Software Requirements Specification

for

Teamwork System

Version 1.3 approved

Prepared by Tian Zhanming, Zhang Qiaomu, Zhang Haibin, Li Jiajia

Software Development Workshop III

2020/3/28

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Zhang Qiaomu, Tian Zhanming, Zhang Haibin, Li Jiajia	2020/03/08	Prepare initial version	1.0
Zhang Qiaomu, Tian Zhanming, Zhang Haibin, Li Jiajia	2020/3/14	Prepare section 3 and section 4.1	1.1
Zhang Qiaomu, Tian Zhanming, Zhang Haibin, Li Jiajia	2020/3/21	Finish all the content and fix the mistakes according to the feedback	1.2
Zhang Qiaomu, Tian Zhanming, Zhang Haibin, Li Jiajia	2020/3/28	Finish a Sequence Diagram for the use case Update Password (student) and Class diagram	1.3

1. Introduction

1.1 Purpose

The purpose of this document is to capture the complete software requirements and preview some elements of the analysis model of the program Teamwork System (T.S.), which is developed by Eagle. It also describes non-functional requirements, design constraints and other factors necessary to provide a complete and comprehensive description of the requirements for the software.

1.2 Document Conventions

This document will be written using two different fonts, Arial for the main content. Time New Roman for varying levels of titles. Main content's font size should be in 11pt. Bold fonts are used for emphasis or as headings. Each heading may contain subheadings in ascending order. Every requirement statement is assumed to have its own priority as to define in a most appropriate way the system behavior. Besides, various figures represent the described system, where it is needed, and serve only for a better understanding of the deployment.

1.3 Intended Audience and Reading Suggestions

This document is intended for any users, developer and tester that needs to understand the basic system architecture and its specifications. Here are the potential uses for each one of the reader types:

- Developer: The developers who wanted to modify the system, must read the requirement and other features definition in order not to misunderstand the functions and do some reduplicative works.
- User: Users should be confirmed to understand the correct use of this system and some essential environment that is needed before using it. Then to find out whether the necessary functions exist, please check product features.
- Tester: Tester must make sure if the functions work well and if those features fit the requirements.

For each one of the reader types to better understand this document, here is a suggestion of the
chapters to read in this document:

- Developer:(1.1 ,1.3 ,2.2 ,2.3 ,2.5 ,2.7 ,3 ,4 ,5 and rest)
- User:(1, 2.2 ,2.3, 2.6, 4.1 ,5)
- Tester:(1 ,2.1 ,2.2 ,2.5 ,2.7 ,3 ,4 ,5)

Overview

1. Introduction: Provide an overview of the application, describe the document structure and point the individual objectives.
2. Overall Description: Provide the specification of the system model, the classes model, the main constraints and the list any assumed factors used within this document.
3. System Features: Provide an analysis of the requirements by feature.

4. **External Interface Requirements**: Provide the visualization of the program and the requirements that are related to hardware, software and networking.
5. **Other Nonfunctional Requirements**: Provide some additional constraints that apply to factors such as performance, safety and security.

1.4 Project Scope

This software aims at helping teachers to calculate the final assessment by a given formula for students' contribution. Besides, its automatically grouping function do assist teachers and students a lot. The benefits of this are that it saves time for teachers from assessing students and the traditional artificial grouping.

As the description above, the goal and objective of the software is set as an automatically-calculate assessment function. With it, teachers won't need to do the calculation by themselves, which may even cause some calculation mistakes.

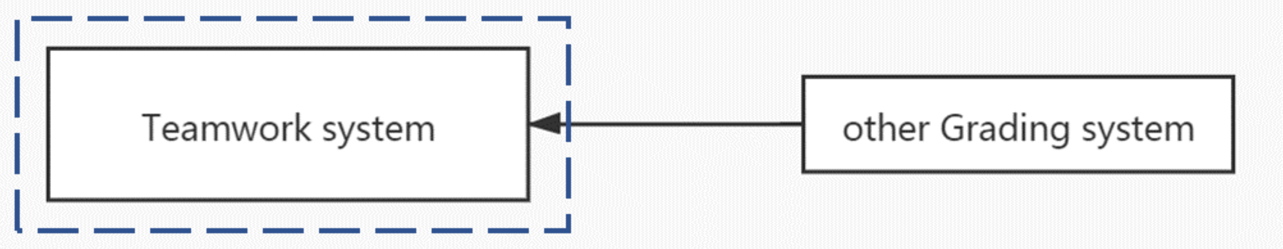
1.5 References

1. Workshop III Teamwork Project V3.
2. "Software Requirements Specification for <Project iTest>," 2018.

2. Overall Description

2.1 Product Perspective

The teamwork system exports the contribution file will be used in another existing Grading system.



Context model of Teamwork System

2.2 Product Features

The major features this program contains are the following:

For teachers:

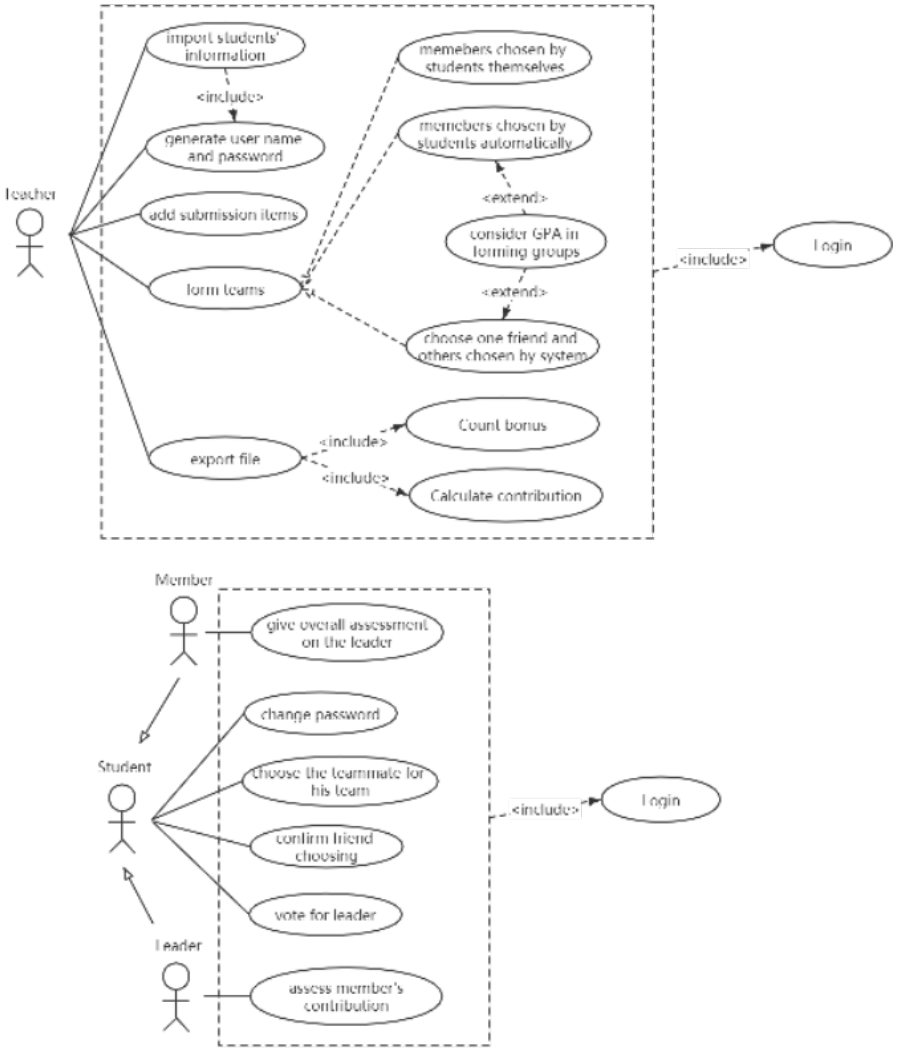
- **Import students' information**: Teacher may import students' information (student name, I.D., email, GPA) for a course. The information can also be imported as an excel file.
- **Generate username and password**: generate user name (student's email) and initial password (same for all) for each student whose name is in the list.
- **Add submission items**: Each submission includes a title (description) and an assessment

percentage. The name and percentage for each submission can be modified. The sum of all submission percentages should be 100%.

- Form teams: Teacher decides the number of members in a team. Furthermore, this application can offer several choices to form teams:
  1. Students themselves choose all the members.
  2. The members are decided by the system automatically.
  3. A student can choose one friend, and others are given by system randomly
  4. GPA might be considered in 1 and 2. Teacher decides it.
- Export file: export a file which lists the contribution for each student in the whole class.

For students:

- Login the system and change password: After a student input the correct username and password, the student can log in the system and can change password.
- Vote for leader: Vote to choose the project leader of the team.
- Choose the friend for his team: Choose his friend for his team if teams are formed using choices 1 and 3 above.
- Confirm team choosing: Confirm when his friend chooses him as a friend using choices 1 and 3 above.
- Assess member’s contribution: The project leader has the right to assess a member’s contribution in each submission if he likes.
- Give overall assessment on the leader: Each member has the right to give the overall assessment on the project leader after the project is finished


Teacher
import students' information
<include>
generate user name and password
add submission items
form teams
export file
memebers chosen by students themselves
memebers chosen by students automatically
<extend>
consider GPA in forming groups
<extend>
choose one friend and others chosen by system
Count bonus
<include>
<include>
Calculate contribution
<include>
Login
Member
give overall assessment on the leader
Student
change password
choose the teammate for his team
confirm friend choosing
vote for leader
Leader
assess member's contribution
<include>
Login

User case model of Teamwork System

2.3 User Classes and Characteristics

For the conventional reason, we name each of the user classes-actors with this format:

- Student: The student is the one that T.S. to form groups and manage the group, such as the election of team-leaders and making an assessment for each other. It is convenient to think that every student represents every user that contributes to a team individually with an assessment from each other.

- Teacher: The Teacher is the one that uses the T.S. to initialize students' accounts, manage submissions in the courses, forming teams and calculate the final assessment of the students. It is convenient to think that every teacher represents every user that lead the corresponding students and know each one's contribution separately.

2.4 Operating Environment

The operating environment is the application that runs on a server, accessible by web browsers. This application is able to run on Linux-based operating system and Microsoft Windows.

2.5 Design and Implementation Constraints

This program is created using python programming language and uses the Django framework for the main modules. Therefore, a minimum P.C. having at least 64mb of RAM and CPU over 400mhz is required to run the online program with good speed. Or the connection stream TCP-IP is used as its the standard gateway for internet applications.

For natural language, the entire system only supports the English language. The UIC organization has full responsibility for maintaining the Teamwork system.

2.6 User Documentation

Requirements document, a design document, and a brief user manual (installation and usage of the application).

2.7 Assumptions and Dependencies

Django framework was used to create the website application and set up the core program. For better understanding the system, we assume the reader is not interested in how the program is coded. We also assume that we have the right to access the account database in UIC to acquire the user information.

3. System Features

3.1 Login

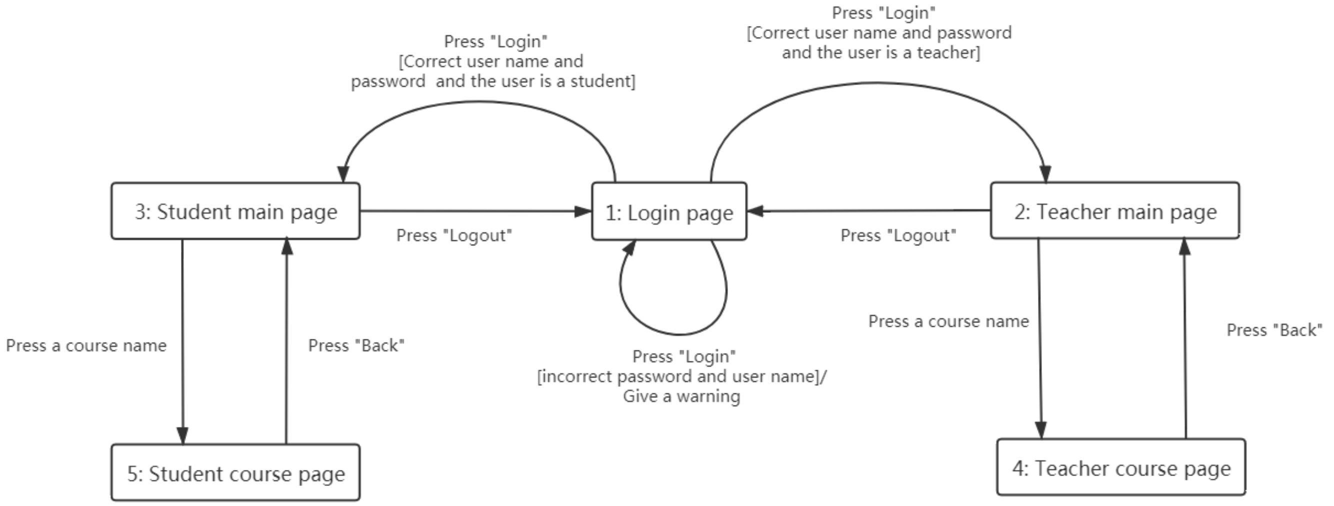
3.1.1 Description and Priority

User should first input their username and password. After pressing the “login” button, the system will search the corresponding user in the database. If no user matches or the password is wrong or empty, a warning message shows up. If username and password match the data in the database successfully, users will get into their main page according to their position (student or teacher). User can also log out by pressing the button “Log out” at the top-right corner on the main page.

After login into the main page, teachers and students can make further operations in their course pages. Besides, students can change their passwords in the Change password page.

Login has a high priority. Students and teachers use the same login page to login to their main page and only if they login can they use other features of this system.

3.1.2 Stimulus/Response Sequences



3.1.3 Functional Requirements

REQ-1: After pressing the “Login” button, if the input username and password are empty, warning message “Please input your username and password!” shows up.

REQ-2: After pressing the “Login” button, if the input username doesn’t exist in the database, a warning message “User does not exist, or password is wrong.” shows up.

REQ-3: After pressing “Login” button, if the user login successfully, as there are two different main pages (teacher’s main page and student’s main page), the corresponding main page will jump out based on the position of the user.

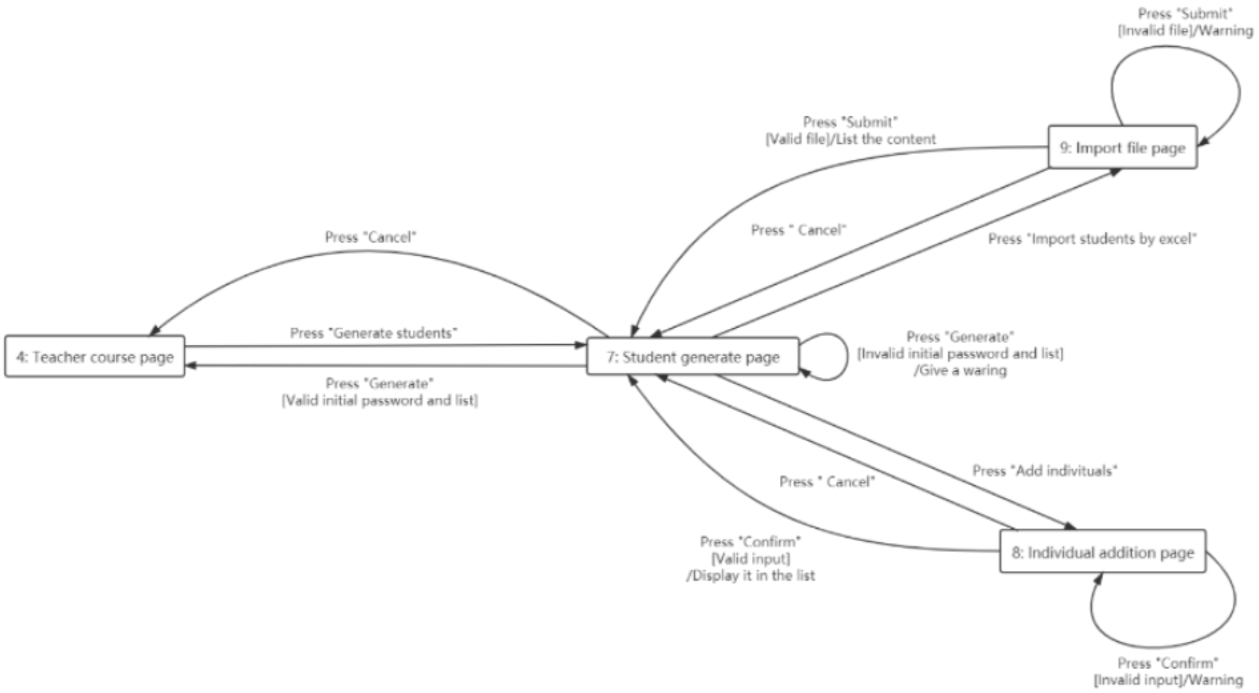
3.2 Accounts Import and Generation

3.2.1 Description and Priority

In a course page, the teacher press “Generate students” to enter the Student generate page. Then the teacher can choose to add students into the list by importing an excel file or by inputting the student’s information individually. The input information of the students will be listed out. Teachers should set up an initial password for all the generated students. After confirming the list and initial password, press “Generate” button and finish generation.

Accounts Import and Generation have a high priority.

3.2.2 Stimulus/Response Sequences



3.2.3 Functional Requirements

REQ-1: After pressing "submit" in "Import file page", The uploaded file should not be empty or a non-excel file and the content should fit the format (name, ID, email, GPA (optional)). If not, warning message "Wrong import!" shows up and the user can choose to cancel uploading the file or upload an excel file again.

REQ-2: After pressing "submit" in "Individual addition page", the input information should be in the correct format and not empty (Except GPA part). If not, warning message "Incorrect student information!" shows up and the user can choose to cancel the adding or input the information again.

REQ-3: After pressing "Generate", student list should not be empty. If empty, the system will do nothing and the warning message "No new students' accounts are generating!" shows up.

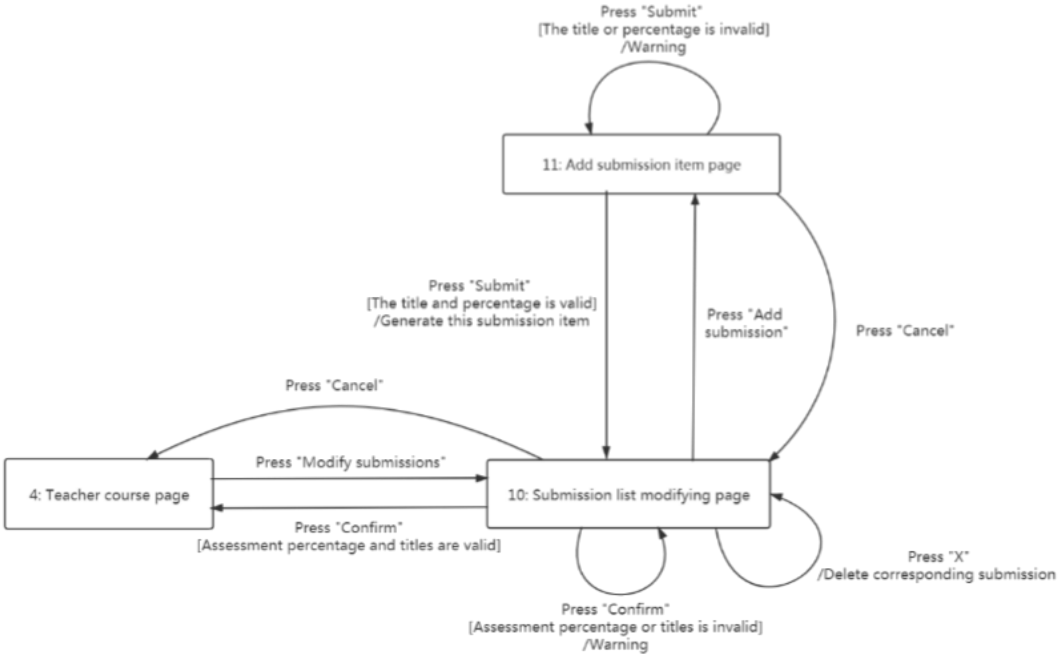
REQ-4: Besides an empty student list, if there are students in the list and no initial passwords for generation, a warning message "No initial password!" shows up.

3.3 Editing Submission Items

3.3.1 Description and Priority

A teacher can add submission items to the course. After getting into the teacher course page, a teacher can modify the whole list of submission items in this course. The teacher may also add submission items or modify the submission items on the existed table. After the input title and percentage, the submission item would be generated. If the teacher wants to modify information in this table, the teacher may directly modify the information, including title and percentage or delete the submission item. Editing Submission Items has a high priority.

3.3.2 Stimulus/Response Sequences


Press "Submit"
[The title or percentage is invalid]
/Warning
11: Add submission item page
Press "Submit"
[The title and percentage is valid]
/Generate this submission item
Press "Add submission"
Press "Cancel"
Press "Cancel"
4: Teacher course page
Press "Modify submissions"
10: Submission list modifying page
Press "Confirm"
[Assessment percentage and titles are valid]
Press "X"
/Delete corresponding submission
Press "Confirm"
[Assessment percentage or titles is invalid]
/Warning

3.3.3 Functional Requirements

REQ-1: In Submission list modifying page, after pressing "Confirm" button, if the sum of the percentage is not 100%, the system will show up a warning message "The sum of the percentage of submissions is not 100%" and provide two choices: Go back to the submission list modification page or jump to the course page without changing the submission list that was just modified.

REQ-2: After adding or modifying one submission, if the title is empty, the system will show up a warning message "The title of this submission is empty". If the percentage is empty, the system will show up a warning message "The percentage of this submission is empty".

REQ-3: After modifying one submission, if the title is empty, the system will show up a warning message "The title of this submission is empty".

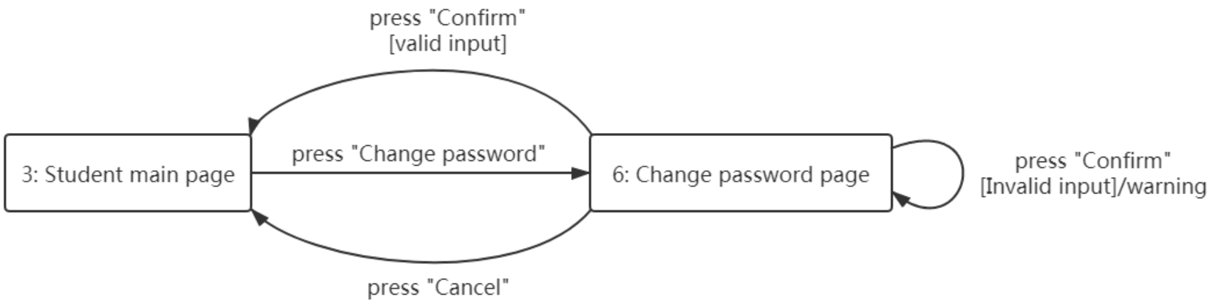
REQ-4: The user in the Submission list modifying page and Add submission item page is able to go back to the last page by pressing "Cancel".

3.4 Change Password

3.4.1 Description and Priority

After login into the main page, students can change their password besides the initial password. Chang Password has the low priority.

3.4.2 Stimulus/Response Sequences



3.4.3 Functional Requirements

REQ-1: If the students input a wrong initial password, after pressing “Confirm”, warning message “Wrong original password!” will show up.

REQ-2: If the students input two different new passwords in “New password” and “Confirm password”, after pressing “Confirm”, warning “New password is not confirmed!”

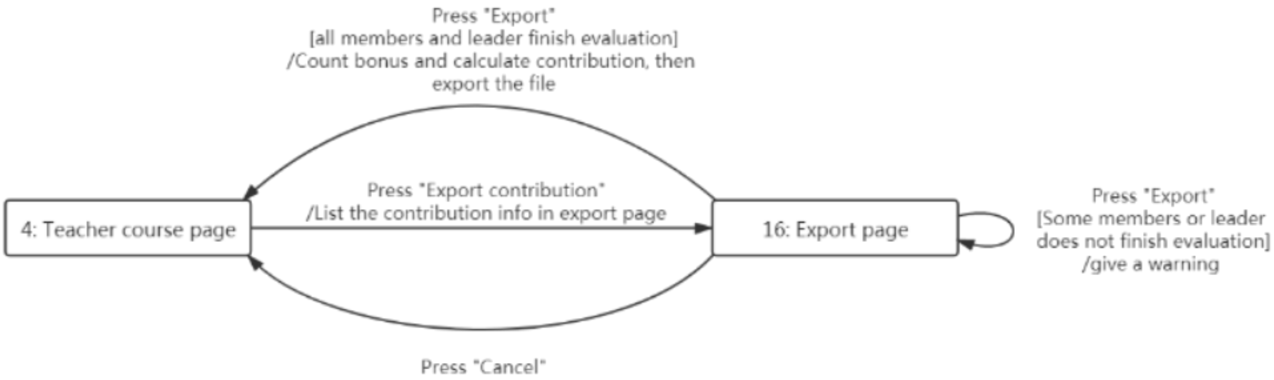
REQ-3: If the student leaves one input block blank, after pressing “Confirm”, warning “Please fill in the blanks!” will show up.

3.5 Export Files

3.5.1 Description and Priority

A teacher has the ability to export the contribution file of the corresponding course. Export Files has a high priority.

3.5.2 Stimulus/Response Sequences


Press "Export"
[all members and leader finish evaluation]
/Count bonus and calculate contribution, then
export the file
4: Teacher course page
Press "Export contribution"
/List the contribution info in export page
16: Export page
Press "Export"
[Some members or leader
does not finish evaluation]
/give a warning
Press "Cancel"

3.5.3 Functional Requirements

REQ-1: After pressing “Export contribution” button, the system will display the whole contribution information on the Export page.

REQ-2: In the Export page, if the user press “Export”, the corresponding export file would be calculated by this way:

1. summarize the contributions from a non-leader member using the following formula.

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

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

2. Count the bonus for the leader: calculate the average.

REQ-3: The exported file has the following columns: name, ID, contribution, bonus.

REQ-4: If some members or leader haven't finished evaluation, the calculation of the contribution will meet some error and warning message "Students haven't finished evaluation".

3.6 Teams Setting (for teacher)

3.6.1 Description and Priority

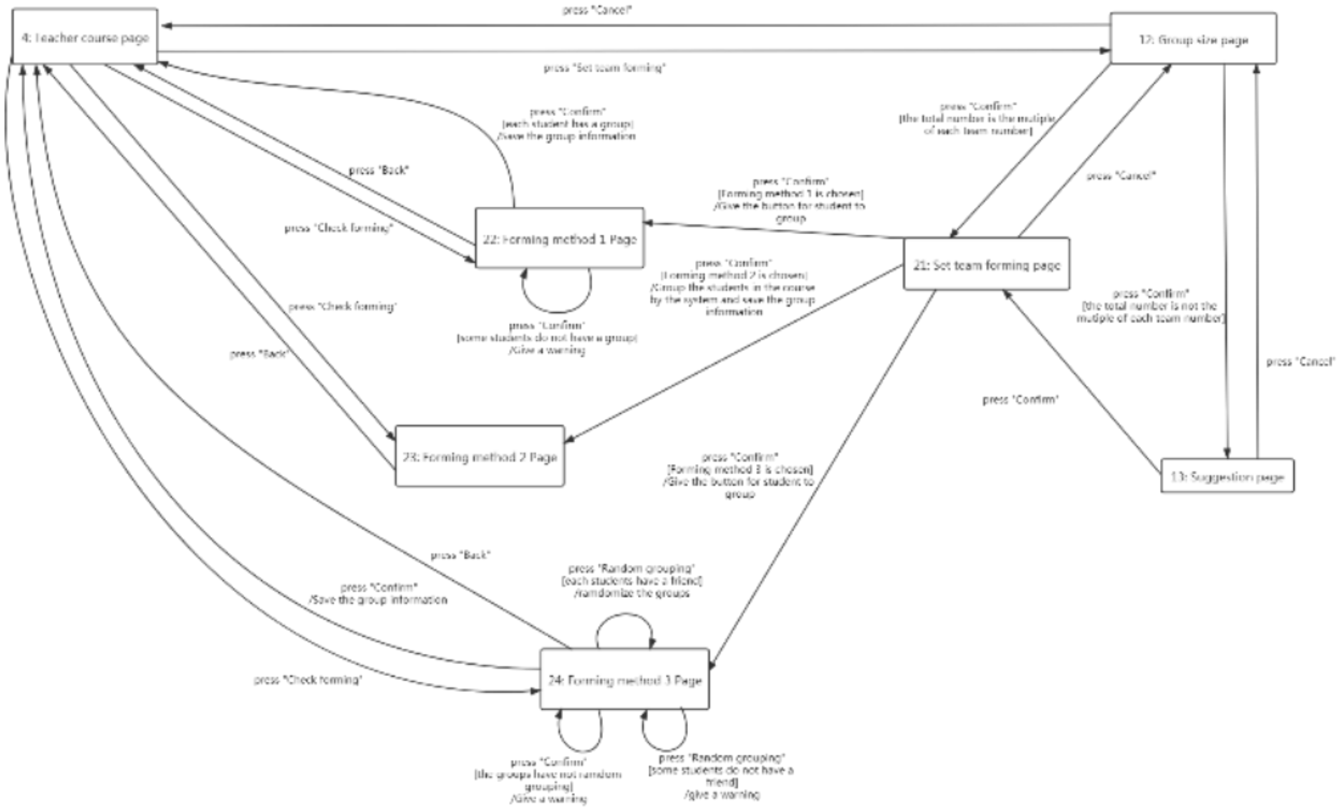
A teacher can set up some the group forming standard in this system. In some case, the suggestion page may show up for the teacher to reset the standard of the group for a better arrangement.

Different methods in forming team have different priority. The three methods are:

i. All the members are chosen by the students themselves.
ii. The members are decided by the system automatically.
iii. A student can choose one friend, and others are given by system randomly
iv. GPA might be considered in ii and iii. It is decided by teacher.

The first and second method has the highest priority. The priority of the second method (consider with GPA) and the third method is in the middle. Using the third method with GPA has a high priority.

3.6.2 Stimulus/Response Sequences


4: Teacher course page
12: Group size page
13: Suggestion page
21: Set team forming page
22: Forming method 1 Page
23: Forming method 2 Page
24: Forming method 3 Page
press "Cancel"
press "Set team forming"
press "Confirm" [each student has a group] /Save the group information
press "Back"
press "Check forming"
press "Check forming"
press "Back"
press "Confirm" [some students do not have a group] /Give a warning
press "Confirm" [the total number is the mutiple of each team number]
press "Cancel"
press "Confirm" [Forming method 1 is chosen] /Give the button for student to group
press "Confirm" [Forming method 2 is chosen] /Group the students in the course by the system and save the group information
press "Confirm" [the total number is not the mutiple of each team number]
press "Cancel"
press "Confirm"
press "Confirm" [Forming method 3 is chosen] /Give the button for student to group
press "Back"
press "Random grouping" [each students have a friend] /ramdomize the groups
press "Confirm" /Save the group information
press "Check forming"
press "Confirm" [the groups have not random grouping] /Give a warning
press "Random grouping" [some students do not have a friend] /give a warning

3.6.3 Functional Requirements

REQ-1: The number of members in a team is decided by the teacher. When the total number is not the multiple of this number, the suggestion should be given in the suggestion page. The suggestion would offer several choices to form teams for students.

REQ-2: In the Set team forming page, user may choose three different forming method.

Forming method 1: All the members are chosen by the students themselves.

Forming method 2: The members are decided by the system automatically.

Forming method 3: A student can choose one friend, and others are given by system randomly

Alternative consideration on GPA on method 2 and 3 is decided by the teacher. Teacher can select whether consider GPA or not in the Set team forming page when choosing forming method 2 and 3.

REQ-3: For Forming method 1 page, after teacher choose the method 1, students would be allowed the empty group list would be shown on the forming method 1 page and waiting for student to form by themselves. Teacher may press “Back” to go back course page for waiting. When students finish grouping by themselves, teacher may press “Check forming” and check the state of forming. If the teacher confirms the forming, the grouping is saved and cannot be changed.

REQ-4: For Forming method 2 page, after teacher choose the method 2, the group list of ramdomized selection by the system will be shown. If the teacher confirms the grouping, the grouping is saved and cannot be changed.

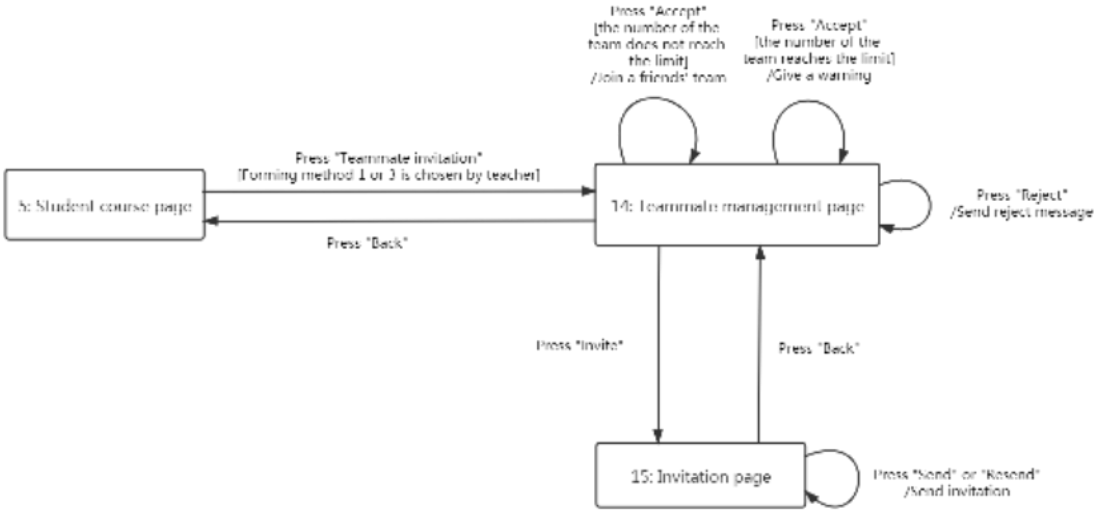
REQ-5: For Forming method 3 page, after teacher choose the method 3, the empty group list would be shown on the forming method 3 page and waiting for student to form by themselves. Teacher may press "Back" to go back course page for waiting. When students finish grouping by themselves, teacher may press "Check forming" and check the state of forming. Then if all the student finds their own friend for grouping, the teacher may press "Random grouping". If the teacher confirms the random grouping, the grouping is saved and cannot be changed.

3.7 Teams Forming (for student)

3.7.1 Description and Priority

After teacher setting up a standard for group forming in a course, students can do the corresponding operations. Teams Forming has a high priority.

3.7.2 Stimulus/Response Sequences



3.7.3 Functional Requirements

REQ-1: If the teacher chooses to use the first or the third method to arrange students groups, students can go to Teammate management page to read the invitation message from other groups. Students can either accept the invitation and reject the invitation on the page. If a student accepts an invitation from another student, and the group which send the invitation is not full, the system will arrange them into a group. Otherwise, a warning message "The group is full" shows up.

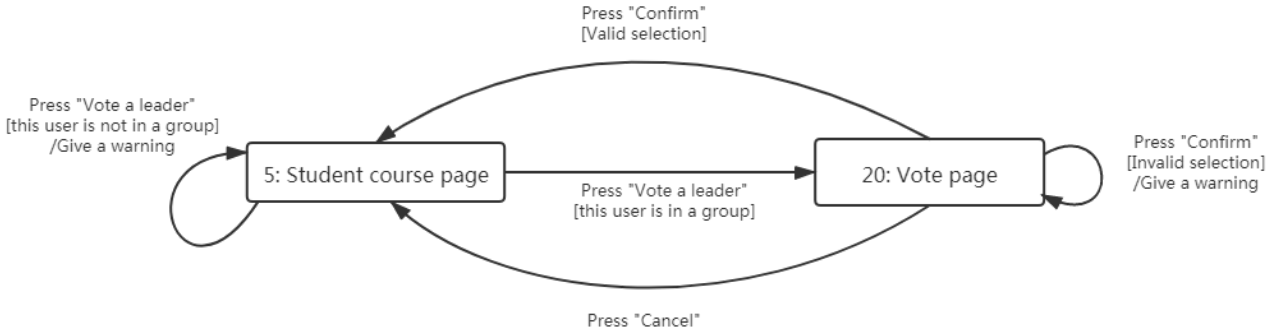
REQ-2: Students can go to the Invitation page to send an invitation to other students. Also, reject-information shows up there. If being rejects, students can resend the invitation again.

3.8 Vote For Leader

3.8.1 Description and Priority

A student in a team may vote for a leader if no leader exists in the team. Vote For Leader has a high priority.

3.8.2 Stimulus/Response Sequences



3.8.3 Functional Requirements

REQ-1: If the user is not in a team, after pressing “Vote a leader”, there should be a warning displayed “You are not in a team!”

REQ-2: If the user in the Vote page does not vote anyone then press “Confirm”, it will be an invalid selection, the system will give a warning “Please select an option!”

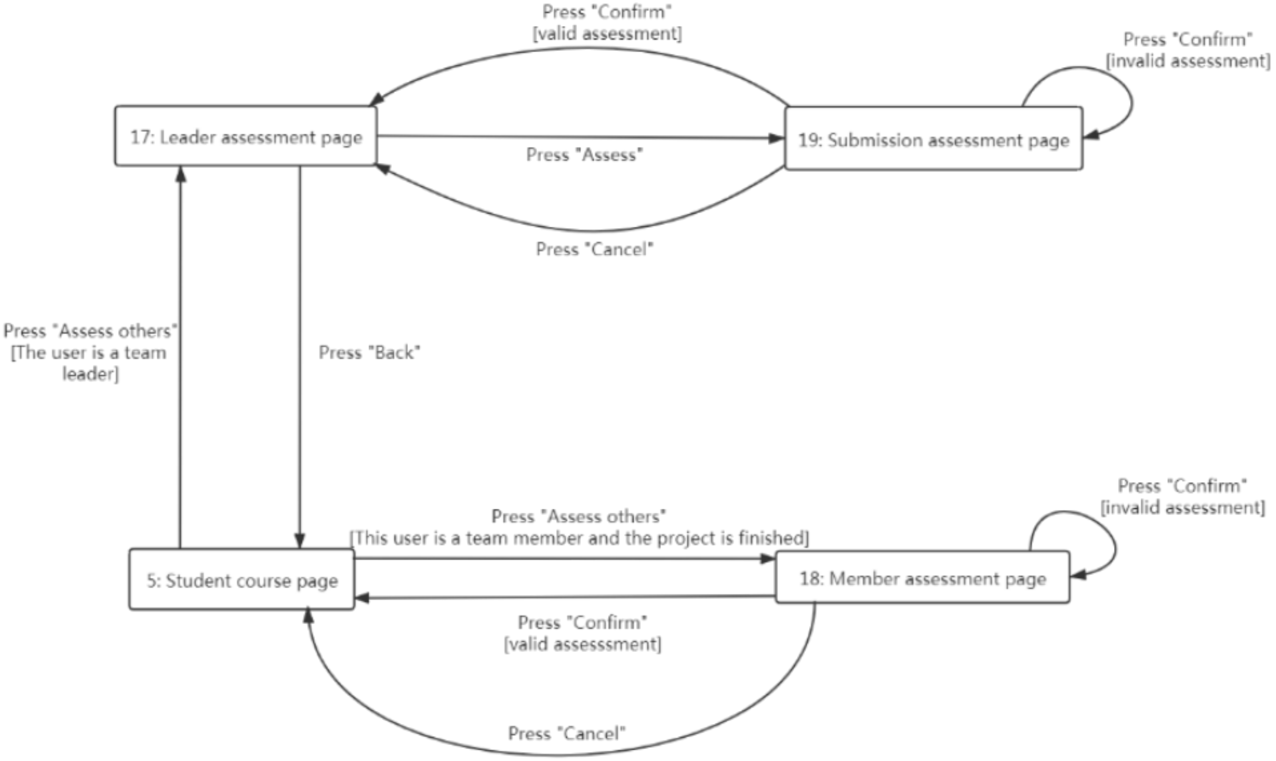
REQ-3: If the user is in the Vote page then press “Cancel”, the user would return to Student course page.

3.9 Assessment Among Teammates

3.9.1 Description and Priority

A project leader has the right to assess members’ contribution in each submission if he likes. The other members have the right to give the overall assessment of their project leader after the project is finished. Assessment Among Teammates has a high priority.

3.9.2 Stimulus/Response Sequences



3.9.3 Functional Requirements

REQ-1: For the team leader, after pressing “Assess others”, he will go to Leader assessment page. The team leader has the ability to assess a member's contribution in each submission by going to the Submission assessment page. The contribution is classified in four levels: (a) full(1.0), (b) fair(0.67), (c) little(0.33), (d) none(0).

REQ-2: For team member, after pressing “Assess others”, he will go to Member assessment page. The team member has the ability to give the overall assessment on their project leader after the project is finished in the Member assessment page. The contribution is classified in five levels: very good <2>, good <1>, fair <0>, bad <-1>, very bad<-2>)

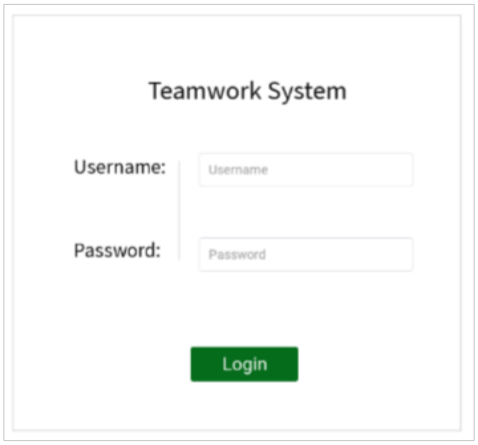
REQ-3: When the user in Submission assessment page, the user is able to go back to Leader assessment page.

REQ-4: When the user in Leader assessment page or Member assessment page, the user is able to go back to Student course page.

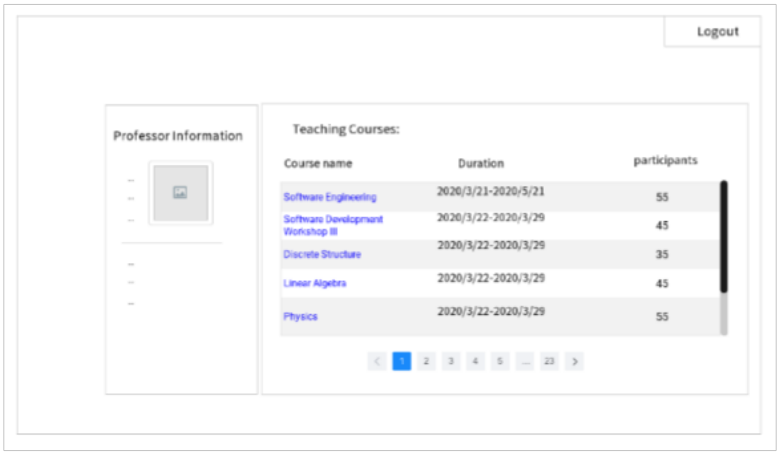
REQ-5: If the team leader or the team member does not finish assessing others submission, then press “Confirm”, then the system will give a warning “Please finish assessment!” and stay still.

4. External Interface Requirements

4.1 User Interfaces

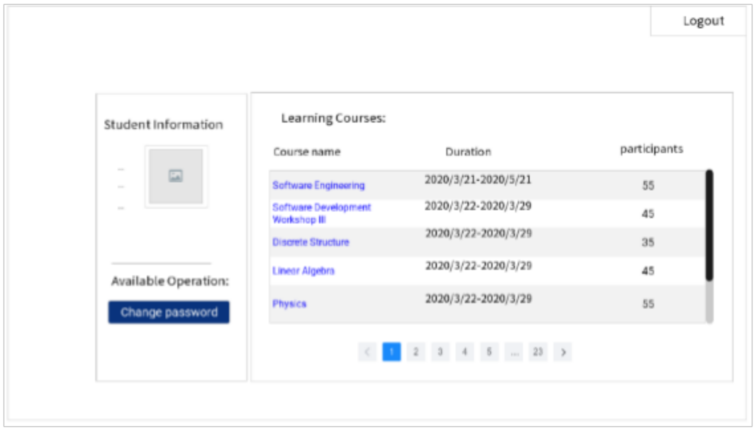


1: Login page



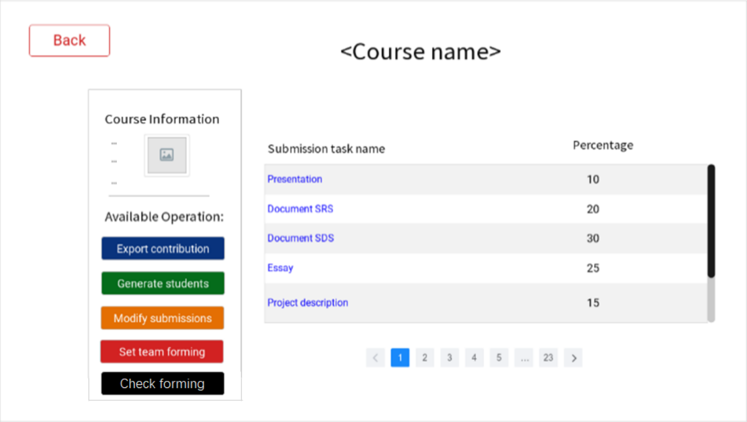
Course name	Duration	participants
Software Engineering	2020/3/21-2020/5/21	55
Software Development Workshop III	2020/3/22-2020/3/29	45
Discrete Structure	2020/3/22-2020/3/29	35
Linear Algebra	2020/3/22-2020/3/29	45
Physics	2020/3/22-2020/3/29	55

2: Teacher main page



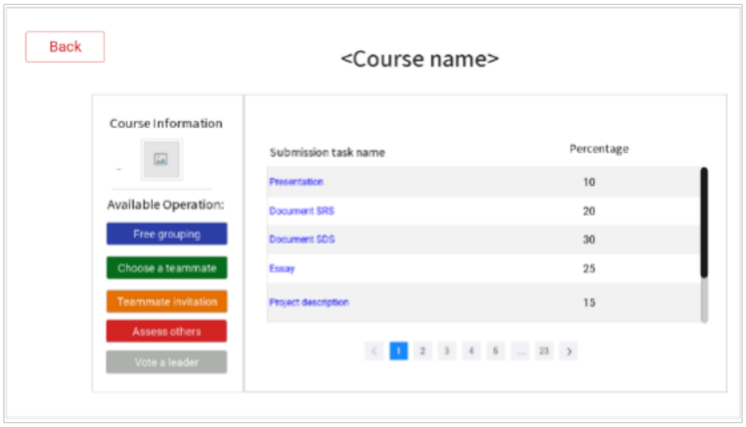
Course name	Duration	participants
Software Engineering	2020/3/21-2020/5/21	55
Software Development Workshop III	2020/3/22-2020/3/29	45
Discrete Structure	2020/3/22-2020/3/29	35
Linear Algebra	2020/3/22-2020/3/29	45
Physics	2020/3/22-2020/3/29	55

3: Student main page



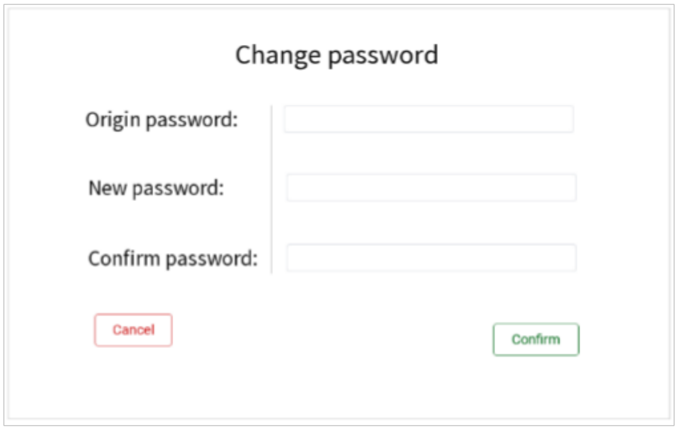
Submission task name	Percentage
Presentation	10
Document SRS	20
Document SDS	30
Essay	25
Project description	15

4: Teacher course page

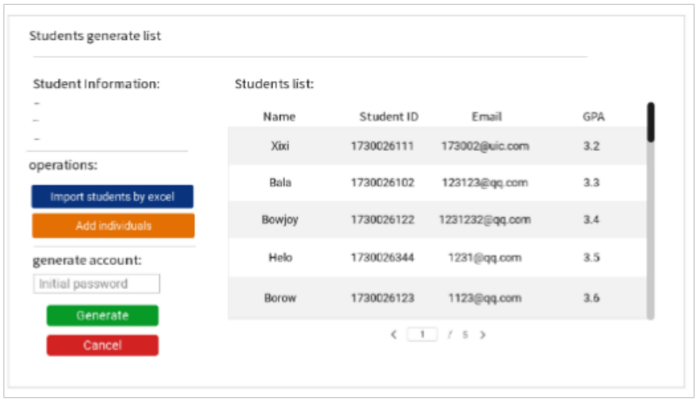


Submission task name	Percentage
Presentation	10
Document SRS	20
Document SDS	30
Essay	25
Project description	15

5: Student course page



6: Change password page



Students generate list

Student Information:

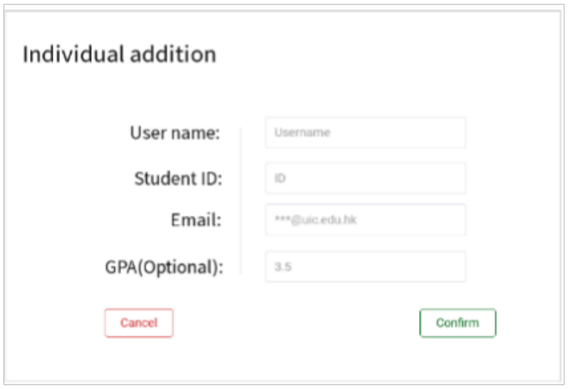
-
-
-

Students list:

Name	Student ID	Email	GPA
Xixi	1730026111	173002@uic.com	3.2
Bala	1730026102	123123@qq.com	3.3
Bowjoy	1730026122	1231232@qq.com	3.4
Helo	1730026344	1231@qq.com	3.5
Borow	1730026123	1123@qq.com	3.6

generate account: Initial password

7: Student generate page



Individual addition

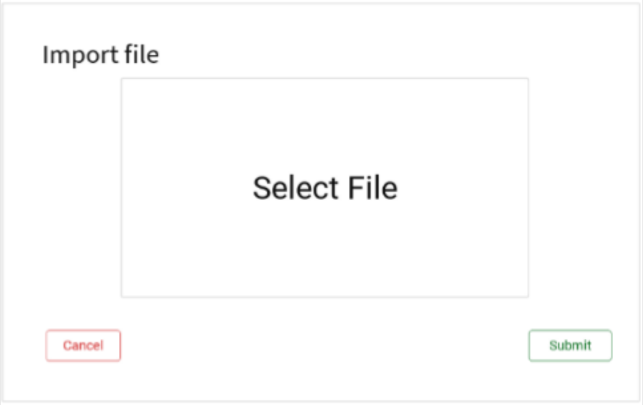
User name: Username

Student ID: ID

Email: ***@uic.edu.hk

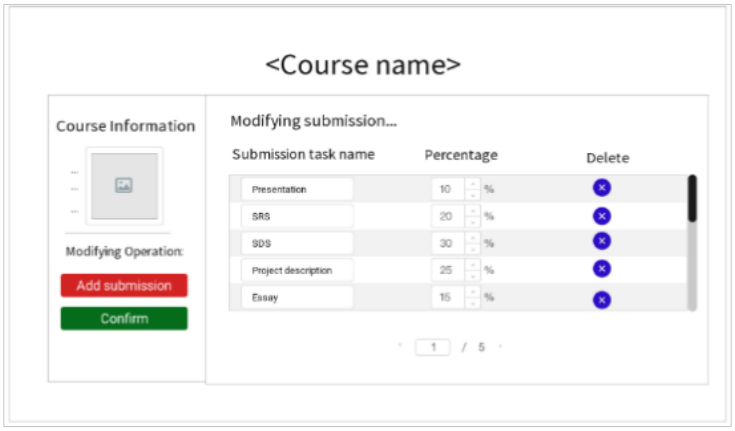
GPA(Optional): 3.5

8: Individual addition page



Import file

9: Import file page



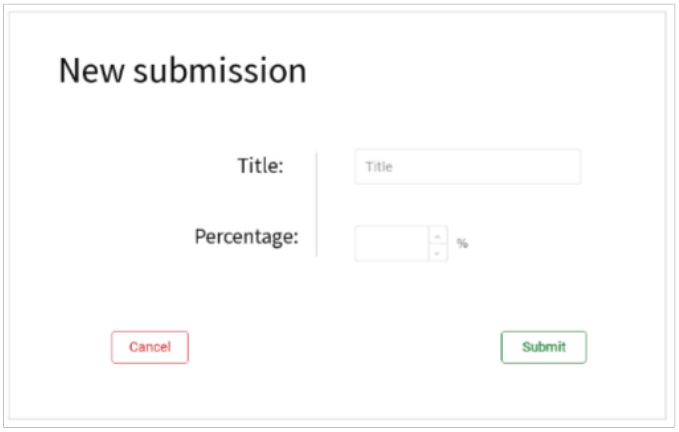
<Course name>

Course Information

Modifying submission...

Submission task name	Percentage	Delete
Presentation	10 %	✕
SRS	20 %	✕
SDS	30 %	✕
Project description	25 %	✕
Essay	15 %	✕

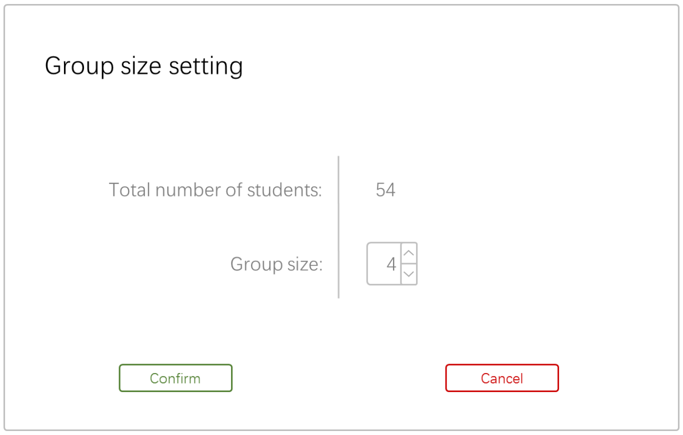
10: Submission list modifying



New submission

Title: Title

Percentage: %



Group size setting

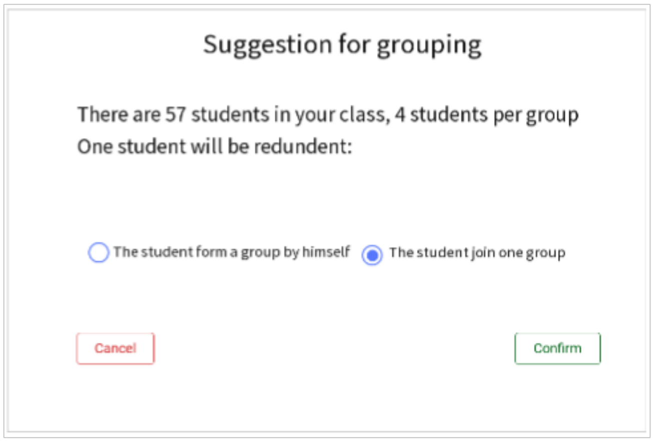
Total number of students: 54

Group size: 4

page

11: Add submission item page

12: Group size page

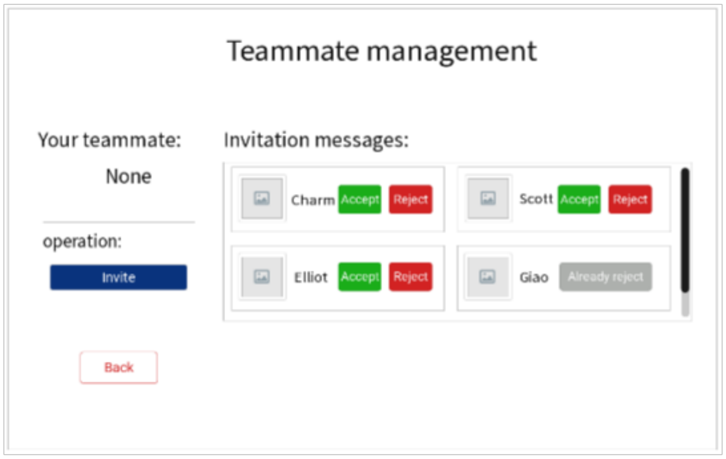


Suggestion for grouping

There are 57 students in your class, 4 students per group
One student will be redundent:

○ The student form a group by himself ● The student join one group

13: Suggestion page



Teammate management

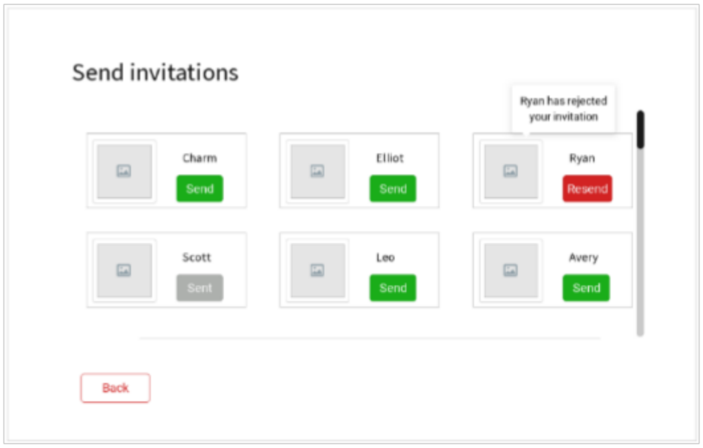
Your teammate: None

Invitation messages:

Charm	Accept	Reject	Scott	Accept	Reject
Elliot	Accept	Reject	Giao	Already reject	

operation:

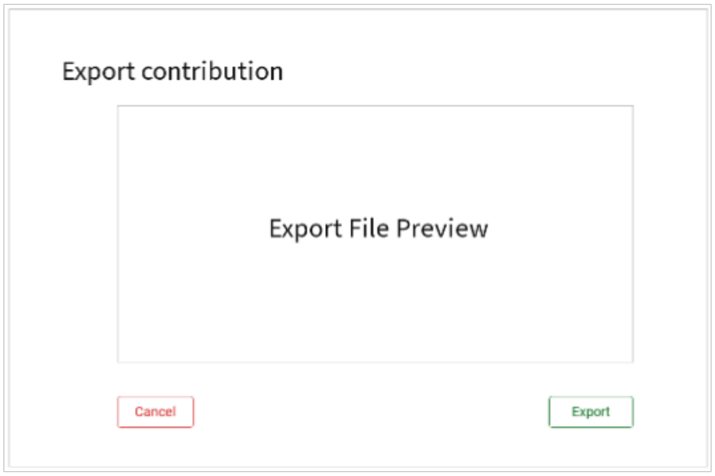
14: Teammate management



Send invitations

Ryan has rejected your invitation

Charm Send	Elliot Send	Ryan Resend
Scott Sent	Leo Send	Avery Send



Export contribution

Export File Preview

page

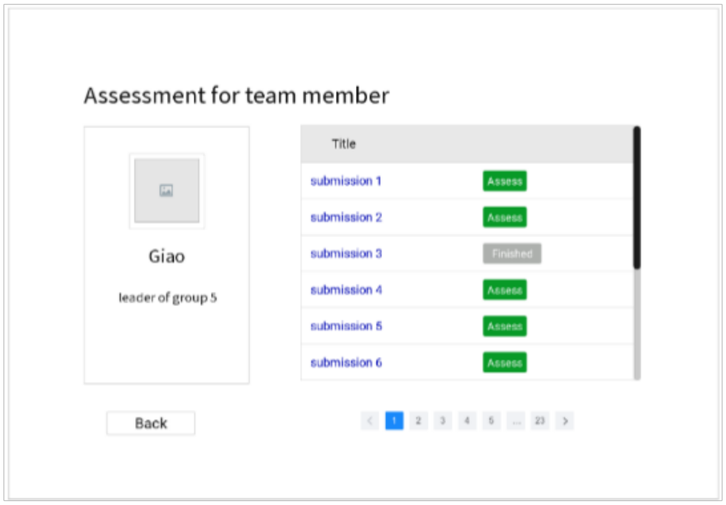
15: Invitation page

16: Export page

17: Leader assessment page

18: Member assessment

page

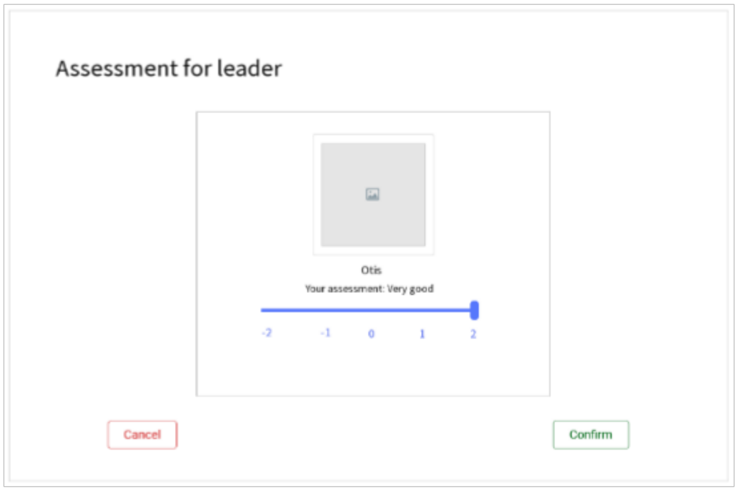


Assessment for team member

Giao

leader of group 5

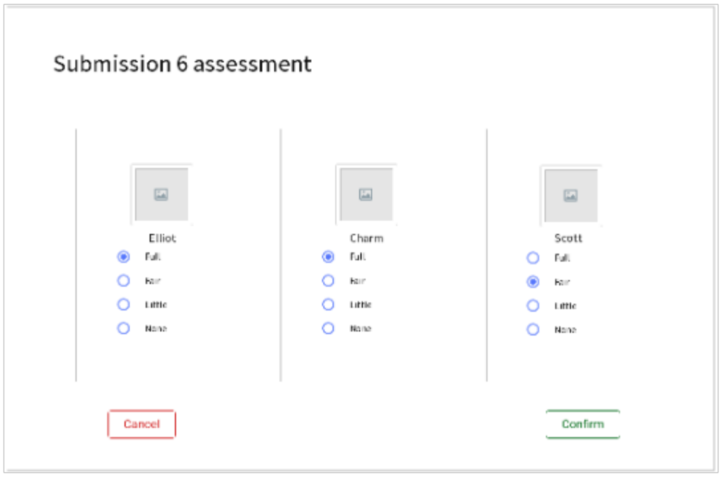
Title	
submission 1	Assess
submission 2	Assess
submission 3	Finished
submission 4	Assess
submission 5	Assess
submission 6	Assess



Assessment for leader

Otis

Your assessment: Very good



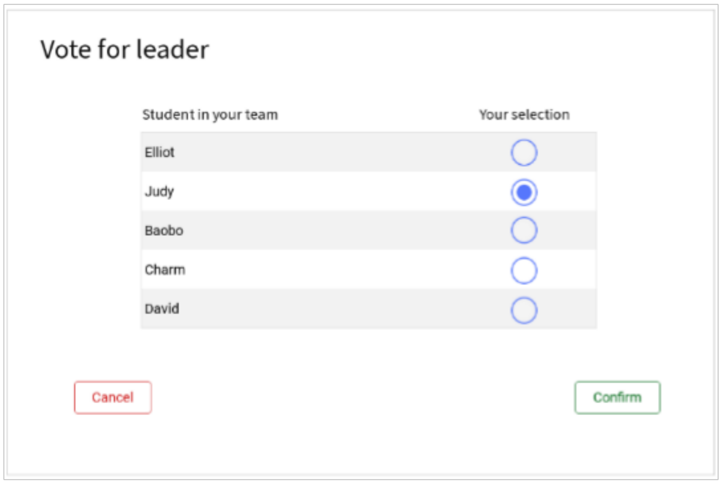
Submission 6 assessment

Elliot
- ● Full
- ○ Fair
- ○ Little
- ○ None

Charm
- ● Full
- ○ Fair
- ○ Little
- ○ None

Scott
- ○ Full
- ● Fair
- ○ Little
- ○ None

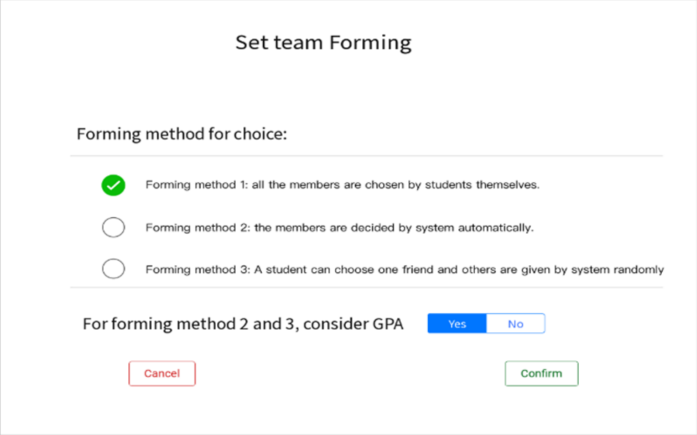
19: Submission assessment page



Vote for leader

Student in your team	Your selection
Elliot	○
Judy	●
Baobo	○
Charm	○
David	○

20: Vote page



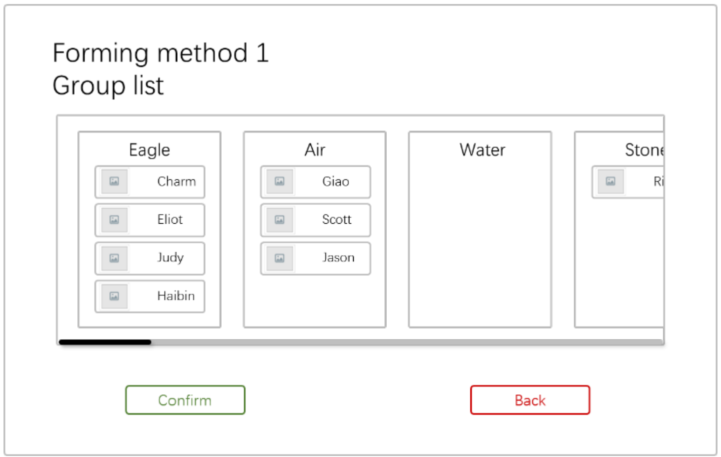
Set team Forming

Forming method for choice:

- ✓ Forming method 1: all the members are chosen by students themselves.
- ○ Forming method 2: the members are decided by system automatically.
- ○ Forming method 3: A student can choose one friend and others are given by system randomly

For forming method 2 and 3, consider GPA [✓ Yes] No

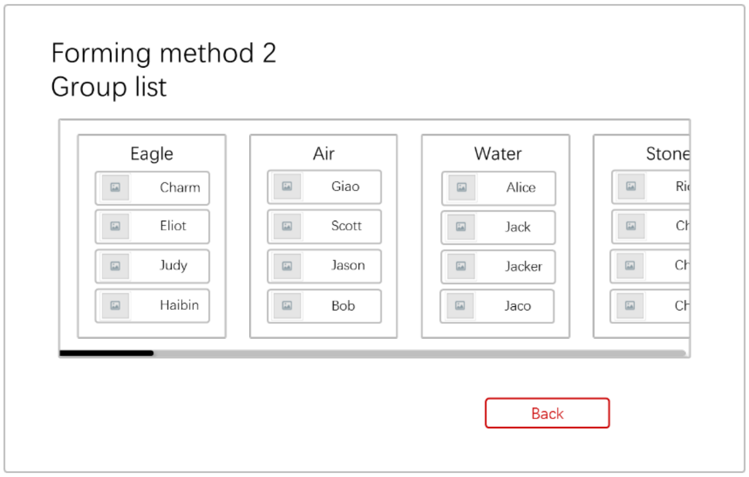
21: Set team forming page



Forming method 1
Group list

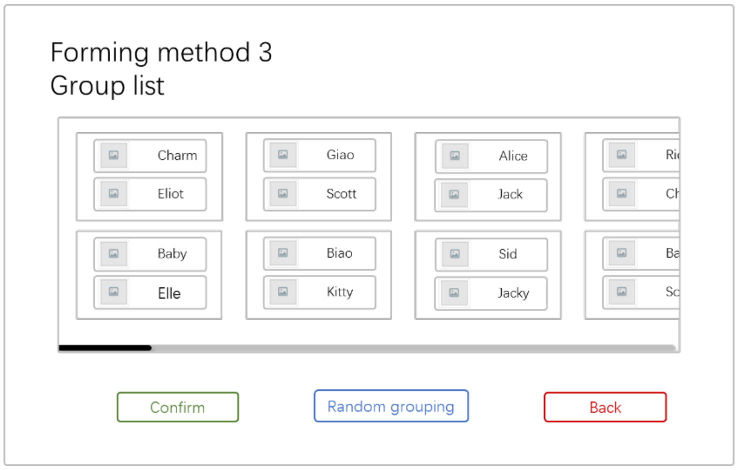
Eagle	Air	Water	Stone
Charm	Giao		Ri
Eliot	Scott		
Judy	Jason		
Haibin			

22: Forming method 1



Forming method 2
Group list

Eagle	Air	Water	Stone
Charm	Giao	Alice	Ri
Eliot	Scott	Jack	Ch
Judy	Jason	Jacker	Ch
Haibin	Bob	Jaco	Ch



Forming method 3
Group list

Charm	Giao	Alice	Ri
Eliot	Scott	Jack	Ch
Baby	Biao	Sid	Ba
Elle	Kitty	Jacky	Sc

page
23: Forming method 2 page
page

24: Forming method 3

4.2 Hardware Interfaces

None.

4.3 Software Interfaces

Software Type	Software	Description
Operating system	Windows 10	We have chosen the Windows operating system for its user-friendliness.
Database	MySQL-like database (MariaDB)	To save students and teachers' records, we recommend using the MariaDB database as it can be set up easily.
Libraries and frameworks	Python 3.6.4(32-bit/64-bit), Django (2.0+)	-
Webserver	Nginx (1.0+)	-

4.4 Communications Interfaces

- This project supports all types of web browsers.
- All features will be tested in Google Chrome (73.0.3683.86(64-bit) or later) based on the system of Microsoft Windows 10.
- All traffic should be encrypted by HTTPS. When running locally, do not enable port mapping to ensure data security.
- The ideal data transfer rate for the smooth user experience is 500kbps-1Mbps. The website will use synchronous requests.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

Within the scope of server processing capabilities, all requests should react accurately, and the order of execution should be scheduled in the order in which the requests are delivered.

5.2 Safety Requirements

None.

5.3 Security Requirements

This program uses object-oriented mechanisms to protect its data passed using methods:

- In term of privacy, all the students could only see their contribution and could not see the contribution of others in the group.
- Students and teachers can log in and change their passwords to keep their accounts safe.

5.4 Software Quality Attributes

- Correctness: All content and information should be correctly displayed and encoded properly. All links need to be correctly connected to the specified location.
- Portability: The website will support new versions of the related browsers.
- Usability: The GUI should be easy to learn and use by the user of any technical background.
- Availability: Checking that the system always has something to function and still pop up error messages in case of component failure. In that case, the error messages appear when something goes wrong so to prevail availability problems.

6. Other Requirements

Currently, no other requirements.

Appendix A: Glossary

T.S.: Teamwork System

Appendix B: Analysis Models

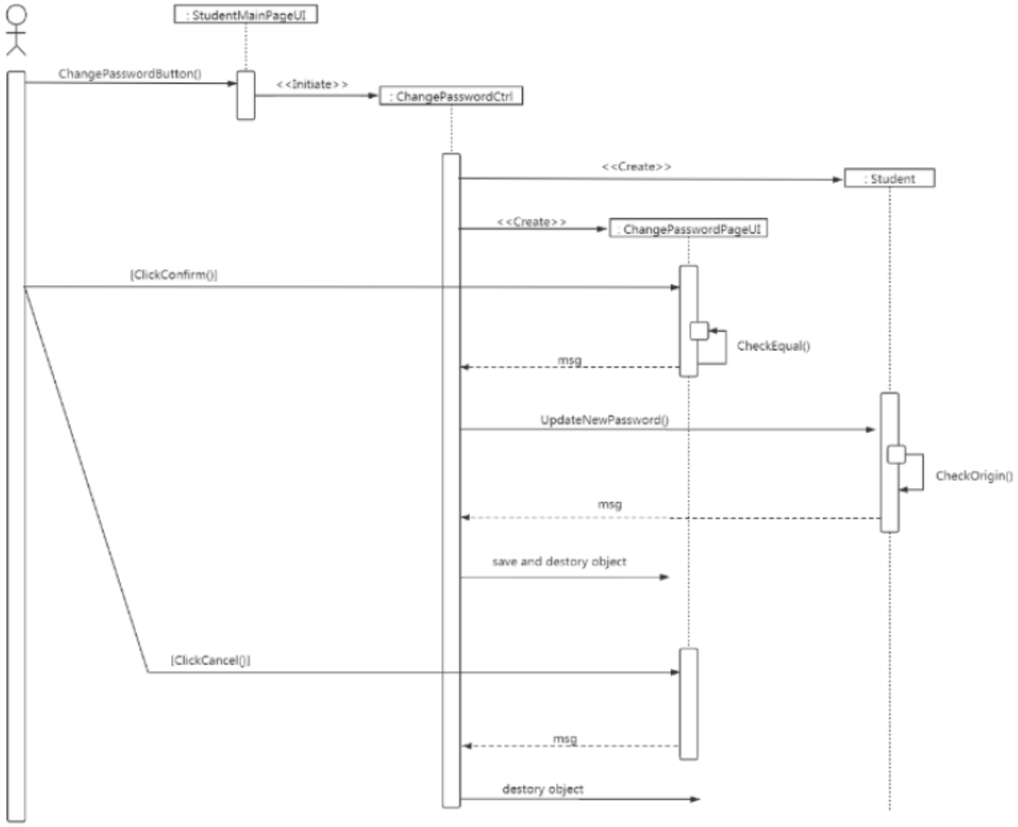
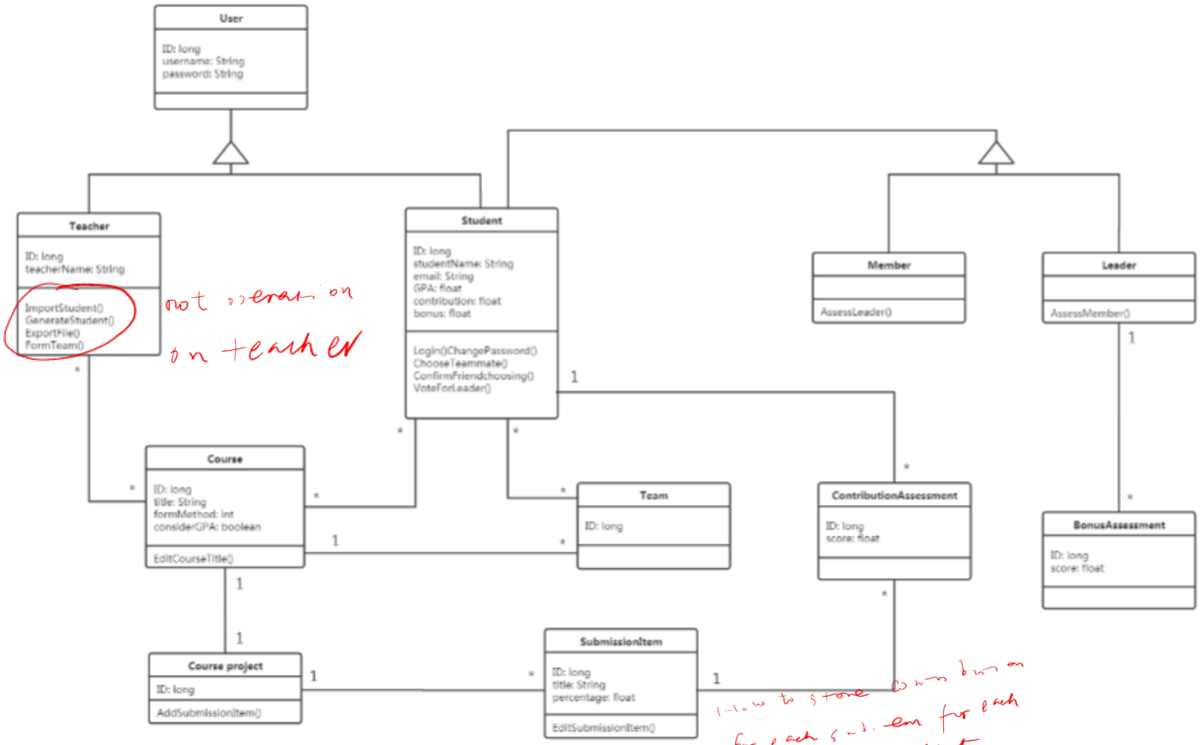


Figure 1 Sequence Diagram for the Update Password (student)



not operation on teacher

How to store contribution for each subitem for each student

Figure 2 Class Diagram

Appendix C: Issues List