

Sisuprojektin dokumentaatio

Group2547: Antti Hakkarainen, Heikki Hohtari, Kilian Kugge

1 Johdanto

Ryhmän toteuttama Sisu sovellus on opiskelijoille, jotka haluavat löytää ja seurata Tampereen yliopiston tarjoamia tutkintoja ja kursseja. Opiskelija voi myös asettaa kursseja suoritetuksi ja näin seurata opintonsa edistymistä.

2 Ominaisuudet

2.1 Perus- ja minimiominaisuudet

- Ohjelma kääntyy
- Ohjelmassa on *vähintään* yksi dialogi-ikkuna (Ohjelmassa on useita)
- Ohjelma osaa näyttää valitun tutkinnon rakenteen pääikkunassa.
- Ohjelmassa on käytetty periytymistä rakenteen osien toteuttamisessa.
- Ohjelmassa on loppudokumentti, josta löytyy: luokkien vastuujako ja ryhmän työnjako
- Ohjelmaan on toteutettu omatoimisesti graafinen käyttöliittymä
- Näytettävää tutkintoa voi vaihtaa ja opiskelijan tutkinnon tilanne esitetään ohjelmassa
- Ohjelma selviää hallitusti asetustiedostojen käsittelyssä tapahtuvista virheistä
- Ohjelmalle on toteutettu yksikkötestejä

2.2 Lisäominaisuudet (mahdolliset lisäpisteet)

- Opiskelija-asetukset. Uutta opiskelijaa luodessa, tai opiskelijan tietoja tarkastellessa avataan asetusikkuna, josta voi asettaa ohjelmaa käyttävän opiskelijan tiedot. Suorituksen tilanne haetaan JSON-tiedostosta ja tallennetaan sinne.
- Tuki usealle eri opiskelijalle. Jokaisen opiskelijan tiedot tallennetaan omaan json-tiedostoon students/ kansioon. Uusia opiskelijoita voi luoda UI:n kautta, ja opiskelijaa voi vaihtaa aina näkyvissä olevasta yläpalkista, milloin vain ohjelman suorituksen aikana.
- Tutkinnon moduuliin kuuluvien pakollisten kurssien esittäminen checkboxilla, joka on aina valittu, ja josta ei voi poistaa valintaa. Näin ollen opiskelija ei voi vahingossa poistaa tutkintoon kuuluvia pakollisia kursseja moduuliin kuuluvia kursseja valitessaan.
- Moduuliin voi hakea uuden kurssin, vapaasanahauulla erillisen hakunäkymän kautta. Kurssi lisätään osaksi moduulia ja merkitään opiskelijan valitsemaksi kurssiksi. Tämä ei valitettavasti tutki, mitä kursseja moduuleihin voi lisätä – joten mihinkä tahansa moduuliin voi lisätä mitä tahansa kursseja.
- Arvosanan asettaminen kurssille. Tarjolla on vaihtoehdot 0-5 tai pass/fail, riippuen siitä, arvostellaanko kurssi arvosana-asteikolla vai hyväksytty-hylätty -asteikolla. Arvosanan voi myös poistaa "remove grade"-napista.
- Opiskelijan opintojen etenemisen aikataulun esittäminen tutkinnon rakenteeseen peilaten. Yläpalkissa on näkyvissä valitun tutkinnon tavoitenoppamäärä, ja suoritettujen kurssien noppamäärä. Etenemisestä on laskettu myös prosentti. Suoritettuun noppamäärään lasketaan vain hyväksytyt kurssit.
- Opiskelijan tutkinto valitaan omasta ruudusta, ja tutkintolistaa voi suodattaa hakusanoilla. Lisäksi tutkintolistan järjestystä voi vaihtaa aakkosjärjestyksen ja käänteisen aakkosjärjestyksen välillä.
- Ohjelma tutkii Internet-yhteyden toimivuutta Funidatan palvelimelle, ja kertoo, jos yhteyden muodostus ei onnistu. Ohjelma pyrittiin toteuttamaan niin, että käyttäjä voi yrittää samaa toimintoa hetken päästä uudestaan, esim. painamalla samaa nappia, tai Tutkinto-ohjelman valintaruudussa "Refresh"-nappia, joka yrittää ladata tutkinto-ohjelman uudestaan Internetistä.
- Erilaisiin ohjelman ajonaikaisiin virhe/huomautus/varmistustilanteisiin, joista käyttäjän olisi hyvä tietää, on tehty ilmoitusikkuna. Virhetilanteissa ikkuna näyttää geneerisen virheilmoituksen, ja "More info"-napin takaa löytää tarkemman virheilmoituksen. Vahvistus- ja huomautustilanteissa lisätietoja ei ole.
- Korruptoituneen opiskelijatiedoston yhteydessä virheilmoitusikkunassa on ylimääräinen nappi, jonka takaa avautuu ikkuna, joka tarjoaa uudessa ikkunassa käyttäjälle toivottavasti häntä piristävää sisältöä.

3 Käyttöohje

3.1 Käyttäjän valinta

Ohjelma avautuessa käyttäjän on ensin tallennettava opiskelijan tiedot. Jos aiempia käyttäjiä on jo tallennettu, voidaan listalta valita oikea käyttäjä painamalla "Choose". Kunkin käyttäjän tiedot tallennetaan omiin tiedostoihinsa. Jos käyttäjätiedostoja on menty muuttamaan ohjelman ulkopuolelta, niin käyttäjän tiedot voivat olla korruptoituneita. Korruptoituneen käyttäjän tietoja ei näytetä ollenkaan listassa, vaan ohjelma antaa virheilmoituksen korruptoituneista tiedoista.

Uuden käyttäjän saa tehtyä painamalla "New Student". Käyttäjälle pitää syöttää etu- ja sukunimi, sekä opiskelijanumero. Kaikkien näistä pitää olla vähintään kolme merkkiä pitkä- ja opiskelijanumeron on oltava enintään 16 merkkiä pitkä. Edellä mainitut kentät eivät hyväksy mitä tahansa merkkejä. Jos käyttäjä yrittää syöttää laitonta merkkiä, niin se ei tule ruudulle näkyviin kyseisessä kohtaa. Käyttäjä tallennetaan "OK" nappia painamalla.

Käyttäjälle pitää asettaa tutkinto-ohjelma. Uuden käyttäjän saa tallennettua ilman tutkinto-ohjelman asettamista, mutta tällöin ohjelman muita toimintoja ei pysty käyttämään. Käyttäjän nimen ja tunnuksen lisäämisen jälkeen tutkinto-ohjelma asetetaan painikkeesta "Select degree program". Tämä avaa uuden näkymän, jossa on listattuna kaikki Tampereen yliopistolla saatavilla olevat tutkinnot. Listaa voi rajata ruudun alalaidassa olevalla Filter hakukentällä. Rajauksen voi poistaa painikkeesta "Clear filter". Tutkinto-ohjelman valinta suoritetaan klikkaamalla haluttua tutkinto-ohjelmaa ja sitten klikkaamalla "Select".

Ruudun ylälaidassa on näkyvillä aina sillä hetkellä valittu käyttäjä. Kaikista näkymistä voi palata käyttäjän tietoihin painikkeesta "View info". Tässä näkymässä käyttäjän nimeä ja valittua tutkintoa voidaan vielä muokata painamalla "Edit info". Jos tutkinto-ohjelmaa muutetaan myöhemmin, niin käyttäjä menettää kaikki aiempaan tutkinto-ohjelmaan tallennetut tiedot. Lisäksi käyttäjää pääsee aina vaihtamaan ruudun ylälaidasta käyttäjää painikkeesta "Change student".

3.2 Tutkinto-ohjelma näkymä

Tutkinto-ohjelmanäkymä avautuu, kun käyttäjä on valittu tai tallennettu. Näkymä jakautuu kahteen palstaan. Vasemmalla on valitun tutkinto-ohjelman moduulien ja kurssien rakenne. Moduuleita voi avata tai piilottaa, niiden vieressä olevasta kolmiosta. Moduulit ja kurssit on merkitty eri väreillä. Kurssia painamalla avautuu kuvan oikealle puolelle kuvaus kyseisen kurssin sisällöstä. Tutkinto-ohjelma näyttää vain tutkinto-ohjelmaan pakollisesti kuuluvat kurssit, sekä käyttäjän valitsemat kurssit.

3.3 Kurssien lisääminen / poistaminen moduuleihin ja niiden valitseminen

Moduuleille asetettuja valinnaisia tai vaihtoehtoisia kursseja ei näytetä tutkinto-ohjelmassa. Opiskelija saa kaikki moduuliin valinnaisina tai pakollisina kuuluvat kurssit klikkaamalla ensin kyseistä moduulia ja sitten alalaidasta "Select courses". Tällöin avautuu moduulin näkymä, jossa kurssin voi asettaa valituksi asettamalla rasti kurssin kohdalle ruutuun. Tämä tarkoittaa sitä, että opiskelija on valinnut kyseisen kurssin osaksi tutkinto-ohjelmaa kyseisessä moduulissa. Pakollisista kursseista ei voi ottaa rastia pois. Myös tässä näkymässä kurssia klikkaamalla tulee oikeaan ruutuun kuvaus kurssista. Muutoksen voi tallentaa ja edelliseen näkymään palata painamalla "Save changes".

Jos halutaan lisätä moduuliin kurssi, joka ei kuulu moduuliin entuudestaan, niin voidaan moduulinäkymästä painaa nappia "Add course". Saman voi tehdä myös tutkinto-ohjelma näkymästä

ensin painamalla moduulia johon kurssin haluaa asettaa ja sitten "Add courses". Tällöin avautuu kurssien haku näkymä. Kurssia pitää hakea hakusanalla. Hakusanassa täytyy olla vähintään kolme merkkiä. Mahdolliset laittomat merkit eivät tule ruudulle näkyviin. Kun käyttäjä on asettanut hakusanan ja painaa "Search" haetaan Sisun verkkorajapinnasta kaikki ne kurssit, joissa kyseinen hakusana esiintyy (kurssin nimessä). Jälleen kurssia klikkaamalla oikealle tulee kuvaus kurssista. Kurssin saa lisättyä osaksi valittua moduulia klikkaamalla ensin kurssia ja sitten "Select course". Näkymä palautuu tutkinto-ohjelma näkymään ja kurssi on heti lisätty valituksi kurssiksi tutkinto-ohjelmaan.

Kurssi voidaan poistaa tutkinto-ohjelman moduulista menemällä kyseisen moduulin näkymään. Käyttäjän pitää klikata kurssia vasemmalla ikkunalla ja sitten painaa "Remove course" painiketta. Tutkinto-ohjelman pakollisia kursseja ei voi poistaa.

3.4 Kurssin suorituksen lisääminen / poistaminen

Kursseille voi asettaa arvosanan, jos se kuuluu johonkin moduuliin. Arvosana voidaan asettaa moduulinäkymästä tai tutkinto-ohjelmaa näkymästä klikkaamalla kurssia ja sitten valitsemalla sille arvosanan pudotusvalikosta ja painamalla "Save grade". Pudotusvalikkoon "Set grade" tulee näkyviin arvosanat, jotka kurssille voi asettaa (esimerkiksi "pass", "fail" tai numeerinen arvosana). Arvosanan voi poistaa samaan tapaan painamalla "Remove grade".

3.5 Muuta

Monessa näkymässä kurssin tiedot näytetään oikeassa laidassa. Joskus kurssille ei ole kuitenkaan tarkemmin määritelty lisätietoja (Sisun verkkorajapinnassa), jolloin kyseisessä ikkunassa saattaa näkyä pelkästään kurssin nimi ja opintopisteet.

Ohjelma vaatii internetyhteyden toimiakseen. Jos internetyhteys puuttuu, katkeaa tai on epävaka, niin ohjelma voidaan sulkea jolloin käyttäjän tulee käynnistää se uudelleen. Näissä tilanteissa käyttäjän on ensin hoidettava internetyhteydet kuntoon ennen ohjelman käytön jatkamista.

4 Ohjelman Toteutus

4.1 Tärkeimmät käytetyt teknologiat

Ohjelma on kirjoitettu Javalla. Tutkinto- ja kurssitiedot haetaan Sisun verkkorajapinnasta. Käyttöliittymä toimii JavaFX-kirjastolla. Testejä ajetaan Junit-kirjastolla. Suurimmaksi osaksi JSON-käsittely on toteutettu käyttäen Jackson-kirjastoa. Ohjelman teossa on käytetty myös apuna Jacoco-testauksen kattavuuspalvelua, sekä "Checkstyle"-koodin tyylin analyysipalvelua.

Ohjelman tietovarasto on Gitlabissa, johon on integroitu CI/CD-toiminto, joka tarkistaa, että:

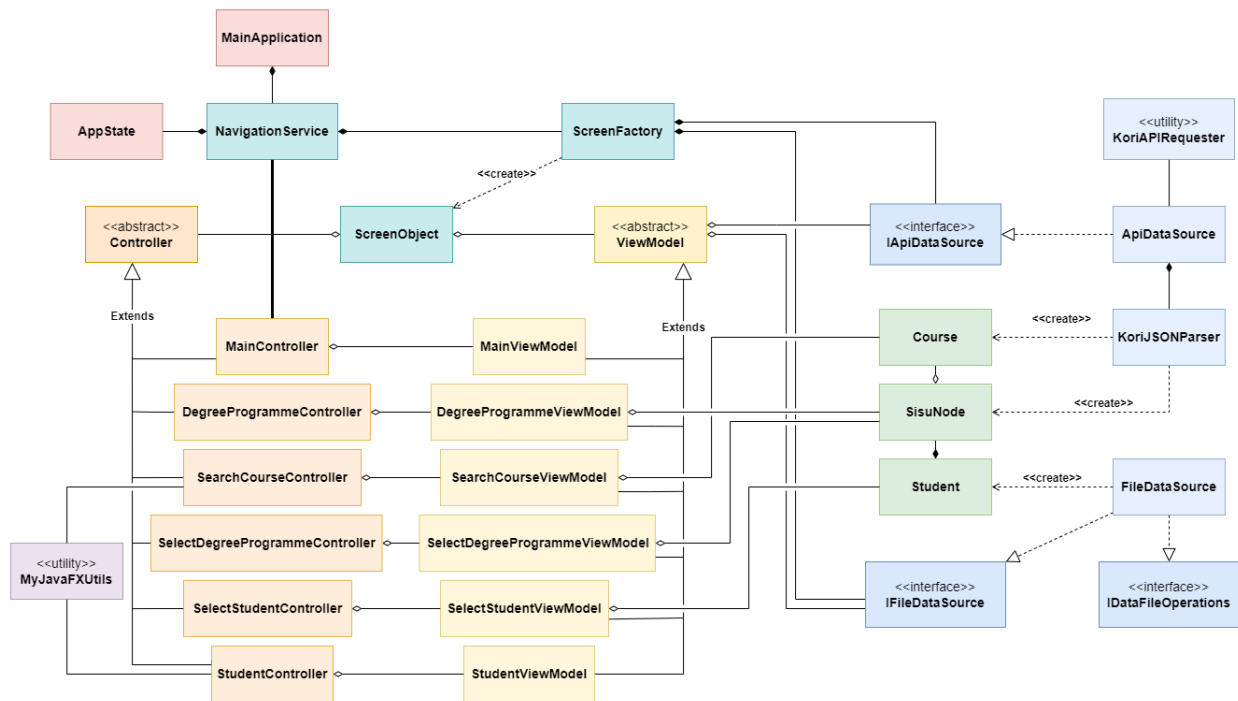
1. Ohjelma kääntyy
2. Ohjelman testit läpäistään

4.2 Ohjelman rakenne

Ohjelman rakenne on toteutettu MVVM mallia käyttäen.

Kaavassa 1 on UML-notaatiota käyttäen kuvattu järjestelmän luokkien väliset yhteydet, ja siten myös järjestelmän rakenne. Värikoodaukset osoittavat miten luokat ovat jaettu eri paketteihin.

1: UML-kaavio järjestelmän rakenteesta



4.3 Tärkeimpien luokkien vastuujako

Tässä luvussa käydään läpi paketti kerrallaan, järjestelmän tärkeimpien luokkien osuudet ja vastuut järjestelmän toiminnassa. Tässä ei siis mainita kaikkia luokkia, vaan karkeat vastuujao.

4.3.1 App-paketti

Tässä paketissa sijaitsee ohjelman pääluokka **MainApplication**, jonka vastuulla on pistää koko ohjelma käyntiin. Luo siis tarvittavat luokat ja kutsuu JavaFX:n start-metodia aloittamaan ohjelman.

4.3.2 Datasource-paketti

Tämän paketin luokat käsittelevät Kori API -tietojen hakemista (**KoriAPIRequester**) ja parsimista (**KoriJSONParser**) **ApiDataSource** -luokan kautta, sekä järjestelmän omien JSON-tiedostojen tallentamista ja hakemista **FileDataSource** -luokan kautta. Tarjoavat rajapinnat näiden käytölle. Luovat Model-paketin luokkien instansseja.

4.3.3 Model-paketti

Model-paketissa on määritelty luokat, jotka kuvastavat kursseja (**Course**), moduuleita (**SisuNode**) ja opiskelijoita (**Student**). Nämä ovat keskeisimpiä järjestelmän käsittelemiä palasia, sillä niistä muodostuu käyttäjän tutkintorakenne.

4.3.4 Navigation-paketti

Paketin vastuulla on järjestelmän eri näkymien luominen (**ScreenFactory**), sekä niiden välillä vaihtaminen ja näkymähistorian ylläpitäminen (**NavigationService**).

4.3.5 View-paketti

View-paketti sisältää MVVM-mallin View osuuden. Paketissa on abstrakti **Controller**-luokka, ja siitä periytyvät controllerit, joiden vastuulla on välittää käyttäjän tekemät toiminnot ViewModeleille, ja päivittää näkymää ViewModeleiden käskyjen perusteella. Jokaiselle järjestelmän näkymälle on oma controller.

4.3.6 ViewModel-paketti

ViewModel-paketti sisältää MVVM-mallin ViewModel-osuuden. Paketin abstraktista **ViewModel**-luokasta periytyvät luokat ovat rajapinta käyttäjänäkymän ja järjestelmän bisneslogiikan välillä. Ne ottavat vastaan tietoa käyttäjän toiminnasta controllereilta, tekevät tiedolla tarpeelliset toimenpiteet, ja käskyttävät controllereita päivittämään näkymää tarpeen mukaan. Jokaiselle näkymälle on oma viewmodel.

5 Ryhmän työnjako

Ryhmän henkilöiden vastualueet asetettiin alussa karkeasti seuraavasti:

- Antti Hakkarainen: Käyttöliittymä
- Kilian Kugge: Sisun verkkorajapinta
- Heikki Hohtari: edellisten väliin jäävät

Kun projekti rakenteeksi valittiin MVVM malli, niin vastualueiksi tarkennettiin:

- Antti Hakkarainen: View (XML tiedostot ja Controllerit)
- Kilian Kugge: Modelit, josta erityisesti Sisun verkkorajapinta
- Heikki Hohtari: ViewModelit ja Modelit siltä osin, kuin ne eivät kuuluneet suoraan Sisua käsitteleviin toimintoihin

Käytännössä projektin aikana ryhmäläiset ovat tehneet tarpeen mukaan muutoksia laajasti myös omien vastualueidensa ulkopuolella.

6 Tiedossa olevat ongelmat tai puutteet

Projektin avoimet issueet löytyvät myös Gitlabista: <https://course-gitlab.tuni.fi/compcs140-spring2023/group2547/-/issues>

Deadlinen vuoksi ainakin seuraavat asiat jäivät avoimeksi:

1. Degree programme-näkymän comboboxissa ei näy suoritettua kurssin arvosanaa, ja oletusarvo on tyhjä, kun kurssien välillä vaihtaa kurssin arvosanan asetettuaan. Tämä on visuaalinen ongelma.
2. Remove course -nappi Degree programme -näkyvässä ei ole tarkistusta onko kurssi pakollinen osa moduulia, vaan se poistaa kurssin.
3. Degree programme -termiä ei käytetä säännöllisesti kaikkialla, vaan on käytetty sekaisin study programme, curriculum, degree programme.
4. Yksikkötestikattavuus jäi pieneksi, kun jotenkin ominaisuuksia ja ongelmia oli turhan paljon toteutettavaksi.
5. TestFX-käyttöliittymätestaus jäi tekemättä vaikka se oli alun perin ideana.
6. Koodaustyylikäytännöt eivät ole yhteneväiset.

7. Kaikkea mahdollista, jota olisi pitänyt kommentoida, ei välttämättä tullut kommentoitua.
Joissain tilanteissa taas kommentteja voi olla liikaa, kommentit eivät välttämättä ole ajan tasalla jne. JavaDoc-kommenttien parametrilistat eivät välttämättä ole ajan tasalla kaikissa metodeissa jne.