CS108 Project Report - Online Grading System

LIU MINGCHEN 17098537-i011-0055

December 30, 2019

1 Introduction

The Online Grading System(OGS) is a web application for MUST students, which using ASP.NET framework, bootstrap framework and SQL server. OGS provides a platform for students and teacher to digitize submit and mark school assignment. Teacher can create account for each student, delete and edit student account information, modify administrator account password, create assignment and mark assignment. Students can limited edit their information, do assignment and view score after marking by teacher.

2 Objective and Solution

2.1 Objective

1. Build database and table to store student account, student information, assignment and question with weight, comment and mark.

2. Teacher can create account for each student, delete and edit student account information, modify administrator account password, create assignment and mark assignment.

3. Students can limited edit their information, do assignment and view score after marking by teacher.

4. All page can redirect to other page for user convenience. When error occurs, prompt user.

2.2 Solution

1. Create Database for storing student and assignment information.

2. Create website: *T-Home.aspx* for welcoming page. Create website: *T-CreaAcc.aspx* for creating account. Create website: *T-ManaAcc.aspx* for managing student account, deleting student account and jumping to editing account website. Create website: *T-EditAcc.aspx* for editing student account information. Create website: *T-AdminAcc.aspx* for modifying administrator account password and jumping to login page. Create website: *T-CreaAss.aspx* for creating assignment. Create website: *T-MarkAss.aspx*, *T-MarkAssIndex.aspx*, *T-MarkDone.aspx* for marking assignment.

3. Create website: *S_Home.aspx* for welcoming page. Create website: *S_Acc.aspx* for editing information. Create Website: *S_DoAss.aspx*, *S_DoAssDone.aspx* for doing assignment. Create website: *S_Mark.aspx*, *S_MarkAss.aspx* for viewing score and detail information after

marking by teacher.

4. Build navigation bar for redirection. Add log out button in navigation bar.

3 Pre-Design

Pre-Design include database design and ASP.NET design.

3.1 Database Design

Constructing the frame about database table, the relationship between tables; schema, data type, primary key and foreign key for connecting to other tables.

Finally, designing seven tables.

1. For features of logging in, build ‘Account’ table to store user ID and password. Add a schema for confirming user identity.

2. For features of storing students information, build ‘StuInfor’ table to store information: Student ID, Student name, Gender, Grade, Academic unit, E-mail and Phone number.

3. For features of storing assignments information, build ‘Assignment’ table to store information: Assignment ID, Questions and Weights.

4. For features of storing students answer, build ‘Answer’ table to store information: Student ID, Assignment ID, each answer. Add a schema to sign weather this assignment have been marked. Among them, Student ID and Assignment ID both are primary key to recognize the answer owner.

5. For features of storing teacher's marking comment, build ‘Comment’ table. Student ID and Assignment ID both are primary key.

6.For features of storing teacher's marking Score, build ‘Score’ table to store marking score, calculated real score and calculated GPA. Student ID and Assignment ID both are primary key.

7.For features of detecting whether student have new assignment, build ‘StuAss’ table. Add a schema to sign weather this assignment has been done. Student ID and Assignment ID both are primary key.

ER schema is shown in figure 1, a logical schema is shown in figure 2.

3.2 ASP.NET Design

In order to facilitate the development and build a friendly GUI, I use bootstrap framework to assist composing and producing websites.

Finally, designing one website for logging in, six websites for student operating and nine websites for teacher operating.

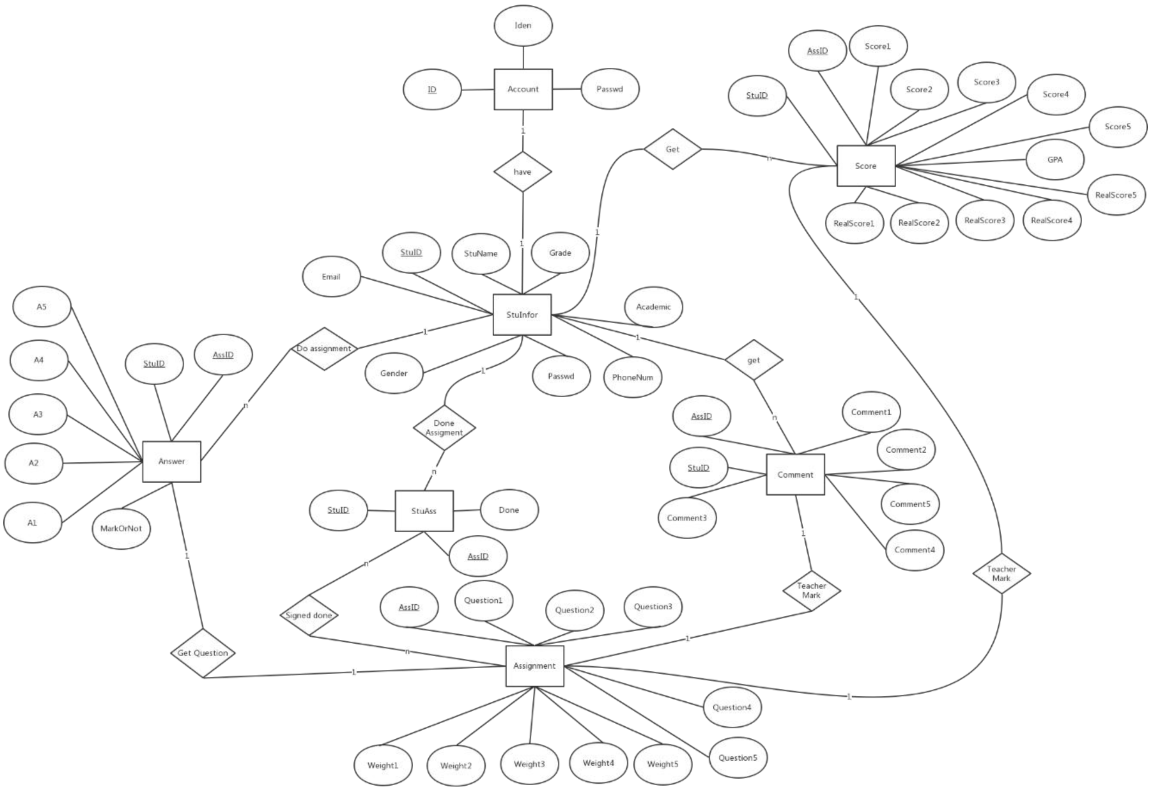


Figure 1: ER Schema

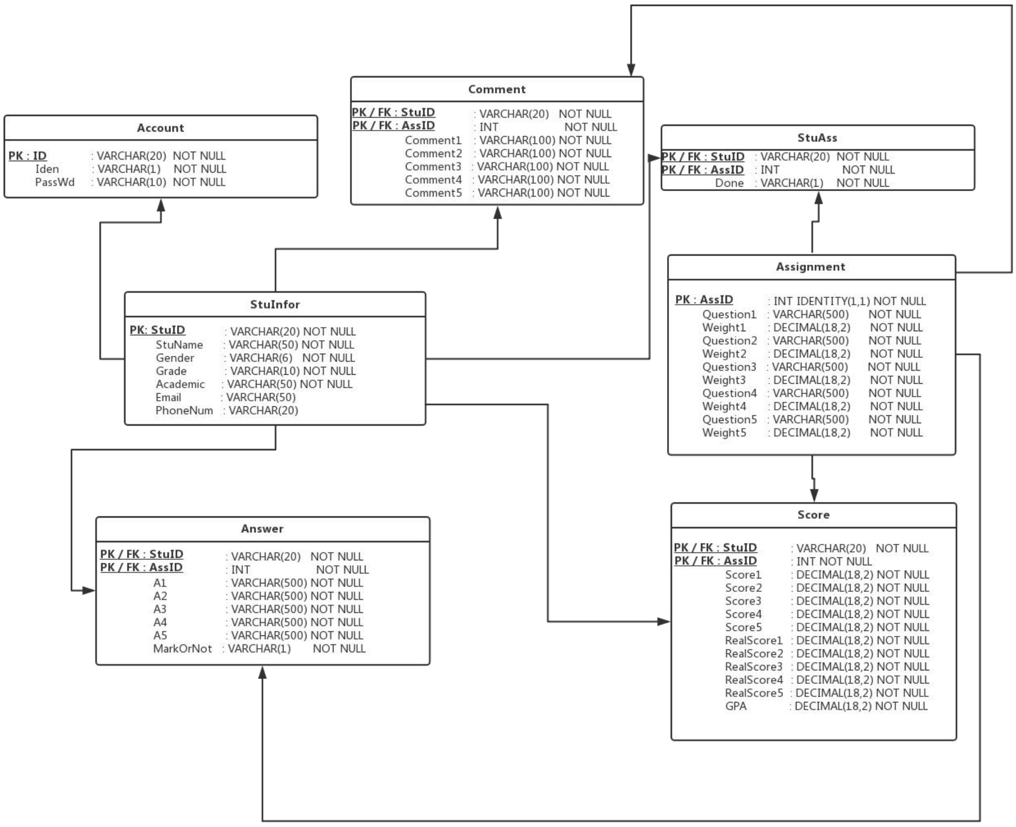

Comment
PK / FK : StuID : VARCHAR(20) NOT NULL
PK / FK : AssID : INT NOT NULL
Comment1 : VARCHAR(100) NOT NULL
Comment2 : VARCHAR(100) NOT NULL
Comment3 : VARCHAR(100) NOT NULL
Comment4 : VARCHAR(100) NOT NULL
Comment5 : VARCHAR(100) NOT NULL
Account
PK : ID : VARCHAR(20) NOT NULL
Iden : VARCHAR(1) NOT NULL
PassWd : VARCHAR(10) NOT NULL
StuAss
PK / FK : StuID : VARCHAR(20) NOT NULL
PK / FK : AssID : INT NOT NULL
Done : VARCHAR(1) NOT NULL
Assignment
PK : AssID : INT IDENTITY(1,1) NOT NULL
Question1 : VARCHAR(500) NOT NULL
Weight1 : DECIMAL(18,2) NOT NULL
Question2 : VARCHAR(500) NOT NULL
Weight2 : DECIMAL(18,2) NOT NULL
Question3 : VARCHAR(500) NOT NULL
Weight3 : DECIMAL(18,2) NOT NULL
Question4 : VARCHAR(500) NOT NULL
Weight4 : DECIMAL(18,2) NOT NULL
Question5 : VARCHAR(500) NOT NULL
Weight5 : DECIMAL(18,2) NOT NULL
StuInfor
PK: StuID : VARCHAR(20) NOT NULL
StuName : VARCHAR(50) NOT NULL
Gender : VARCHAR(6) NOT NULL
Grade : VARCHAR(10) NOT NULL
Academic : VARCHAR(50) NOT NULL
Email : VARCHAR(50)
PhoneNum : VARCHAR(20)
Answer
PK / FK : StuID : VARCHAR(20) NOT NULL
PK / FK : AssID : INT NOT NULL
A1 : VARCHAR(500) NOT NULL
A2 : VARCHAR(500) NOT NULL
A3 : VARCHAR(500) NOT NULL
A4 : VARCHAR(500) NOT NULL
A5 : VARCHAR(500) NOT NULL
MarkOrNot : VARCHAR(1) NOT NULL
Score
PK / FK : StuID : VARCHAR(20) NOT NULL
PK / FK : AssID : INT NOT NULL
Score1 : DECIMAL(18,2) NOT NULL
Score2 : DECIMAL(18,2) NOT NULL
Score3 : DECIMAL(18,2) NOT NULL
Score4 : DECIMAL(18,2) NOT NULL
Score5 : DECIMAL(18,2) NOT NULL
RealScore1 : DECIMAL(18,2) NOT NULL
RealScore2 : DECIMAL(18,2) NOT NULL
RealScore3 : DECIMAL(18,2) NOT NULL
RealScore4 : DECIMAL(18,2) NOT NULL
RealScore5 : DECIMAL(18,2) NOT NULL
GPA : DECIMAL(18,2) NOT NULL

Figure 2: Logical Schema

4 Implementation

Implementation include database implement and ASP.NET implement. Two implement will connect by information exchange.

4.1 Database Implement

Firstly, create database name 'OnlineGrade'. Create data file and log file, file growth is 5MB.

Then create seven tables just discussed above.

'StuInfor' table: I think student ID, Student name, Gender, Grade, Academic unit are essential for identifying student, so all those are 'NOT NULL'. E-mail and phone number can help teacher to contact the student so it can be 'NULL'. Also add a password schema to store account password. Because the system is designed for MUST undergraduate students, grade limited to 'Freshman','Sophomore', 'Junior' and 'Senior'. Academic Unit limited to the academic units in our school, which implemented in ASP.NET websites.

'Assignment' table: Assignment ID designed to increase 1 automatically after assignment have created. There are five questions in each assignment, so designed schema as five questions and weights. Of course, all schema are 'NOT NULL'. Weight must be positive number, so add check constraint to limit all weights are larger or equal than zero. Because weight might have decimal number, so set data type of weight schema to DECIMAL which keep two decimal places.

'Answer' table: Student ID and Assignment ID both are primary key which means this is the only answer to an assignment of a person. Student ID is foreign key for 'StuInfor' table because the answer must submit from exist student account. Assignment ID is foreign key for 'Assignment' table because the answer must correspond to the arranged assignment. 'Comment' table, 'Score' table, 'StuAss' table also use those two ID to be primary key and foreign key. The reason is same with 'Answer' table, so after the introduction of left tables, I will omit explaining this. To inform teacher to mark assignment, add 'MarkOrNot' schema to sign weather this assignment have been marked. This schema will fill in by ASP.NET website. To prevent accident, also add constraint to limited value.

'StuAss' table: Student ID and Assignment ID both are primary key. The table for signing weather the assignment have been done. System will display assignment or prompt information according to the value of 'Done' schema. Add constraint to check 'Done' value: 'Y' is yes, 'N' is no.

'Comment' table: Student ID and Assignment ID both are primary key. Just store each comment for each question in an assignment.

'Score' table: Student ID and Assignment ID both are primary key. I designed an standard: "The full score of each question is 100 points, the finally score for each question will calculated by the product of score(teacher mark) and weight of each question, the GPA is the division between sum of real score and sum of weight." Teacher need to mark each question according to this standard. Schema 'Score1' to 'Score5' is the score of each question teacher mark. Schema 'RealScore1' to 'RealScore5' is the product of score(teacher mark) and weight of each question. Schema 'GPA' is the division between sum of real score and sum of weight.

Because all score and GPA may have decimal number, so set all data type to DECIMAL.

'Account' table: The table to store all account ID and password, also have schema 'Iden' to identify the identity of user. Add constraint to 'Iden', which limited to 'T'(represent teacher) and 'S'(represent students).

All table must be created before system first use. I designed default administrator ID and password both are 'admin'. So add initial administrator account data into 'Account' table which enable teacher to start use the system.

4.2 ASP.NET Implement

Asp.Net implement include Log interface design, teacher interface design and student interface design.

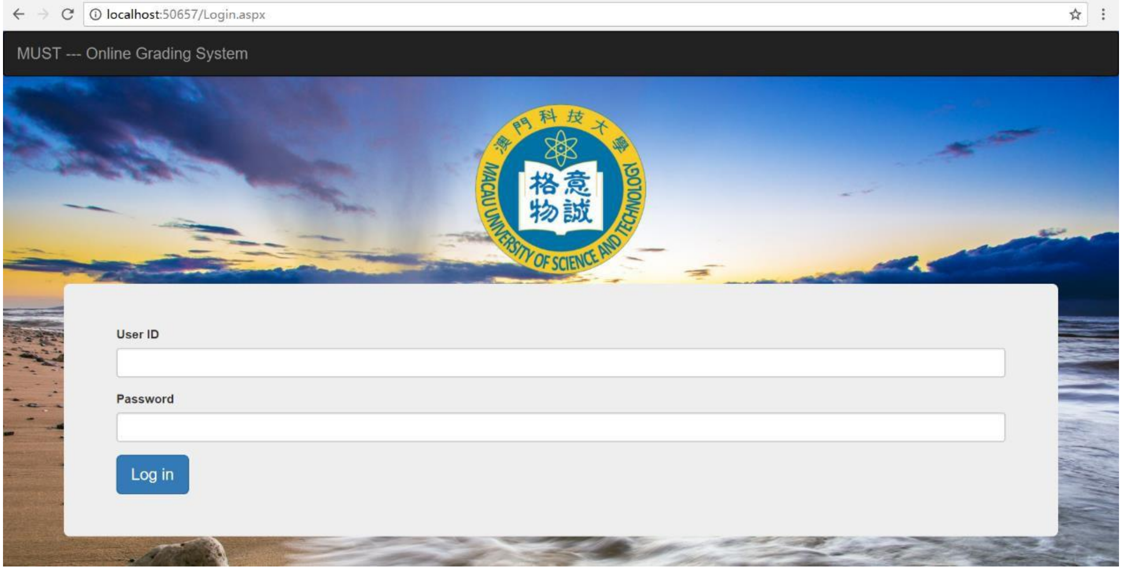
All websites have the navigation bar for the convenience of user. Teacher and student have different navigation bar because they have different permissions and features. When click navigation bar button, page will jump to corresponding function page.

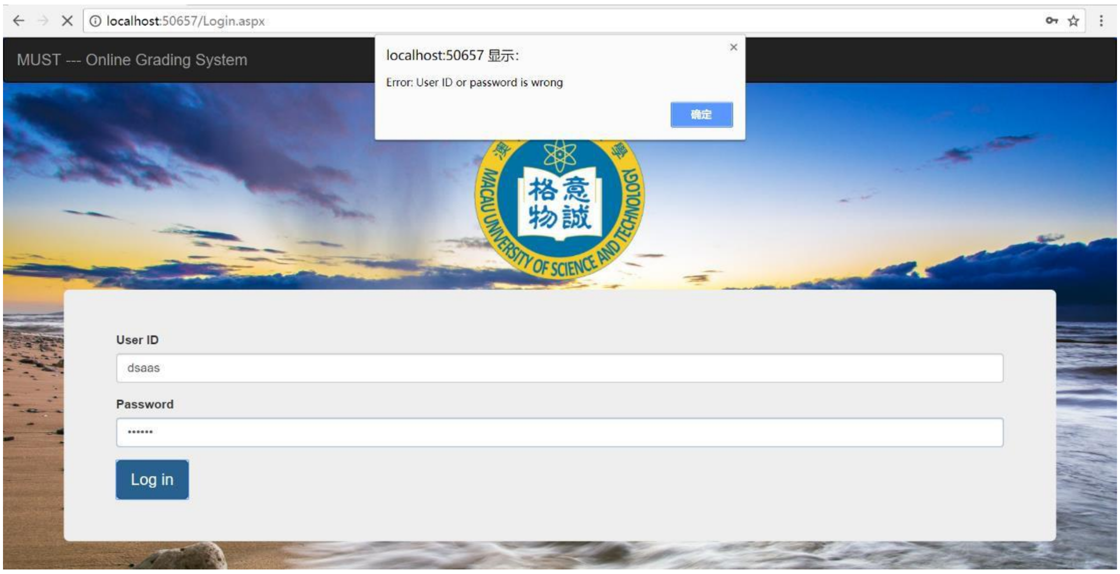
First page is Log page, after detect right account, user will jump to different page according to the ID identity. After Logging in, the user ID will transfer all page(which using cookie technology) until log out.

4.2.1 Log Interface Design

Login.aspx : In order to be consistent with other interfaces, Log page include a useless navigation bar. Use bootstrap framework to design interface, add two ASP textbox to get user ID and password. When user click 'Log in' button, 'Log_Click' function will execute. Connecting to database and compare database data from 'Account' table and user input data. If right, search the identity of user ID from 'Account' table and jump to corresponding Home pages. Also add user ID to cookie for later use. If wrong or some textbox are unfilled, reset two textbox to initial state and prompt user to enter again. Figure 3 shown the log page and Figure 4 shown the wrong prompt.

Figure 3: Log Page(Login.aspx)





localhost:50657 显示:

Error: User ID or password is wrong

确定

User ID

dsaas

Password

••••••

Log in

Figure 4: Wrong Prompt(Login.aspx)

4.2.2 Teacher Interface Design

Teacher use case is shown as Figure 5. In addition to the figure said, All page can click navigation bar to jump to other accessable page. All page will return to Log page after click ‘Login Out’.

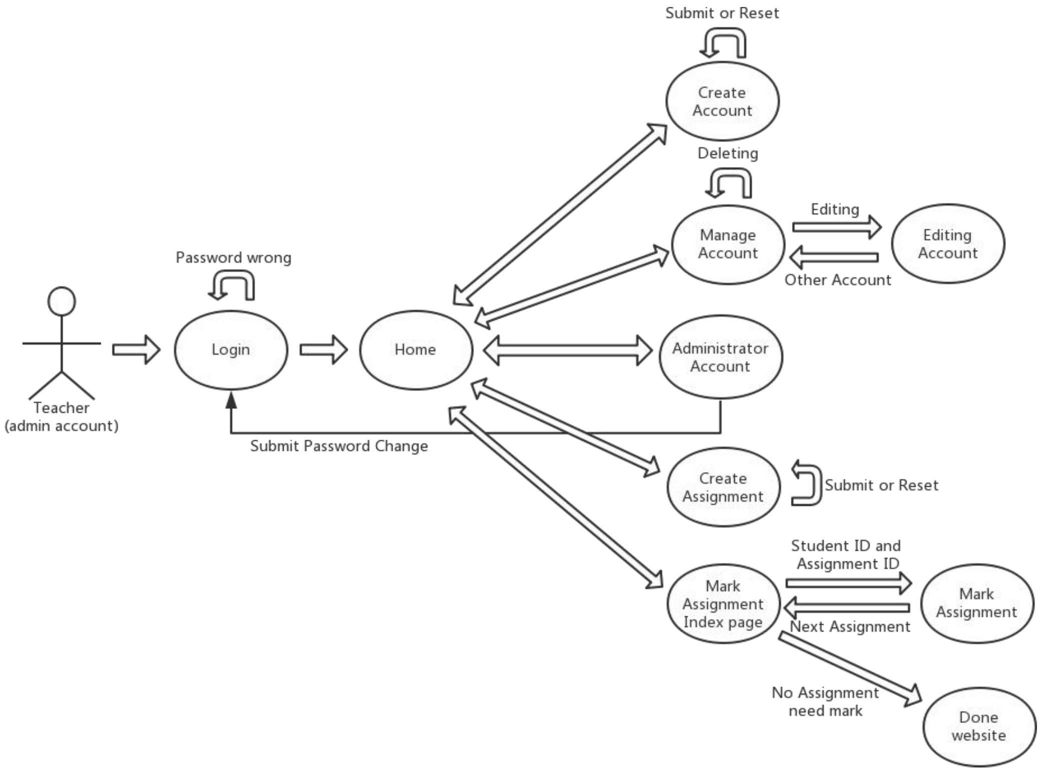
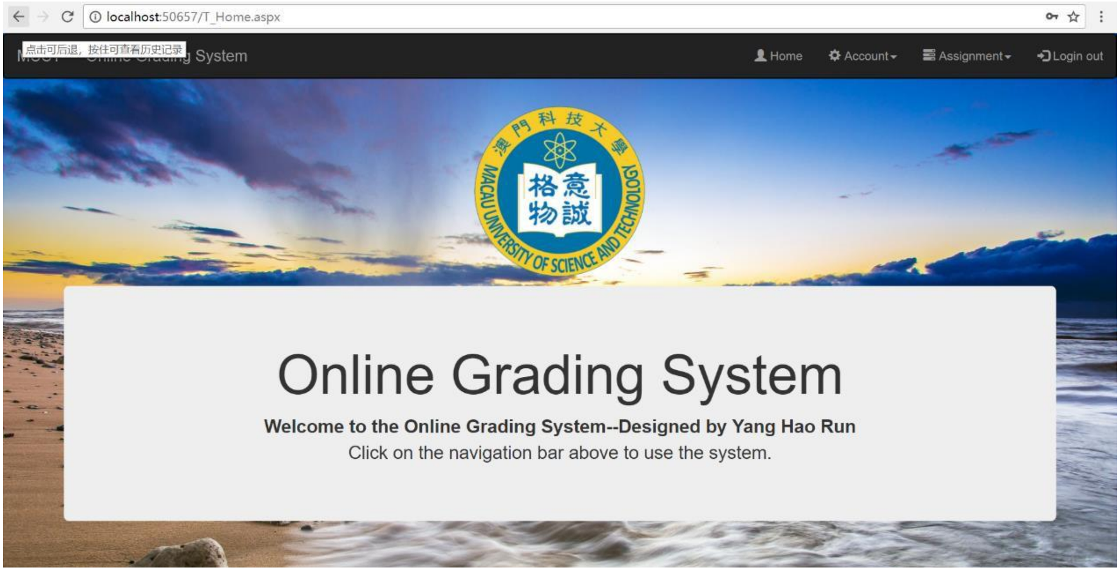


Figure 5: Teacher Use Case

T-Home.aspx : A Welcome page. Prompt user to select function above the navigation bar. Shown as Figure 6.



Online Grading System

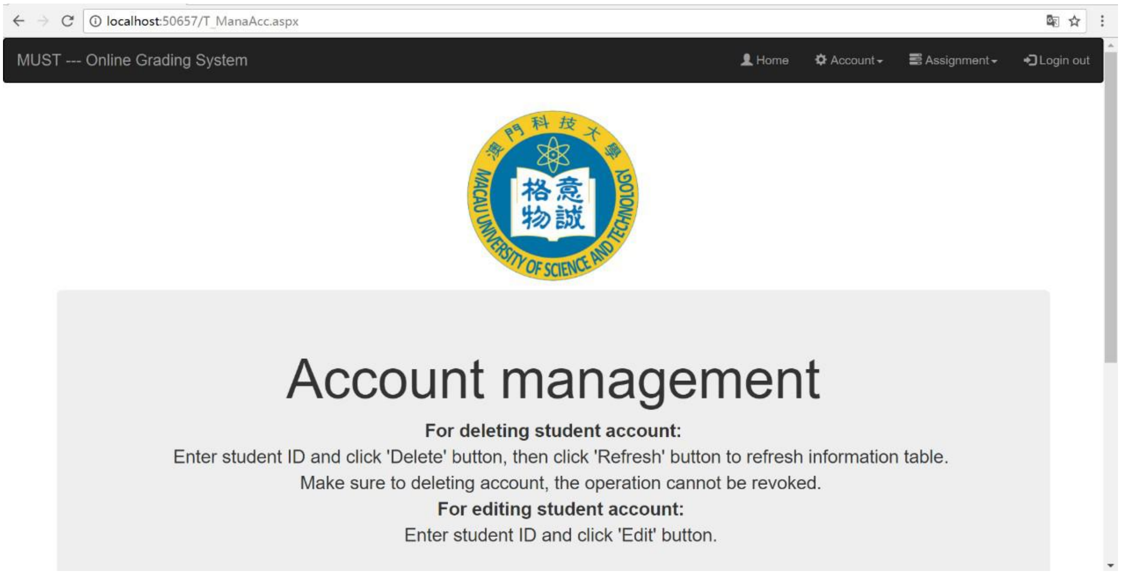
Welcome to the Online Grading System--Designed by Yang Hao Run

Click on the navigation bar above to use the system.

Figure 6: Home page(T-Home.aspx)

All other page have title and body. Title is to introduce the function and how to use it. Shown as Figure 7. Each page body will introduce below.

T_CreaAcc.aspx : Page for filling student information and creating account. Use textbox, RadioButtonList(Gender, Grade), DropDownList(Academic Units) to get input from user. 'Reset' button reset all input module. 'Submit' button connect database and insert data into 'StuInfor' table. At default, the password of new student account always is '123456'. Student can change password in weibsite *S_Acc.aspx*. If success, prompt "Successful" and reset all input module. If fail, prompt error message and reset. The part of interface of this page is shown as Figure 8.


localhost:50657/T_ManaAcc.aspx
MUST --- Online Grading System
Home
Account
Assignment
Login out

Account management

For deleting student account:

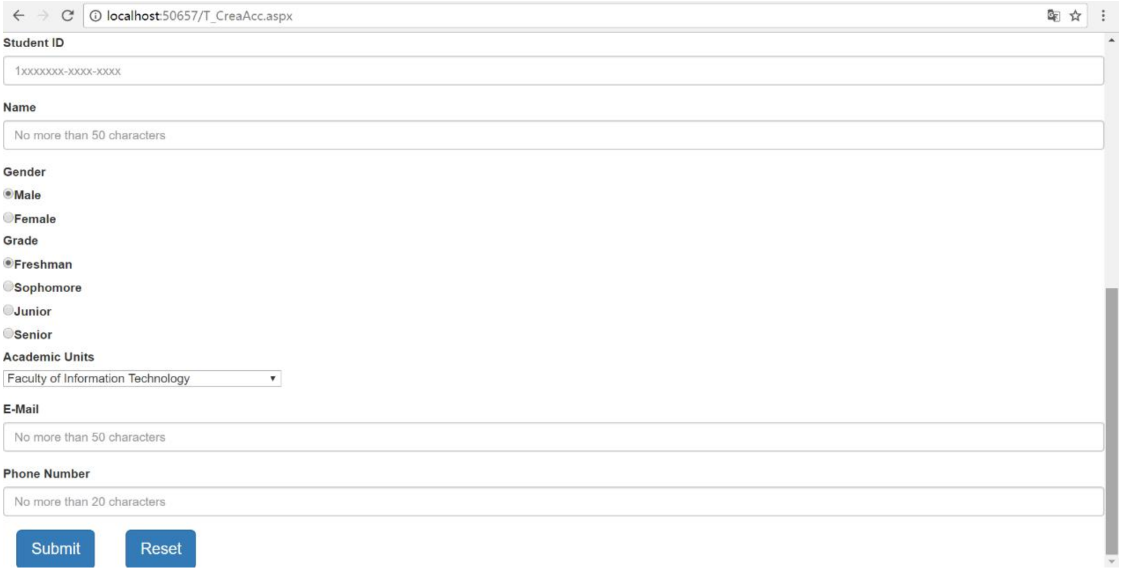
Enter student ID and click 'Delete' button, then click 'Refresh' button to refresh information table.

Make sure to deleting account, the operation cannot be revoked.

For editing student account:

Enter student ID and click 'Edit' button.

Figure 7: Introduce page(T_ManaAcc.aspx)



localhost:50657/T_CreaAcc.aspx

Student ID

1xxxxxxx-xxxx-xxxx

Name

No more than 50 characters

Gender

● Male

○ Female

Grade

● Freshman

○ Sophomore

○ Junior

○ Senior

Academic Units

Faculty of Information Technology ▾

E-Mail

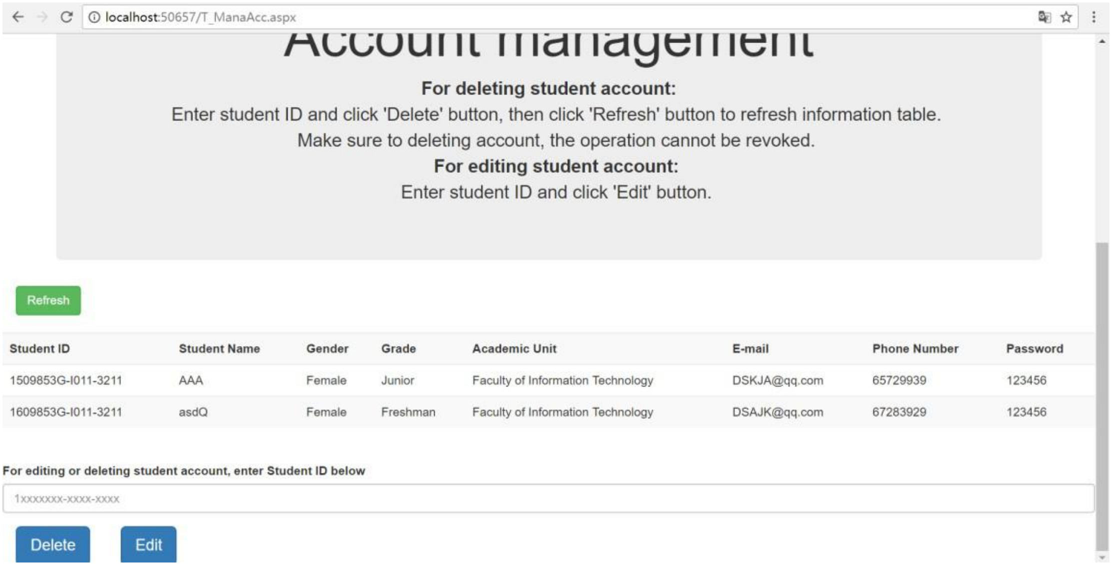
No more than 50 characters

Phone Number

No more than 20 characters

Figure 8: Input module(T_CreaAcc.aspx)

T_ManaAcc.aspx : Page for managing student account. This page will display all student information. Use textbox to get student ID from user. Connecting database and find student ID data from 'StuInfor' table. If find, deleting account or jump to editing account page. Also will transfer student ID to *T_EditAcc.aspx* by cookie technology. If not find, prompt User and reset textbox. Noticeable, there are several foreign key reference 'StuInfor' table. If deleting 'StuInfor' table in front of those table who have 'StuInfor' table foreign key, it will fail and return error message. So importantly, in deleting progress, 'StuInfor' table delete at last. After deleting account, should click 'Refresh' button to refresh page for updating shown data. The part of interface of this page is shown as Figure 9.



localhost:50657/T_ManaAcc.aspx

Account management

For deleting student account:

Enter student ID and click 'Delete' button, then click 'Refresh' button to refresh information table.

Make sure to deleting account, the operation cannot be revoked.

For editing student account:

Enter student ID and click 'Edit' button.

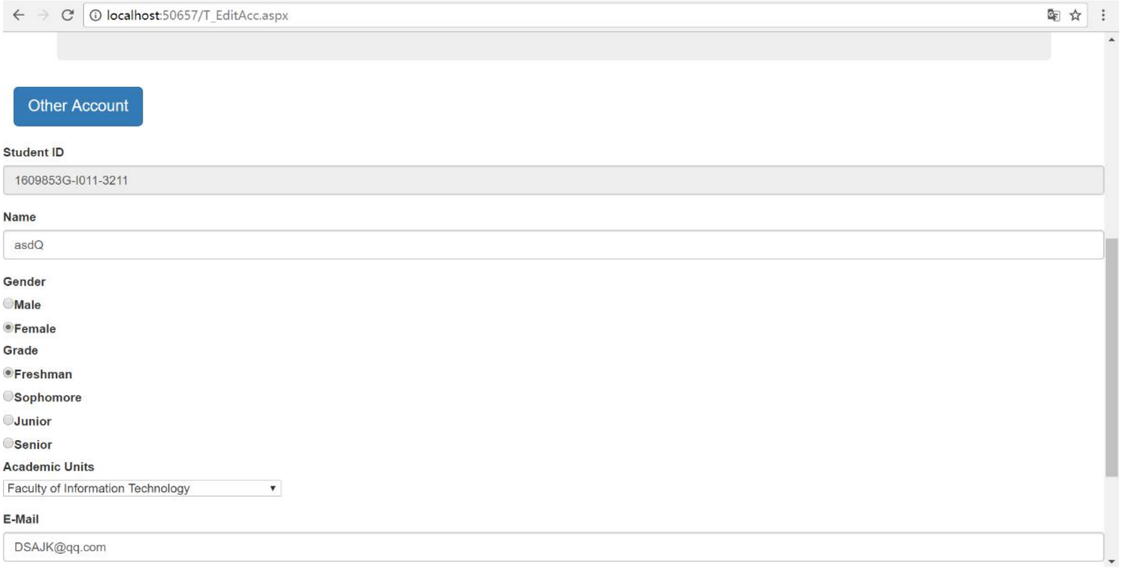
Student ID	Student Name	Gender	Grade	Academic Unit	E-mail	Phone Number	Password
1509853G-I011-3211	AAA	Female	Junior	Faculty of Information Technology	DSKJA@qq.com	65729939	123456
1609853G-I011-3211	asdQ	Female	Freshman	Faculty of Information Technology	DSAJK@qq.com	67283929	123456

For editing or deleting student account, enter Student ID below

1xxxxxxx-xxxx-xxxx

Figure 9: The part of interface(T_ManaAcc.aspx)

T_EditAcc.aspx : Page for editing student account. Teacher have permissions to modify all information. This page get student ID from cookie, and display information to all modifiable textbox, RadioButtonList(Gender, Grade) and DropDownList(Academic Units). Teacher can modify information and submit to update database. Compare with creating account, this page can modify student's password. Also can click 'Other Account' button to return *T_ManaAcc.aspx.* The part of this page is shown as Figure 10.



localhost:50657/T_EditAcc.aspx

Other Account

Student ID

1609853G-I011-3211

Name

asdQ

Gender

○Male

●Female

Grade

●Freshman

○Sophomore

○Junior

○Senior

Academic Units

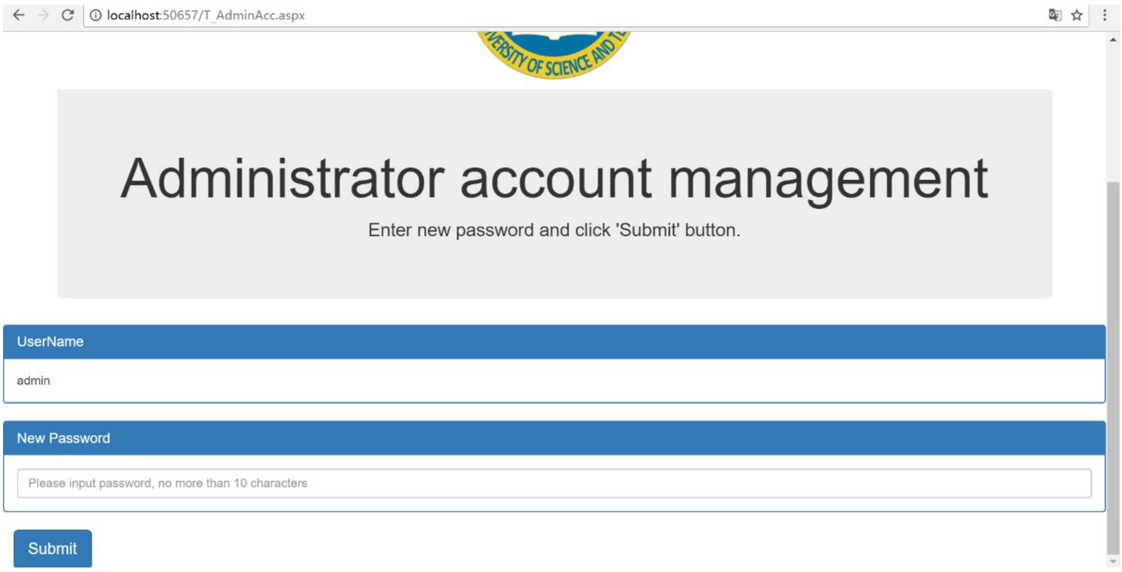
Faculty of Information Technology ▾

E-Mail

DSAJK@qq.com

Figure 10: The part of interface(T_EditAcc.aspx)

T_AdminAcc.aspx : Page for editing password of teacher account. For easily administrated, the user name of administrator account must be 'admin'. After click 'Submit', will update database, prompt any error message, return to Log page. The part of this page is shown as Figure 11.



localhost:50657/T_AdminAcc.aspx

Administrator account management

Enter new password and click 'Submit' button.

UserName

admin

New Password

Please input password, no more than 10 characters

Submit

Figure 11: The part of interface(T_AdminAcc.aspx)

T_CreaAss.aspx : Page for creating assignment. Each assignment have five question and weights. Each question have different color of frame for easily recognize. The part of this page is shown as Figure 12.

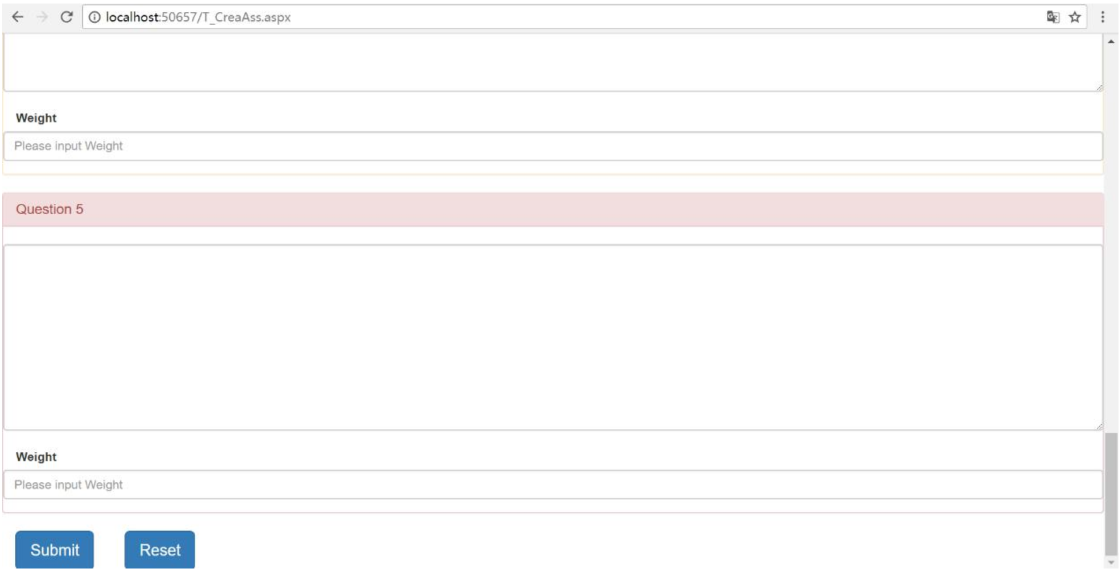
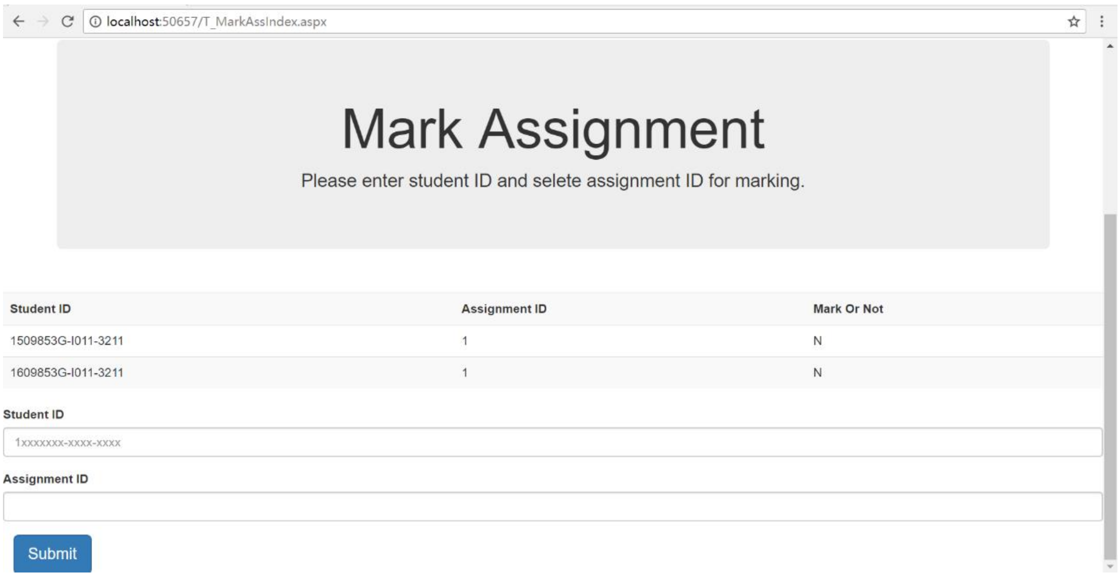


Figure 12: The part of interface(T_CreaAss.aspx)

T_MarkAssIndex.aspx : Page for the marking interface index page. Show all submitted assignment marking state. Enter and transfer student ID and assignment ID to *T_MarkAss.aspx* for marking. If all assignment have been mark. Jump to *T_MarkDone.aspx* to prompt user. The part of this page is shown as Figure 13



Mark Assignment

Please enter student ID and selete assignment ID for marking.

Student ID	Assignment ID	Mark Or Not
1509853G-I011-3211	1	N
1609853G-I011-3211	1	N

Figure 13: The part of interface(T_MarkAssIndex.aspx)

T_MarkAss.aspx : Page for marking assignment. Same as *T_CreaAss.aspx*, *T_MarkAss.aspx* also have different color of frame for easily recognize. What's more, This page use panel-group and collapse plug-ins to show each question, weight and answer. Teacher also enter comment and score inside each panel. After marking, can return *T_MarkAssIndex.aspx* to mark next assignment. This page also will calculate real score and GPA, then update database. The part of this page is shown as Figure 14.

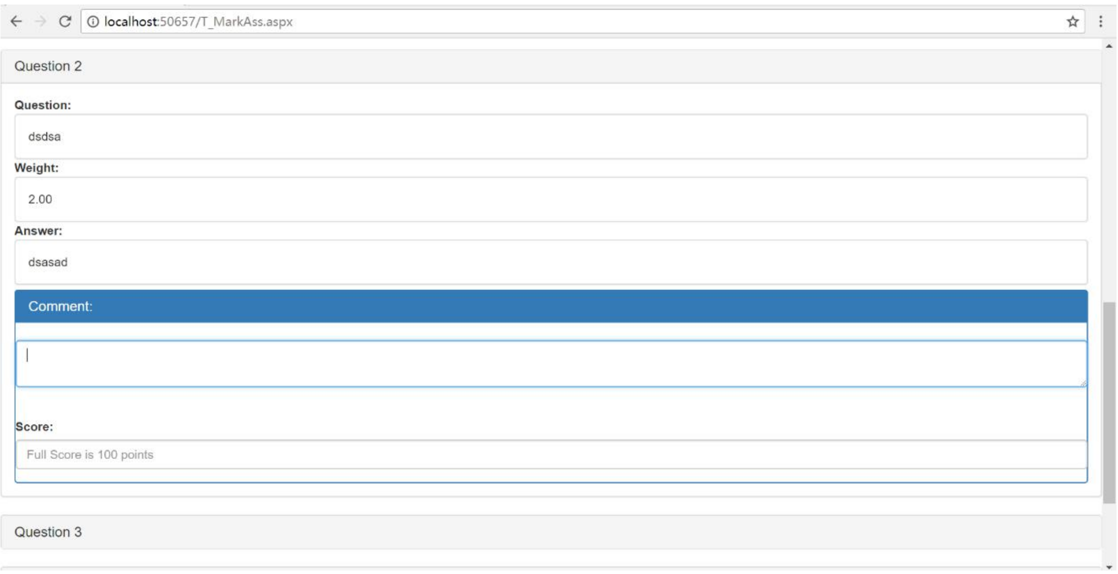
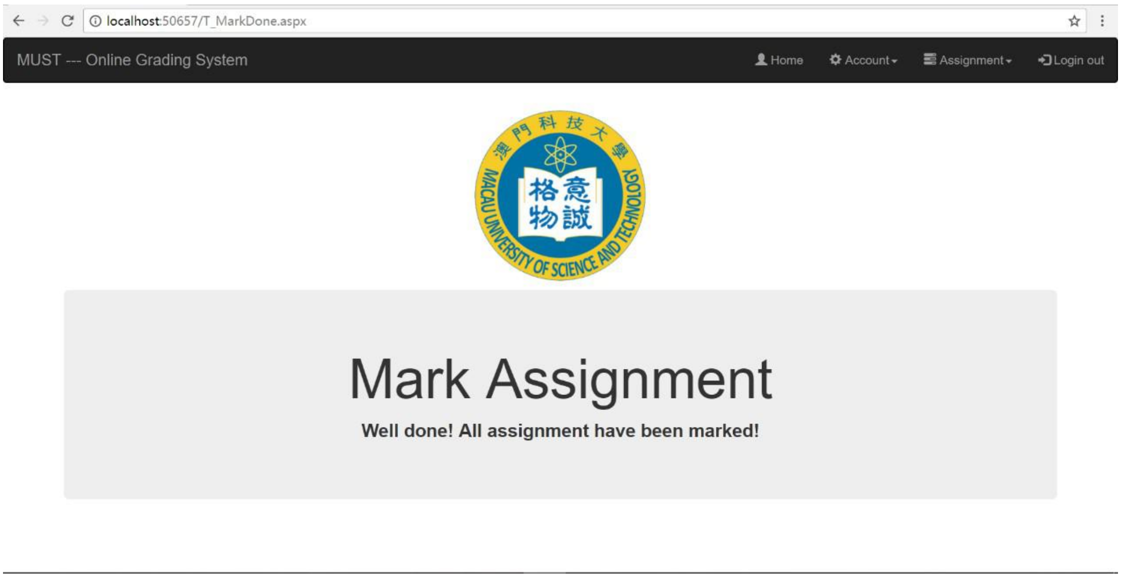


Figure 14: The part of interface(T_MarkAss.aspx)

T_MarkDone.aspx : Page for prompting teacher all assignment have been marked. This page is shown as Figure 15.



localhost:50657/T_MarkDone.aspx
MUST --- Online Grading System
Home
Account
Assignment
Login out
Mark Assignment
Well done! All assignment have been marked!

Figure 15: The interface(T_MarkDone.aspx)

4.2.3 Student Interface Design

Student use case is shown as Figure 16. Same as Teacher interface, all page can click navigation bar to jump to other accessable page. All page will return to Log page after click ‘Login Out’. User ID is Student ID, will transfer to all page with cookie until log out.

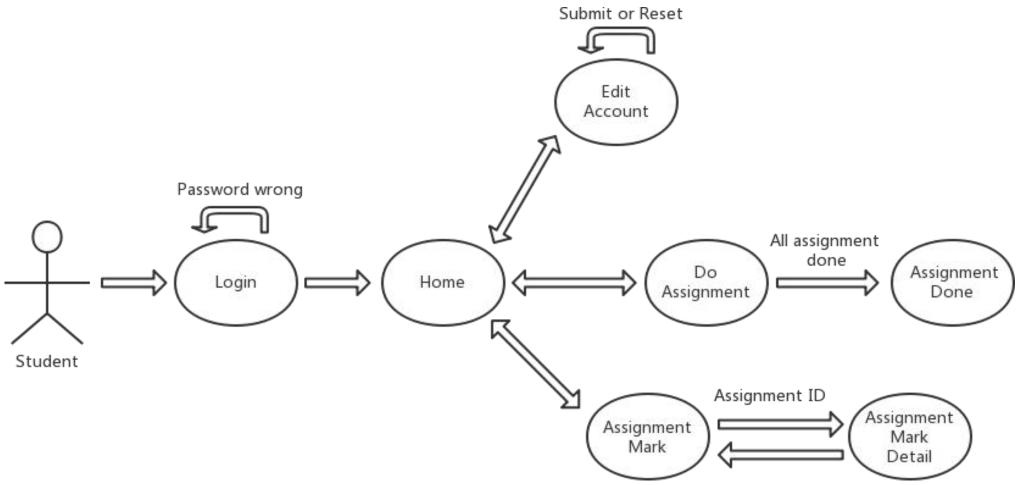
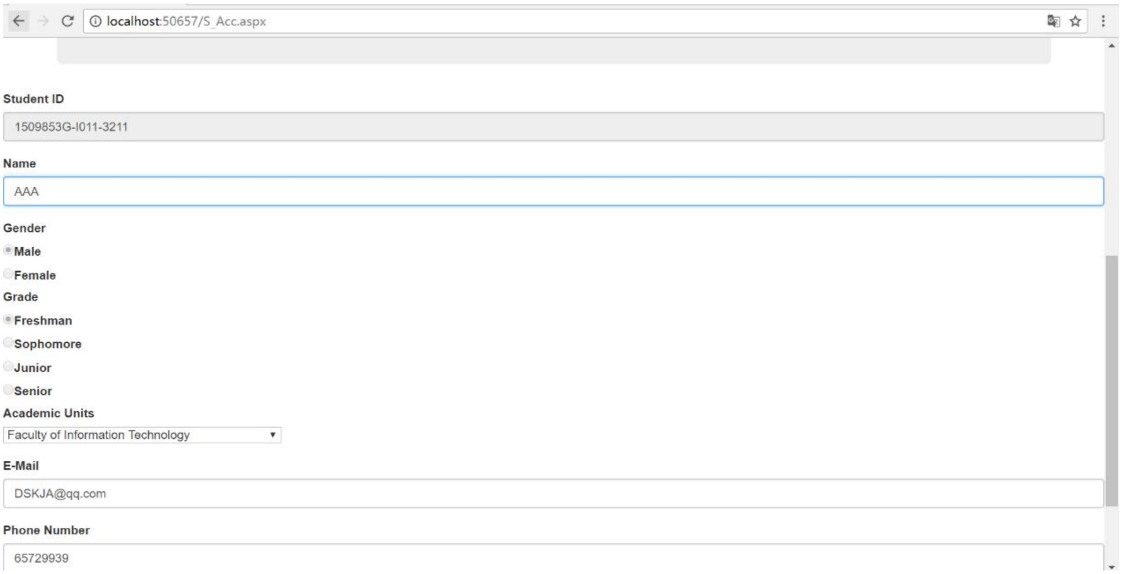


Figure 16: Student use case

S_Home.aspx : A Welcome page same as *T_Home.aspx .*

S_Acc.aspx : Page for editing account information. Student only have permission to modify student name, E-mail, Phone number and account password. Other information just display but can't modify. The part of this page is shown as Figure 17.



Student ID
1509853G-I011-3211

Name
AAA

Gender
● Male
○ Female

Grade
● Freshman
○ Sophomore
○ Junior
○ Senior

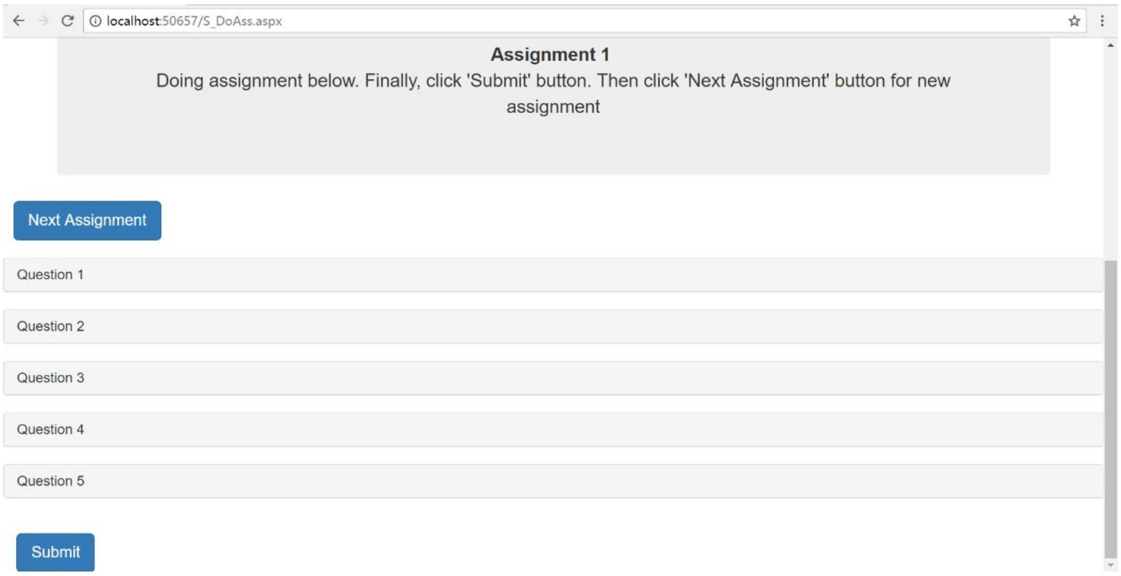
Academic Units
Faculty of Information Technology ▾

E-Mail
DSKJA@qq.com

Phone Number
65729939

Figure 17: The interface(S_Acc.aspx)

S_DoAss.aspx : Page will display question, weight and allow student to answer. Use panel-group and collapse plug-ins for displaying and entering. After submit, Student can click 'Next Assignment' to write next assignment. If no Assignment to do, jump to *S_DoAssDone.aspx* to inform student. The part of this page are shown as Figure 18 and Figure 19.

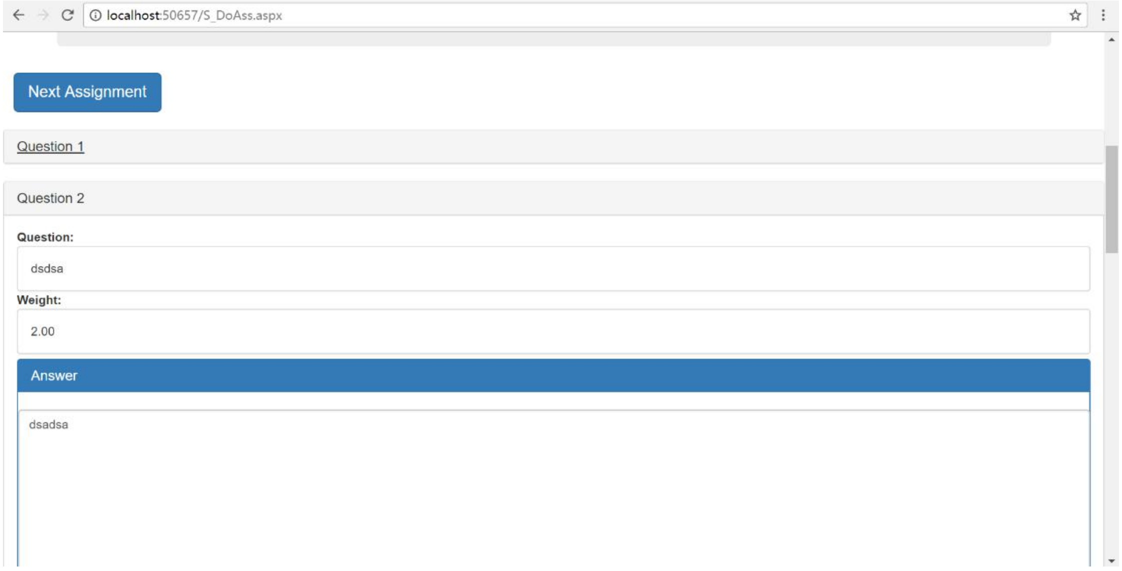


localhost:50657/S_DoAss.aspx

Assignment 1

Doing assignment below. Finally, click 'Submit' button. Then click 'Next Assignment' button for new assignment

Figure 18: The interface before folding (S_DoAss.aspx)



localhost:50657/S_DoAss.aspx

Question: dsdsa

Weight: 2.00

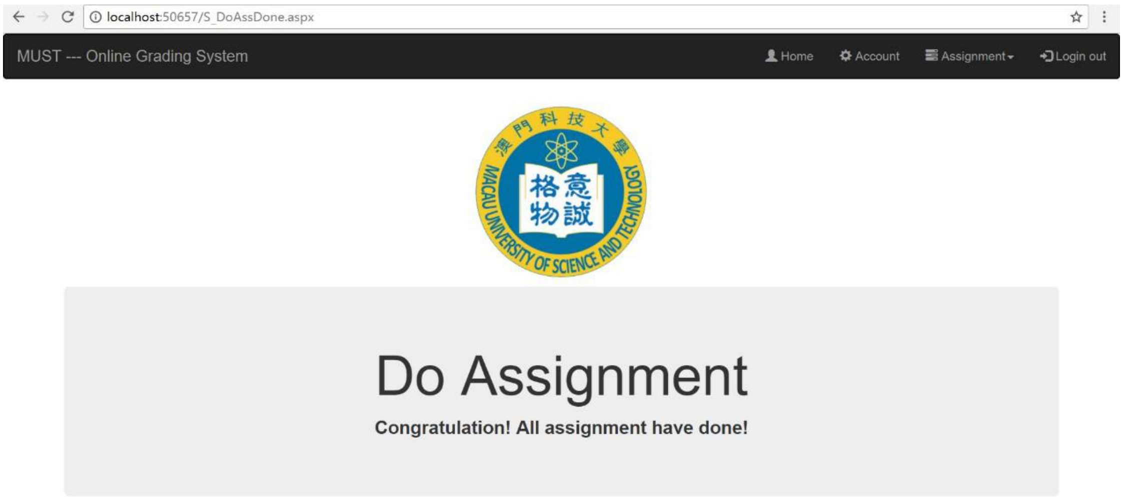
Answer: dsadsa

Figure 19: The interface after unfolding (S_DoAss.aspx)

S_DoAssDone.aspx : Page for informing student all assignment have been done. This page is shown as Figure 20.

S_Mark.aspx : Page for display marked assignment score. Calculate average score and display. Transfer assignment ID to *S_MarkAss.aspx* to view assignment detail. The part of this page is shown as Figure 21.

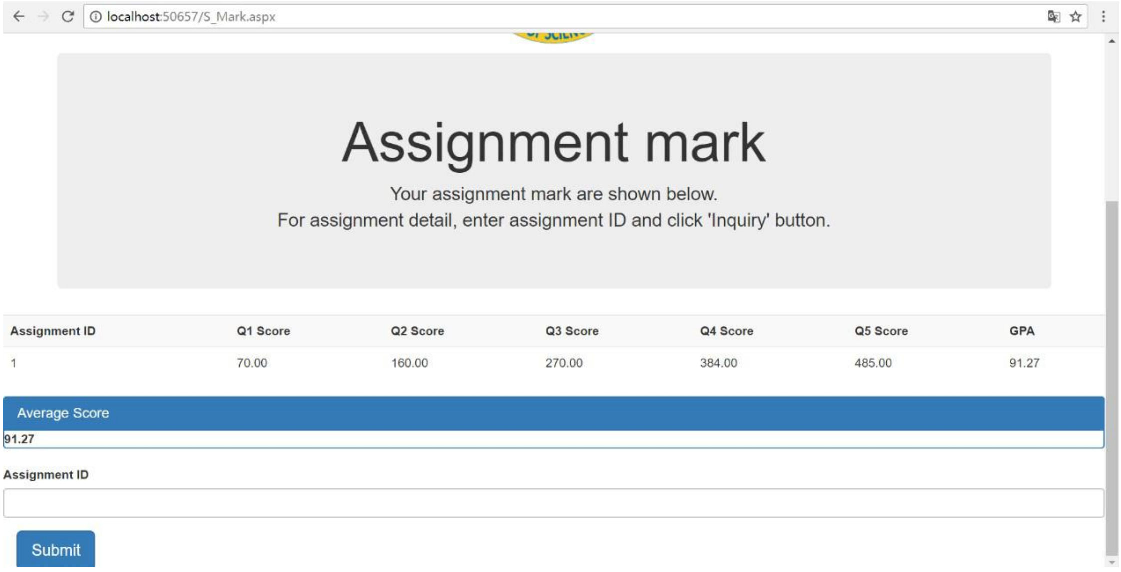
S_MarkAss.aspx : Page for display detail marked assignment. Student can view real score and comment of each question. This page also use panel-group and collapse plug-ins to show information. After viewing, can return S_Mark.aspx to view another assignment detail. The part of this page is shown as Figure 22.


localhost:50657/S_DoAssDone.aspx
MUST --- Online Grading System
Home
Account
Assignment
Login out

Do Assignment

Congratulation! All assignment have done!

Figure 20: The interface (S_DoAssDone.aspx)



Assignment mark

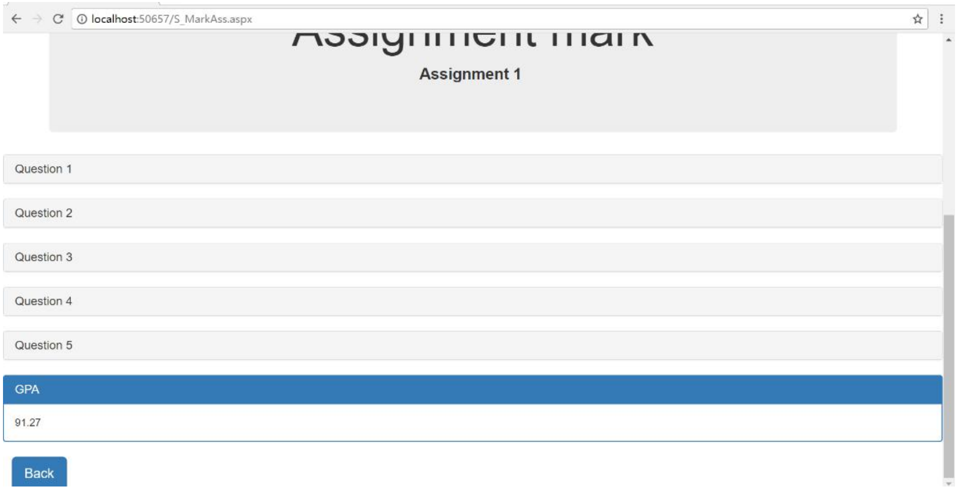
Your assignment mark are shown below.
For assignment detail, enter assignment ID and click 'Inquiry' button.

Assignment ID	Q1 Score	Q2 Score	Q3 Score	Q4 Score	Q5 Score	GPA
1	70.00	160.00	270.00	384.00	485.00	91.27

Average Score
91.27

Assignment ID

Figure 21: The interface (S_Mark.aspx)



Assignment mark

Assignment 1

GPA
91.27

Figure 22: The interface with folding (S_MarkAss.aspx)

5 User Manual

All operation should start from Login page, or will get error. All password limit 10 characters.

5.1 Teacher

Initial user name is ‘admin’, password is ‘admin’.

Create Account: Enter information and submit. All new student account have default password ‘123456’. After submit, also can continue entering information to create next account.

Manage Account:

a) Deleting: Enter student ID and click ‘Delete’, Then click ‘Refresh’ to update information in this page. THIS OPERATION WILL DELETE ALL INFORMATION AND CAN’T WITHDRAW.

b) Editing: Enter student ID and click ‘Edit’. Edit student information and click ‘Submit’ to update database. Click ‘Reset’ to set all input be the same as the data in database. Click ‘Other Account’ to return index page.

Administrator Account Management: Enter new password and click ‘Submit’. MUST REMEMBER NEW PASSWORD BEFORE CLICK ‘Submit’. After submit, you will need to log again immediately.

Assignment Creation: Enter question and weight(the each question weight of assignment). After submit, also can continue entering information to create next assignment.

Mark Assignment: Enter student ID and assignment ID then click ‘Submit’ to mark. Enter comment and score(the full score for each QUESTION is 100 points). Click ‘Submit’ to submit data. Click ‘Next Assignment’ to mark next assignment.

5.2 Student

Initial user name is your student ID value, password is ‘123456’.

Edit Account: You only have permission to modify student name, E-mail, Phone number and account password. Other information just display but can’t modify. Click ‘Submit’ to update database. Click ‘Reset’ to set all input be the same as the data in database.

Do Assignment: Click ‘question’ to unfolding question and enter answer. Click ‘Submit’ to update database. After done current homework, click ‘Next Assignment’ to do next assignment.

Mark Assignment: View your score and average mark. Enter your assignment ID to view assignment detail. Click ‘Back’ button to return last page.

6 Other

Description of the queries and transactions are shown in ***ProjectSQL.sql*** file. Code fragments with explanation are shown in all code file.