Architecture Design

for

Contribution Assessment System

Version 1.0 approved

Prepared by Winnie/ Carson/ Mario /Sherry

Fish

2020/04/07

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Winnie/ Carson/ Mario /Sherry	2020/04 /07	Section 1-7	1.0

1. Overview

1.1 Project description

This project is for teachers to divide students into groups and calculate their contribution in group projects. It would also export the contributions of students as a file, which could be used in another grading system.

1.2 References

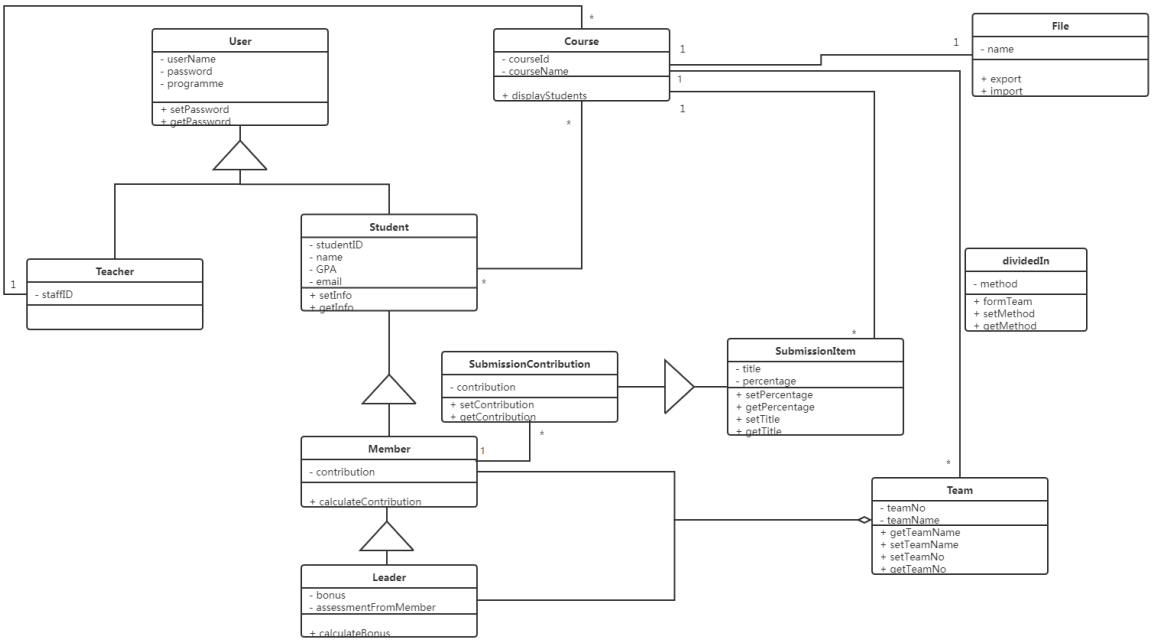
“Fish_181210_assignsubmission_file_Fish_Teamwork_System_SRS_2”.

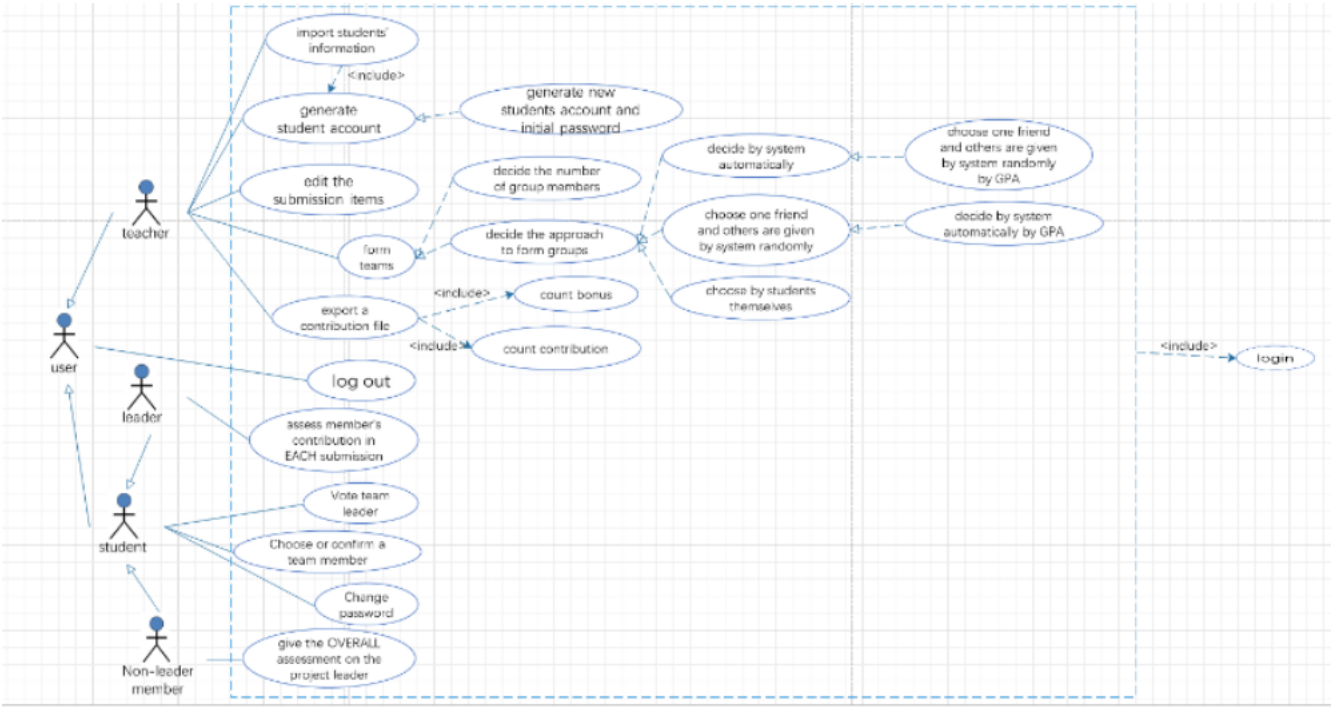
1.3 Design purpose

This design is for designer to identify subsystem, establish relationships between subsystem, and between objects in one subsystem, so that it would be clearer for designer to know about the structure of the whole system, how to develop it. This could maximum the data utilization and each subsystem could be managed and implemented well, and communicate to each other.

2. Overall description

2.1 Use case diagram and class diagram

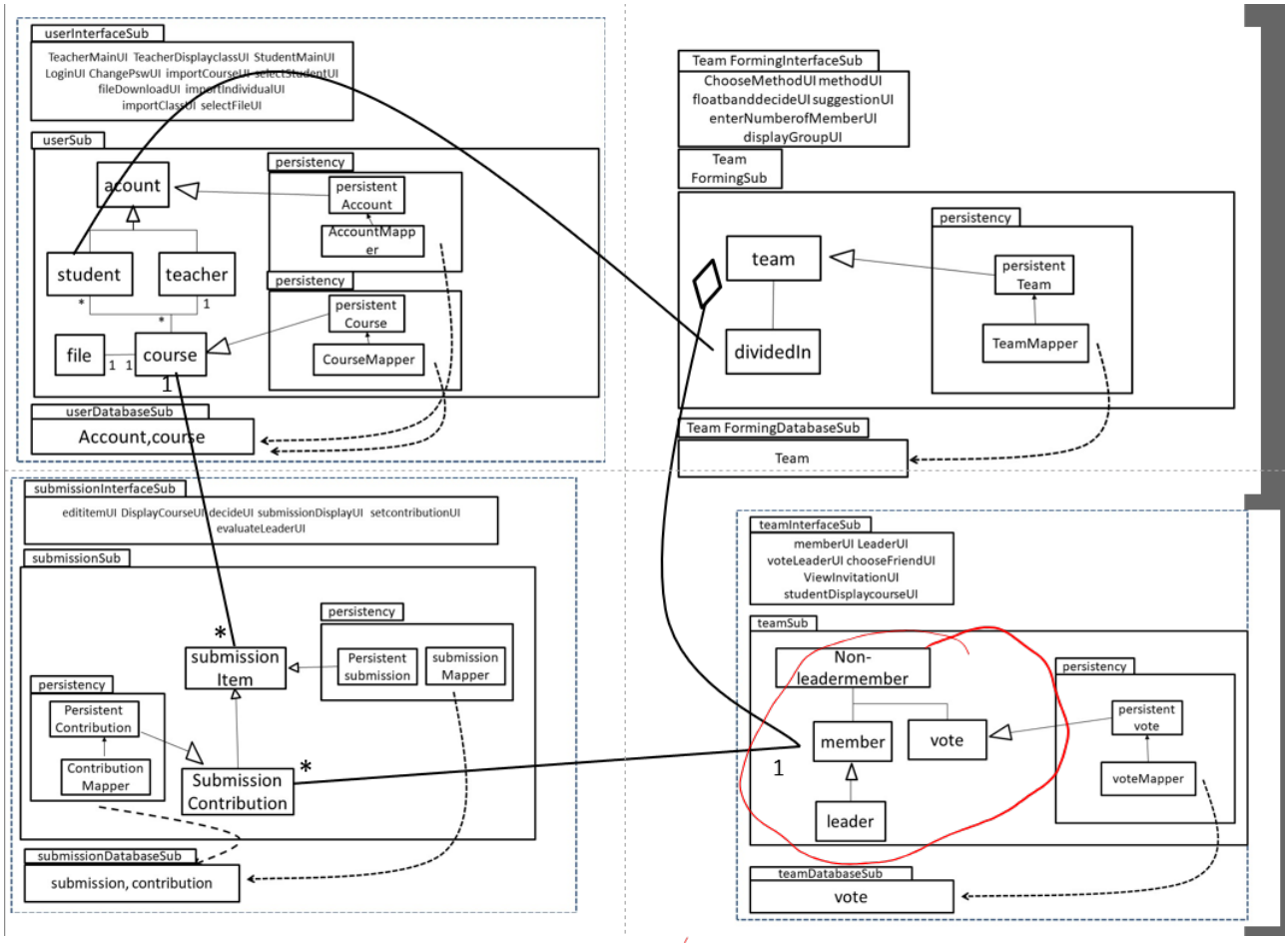




2.2 Design model

We use MVC 3-layered model in this design, in which each model shows different features, so that all subsystems can not only share the same application logic and storage with different interfaces, also the model would not depend on any view or controller subsystems, so that when a change in the model is detected, that change would be displayed in the view.

2.3 System architecture



In this diagram, not necessary so many details

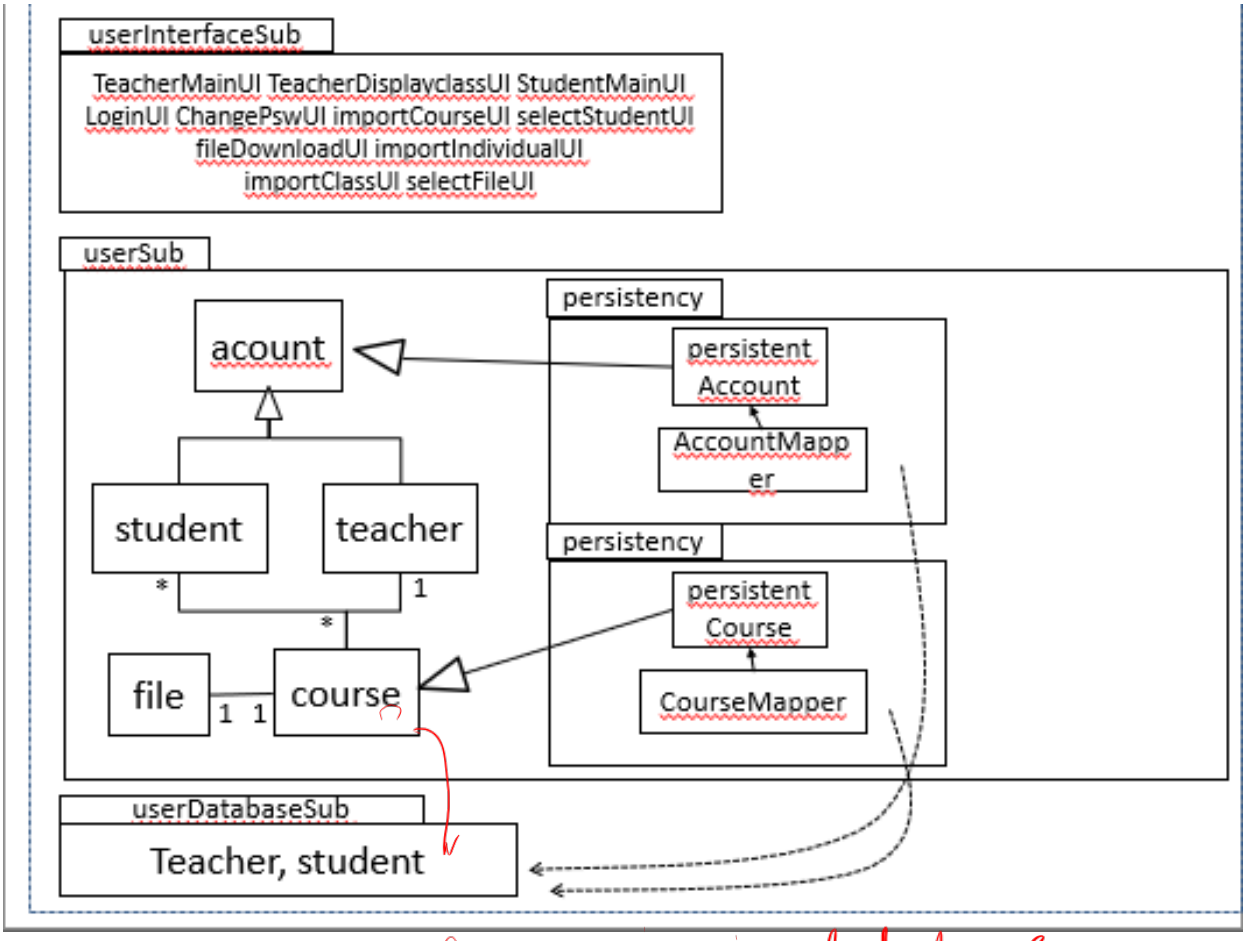
where are these classes from?

The system would divide into 4 subsystems:

- User subsystem including functions of login, logout, changing password, importing information into the system database, generating accounts and initial passwords, and export information stored in system database as a file.
- Submission subsystem, including function of editing submissions.
- grouping subsystem, including functions of dividing students into teams, and students inviting their friends.
- Team subsystem, including functions of voting leader and assessment by leader and by non-leader members.

3. System architecture

3.1 User subsystem



Course is in database

3.1.1 Description

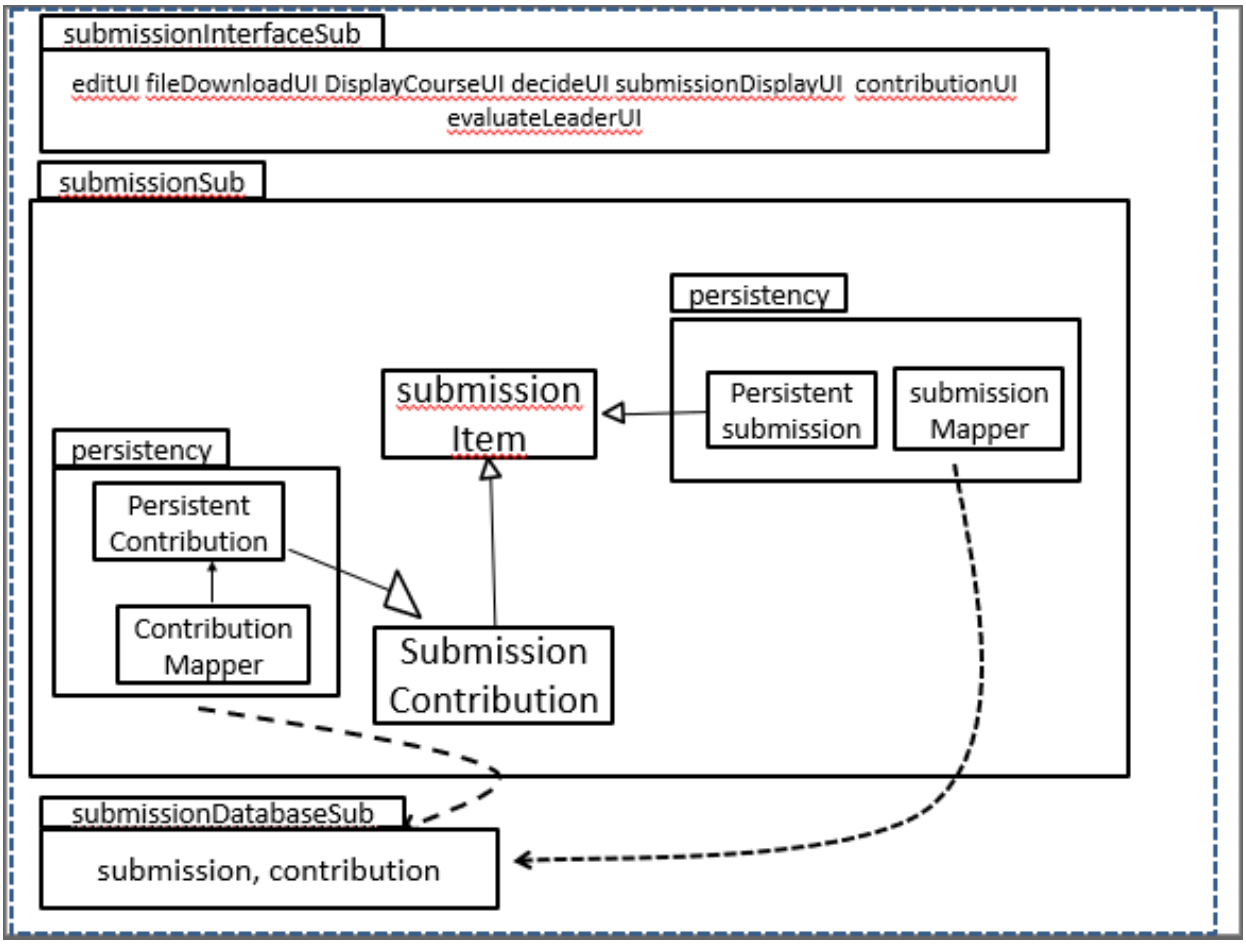
This user subsystem includes TeacherMainUI, TeacherDisplayclassUI, StudentMainUI, LoginUI, ChangePswUI, importCourseUI, selectStudentUI, fileDownloadUI, importIndividualUI, importClassUI, and selectFileUI, userSub, which includes objects of account, student, teacher, file and course, and userDatabaseSub, which has the table of students and teachers.

3.1.2 Database

There are two tables in this database, including students and teachers. In students table, it has attributes of account, password, course, gpa, email, name and ID. Table of teachers would have attributes of account, password, course, name and email.

+ tables

3.2 Submission subsystem



3.2.1 Description

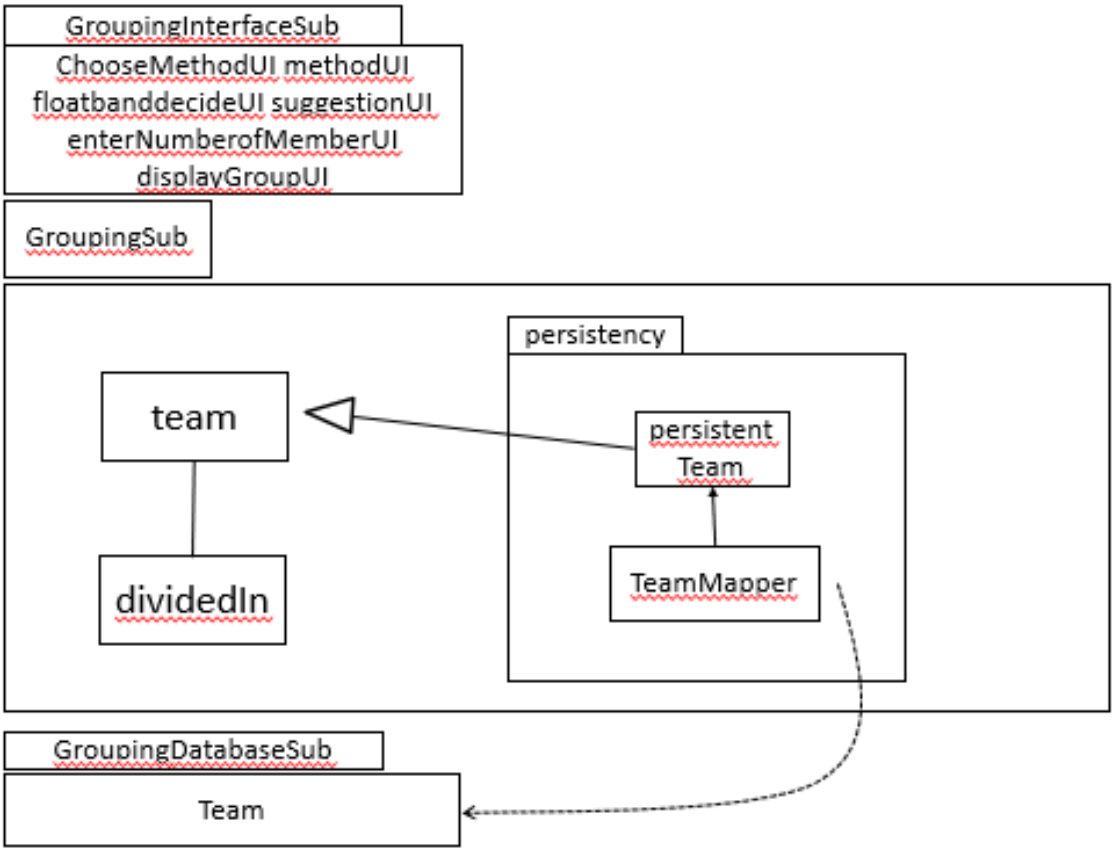
The grouping subsystem includes submissionInterfaceSub, which includes edititemUI, DisplayCourseUI, decideUI, submissionDisplayUI, setcontributionUI, and evaluateLeaderUI, submissionSub, which includes objects of submission item and submission contribution, and submissionDatabaseSub, which includes table of submission and contribution.

3.2.2 Database

The submissionDatabaseSub would have the table of submission, which has the attributes of course, submissionID, submission name and submission contribution.

one table or two tables ?

3.3 Grouping subsystem



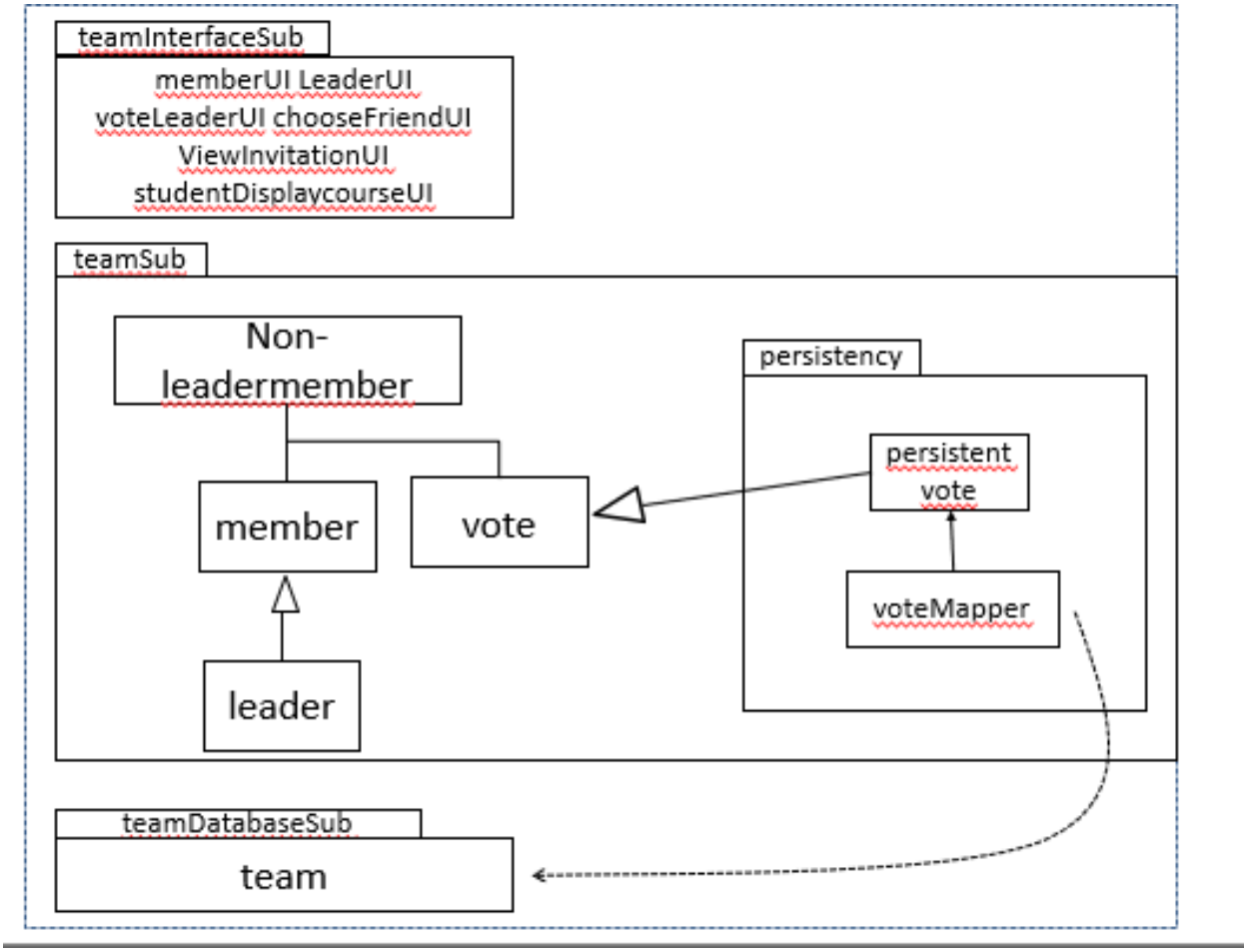
3.3.1 Description

Grouping subsystem includes two functions, help to divide the students and let the students to invite their friends.

3.3.2 Database

The database contains all the information about all the teams, including the ID, the position, the contribution and the bonus of every team members and the team number and the belonging course of the team.

3.4 Team subsystem



No tables for leader or member ?

3.4.1 Description

Team subsystem includes two functions, to vote the team leader and assess the members and leaders according to the team members' position in the team.

3.4.2 Database

The database contains all the vote record of each team members. Specially, since the assessment will be design as multiple-choice question for team members to select, the result will be also recorded as vote record as shown in the database.

4. Assessment

4.1 Stability

The system is stable since the users perform a series of operations by calling functions and passing parameters. If certain changes are made to certain components, the changes will be reflected in all of relevant areas.

4.2 Reusability

Some functions of the system have strong reusable capabilities, such as the function of group formation. But the reusability of the calculation part of the specific contribution value is still questionable. Whether the new calculation method and usage are suitable for our current function will affect the level of reusability.

4.3 Scalability

The system can add additional functions, data sets or sub-functions for expansion. This can be done by adding subsystems or directly adding functions.

5. Alternative design (optional)

6. More considerations

The document is intended for developers, project managers, testers and documentation writers. The Architecture Design file contains the overview, the overall description, and its main part, system architecture and so on. Though it is based on the latest version of SRS, we suggest you to finish the overview and overall description part then read the part that is most relevant to you. The system architecture section which explained the subsystems in the project is especially for developers to understand the system better.

7. Appendix