Software Requirements Specification

for

Teamwork System

Version 1.4 approved

Prepared by Winnie/ Carson/ Sherry/ Mario

Fish

2020/3/30

Table of Contents

1. Introduction 1
   1.1 Purpose 1
   1.2 Document Conventions 1
   1.3 Intended Audience and Reading Suggestions 1
   1.4 Project Scope 1
   1.5 References 1
2. Overall Description 1
   2.1 Product Perspective 1
   2.2 Product Features 2
   2.3 User Classes and Characteristics 3
   2.4 Operating Environment 3
   2.5 Design and Implementation Constraints 3
   2.6 User Documentation 3
   2.7 Assumptions and Dependencies 3
3. System Features 3
   3.1 Login 3
   3.2 Import students' info, including students' account generation 4
   3.3 Edit submission items 5
   3.4 Form teams 6
   3.5 Export a contribution file 8
   3.6 Change password 8
   3.7 Choose team member 9
   3.8 Vote team leader 10
   3.9 Evaluate team members 10
4. External Interface Requirements 12
   4.1 User Interfaces 12
   4.2 Hardware Interfaces 14
   4.3 Software Interfaces 14
   4.4 Communications Interfaces 14
5. Other Nonfunctional Requirements 15
   5.1 Performance Requirements 15
   5.2 Safety Requirements 15
   5.3 Security Requirements 15
   5.4 Software Quality Attributes 15
6. Other Requirements 15

Revision History

Name	Date	Reason For Changes	Version
Winnie/ Carson/ Sherry/ Mario	2020/3/6	Write the SRS except section 3 and section 4.1	1
Winnie/ Carson/ Sherry/ Mario	2020/3/15	Write section 3 and section 4.1, modified section 1.3, 2.1, 2.2, 4.2.	2

Winnie/ Carson/ Mario /Sherry	2020/3/22	Finish section 3 and section 4.1, slightly modified section 1.5, 2.2 and 5.	1.3
Winnie/ Carson/ Mario /Sherry	2020/3/30	Modified section 3, 4 and 6.	1.4

1. Introduction

1.1 Purpose

The document is the software requirement specification. It's written for list all the system feature and requirement during the developing and coming usage.

1.2 Document Conventions

First level headings are 18pt big and the Secondary titles are 14pt big, both bolded. The tertiary titles are 12pt big. The font type of all the headings and titles is Times New Roman. For the contents, the font size is 11pt and the font type is Arial. The text in italics and parentheses are explanatory notes. The context model in section 2.1 and the use case diagram in section 2.2 is drawn by "Edraw" *(a Graphic design software).*

1.3 Intended Audience and Reading Suggestions

The document is intended for developers, project managers, users, testers and documentation writers. This SRS contains the overview section, system feature section, interface requirements and other nonfunctional requirements. Please read the overview sections and then proceed through the sections that are most pertinent to you. The system feature section which explained the features of the system is especially for users to understand the system better.

1.4 Project Scope

It's a web application which can help teachers to form teams and then assess the contributions from team members during the process of task fulfillment in a course project.

1.5 References

Workshop III Teamwork Project V4.

2. Overall Description

This section will give an overview of the whole system, which would be explained in its context to show its operating environment and introduce the basic functionality of it.

2.1 Product Perspective

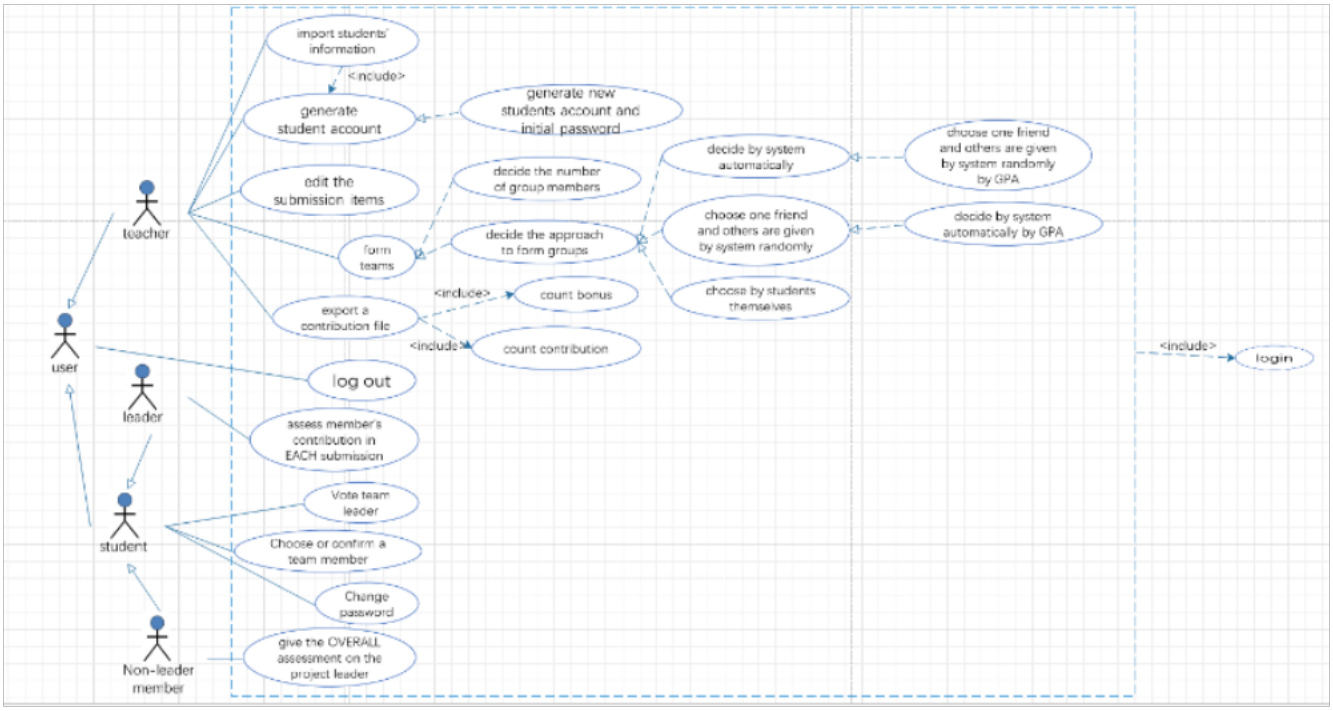
This system is a web application that would help teachers to divide students into groups and calculate their contribution in group projects. Since this system would collect information of lots of students, team and team work, the student information database and the team information database are necessary. The whole system consists of a group forming system, an assessment system, a contribution calculating system, the security system and the maintenance system to

implement the functionalities of this system.

A Teamwork system

2.2 Product Features

Teachers could first use this web application to divide students into groups, including deciding the number of members in one group and the approach to form groups. There are four approaches for teachers to choose, including letting students to form their group, decided by system automatically, a student choosing a friend as a group member and other members given by system randomly and divided by system according to the GPA. Also, the system would record their group information and collect their group project submission, so that teachers could be able to manage their submission in this system. Finally, teachers could export the list file of contribution of all students, which could be used in another Grading system.



User case diagram

2.3 User Classes and Characteristics

The system would have two groups of users, including teachers and students. Teachers would be able to import students' information into the system and generate accounts and initial passwords for them. Teachers would also decide the approach and the number of members in one group to divide students into groups. They could also manage students' submission and export the list file of students' contribution.

Students could login the system and change their password. With permission, students may be able to form the groups by themselves or choose a friend as their group member. After the group formed, they could vote to decide a team leader. The team leaders have the right to assess members' contribution and non-leader members have the right to assess their team leaders.

2.4 Operating Environment

It's a web application so it runs on a server, accessible by browsers.

2.5 Design and Implementation Constraints

Except for that students could use other public Internet when they submit their project, users need to use the Internet in the campus in most of the situations, because most of the functionalities in this system need to access to those two databases.

In order to avoid the problem that students' information leakage, the system must insure the security of the database and develop the security system for the whole system.

2.6 User Documentation

The user documentation would consist of the user manuals, introduction of each functionality, troubleshooting section, and online help, which would include the tutorial videos that teaches user that how to use each functionality in this system. The documentation would be delivered to the user both in a hard copy and the pdf file, with the hyperlink to the tutorial webpage.

2.7 Assumptions and Dependencies

The system would be available under the assumptions that the system is permitted to access to the specific database by using the campus' Internet.

3. System Features

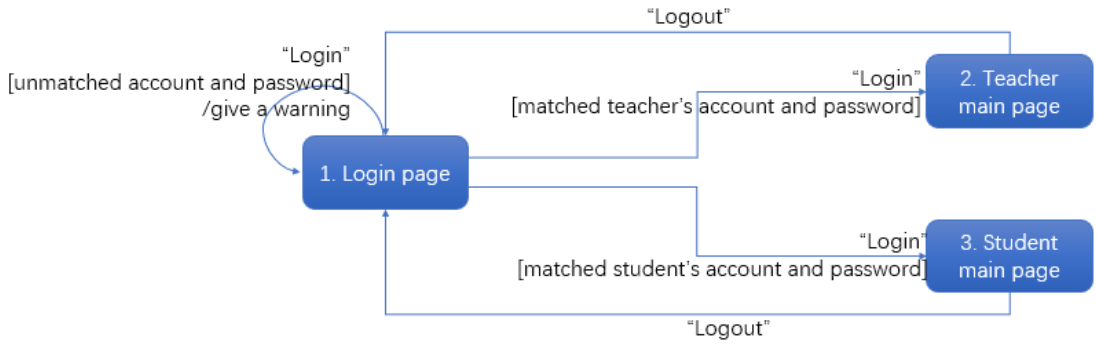
3.1 Login

3.1.1 Description and Priority

This function is for teachers and students to login. After login, system should display different main pages for them.

The priority of this function is high, without it, the system cannot implement its further functions correctly

3.1.2 Stimulus/Response Sequences



3.1.3 Functional Requirements

REQ-1: Type in the account and password, and press "Login". The system should check if the account is valid and if the password matches the account. If matched, then system allows users to login and go to the specific main page. Otherwise, a warning should be given.

REQ-2: Users can logout on their main page. After press "Logout", system should make the account logout.

REQ-3: TBD

3.2 Import students' info, including students' account generation

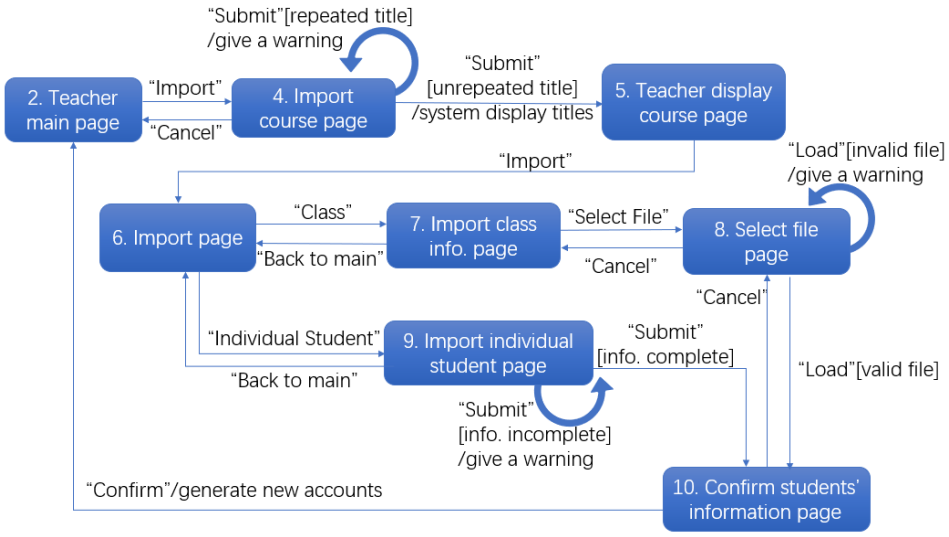
3.2.1 Description and Priority

This function is for teachers to import students' information and generate their accounts. Before importing students' information, course must be imported first. Teacher can edit the course title as well.

Teachers choose to import for the whole class or for some students are enrolled into the class later. For the whole class, the information is imported with an excel file. For those students, their information can be typed by the teacher individually. The system would generate accounts for each student whose information is new to the database. No account will be generated twice.

The priority of this function is high, without it, students cannot login the system and the further functions of the system cannot be implemented.

3.2.2 Stimulus/Response Sequences



3.2.3 Functional Requirements

REQ-1: When "Import" pressed on the teacher's main page, the system will see whether courses have been imported. If yes, teachers can go to the course display page. If no, teachers have to import courses first.

REQ-2: If teachers submit the course title which already existed, a warning will be given. If the title is a new one, the system will go to the display page and display the title.

REQ-3: If teachers want to edit the course title again, it can be reached from the display page.

REQ-4: After teachers choose to import the information of the whole class by pressing "Class". If the file selected is invalid data file, the system should give a warning. If the file is valid, the students' information confirm page should show up.

REQ-5: If teachers press "Confirm", the system should generate those new accounts and initial password. If "Cancel", then go back to the select file page.

REQ-6: If teachers want to add student information individually, they could press "Individual Student" to import the student's information. If the all the information is typed, the confirm page should show up as REQ-5.

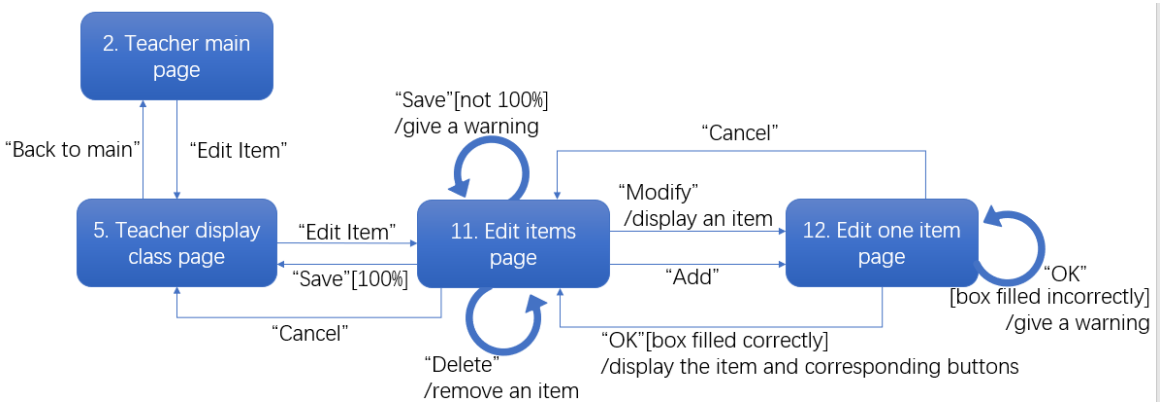
REQ-7: TBD

3.3 Edit submission items

3.3.1 Description and Priority

The function is for teachers to add, modify and delete the submission items of the course.

The function priority is high, because this is for system to collect the submission information of all students to calculate their contribution.



3.3.2 Stimulus/Response Sequences

3.3.3 Functional Requirements

REQ-1: If teachers click "Cancel" in the edit one items page, system should go back to the edit items page, without saving any changed information or new added submission.

REQ-2: Teachers select a submission, and press "Modify". Enter the information which need to be changed, and press "OK". If the entered information is valid, the change will be saved in that submission. Otherwise, a warning will be given.

REQ-3: Teachers select a submission and press "Delete". The selected submission will be deleted.

REQ-4: Teachers click "Add" in the edit items page, and a add new submission would show in the edit one item page. If the information of the new added submission is not complete, the system would give a warning.

REQ-5: If the sum of percentage of each submission takes is not equal to 100%, the system should give the warning; otherwise the system saves the change.

REQ-6: Teachers press "Cancel" in edit items page to go back to teacher display course page.

REQ-7: TBD

3.4 Form teams

3.4.1 Description and Priority

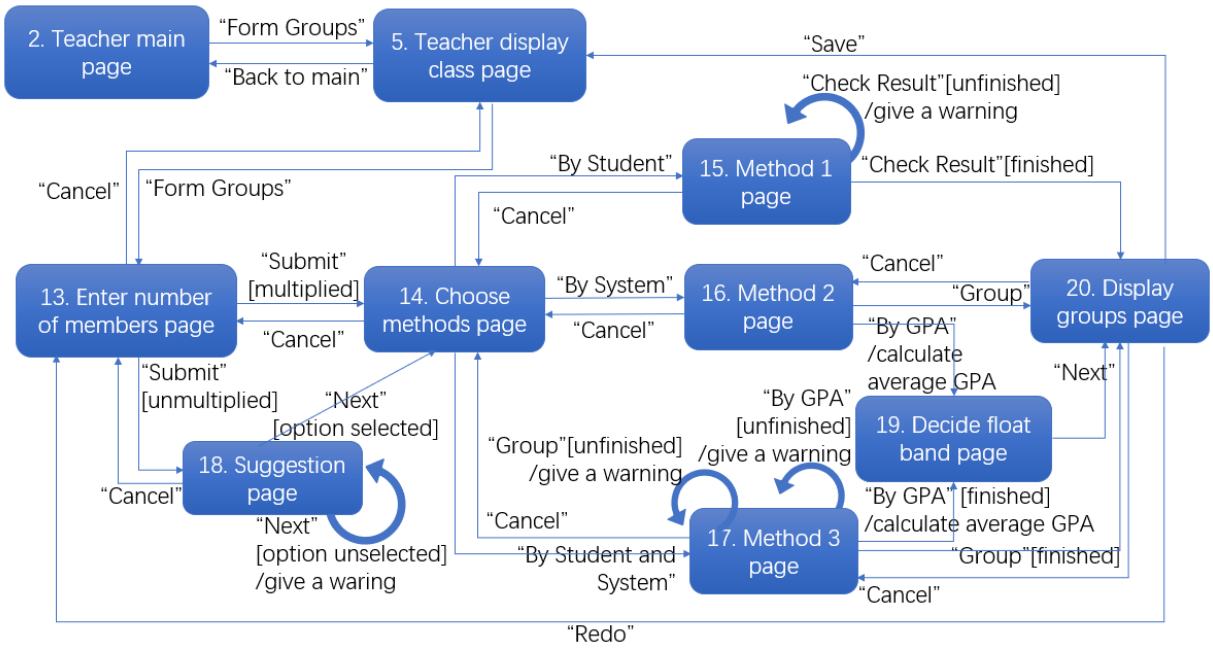
This function is for teachers to divide students in a class into several groups for their group projects. It would ask teacher to decide the number of members in a group and then give suggestion for teacher to decide the groups to have other number if the number of students is not the multiple of the decided number.

Then, system would let teacher to decide how to divide the group. This application can offer several choices to form teams for students. *Method 1.* All the members are chosen by students themselves. *Method 2.* The members are decided by system automatically. *Method 3.* A student can choose one friend and others are given by system randomly. *GPA* might be considered in *Method 2* and *Method 3*. It is decided by teacher. If GPA is considered, the system will calculate the average GPA for the whole class, and teacher can decide the floating bands around the average GPA. In

the above methods, *Method 1* and *Method 2* have highest priority. The priority for *Method 3* is middle. The priority for *Method 2* with *GPA* is middle too. The *Method 3* with *GPA* has lowest priority.

However, this function priority is high, because after the teacher decides how to form the group and the students grouped.

3.4.2 Stimulus/Response Sequences



3.4.3 Functional Requirements

REQ-1: After enter the page to form groups, system would first show the list of the classes to let teachers select which class of students to divide into groups.

REQ-2: After teachers enter the number of members for each group, if the total number of students is not the multiple of the number teacher enters for members in each group, the system should give solutions for teachers to choose in the suggestion page.

REQ-3: If not solution is selected in the suggestion page, the system would give a warning. After submit the solution, the system would go to the choose methods page to let teachers select the method to divide groups.

REQ-4: If teachers select to form the groups by students, then system get the group forming result decided by students and shows it in the display groups page.

REQ-5: If teachers select to divide groups by system automatically, system would let teachers decide whether according to GPA. Then, system combines all needed information together to divide groups and display the division result in the display groups page.

REQ-6: If teachers select to divide groups both by students and by system, system should first let teachers decide whether according to GPA. Then system gets friends information from students and combines with all needed information to divide groups and display the division result in the display groups page.

REQ-7: In the display groups page, if teachers confirm the result and press “Save”, then system would save the groups information in the database; If press “Redo”, then system would give the division result and to back the enter number page.

REQ-8: TBD

3.5 Export a contribution file

3.5.1 Description and Priority

This function is for teachers to export the contribution list of students as a file, which contains name, ID, contribution and bonus of each student. This file could be used in the Grading system. Also, before system export the file, it would first calculate the contribution for each student whose information is recorded in the system according to the following formula:

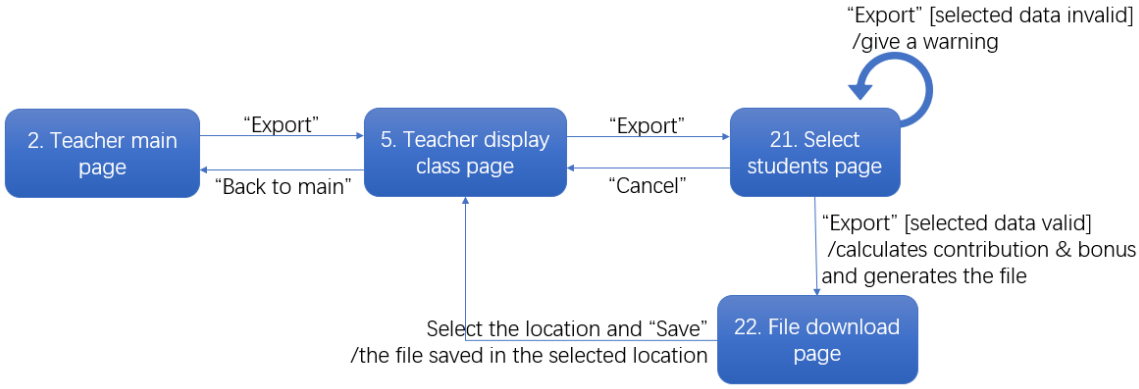
$$\text{Overall Contribution} = \sum_{i=1}^{n} P_i * C_i$$

Where pi is the assessment percentage for submission i, ci is the contribution of a student in submission i and n is the total number of submissions.

Also, it would count the bonus for the leader: calculate average.

The priority of this function is high, because without it, teachers could not export the contribution list of students.

3.5.2 Stimulus/Response Sequences



3.5.3 Functional Requirements

REQ-1: If teachers press “Export”, system would let teachers choose which class information to export. If teachers make no selection, the system would give a warning. Otherwise, the system would export the file and save it in the selected location in the computer.

REQ-2: If teachers press “Cancel” in the students selecting page, then the system would go back to the teacher display class page.

REQ-3: TBD

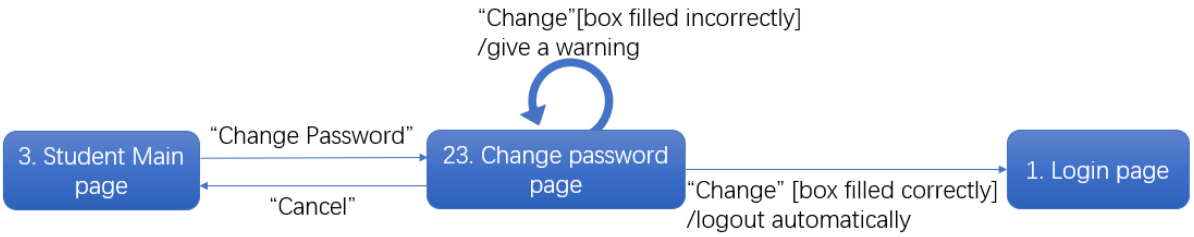
3.6 Change password

3.6.1 Description and Priority

This function is for students to change their password.

The priority of this function is middle. It is to protect the information of student account.

3.6.2 Stimulus/Response Sequences



3.6.3 Functional Requirements

REQ-1: After students enter the old password and new password, system should check if the old password is correct and the two times of new password are consistent. If there is a mistake – the old password is incorrect, the two times of new password are inconsistent or one of the boxes is empty, then the system would give a warning. Otherwise, the system would save the new password into the database.

REQ-2: TBD

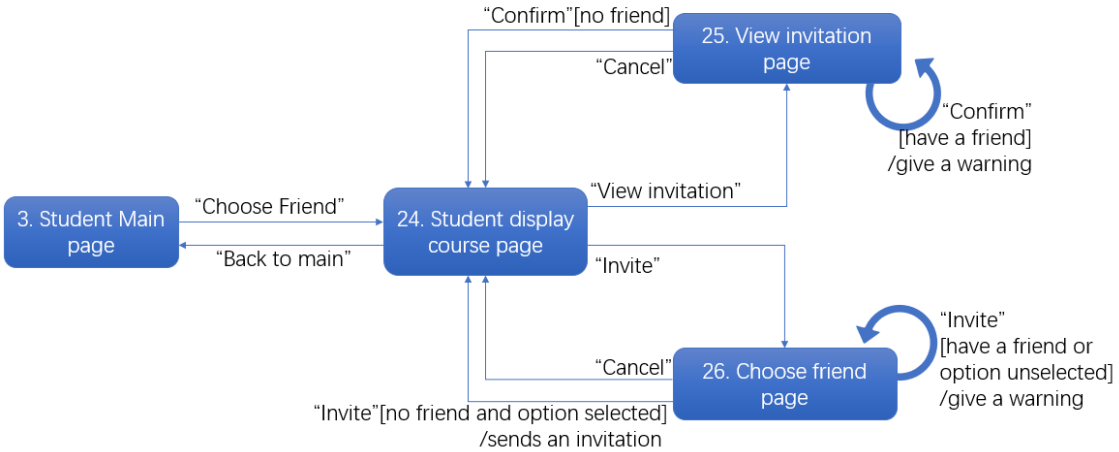
3.7 Choose team member

3.7.1 Description and Priority

This function is for students to choose one friend as their team member, which is a part of the *Method 1* and *Method 3* in 3.4 to form groups.

The priority of this function is high.

3.7.2 Stimulus/Response Sequences



3.7.3 Functional Requirements

REQ-1: Students press “Choose Friend” to go to the student display course page.
REQ-2: Students could select “Invite” to invite friend if they have no friend. If one has already had a friend, then if he invites friend, the system would give a warning.
REQ-3: Students could select “View Invitation” to confirm invitation from others if they have no friend. If one has already had a friend, then when he confirms the invitation, the system would give a warning.
REQ-4: TBD

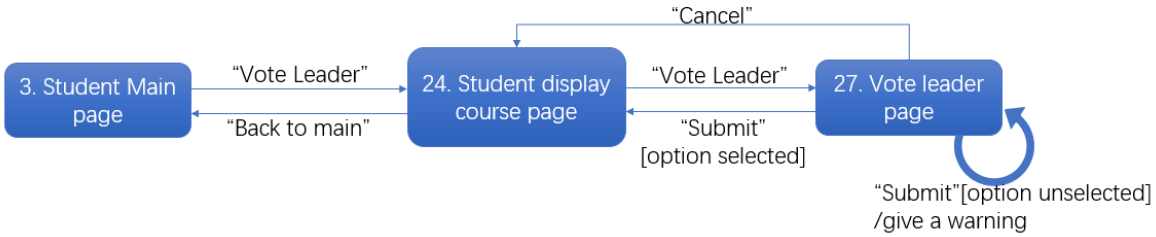
3.8 Vote team leader

3.8.1 Description and Priority

This function is for students in a group to vote and choose their group leader.

The priority of this function is high, because the assessment of the group leader is the important part for system to calculate the contribution of students.

3.8.2 Stimulus/Response Sequences



3.8.3 Functional Requirements

REQ-1: Students press “Vote Leader” and go to the student display course page.
REQ-2: Students make their selections and click “Submit”, if the selection is valid, then system would go to the result display page. If students make no selection and submit, then the system would give a warning and stay in the current page. If students press “Cancel”, then system would go back to the student display course page.
REQ-3: TBD

3.9 Evaluate team members

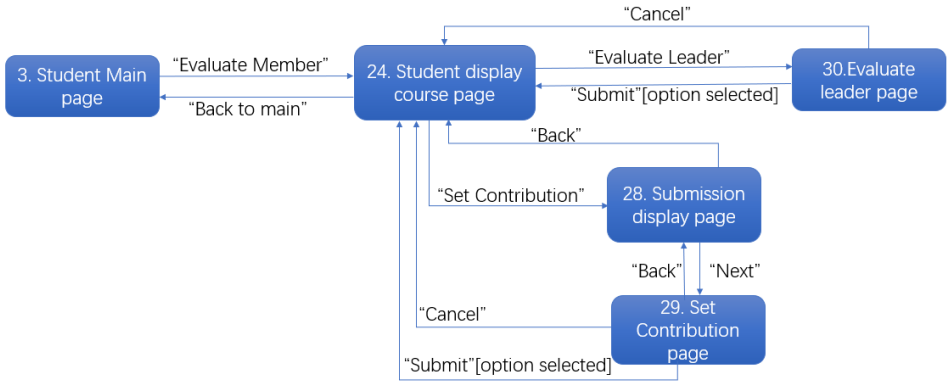
3.9.1 Description and Priority

The function is for group members to evaluate themselves. Leader give assessment to each group member in EACH submission item if he likes. The contribution is classified in four levels: (a) full(1.0), (b) fair(0.67), (c) little(0.33), (d) none(0). If the leader does not give the contribution, the contribution is full by default.

The other members have right to give the OVERALL assessment on their project leader after the project is finished (very good <2>, good <1>, fair <0>, bad <-1>, very bad<-2>). If a member does not give the assessment to his leader, the assessment is fair by default.

The priority of this function is high, because the assessment is the necessary data for system to calculate the contribution and bonus for students.

3.9.2 Stimulus/Response Sequences



3.9.3 Functional Requirements

REQ-1: If the account is belonged to a group leader, then the system would show the "Evaluate Member" in the course display page.

REQ-2: Leaders could go to the set contribution page to evaluate his group members. After a submission is selected, then leaders can set contribution for himself and his members. The selections would be saved after "Submit" pressed. If the leader makes no selection for one member and presses "Submit", the contribution is full by default.

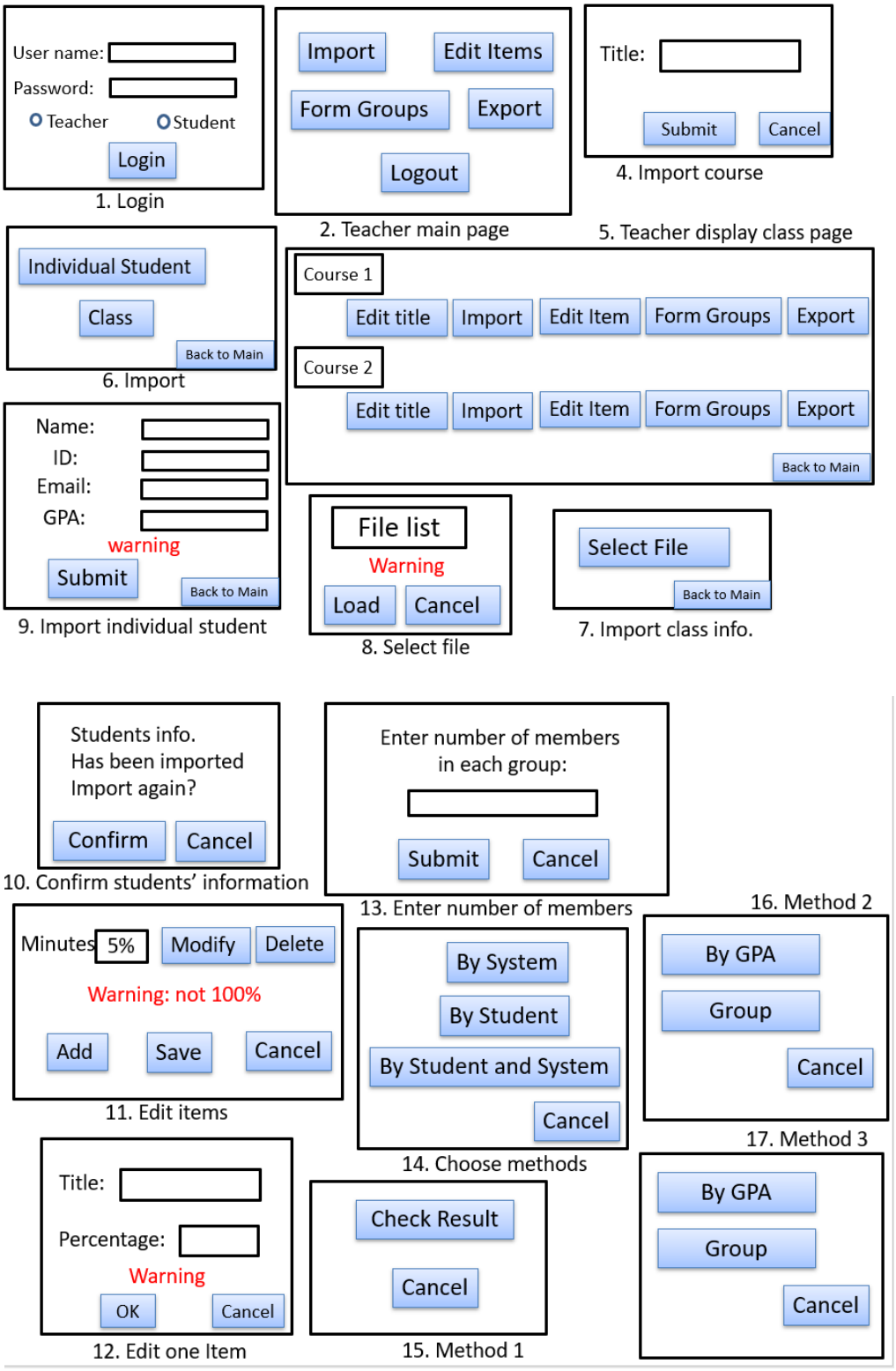
REQ-3: If the account is belonged to a non-leader member, then he could only access to the evaluate leader page through the button "Evaluate Leader". The selections would be saved after "Submit" pressed. If the member makes no selection for the leader and presses "Submit", the assessment is fair by default.

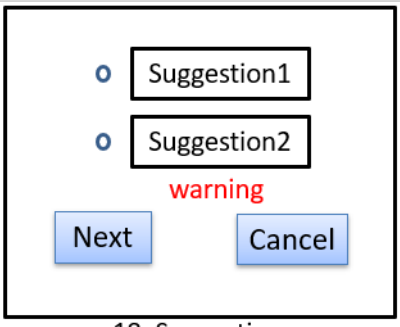
REQ-4: After the evaluation of leader is over, the system would calculate students' contribution and bonus and record the result in the database.

REQ-5: TBD

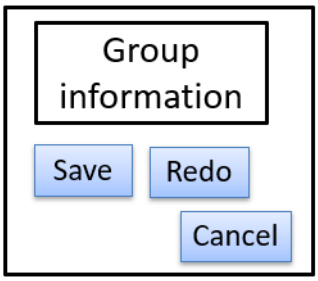
4. External Interface Requirements

4.1 User Interfaces



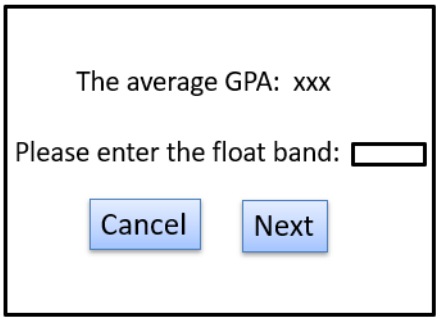


18. Suggestion

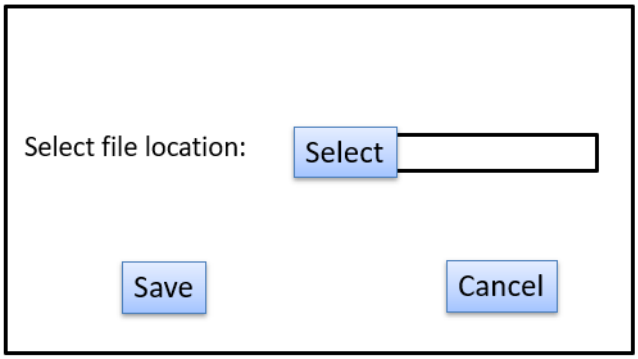


20. Display group

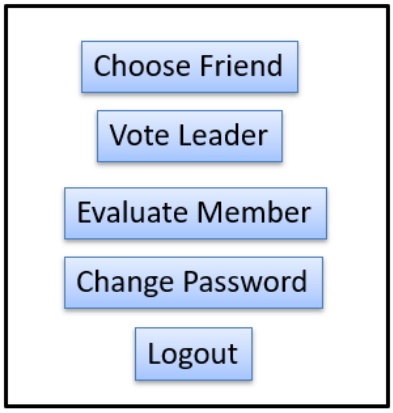
21. Select students



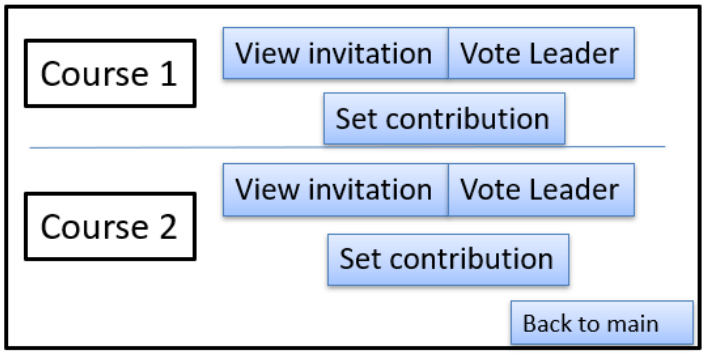
19. Float band decide page



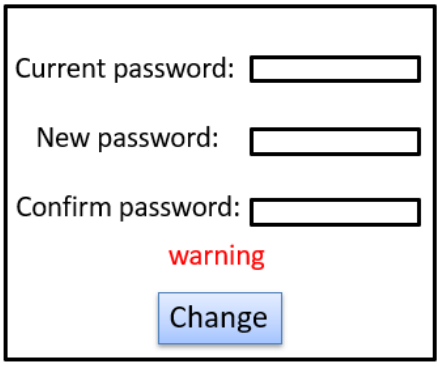
22. File download



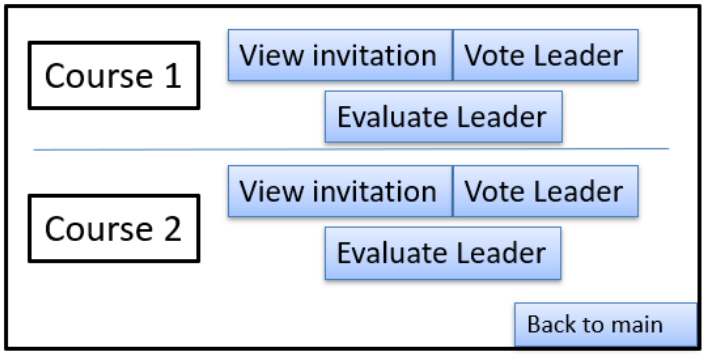
3. Student main page



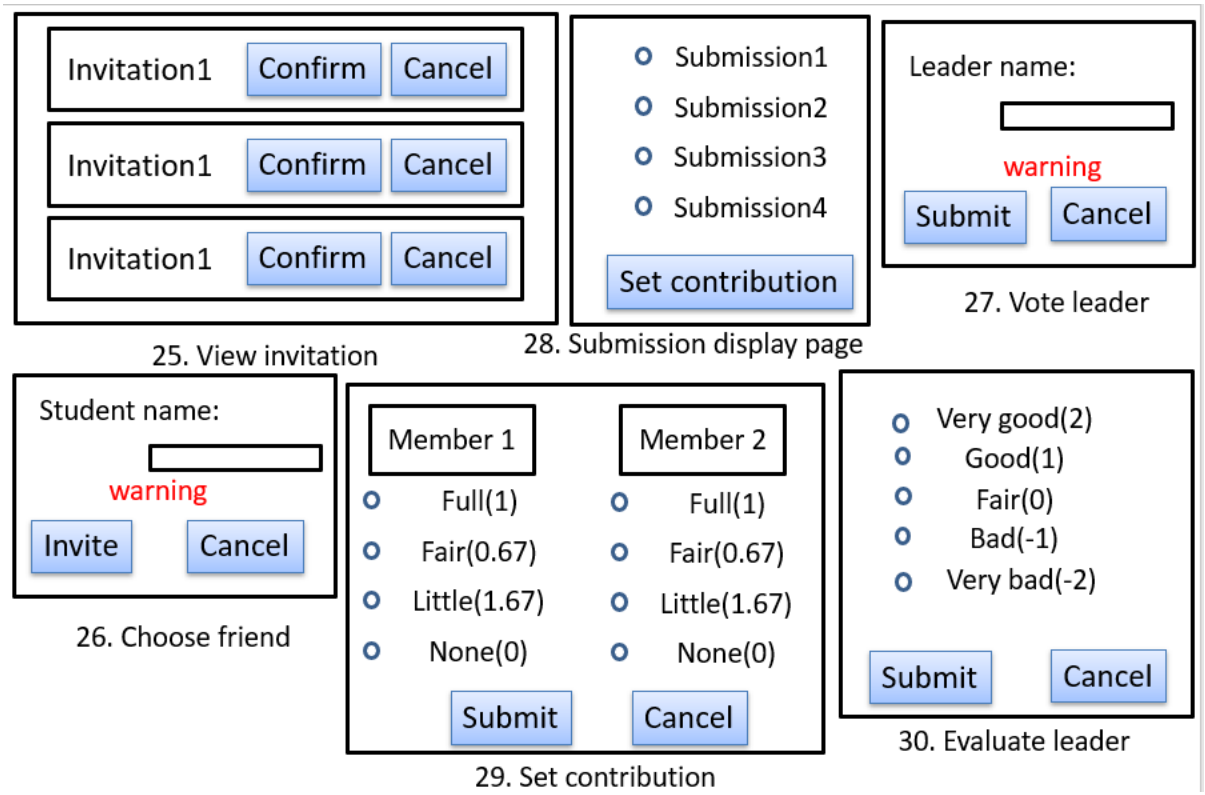
24. Student display course page(leader)



23. Change password



24. Student display course page(member)



4.2 Hardware Interfaces

None.

4.3 Software Interfaces

This software product can be applied on many different operating systems since it's a web application. The communication is based on transfer protocol like TCP/IP and so on. The basic programming language of this software is python, it takes flask frame for the main interface.

4.4 Communications Interfaces

The student information is imported to the web as an excel file. Besides, exported contribution file will be used in another existing Grading system.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

For real-time requirement, every search action in this software should be finished within 1 second. The system should work well in different operating systems and have consistent style and operations.

5.2 Safety Requirements

N/A

5.3 Security Requirements

Students can only change their passwords after they login, and they need to login again after changing passwords. Only team leaders have right to assess members' contribution in each submission. The system also needs to protect users' information, preventing message leaking.

5.4 Software Quality Attributes

This software should guarantee usability, it should be easy to use, and user manual should be provided if necessary. Reusability and maintainability should also be considered, which means the software should be easy to update and add new requirements in the future. Because the exported contribution file needs to be used in another existing grading system, the exported files must be compatible with that system.

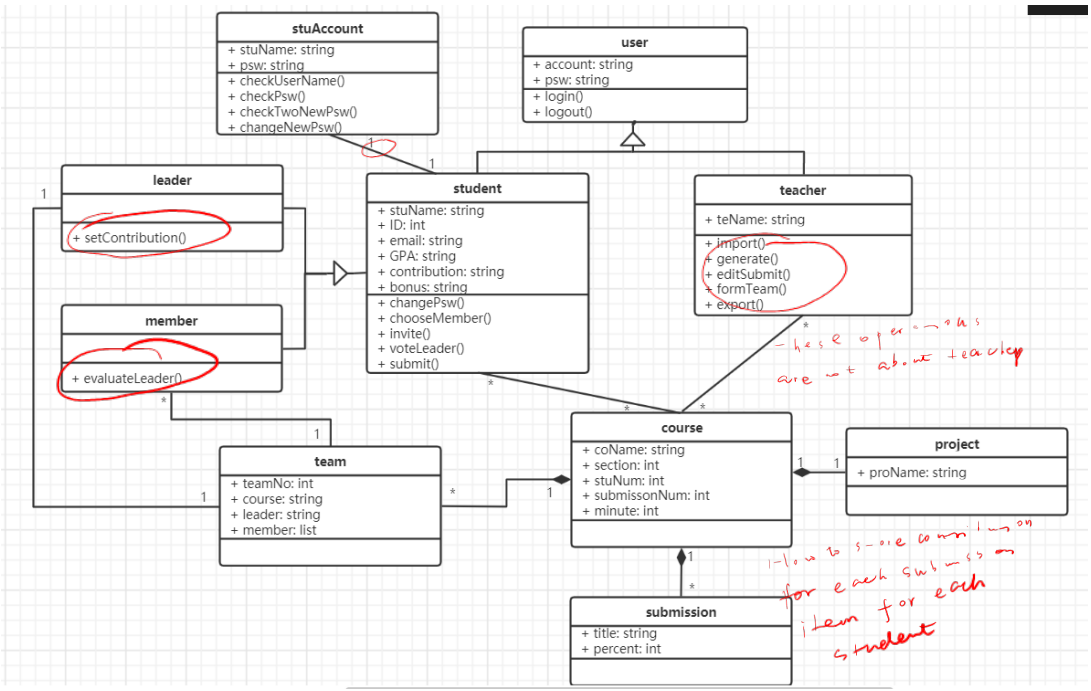
6. Other Requirements

None.

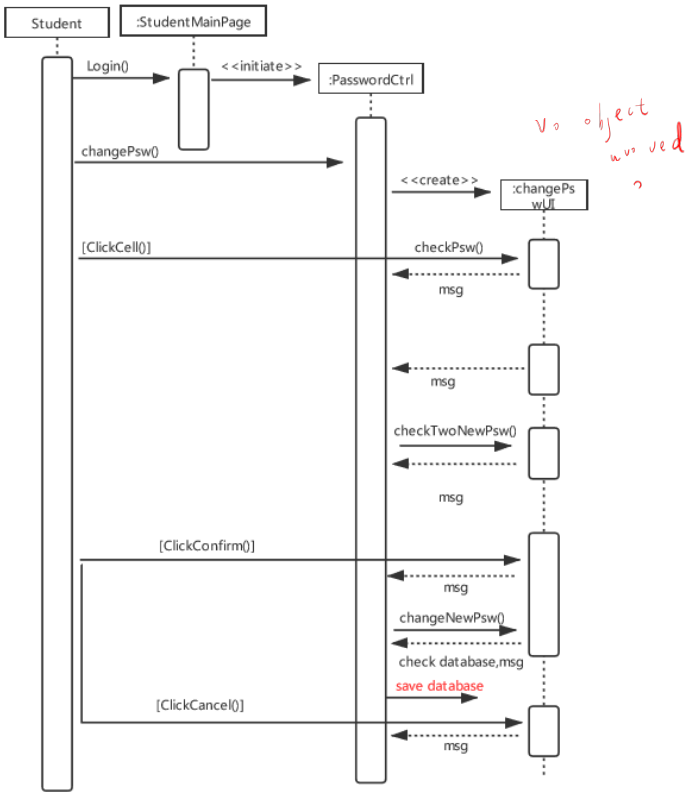
Appendix A: Glossary

None.

Appendix B: Analysis Models



Class diagram



Sequence Diagram for the use case Update Password (student)

Appendix C: Issues List

N/A