka tarjoaa yleisiä ominaisuuksia opinto-ohjelman kursseille ja moduuleille.

- **DegreeModuleParser**

- luokka vastaa JSON-dataan liittyvien tutkinto-ohjelman kurssien ja moduulien jäsentämisestä. Se hakee tutkinnon nimet ja tutkinnon yksityiskohdat ID:n perusteella, luo DegreeModule-objekteja ja tallentaa ne listaan.

- **DisplayPlan**

- luokka vastaa toisen välilehden sisällöstä käyttöliittymässä, joka näyttää opintosuunnitelman hierarkkisessa puussa. Luokka käyttää TreeView- ja TreeItem-luokkia, joilla esitetään opintosuunnitelmaa ja sen sisältöä

- **DisplayStudent**

- Kysyy ja näyttää opiskelijan tiedot ensimmäisellä välilehdellä. sisältää useita käyttöliittymäelementtejä, kuten tekstikenttiä, pudotusvalikoita ja painikkeita

- **PrimaryView**

- Hallinnoi pääkäyttöliittymän näkymää. Luo ja alustaa TabPane-objektin, joka sisältää kaksi välilehteä: "Opiskelijan tiedot" ja "Opintojen rakenne".

- **Sisu**
 - Sisu-luokka, joka on pääluokka Sisu-sovellukselle. Sovelluksen käynnistyessä luo käyttöliittymän, asettaa sen päänäytöksi ja käynnistää sovelluksen
- **SisuApiHandler**
 - Luokka sisältää metodeja JSON-tietojen hakemiseen Sisu API:lta. Toteuttaa iAPI-rajapinnan ja tarjoaa metodit tutkinto-ohjelmien, moduulien ja kurssiyksiköiden hakemiseen API:lta.
- **iAPI**
 - rajapinta, joka määrittelee yhden metodin, getObjectFromApi.