PROGRAMMING 3: GROUP PROJECT

G2470

LAURI PARONEN, 150535467

MANU MÄKI-RAJALA, H292036

TURKKA SALONEN, 150272076

Overview:

This application permits its user to track the process of their studies. The program interacts with the Sisu API.

In the opening screen, the user enters their name, student number, and the approximated duration of their studies. As the user presses continue, the name and student number are saved into a JSON file entitled *userData.json* (into which the degree module that the user chooses will also be saved), and the program proceeds to the main window.
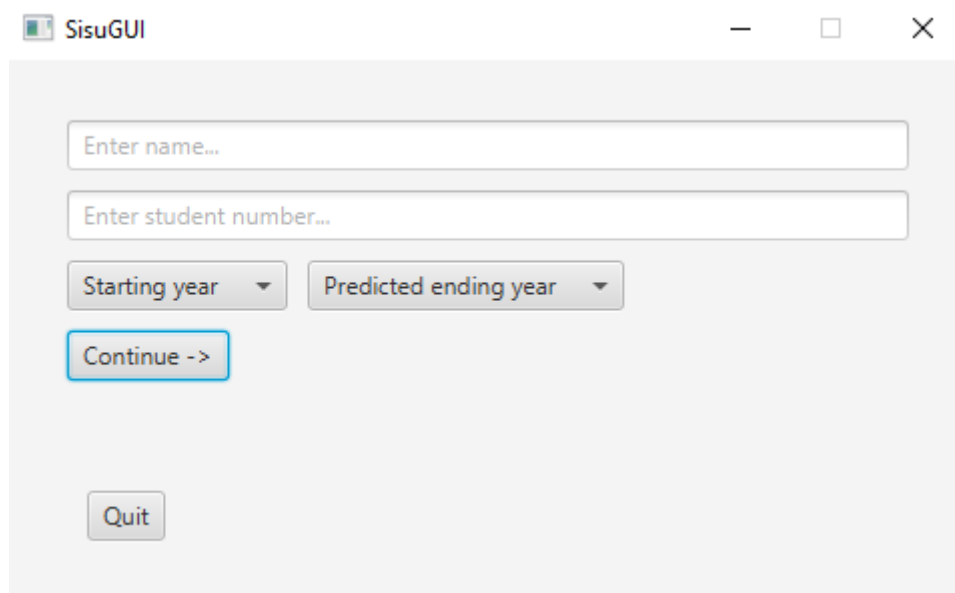

Figure 1 The starting screen of the application.

In the main window, the user can view a drop-down list of all the degree modules available in Sisu, and select any degree module for closer inspection in a tree view menu. In this window the user can also choose to view their profile from the *View profile* button, as well as save the degree module information into the *userData.json* file. In the dialog window opened by the *View profile* button, the user sees their name and student number, and has the possibility to change them.

When the user selects a degree from the drop-down menu, the program sends a request to the Sisu API and recursively fetches the information about the submodules beginning from the abstract base class *DegreeModule*. From these is formed the tree view visible in figure 2.
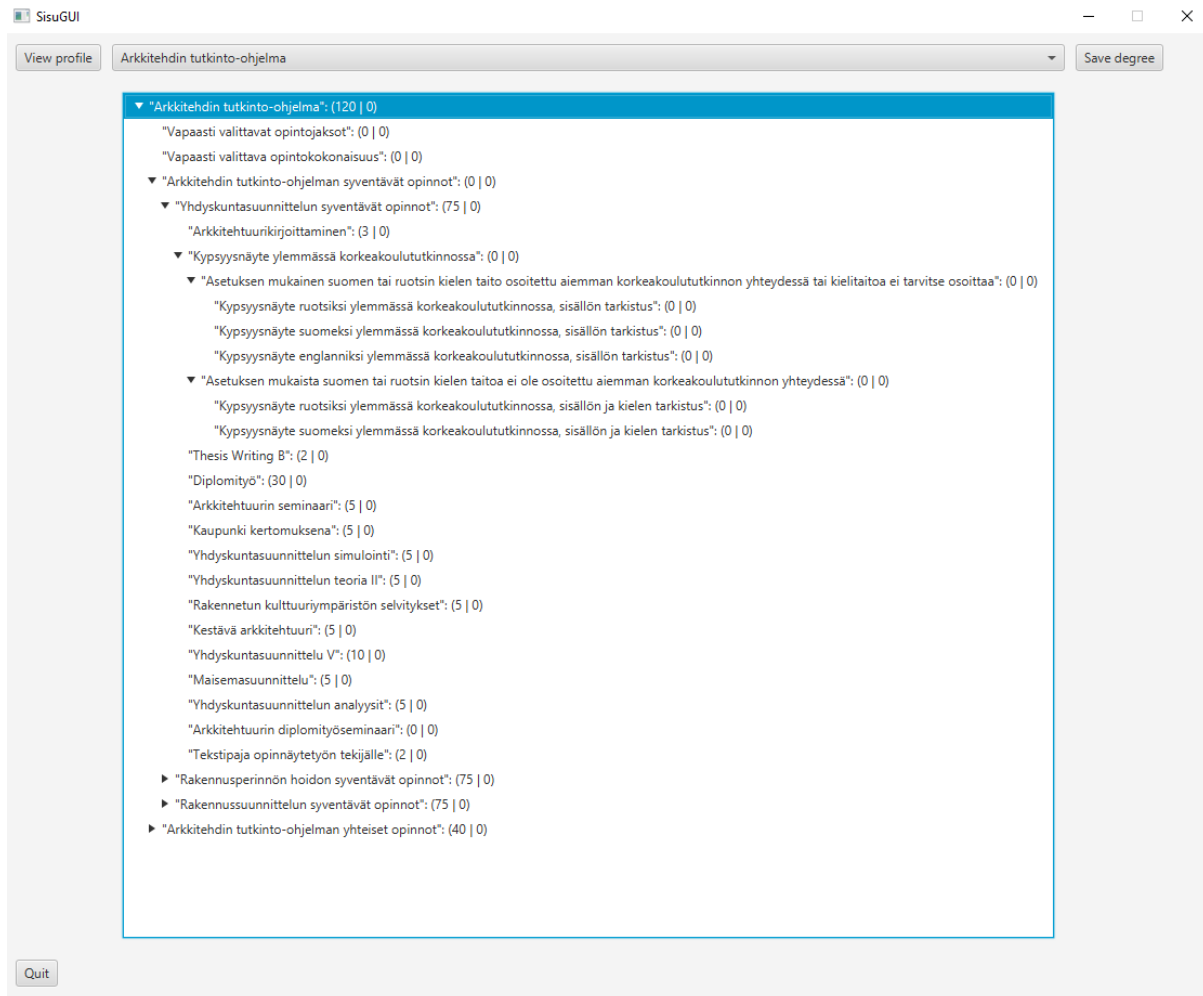
Figure 2 The main window of the application, showing the tree view structure when a degree programme is selected.
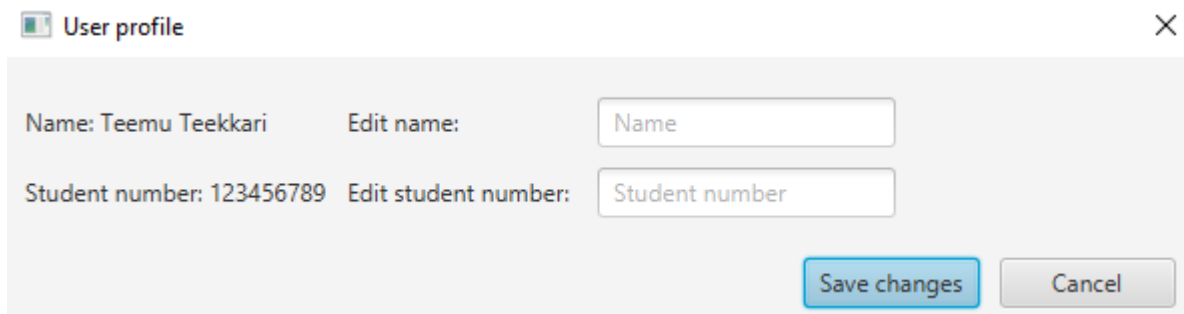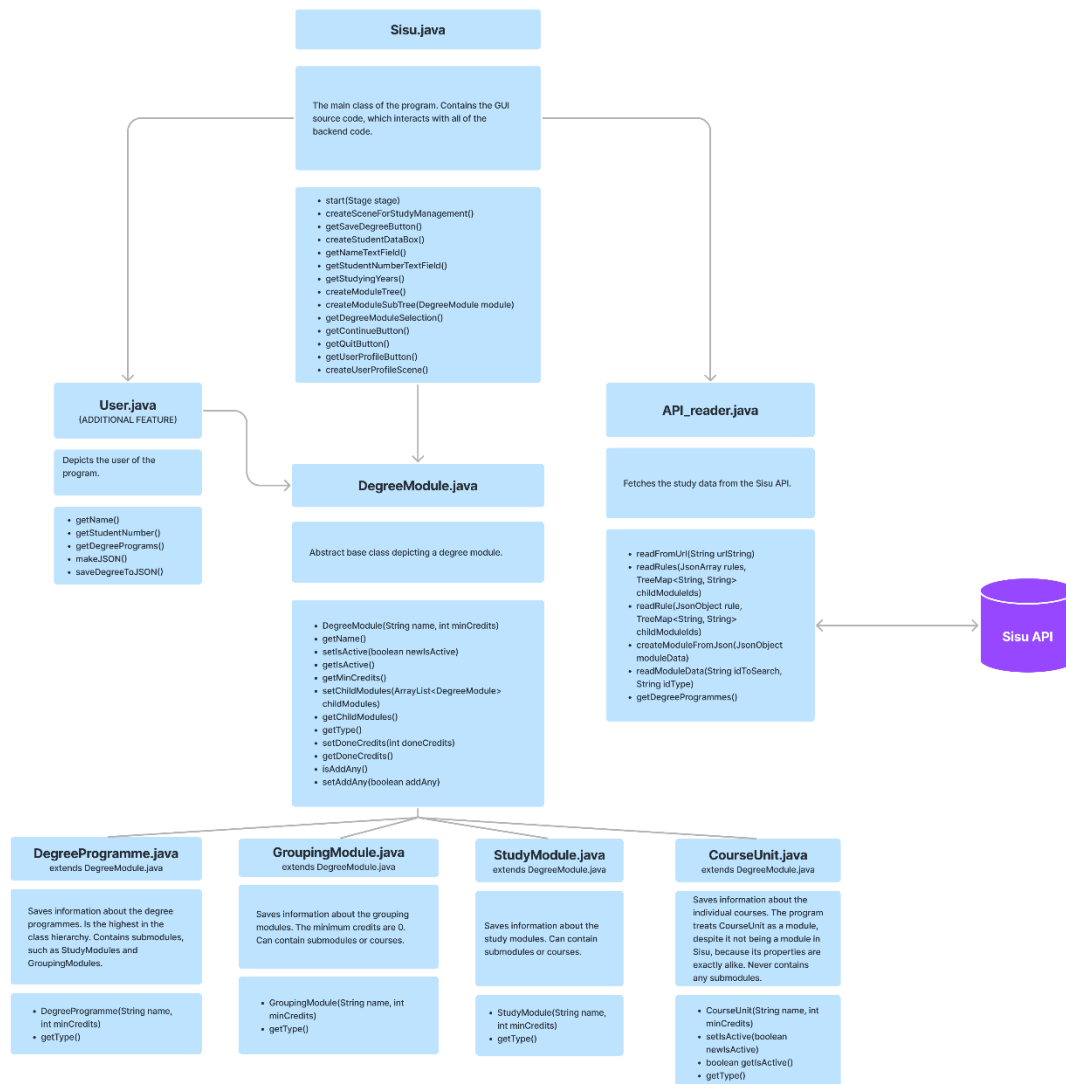

Figure 3 The user profile dialog.

Class diagram:

Sisu.java

The main class of the program. Contains the GUI source code, which interacts with all of the backend code.

- start(Stage stage)
- createSceneForStudyManagement()
- getSaveDegreeButton()
- createStudentDataBox()
- getNameTextField()
- getStudentNumberTextField()
- getStudyingYears()
- createModuleTree()
- createModuleSubTree(DegreeModule module)
- getDegreeModuleSelection()
- getContinueButton()
- getQuitButton()
- getUserProfileButton()
- createUserProfileScene()

User.java
(ADDITIONAL FEATURE)

Depicts the user of the program.

- getName()
- getStudentNumber()
- getDegreePrograms()
- makeJSON()
- saveDegreeToJSON()

DegreeModule.java

Abstract base class depicting a degree module.

- DegreeModule(String name, int minCredits)
- getName()
- setIsActive(boolean newIsActive)
- getIsActive()
- getMinCredits()
- setChildModules(ArrayList<DegreeModule> childModules)
- getChildModules()
- getType()
- setDoneCredits(int doneCredits)
- getDoneCredits()
- isAddAny()
- setAddAny(boolean addAny)

API_reader.java

Fetches the study data from the Sisu API.

- readFromUrl(String urlString)
- readRules(JsonArray rules, TreeMap<String, String> childModuleIds)
- readRule(JsonObject rule, TreeMap<String, String> childModuleIds)
- createModuleFromJson(JsonObject moduleData)
- readModuleData(String idToSearch, String idType)
- getDegreeProgrammes()

Sisu API

DegreeProgramme.java
extends DegreeModule.java

Saves information about the degree programmes. Is the highest in the class hierarchy. Contains submodules, such as StudyModules and GroupingModules.

- DegreeProgramme(String name, int minCredits)
- getType()

GroupingModule.java
extends DegreeModule.java

Saves information about the grouping modules. The minimum credits are 0. Can contain submodules or courses.

- GroupingModule(String name, int minCredits)
- getType()

StudyModule.java
extends DegreeModule.java

Saves information about the study modules. Can contain submodules or courses.

- StudyModule(String name, int minCredits)
- getType()

CourseUnit.java
extends DegreeModule.java

Saves information about the individual courses. The program treats CourseUnit as a module, despite it not being a module in Sisu, because its properties are exactly alike. Never contains any submodules.

- CourseUnit(String name, int minCredits)
- setIsActive(boolean newIsActive)
- boolean getIsActive()
- getType()

The User-class is an additional feature, which saves information about the user. Due to scheduling-related difficulties, we could not finish the full intended functionality of this feature: in this phase, the user-selected degree programme is appended into the *userData.json* file, but the user cannot yet view it in the profile dialog. In addition to the degree information, the file contains the user's name and student number, which is editable in the profile dialog shown in figure 3.

For the same reason, the modules implemented in the program store extraneous information that is not entirely relevant to the current functionality of the program; however, we decided to leave these in the source code, to ease further development. Currently, the user also cannot

select a course; in the present phase, only the hierarchy is shown; but the classes for the modules are implemented in such a way that this functionality can be added easily.

Distribution of work:

Lauri	Turkka	Manu
User.java	GUI design and implementation	API_reader.java
Documentation and class diagram	Aid with frontend-to-backend communication	All module class implementations
GUI communication with User backend		GUI communication with the above parts