

# Ohjelman dokumentti

Emma Mannila, Pinja Myllymäki, Väinö Anttalainen

Tässä dokumentissa on esitelty ohjelman toiminta lyhyesti, ohjelman luokkakaavio, ohjelman vapaaehtoiset lisäominaisuudet, ryhmän työnjako sekä ohjelman puutteet ja rajoitteet. Ohjelmalle on kirjoitettu myös Javadoc-dokumentti, josta löytyy tarkemmat metodien ja luokkien kuvaukset, sekä näiden ehtoja. Käyttäjän täytyy kuitenkin generoida Javadocit itse, koska niitä ei löydy Gitistä.

## Ohjelman toiminta

Ohjelman käynnistyessä aukeaa valikko, josta käyttäjä voi joko kirjautua sisään opiskelijanumerolla tai luoda uuden käyttäjän valitsemallaan nimellä, opiskelijanumerolla sekä tutkinnolla. Jos jokin näistä kohdista puuttuu tai opiskelijanumero on jo varattu, ei käyttäjä pääse eteenpäin. On myös huomattava, että kun käyttäjä on valikossa "Lisää uusi opiskelija", valikosta ei pääse takaisin valikkoon, josta voi kirjautua sisään opiskelijanumerolla. Tällöin ohjelma pitää käynnistää uudestaan.

Kun käyttäjä on kirjautunut sisään tai luonut uuden käyttäjän, avautuu sivu, jossa hän voi tarkastella opintojen etenemistä, keskiarvoa, vaihtaa tutkintoa, tallentaa ja poistua tai palata kirjautumissivulle.

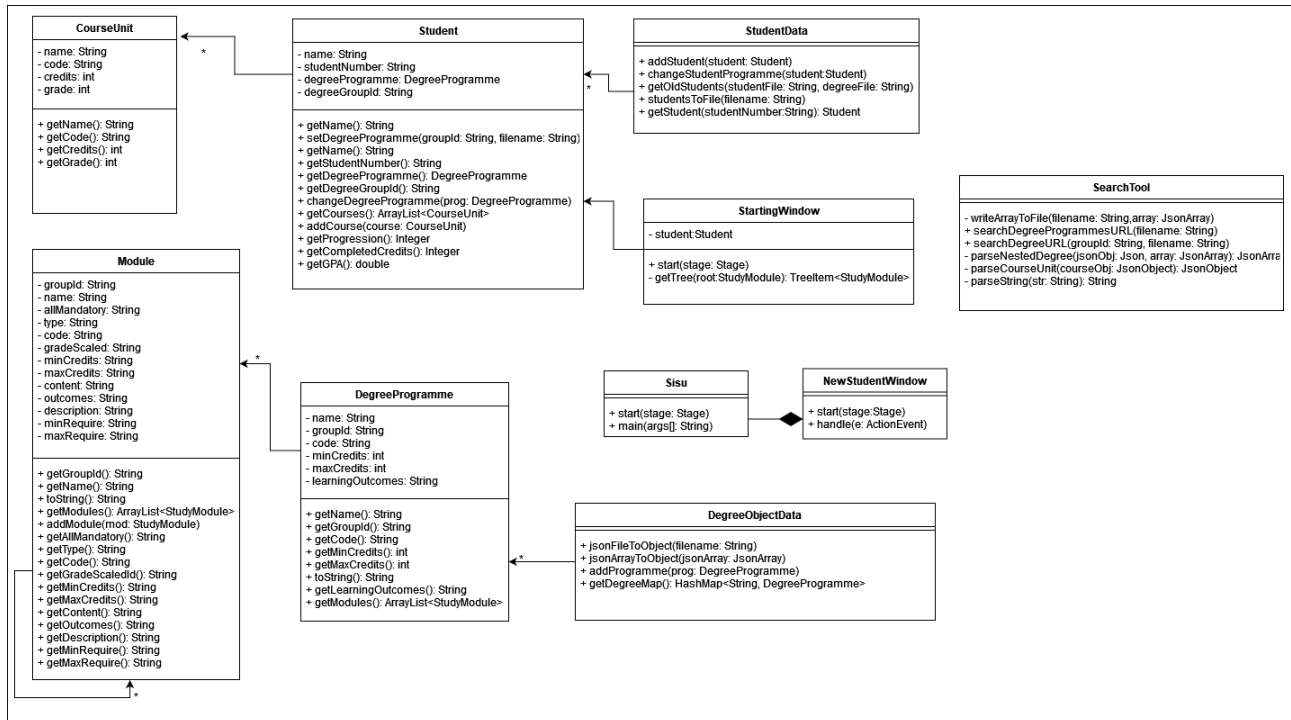
Välilehteä vaihtamalla käyttäjä pääsee näkymään, jossa hän voi tarkastella tutkintonsa rakennetta. Tutkinnon rakenne sisältää kaikki Sisun APIsta löydetty tutkintorakenteen osat ja kurssit.

Ohjelma näyttää moduulista erilaisia tietoja, kun moduulia klikkaa. Mikäli kurssin arviointi on asteikolla 0–5, voi käyttäjä merkata kurssista saadun arvosanan ennen kurssin merkkäämistä suoritetuksi. Mikäli kurssista on mahdollista saada vaihteleva määrä opintopisteitä, myös tämän pystyy valitsemaan ennen kurssin lisäämistä suoritetuksi.

Käyttäjä voi halutessaan vaihdella välilehdeltä toiselle vapaasti, mutta ohjelma tulee lopettaa "Tallenna ja poistu" -napista, mikäli muutokset halutaan tallentaa. Käyttäjää voi vaihtaa sulkematta ohjelmaa käyttämällä "Tallenna ja palaa kirjautumissivulle" -nappulaa. APIsta haetut tutkinnot tallentuvat "muistiin", vaikkei opiskelijan tietoja tallennettaisiinkaan.

Ohjelma voidaan suorittaa, vaikkei mitään koodissa viitattuja tiedostoja ole vielä luotu. Ei voida kuitenkaan taata, että ohjelma toimii odotetusti, mikäli tiedostot sisältävät vääränlaista dataa tai vääränlaisessa formaatissa. Ei ole siis suositeltavaa muuttaa tiedostojen sisältöä ohjelman suorituksen aikana tai sen jälkeen.

## Ohjelman luokkakaavio



## Ohjelman vapaaehtoiset lisäominaisuudet

### • Moduulien katselunäkymä

Moduulia klikkaamalla näytetään siitä lisätietoja. Moduulilla tarkoitetaan tutkinnon puurakenteen osaa, jota voidaan klikata. Lisätietoihin kuuluu moduulin tyypistä riippuen moduulin nimi, koodi, arviointiasteikko, moduulista saatavat/moduuliin tarvittavat opintopisteet, moduulin sisältö ja oppimistavoitteet. ”Valitse osa” -moduuli näyttää lisäksi tietoa siitä, kuinka monta alimoduulia on suoritettava. Jokaiselle moduulille ei välttämättä löydy jotakin lisätietoa, jos sitä ei ole merkitty APIssa. Tällöin lisätiedoissa voi esiintyä null, merkintöjä.

### • Kurssien suorittaminen

Kun käyttäjä klikkaa kurssia, näytetään kurssista lisätietoja (kts. lisäominaisuus Moduulien katselunäkymä). Alakulmaan ilmestyy nappula, jota painamalla käyttäjä merkkää kurssin suoritetuksi. Mikäli kurssin arviointi on asteikolla 0–5, voi käyttäjä merkata kurssista saadun arvosanan ennen kurssin merkkäämistä suoritetuksi. Mikäli kurssista on mahdollista saada vaihteleva määrä opintopisteitä, myös tämän pystyy valitsemaan ennen kurssin lisäämistä suoritetuksi. Opintopisteiden määrä ja arvosana valitaan spinnerien avulla. Ne antavat käyttäjän valita vain kurssille määriteltyjen opintopisteiden ja arvosanan mukaisia arvoja. Lisäksi käytyjä kurseja havainnollistetaan puurakenteeseen ilmestyvällä ikonilla.

- **Opiskelijan tallennus**

Kun uutta opiskelijaa luodaan, annetaan sille yksilöllinen opiskelijanumero, jonka avulla opiskelijat voidaan yksilöidä. Opiskelijan tiedot ja hänen käymänsä kurssit tallennetaan JSON-tiedostoon, josta ne luetaan ohjelman uudelleen käynnistyessä. Opiskelijaa ei ole mahdollista poistaa ohjelman kautta.

- **Opiskelijan tutkinnon tilanne**

*HUOM. tämä ei välttämättä ole vapaaehtoinen lisäominaisuus riippuen siitä, mitä Plussassa esitetty ominaisuus ”opiskelijan tutkinnon tilanne esitetään ohjelmassa” tarkoittaa. Ominaisuus on kuitenkin varuiksi listattu dokumenttiin.*

Ohjelma laskee opiskelijan suorittamien kurssien opintopisteet ja esittää ne muodossa ”opiskelijan op:t” / ”tutkinnon vähimmäis op määrä” (esim. 75/180). Samalla edistymistä havainnollistetaan visuaalisesti palkin avulla, joka täyttyy opintopisteiden tahtiin. Myös opiskelijan keskiarvo lasketaan ja esitetään käyttöliittymässä.

- **Tutkintojen tallennus**

APIsta haetut tutkintorakenteet tallennetaan JSON-tiedostoon, josta ne luetaan olioiksi ohjelman käynnistyessä. Tämä mahdollistaa ohjelman nopeamman toiminnan, kun tietoja ei tarvitse aina hakea APIsta.

## Ryhmän työnjako

Ohessa esitetty ryhmän työnjako taulukkomuodossa. Koska usein kävi niin, että yhtä luokkaa/tehtävää oli tekemässä useampi henkilö, on taulukkoon merkattu merkillä ”X” henkilö, jolla oli isoin panos kyseiseen tehtävään ja merkillä ”o” henkilö(t), jolla oli pienempi panos tehtävään. Usein työt jaettiin Trellon avulla jäsenien kesken, mutta loppua kohden työnteko muuttui joustavammaksi, eikä työnjakoa suunniteltu yhtä paljon.

	Emma	Pinja	Väinö
CourseUnit.java		X	o
DegreeObjectData.java	o	o	X
DegreeProgramme.java	o	X	o
SearchTool.java		X	o
Sisu.java	X	o	
StartingWindow.java		X	o
Student.java		X	o
StudentData.java	X		o
Module.java	o	X	X
package-info.java			X
JUnit-testit		X	o
Javadoc			X
Luokkakaavio		X	

## Ohjelman rajoitteet, ongelmat ja puutteet

- Ohjelmaan ei ole toteutettu TestFX-testejä.
- Käyttäjä ei voi hakea, listata tai merkitä suoritetuksi vapaavalintaisia opintoja tai tutkintokokonaisuuksia, joita ei ole API:ssa määritelty tutkintoon kuuluvaksi.
- Ohjelma ei tarkista, onko opiskelijan käymät kurssit sääntöjen mukaisia. Esim. jos kahdesta rinnasteisesta tutkinnon osasta pitää valita vain toinen, voi käyttäjä silti merkitä kursseja suoritetuksi kummastakin tutkinnon osasta.
- Koska jotkin tutkinnot pitävät sisällään runsaasti erilaisia alamoduuleita ja rakenteita, ja koska tutkinnoista tallennetaan myös kirjallista tietoa (tutkinnon/kurssin sisältö ja oppimistavoitteet), voi tutkinnot sisältävän JSON-tiedoston koko kasvaa aika isoksikin varsinkin, jos monia eri tutkintoja haetaan.
- Ohjelma ei tarkista sitä, että tiedostojen sisältö on oikeanlaista.
- Koodin ymmärrettävyyttä ja uudelleenmuokkausta vaikeuttaa varsinkin luokissa SearchTool, Sisä ja StartingWindow luokkien pitkä ja hieman monimutkainen toteutus. Lisäksi koodia ei ole Javadocin lisäksi kommentoitu, mikä vaikeuttaa luettavuutta monimutkaisissa kohdissa.
- Mikäli käyttäjä merkitsee suoritetuksi kurssin, joka löytyy muualta samasta tutkinnosta, ei vihreää merkintää tule muiden kuin vain klikatun kurssin nimen viereen. Käyttäjän kirjautuessa uudestaan sisään kaikki samat kurssit kuitenkin päivittyvät näyttämään vihreää merkkiä. Tämä on kuitenkin vain kosmeettinen haitta, joka ei vaikuta ohjelman toiminnallisuuteen.