

**Software Engineering**

**Course Project**

**Student-Course-Selection System**

1. **Introduction**

In this course, we will learn a systematic software development method and use what we have learned in the course to complete a practical student course-selection system project. Client-based, B/S(recommended), C/S

Usually, here are three roles in the system, student, teacher and administrator. Students are responsible for selecting courses. Teachers add, view and confirm courses. **Administrators** are responsible for importing information of students and teachers and arranging courses (subject TimeTable).

The whole system is divided into many function points and constraint points. **The more points you implemented, the higher the score will be.**

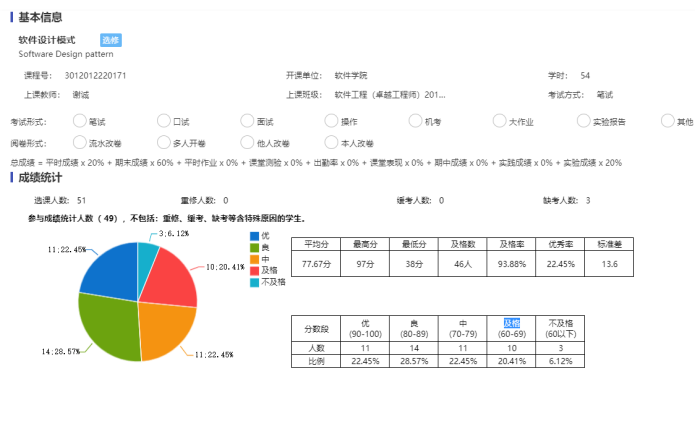
1. **Function (Functional requirements)**

Here are some common functions that need to be implemented. (Baseline)

* Log in/Log out
* Authentication? To verify whether the user has the right to access the page.
* Create semester (administrator), e.g. Fall-2020
* Create/Update/Delete a course (teacher) which he created.
* Arranging Courses and Teachers
* Select or withdraw a course (student)
* Show the class of a Course (Teachers)
* Show student’s TimeTable (students)
* Show Teacher’s TimeTable (Teachers)
* Scoring a student in a Class (Teachers)
* Viewing student’s courses’ score (students)

For student, the following functions can be considered to be added to the personal implementation of the system. (Optional)

* Credit statistics page for teacher. Example:



* Compulsory courses, Elective courses

For teacher, the following functions can be considered to be added to the personal implementation of the system. (Optional)

* Export selected student list (excel)
* Statistical chart of course selection

For administrator, the following functions can be considered to be added to the personal implementation of the system. (Optional)

* Manage the information of departments
* Arrange the course sequence to avoid conflict

1. **Constraint**

In order to make the system more reasonable and robust, we declare the following constraints.

* Students are not allowed to select the course with conflicting opening times.
* Courses can only be selected between September 1st and October 1st.
* The current course can only be selected when the number of students does not reach 50.
* No more than 4 courses can be taught by a teacher in one semester.

We can see that the last three constraints are static. **If you can change them to dynamically configurable conditions, you will get extra points.**

1. **Final Materials**

Throughout the course, you will submit the following documents periodically.

* Requirement Document
* Design Document
* Development Document
* Software and Source Code