

---

# CS3425 Course Project

## Online Exam System

### (Phase2 and Demo)

---

#### Table of Contents

---

<b>Important information .....</b>	<b>2</b>
<b>Tips for phase2 .....</b>	<b>2</b>
<b>Phase Two Tasks (110 points) .....</b>	<b>4</b>
1. Instructor functions .....	4
2. Student functions in the web browser .....	5
3. What to turn in to Canvas in Phase2? report.pdf .....	6
4. Create archive file onlineExam.zip .....	6
<b>Project Demonstration (40 points) .....</b>	<b>7</b>
Time and location .....	7
How to prep for the demonstration .....	7
<b>Appendix: Sample User interfaces.....</b>	<b>8</b>
Login Screens .....	8
Instructor Interfaces.....	9
Student Interfaces.....	11

## Important information

---

This project will be implemented in 2 phases with a required project demonstration to TA.

Phase 1: Database Design and Admin Functions .

Due **Week11** Monday 11:59pm.

Phase 2: User Function Implementations .

Due **Final week Wed** 11:59pm.

Demo to TA: Show selected functions

Due **Final Week Tuesday** 11:50pm. **This is subject to change due to TA's schedule.**

## Tips for phase2

---

1. It is good practice to separate the database functions, business logic and user interface in different files. Since our course project does not focus so much on the user interface, you may combine the business login and user interface together.

For example:

- a. A file for all database functions
- b. A file for student functions
- c. A file for instructor functions
- d. A file for common functions between students and instructors.

If one file needs to use functions from the other file. Use:

```
require "FILENAME "
```

2. Build your application bottom up, not top down. Get the core function working first. Don't worry about how to collect the user input first.

For each student or instructor function, do the following

- a. First, get the SQL statements working in MySQL Workbench before you use it in the PHP.
  - i. Use hardcoded data or use variables in MySQL workbench for the user input data
  - ii. Start a transaction whenever the function involves more than one SQL statements.
- b. Then, get it working in PHP code
  - i. Create a function with arguments  
Use prepared statement for statement involving user input  
Don't hard code the value in the function.

Example1:

```
Function get_exams($course_id) {  
...  
return ...  
}
```

```
Function save_responses($course_id, $exam, $id, $responses) {  
...  
return ...
```

```
}
```

- ii. Call the function and display the result

Again, for the data that should be read from user input, set the value as hardcoded first.

Example1:

```
$course_id='CS1000'  
$exams = get_exams($course_id)  
Print_r($exams)
```

Example 2:

```
$course_id='CS1000'  
$exam= ...  
$id = ...  
$responses= array(  
    "Q1" => "A",  
    "Q2" => "B",  
);  
save_response($course, $exam, $id, $responses);
```

3. After you have the core SQL functions ready, now create the user interface
  - a. Use POST method
  - b. Change the above hardcoded variable to be `$_POST[...]`
  - c. Call appropriate function when user clicked some "submit" buttons
4. Add login function and use sessions
  - a. Force user to login
  - b. Keep the login user id and other needed information in `$_SESSION`

## Phase Two Tasks (110 points)

In phase two, the instructor and student functions will be implemented. While you are programming, you will need to pay attention to whether **transaction** should be used, how to prevent **SQL injection** and how to handle **password** safely.

### 1. Instructor functions

---

Due to limited time, we will only implement a few functions for instructor. You may create exams with any method of your choices. Such as insert statement, procedures, load data file, etc.

- 1) Login and logout
  - a. The instructor has to login first
  - b. The system must force the instructor reset the password when they login to the system for the first time.
  - c. (Optional) The instructor should be able to change their password later. Due to time limit, you may choose not to implement this function.

This is same as student's login function. Please note the system won't allow two different users logged in simultaneously due to the usage of SESSION in php. The user has to logout (or you destroy the SESSION somehow manually) before another user can login.

- 2) Main page for instructors

After the instructor login,

  - a. Display a welcome message and a logout button
  - b. List all the courses that the instructor is teaching
  - c. List the exams that were created for the course.
  - a. A input form with button "Review Exam Problems" and Check "Exam Score" . If you use different web pages to implement these functions, include "Go home" button to go back to the home page.

Redirect user to the login page if no instructor has not logged in yet.

- 3) Display the questions for an exam

Instructor enters the course and exam name, the system list all the problems in the exam including choices, correct answers, points.
- 4) Display the score for an exam

Instructor enters the course and exam name, the system list

  - Statistic information about the exam:
    - min, max, avg
    - How many students in the class, how many completed the exam
  - The score of each students

## 2. Student functions in the web browser

---

You shall design and implement the following functions for Students as a Web applications

### 2) Login and logout

- a. The instructor has to login first
- b. The system must force the instructor reset the password when they login to the system for the first time.
- c. (Optional) The instructor should be able to change their password later. Due to time limit, you may choose not to implement this function.

### 3) Student main page

After the student login, display the following information

- a. Display a welcome message and a logout button
- b. display Show what classes the student is taking.
- c. Show the exams for all the courses that the student is taking and when the student took it and the score they got.
- d. Show what classes that the student is not taking.
- e. A input form with button "Register New Courses", "Take Exam", "Check Score". If you use different web pages to implement these functions, include "Go home" button to go back to the home page.

Redirect user to the login page if no instructor has not logged in yet.

### 4) Register for a new class

Allow the student to enter class name and register the students into the class.

### 5) Take an open exam for a class.

Ask student to enter the class name and exam name, then display the questions for students to answer.

Please note the system grades the exam automatically after student submits. This could be done by using a trigger automatically, or calling a stored procedure explicitly in your code.

### 6) Check the result for exams.

Ask students to enter the class name and exam name, list the following

- The total numeric score and percentage
- Student's answer, solution, the point that student get for each problem.

### 3. What to turn in to Canvas in Phase2? [report.pdf](#)

---

Please include the following in your report.pdf

- 1) (40p) Show how the above instructor function work. Use screenshots.
- 2) (50p) Show the screen shot for the above required student functions. Use screenshots.
- 3) (5p) Describe how transactions are used in your project. Copy the related code.
- 4) (5p) Describe how your code prevent SQL injection. Copy the related code.
- 5) (5p) Describe how you encrypt the password and force user to reset their password when they login for the first time. Copy the related code.
- 6) (5p) Describe how you use SESSION.

### 4. Create archive file [onlineExam.zip](#)

---

No need to submit the zip file. You might want to save every you did in a zip file for your own reference.

- 1) Table creation SQL statements
- 2) PSM (Stored procedures, Functions source code) and Trigger code
- 3) Web HTML and PHP code

## Project Demonstration (40 points)

---

The purpose of this demo is to check your user interface and functions. You will be asked by the TAs to perform some selected functions. It will last about 10 minutes.

### Time and location

---

The demo will happen during the exam week. Please sign up for the 10-minute meeting in this google sheet [The link will come later ....]. Please select a slot in that works for you and your partner, and write down your group number following by team member's names.

**LOCATION:** Please note the meeting will be in Zoom, not in person.

### How to prep for the demonstration

---

Please set up your environment (database, web browser) before the meeting, then enter the Zoom meeting at your selected time.

Please have some good data in your database prior to the exam.

The possible functions that you need to present during the meeting will be selected below:

Admin:

- create an exam for a course

Student:

- login and change password
- main page
- register a course
- take an exam
- check the exam status

Instructor:

- main page
- Review exam questions
- Check grades for an exam

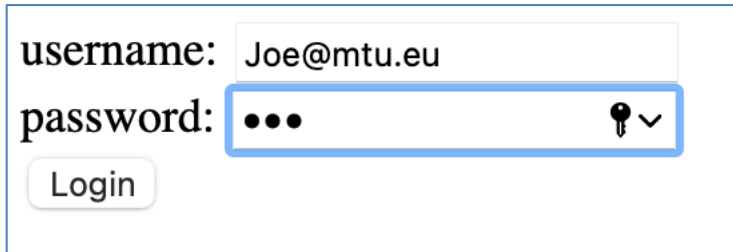
## Appendix: Sample User interfaces

---

### Login Screens



---

1. Login



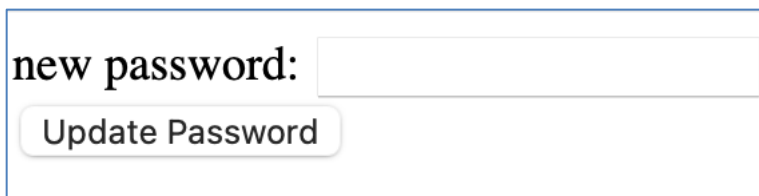
A login form UI mockup. It features two input fields: 'username:' with the value 'Joe@mtu.eu' and 'password:' with three dots indicating a masked password. A blue rectangular highlight is placed around the password input field. To the right of the password field is a key icon and a downward arrow. Below the password field is a 'Login' button.

username: Joe@mtu.eu

password: ●●●  

Login

2. Reset password upon first login



A 'Reset password' form UI mockup. It contains a single input field labeled 'new password:'. Below the input field is an 'Update Password' button.

new password:

Update Password



### 1. Main screen

Dear instructor Linda@mtu.edu, Welcome!

[logout](#)

Here are the courses you are teaching and the exams that you have created for each course

id	title	credit	exam_name	open_time	close_time	total_points
CS1000	Expolore	1.0	final	2022-10-20 09:00:00	2022-12-19 09:00:00	15.0
CS1000	Expolore	1.0	midterm	2022-10-20 09:00:00	2022-12-19 09:00:00	15.0

Please enter the course id and the exam name to see the score of students.

Course:

Exam:

[Check Score](#)

[Review Exam](#)

[Create Exam](#)

### 2. Check Score for an exam

Please enter the course id and the exam name to see the score of students.

Course:

Exam:

[Check Score](#)

[Review Exam](#)

[Create Exam](#)

c_id	total	exam_name	Completed	Mininum	Maximum	Average
CS1000	3	midterm	2	5.0	8.0	6.50000

id	name	start_time	end_time	score
S1	Mary	2022-10-01 10:00:00	2022-10-01 10:20:00	5.0
S2	Joe	2022-10-02 09:00:00	2022-10-02 10:20:00	8.0

[Go Back](#)

### 3. Review Exam

Here are questions for exam midterm

Q1: What is  $3+5$ ?

A:28 (Correct)

B:20

C:18

D:15

Q2: What is  $5*5$ ?

A:225 (Correct)

B:100

C:30

Q3: How many inches of rain falling in Houghton??

A:1

B:10

C:20

D:2 (Correct)

[Go Back](#)

### 4. Create exam

This function is not implemented yet

[Go Back](#)

## Student Interfaces

### 1. Main page

Dear student Joe@mtu.edu, Welcome!

[logout](#)

Here are the classes that you are taking

c_id	title	credit	name
CS1000	Expolore	1.0	Linda

Here are the exams in each course and your score

c_id	exam_name	open_time	close_time	total_points	start_time	end_time	score
CS1000	final	2022-10-20 09:00:00	2022-12-19 09:00:00	15.0			
CS1000	midterm	2022-10-20 09:00:00	2022-12-19 09:00:00	15.0	2022-10-02 09:00:00	2022-10-02 10:20:00	8.0

Here is the list of classes that you are not enrolled in yet.

id	title	credit	instructor_id
CS2321	Data Structures	3.0	T2

To register new courses, please type the course id, then click the "Register New Course" button.  
To take an exam , please type the course id and the exam name, then click the "Take Exam " button.  
To check the exam score, please type the course id and the exam name, then click the "Check Score " button.

Course Id:

Exam Name:

[Regist New Courses](#) [Take Exam](#) [Check Score](#)

### 2. Register a class

Course Id:

Exam Name:

[Regist New Courses](#) [Take Exam](#) [Check Score](#)

You have been added to CS2321

[Go Back](#)

### 3. Take an exam

Course Id:

Exam Name:

Here are the questions for the exam. Click Submit after you finish.

Q1: What is 13+15?

- ☒ A:28  
☐ B:20  
☐ C:18  
☐ D:15

Q2: What is 15\*15?

- ☐ A:225  
☒ B:100  
☐ C:30

Q3: How many inches of snow falling in Houghton??

- ☐ A:10  
☐ B:100  
☒ C:150  
☐ D:250

### 4. Check the score

score	start_time	end_time	duration_in_sec
5.0	2022-11-05 13:37:30	2022-11-05 13:37:51	21

q_id	description	yourAnswer	yourPoint	CorrectAnswer
Q1	What is 13+15?	A	5	A
Q2	What is 15*15?	B	0	A
Q3	How many inches of snow falling in Houghton??	C	0	D