

320 Requirements

Trevor's Team

October 17, 2014

Abstract

Purpose of System:

The purpose of this system is to be an online tutor for students at the University of Massachusetts Amherst. This learning tool would help teach the students and allow them to create ER diagrams and submit them as homework assignments. They can then be graded by solutions uploaded by the instructor of a course. Feedback will then be provided to help the student learn from their mistakes. The benefits of this system is that it makes submitting learning in the course and submitting homework much easier for both the students and instructor.

Scope:

This document will clearly outline the requirements and the system requirements of ER Diagram Tutoring System. This document should cover all key concepts and fundamental requirements. The requirements should satisfy the standards of the client and provide accuracy in the functionality of the system.

Contents

1	Data Dictionary	3
2	Context Level Diagram	4
3	Level 0 Diagram	5
4	Functional Requirements	6
4.1	Allow Student to Draw a Diagram	6
4.2	Submit Answer	6
4.3	Allow Teacher to Create and Upload Questions/Feedback	7
4.4	Question Browsing	7
4.5	Add User	8
5	Environmental Requirements	8
5.1	ToolBox for Drawing Diagram	8
5.2	Save Progress of Current Assignment	8
5.3	List of Available Questions and Their Status	9
5.4	Open Space to Draw Diagram	9
5.5	Compatibility With Multiple Browsers	9
6	Performance Requirements	9
6.1	Correctness Analysis	9
6.2	Question Loading	9
7	Safety/Security Requirements	9
7.1	Student Navigates to the Main Page	9
7.2	Professor Navigates to the Main Page	10
7.3	Student Closes Application With Unsaved Answers in the Window	11
7.4	Student Submits Page with Blank Answers	11
7.5	Student Submits Assignment That Was Already Submitted . . .	11
7.6	Student or Professor Tries to Log Into Someone Else's Account .	12
8	Robustness Use Cases	12
8.1	Student Enters Diagram	12
8.2	Student Enters Variant of Incorrect Diagram	13
8.3	Professor or Content	13
8.4	System Should Never Crash	14
8.5	UI Should React Correctly To Student and Professor Actions . .	14
9	Accuracy Requirements	14
9.1	Student Submits Correct Diagram	14
9.2	Student Submits Incorrect Diagram	15

10 Login/Logout System Use Cases	16
10.1 Student or professor correctly logs into the system	16
10.2 Student or professor incorrectly logs into the system	16
10.3 Student or professor correctly logs out of the system	17
11 Professor UI Requirements	17
11.1 Professor Clicks on Side or Top Navigation Buttons	17
11.2 Professor Clicks on the Assignment Submission Ticker	17
11.3 Professor Clicks on the Class or Student Ticker	18
11.4 Professor Clicks to Expand the Assignment Creation Sandbox . .	18
11.5 Professor View	19
11.6 Create	20
11.7 Login	21
11.8 Student	22

1 Data Dictionary

Student Info	Pay Info	New Class	Class Availability
<ul style="list-style-type: none"> - Name - Student ID - SSN - Date of Birth - Address - Telp. Number - Gender - Email - Classes Taken - Grades 	<ul style="list-style-type: none"> - Student ID - Amount - Name - Address - Phone - EMail 	<ul style="list-style-type: none"> - Student ID - Transcript - Section - Discussion Section - Class ID - Professor 	<ul style="list-style-type: none"> - Student ID - Class ID - Seat Availability - Class Capacity
Proof of Registration	Send Payment Due	Billing Info	Class Roster
<ul style="list-style-type: none"> - Student ID - Day - Time - Room - Class ID 	<ul style="list-style-type: none"> - Student ID - Name - Amount - Receipt ID 	<ul style="list-style-type: none"> - Student ID - Name - Address - Phone - Email - Amount 	<ul style="list-style-type: none"> - Student ID -Name Major

2 Context Level Diagram

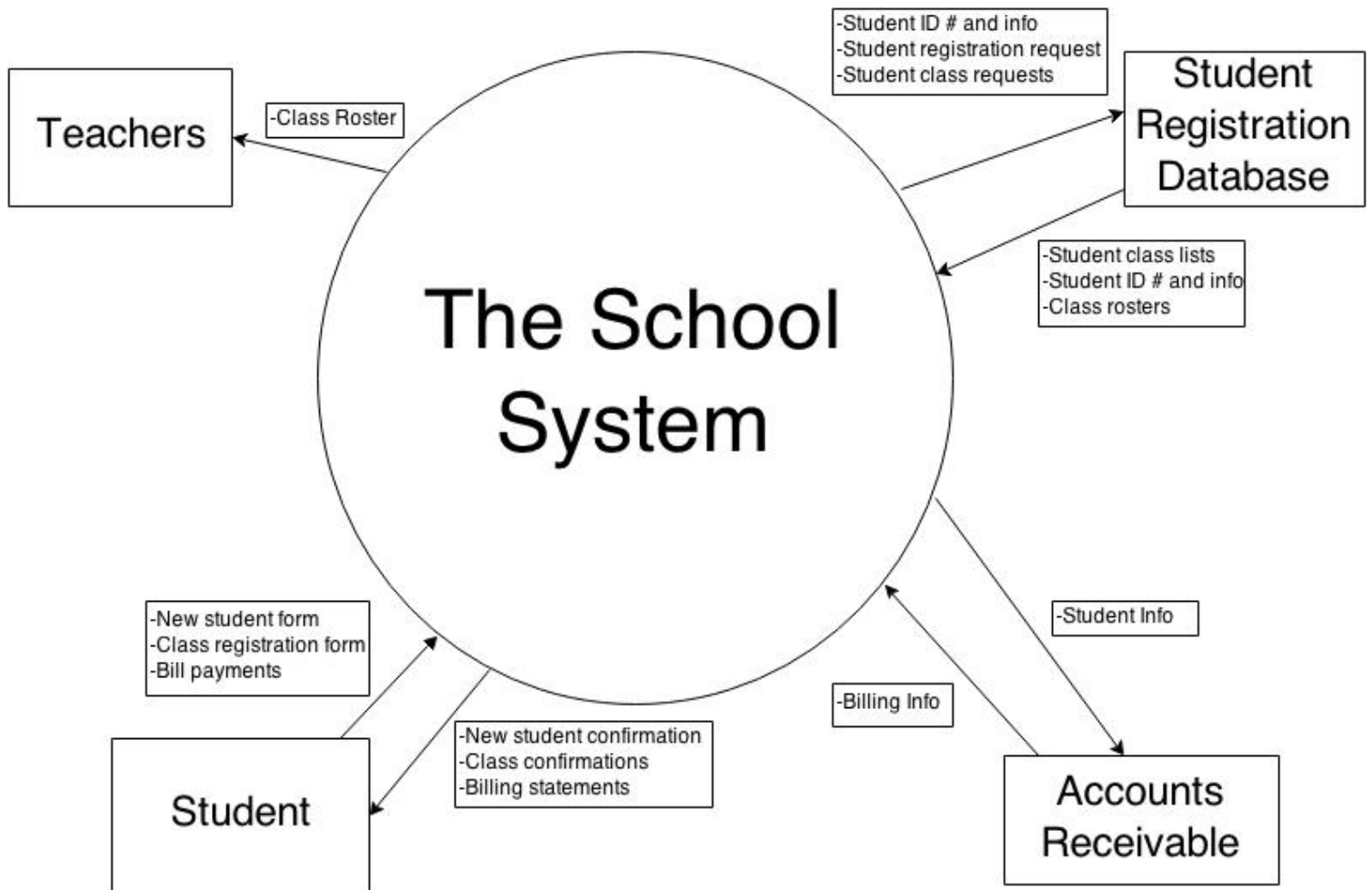


Figure 1: DFD Context Diagram

3 Level 0 Diagram

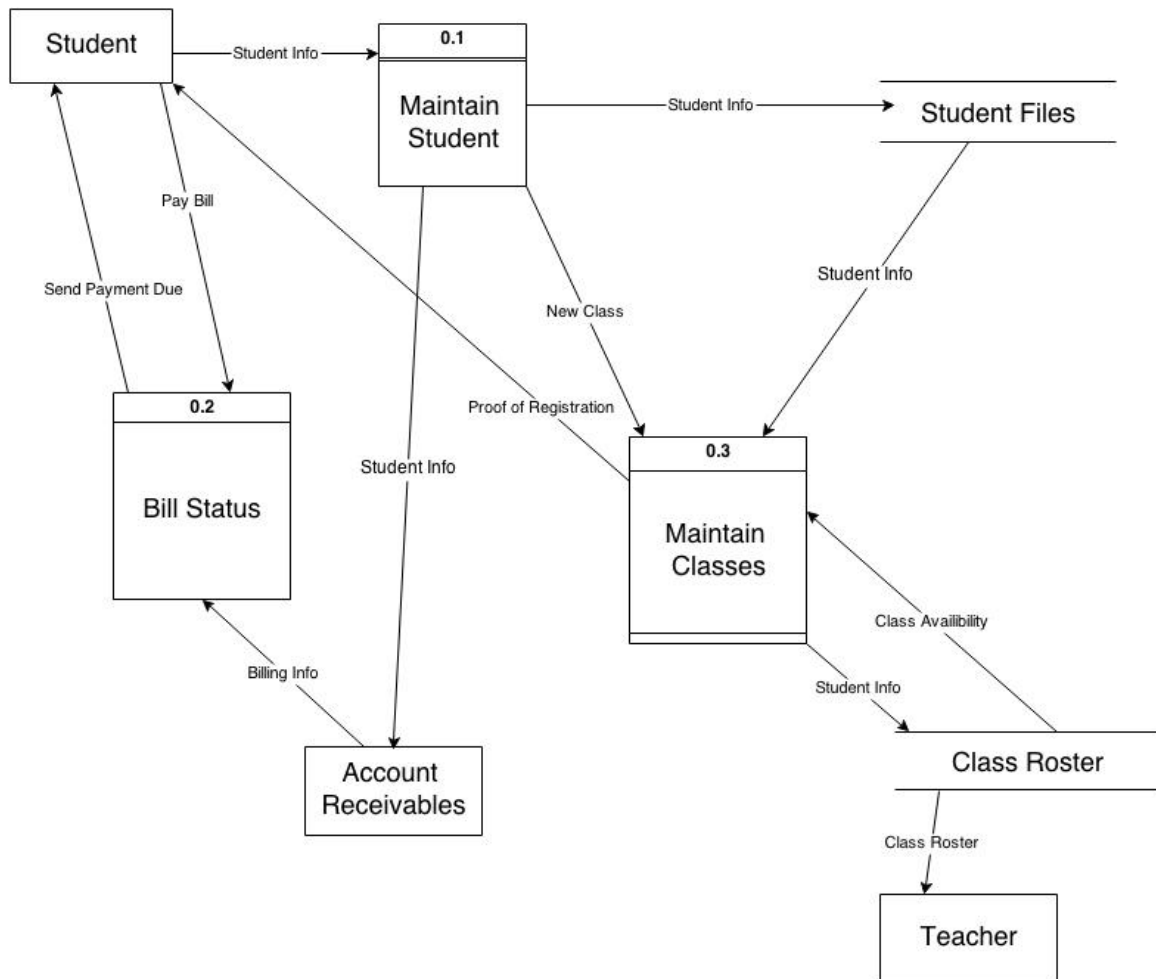


Figure 2: DFD Level 0 Diagram

4 Functional Requirements

4.1 Allow Student to Draw a Diagram

Primary Actor: Student

Precondition: Student is logged in and viewing a question

Trigger: Student chooses a question to work on

Success end condition: Student can submit answer

Failed end condition: Diagram is not drawn

Steps:

1. Empty space available to draw diagram
2. Student picks a diagram type (Chen's/Crow's Foot)
3. Student picks shapes from toolbox and places in diagram
4. Student places text inside shapes
5. Student places links between shapes

Exceptions:

1. Teacher draws diagram that students must edit
*Space will not be empty, student will be able to make changes to present diagram
3. Student wants to remove/change something previously done
*Nothing in the diagram is permanent, anything can be removed or edited.

4.2 Submit Answer

Primary Actor: Student

Precondition: Student is ready to submit answer

Trigger: Pressing the submit button

Success end condition: Feedback and/or grade is given

Failed end condition: Nothing is given back to student

Steps:

1. Student submits answer by pressing the submit button
2. System processes answer and decides validity
3. Outputs feedback and/or grade for student to see
4. Answer is saved into database

Exceptions:

1. Student leaves area blank
*Answer is just marked as wrong
3. Teacher never properly input the correct answer or feedback
*Student is told that there's nothing to report

4.3 Allow Teacher to Create and Upload Questions/Feedback

Primary Actor: Teacher

Precondition: Teacher is logged in a specific course.

Trigger: Teacher selects a specific Homework to create a question.

Success end condition: Question is uploaded and students are able to see it.

Failed end condition: Students don't see the question properly.

Steps:

1. Teacher clicks on a "create" button to create the question.
2. Teacher select the number of the question that he/she wants to post for a previously chosen homework.
3. Teacher writes the database schema of the ER diagram that is going to be drawn by the student, in an available textbox.
4. Teacher provides a general feedback in a textbox provided.
5. Teacher clicks on a "add" button to add the question of the homework.
6. Teacher clicks on a "preview" button to see the students point of view of the entire homework.
7. Teacher clicks on a "submit" button to submit the entire homework.

Exceptions:

1. Teacher provides the ER diagram that the student is to write the schema for.
2. Teacher provides either Crow's Foot or Chen diagram and the student has to convert it into the opposite diagram.

4.4 Question Browsing

Primary Actor: Student

Precondition: Student is logged in and is in a quiz

Trigger: The student attempts to select a new question to attempt to answer

Success End Condition: Question that the student selects successfully loads

Failed End Condition: The question that is selected does not load

Steps:

1. Student selects the question they would like to answer from a list on the top of the page.
2. Webpage for the selected question loads
3. Student can read and answer question

Exceptions: None

4.5 Add User

Primary Actor: Teacher

Precondition: Teacher has student information

Trigger: Teacher inputs student's name and ID

Success end condition: Student can view new/taken quizzes

Failed end condition: Student is not in the system

Steps:

1. Student info added to database
2. Student logs in with information
3. Student's quizzes are retrieved from database
4. Student can browse through previous and new quizzes

Exceptions:

4. Teacher can block upcoming quizzes, only make them available at specific date/time
*Student will be able to view the quiz once the date and time are reached

5 Environmental Requirements

5.1 ToolBox for Drawing Diagram

A ToolBox will help the student draw the diagram. The student will first select whether he wants to draw a Chen's Diagram or a Crow's Foot Diagram. The toolbox will then adjust accordingly, displaying the shapes and edges that are common to the chosen diagram. This will include squares, circles, directed edges, and tick marks (for Crow's Foot).

5.2 Save Progress of Current Assignment

When a student logs in, he/she should be able to start a quiz. As the student progresses through the quiz, the answers answered by the student should be saved in the students personal database. In that way, if he/she wants to stop and continue with the quiz other time, he can do so without starting all over again.

5.3 List of Available Questions and Their Status

While a student is answering questions, there will be a list of all the questions listed horizontally at the top of the page. Next to each question number will be a small image. The image will either be a green check mark if the question is answered correctly, a red x if the question is answered incorrectly or a black question mark if the question has not been answered yet. When the status of the question changes then so will the image next to the question number.

5.4 Open Space to Draw Diagram

After a student selects which question they want to work on a white drawing space will be given to them in which to draw their answer. Off to the side of this space will be the toolbox which they use to do the drawing. There will also be a submit button off to the side somewhere which is what the student is to press when they're finished with their answer.

5.5 Compatibility With Multiple Browsers

The program will open and be usable with all major internet browsers. Including Safari, Chrome, Firefox, Internet Explorer.

6 Performance Requirements

6.1 Correctness Analysis

The ER diagram drawn by the student generates a database schema that will be compared with schema provided by the professor in the question. If they are equal, then the solution provided by the student is correct, otherwise is wrong. This should be done in a matter of seconds.

6.2 Question Loading

When the student selects a new question to attempt to answer the question should load within 5 seconds when the server is not busy and should load within 10 seconds when the server is busy.

7 Safety/Security Requirements

7.1 Student Navigates to the Main Page

Description: The student reaches the main page of an assignment.

Primary Actor: Student

Secondary Actor: Content author

Steps:

1. The page loads, displaying the correct questions for the assignment selected.
2. All questions load and display properly.
3. The student response areas load and allow for proper student input.

Successful Post Conditions:

- The page displayed is for the correct class and assignment.
- The student gains no access to non-student interfaces.
- The submission of answers writes only to the students record for the assignment.

Exceptional Condition:

- The student attempts to access a page he/she should not have access to - the student is displayed an appropriate warning message and is not given access to any assignment or interface that he/she is not meant to have access to.

7.2 Professor Navigates to the Main Page

Description: The professor reaches the main page of an assignment.

Primary Actor: Professor

Secondary Actor: Content author

Steps:

1. The page loads, displaying the tools for adding or modifying an assignment and the information corresponding to student-submitted scores.
2. All questions load and display properly.
3. The content-create pallet allows for the full range of content creation.
4. The data corresponding to student submissions accurately reflects the data currently stored for each student.

Successful Post Conditions:

- The page that is displayed shows the correct class and assignment.
- The professor is shown the assignment creation and management page.
- The content changes that are made only apply to the currently selected assignment.

Exceptional Condition:

- The professor attempts to access a page he/she should not have access to - the professor is displayed an appropriate warning message and is not given access to any assignment or interface he/she is not meant to have access to.

7.3 Student Closes Application With Unsaved Answers in the Window

Description: The student has entered a response to a question without saving or submitting the assignment.

Primary Actor: Student

Secondary Actor: Professor

Steps:

1. The student enters a full or partial response to an assignment question.
2. The student does not save or submit the assignment.
3. The student closes the assignment page.

Successful Post Condition:

- The student is shown a prompt asking if he/she is sure about wanting to leave the page with unsaved responses.

7.4 Student Submits Page with Blank Answers

Description: The student submits an assignment with a response that was left blank.

Primary Actor: Student

Secondary Actor: Professor

Steps:

1. The student accesses the page.
2. The student inputs and saves responses to some questions.
3. The student submits the assignment with some responses that are left entirely blank.

Successful Post Condition:

- The student is presented with a prompt asking he/she is sure about wanting to submit a blank response to a question.

7.5 Student Submits Assignment That Was Already Submitted

Description: After a student submits an assignment, he/she tries to submit the same assignment again.

Primary Actor: Student

Secondary Actor: Professor

Steps:

1. The student completes an assignment.

2. The student completes the same assignment with different answers.
3. The student submits the assignment again.

Successful Post Condition:

- The student is presented with a prompt asking if the/she wants to overwrite the submitted assignment - assuming this occurs before the assignment is closed.

Exceptional Condition:

- If the submission comes after the assignment is closed, the student is informed that the assignment is closed and no subsequent submissions are allowed.

7.6 Student or Professor Tries to Log Into Someone Else's Account

Description: The student or professor tries to log into the system with the NetID of another student or professor.

Primary Actors: Student, Professor

Secondary Actor: System

Steps:

1. The student or professor enters someone else's NetID into the system.
2. The student or professor enters a fake password into the system.
3. The student or professor clicks the "Login" button or the enter key.

Successful Post Conditions:

- The system prohibits the student or professor from logging into the system with a NetID and password that do not match.
- The system sends a reply that says that the login was unsuccessful.
- The system prompts the user to try entering a username and password again.

8 Robustness Use Cases

8.1 Student Enters Diagram

Description: The student submits a diagram in either Chen or CF that is correct but is not the same as the instructor's.

Primary Actors: Student, System

Secondary Actor: Professor

Steps:

1. The student enters a diagram into the system.
2. The system checks the diagram against the professor-submitted diagram(s).

Successful Post Condition: The system marks the student correct.

Exceptional Conditions:

- The student is unable to submit a diagram as an answer to a question.
- The system cannot process the student's submitted diagram so that it can be compared to the professor's uploaded answer.
- The system cannot send the student a response that says if his/her diagram was correct or incorrect.

8.2 Student Enters Variant of Incorrect Diagram

Description: The student submits a diagram in either Chen or CF that is incorrect.

Primary Actors: Student, System Secondary Actor: Professor

Steps:

1. The student enters a diagram into the system.
2. The system checks the diagram against the professor-submitted diagram(s).

Successful Post Condition:

- The system marks the student incorrect.

Exceptional Conditions:

- The student is unable to submit a diagram as an answer to a question.
- The system cannot process the student's submitted diagram so that it can be compared to the professor's uploaded answer.
- The system cannot send the student a response that says if his/her diagram was correct or incorrect.

8.3 Professor or Content

Author enters unspecified answer format

Description: The professor or content author enters an answer into the system without specifying a format.

Primary Actor: Professor (or Content Author)

Secondary Actor: System

Steps:

1. The professor submits an answer into the system.
2. The system recognizes that the professor did not submit an answer format.

Successful Post Condition:

- The system sends a prompt to the professor specifying that the answer was not recorded in the system and that he/she must submit an answer format.

8.4 System Should Never Crash

Description: Nothing the student does should crash the browser. The system may not respond or react correctly, but it will not crash the students browser or machine.

8.5 UI Should React Correctly To Student and Professor Actions

Description: The UI should correctly react to what the student and professor's actions command.

Primary Actor: Student, Professor

Secondary Actor: System

Steps:

1. The student or professor clicks buttons/links or drags items from a template to the answer submission.
2. The system follows the links or correctly places the dragged items in the submission box.

Successful Post Condition:

- The system view refreshes so that the student or professor sees his/her actions take place in real time.

Exceptional Condition:

- A specific student or professor action was not accounted for when the UI was created - the UI does not have the ability to make a response to that action.

9 Accuracy Requirements

9.1 Student Submits Correct Diagram

Description: The student answers a question by submitting the correct diagram.

Primary Actors: Student, System

Secondary Actor: Professor

Steps:

1. The student enters a diagram as an answer to a question.

2. The system compares that diagram to the correct diagram that was uploaded by the professor.
3. The system recognizes that the diagram submitted matches the diagram that was uploaded by the professor.

Successful Post Condition:

- The system marks the student's answer to the question as correct.

Exceptional Conditions:

- The student is unable to submit a diagram as an answer to a question.
- The system cannot process the student's submitted diagram so that it can be compared to the professor's uploaded answer.
- The system cannot send the student a response that says if his/her diagram was correct or incorrect.

9.2 Student Submits Incorrect Diagram

Description: The student answers a question by submitting an incorrect diagram.

Primary Actors: Student, System

Secondary Actor: Professor

Steps:

1. The student enters a diagram as an answer to a question.
2. The system compares that diagram to the correct diagram that was uploaded by the professor.
3. The system recognizes that the diagram submitted matches the diagram that was uploaded by the professor.

Successful Post Condition:

- The system marks the student's answer to the question as incorrect.

Exceptional Conditions:

- The student is unable to submit a diagram as an answer to a question.
- The system cannot process the student's submitted diagram so that it can be compared to the professor's uploaded answer.
- The system cannot send the student a response that says if his/her diagram was correct or incorrect.

10 Login/Logout System Use Cases

10.1 Student or professor correctly logs into the system

Description: The student or professor logs into the system with the correct username and password.

Primary Actors: Student, Professor

Secondary Actor: System

Steps:

1. The student or professor enters his/her correct NetID as a username.
2. The student or professor enters his/her correct password.
3. The student or professor clicks the “Login” button or the enter key.

Successful Post Condition:

- The student or professor is successfully logged into his/her account on the system and can see his/her homepage on the screen.

Exceptional Condition:

- The system does not recognize that the correctly entered username and password are associated with an existing account.

10.2 Student or professor incorrectly logs into the system

Description: The student or professor logs into the system with an incorrect username and/or password.

Primary Actors: Student, Professor

Secondary Actor: System

Steps:

1. The student or professor enters an incorrect NetID as a username. and/or
2. The student or professor enters an incorrect password.
3. The student or professor clicks the “Login” button or the enter key.

Successful Post Conditions:

- The system prohibits the student or professor from logging into the system with a NetID and password that do not match.
- The system sends a reply that says that the login was unsuccessful.
- The system prompts the user to try entering a username and password again.

10.3 Student or professor correctly logs out of the system

Description: The student or professor is able to successfully log out of the system.

Primary Actors: Student, Professor

Secondary Actor: System

Steps:

- The student or professor clicks the “Log Out” button.
- Successful Post Condition:
- The student or professor receives a reply from the system that says that the log out was successful.

Exceptional Condition:

- The “Logout” button does not have the functionality to successfully log out the student or professor.

11 Professor UI Requirements

11.1 Professor Clicks on Side or Top Navigation Buttons

Description: While on any page containing navigation links in the side or top bars, the professor clicks on a nav link

Primary Actors: Professor

Steps:

1. The professor is logged into any of the main pages containing nav links
2. The professor clicks on any link

Successful Post Conditions:

- The professor is directed to the page intended by the link

Exceptional Condition:

- The professor is already on the page of the link they are clicking - nothing happens and the page does not change.

11.2 Professor Clicks on the Assignment Submission Ticker

Description: From the main landing page the professor clicks on the assignment submission ticker

Primary Actor: Professor

Steps:

1. The professor is on the main page after login

2. The professor clicks on the Assignment Submission Ticker Pane

Successful Post Condition:

- The professor is redirected to a larger detailed view with a time directed log of student submission stats for assignments the professor has created, or is linked to through class ownership

Exceptional Post Condition:

- The professor has no submissions in his feed - he is directed to the log but is not shown any information

11.3 Professor Clicks on the Class or Student Ticker

Description: From the main landing page the professor clicks on enrolled student/ class ticker pane

Primary Actor: Professor

Steps:

1. The professor is on the main page after login
2. The professor clicks on the enrolled student/ class ticker pane

Successful Post Condition:

- The professor is redirected to a larger detailed view with the option to sort by class or student name to display stats pertaining to course enrollment and individual student assignment histories

11.4 Professor Clicks to Expand the Assignment Creation Sandbox

Primary Actor: Professor

Steps:

1. The professor is on the main page after login
2. The professor clicks to expand the assignment creation sandbox

Successful Post Condition:

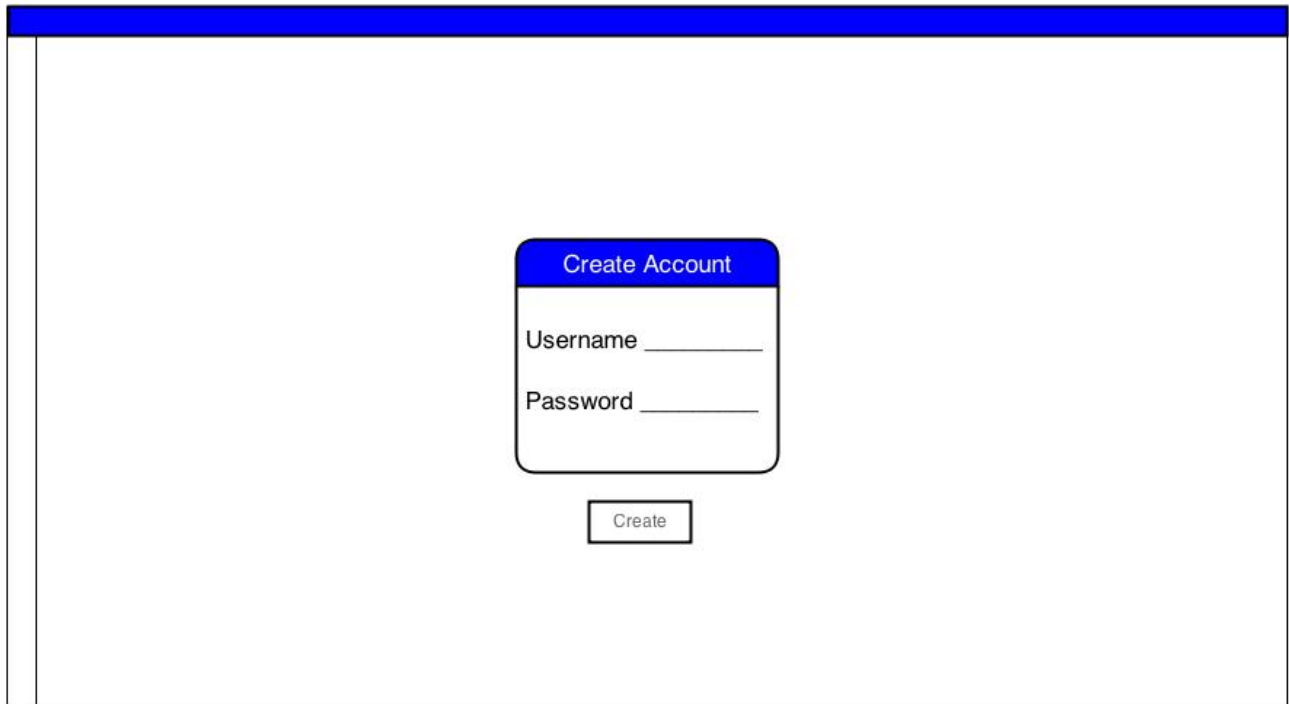
- The professor is taken to the assignment creation page and any potential diagrams or questions created in the sandbox are stored and transferred to the assignment creation window

11.5 Professor View

	Navigation	Navigation	Navigation	Navigation	Navigation	Navigation
Account Administration	Assignment Submission Ticker					
Class Listing						
Assignment Listing	Assignment Creation Pallet		Enrolled Student Ticker		Class Ticker	
Student Directory						
Create New Assignment						
Modify Assignment						
Logout						

Figure 3: Professor's User Interface

11.6 Create



The image shows a user interface for creating an account. It features a blue header bar at the top. Below the header, there is a central form titled "Create Account" in a blue box. The form contains two input fields: "Username" and "Password", each followed by a text entry line. Below the form is a "Create" button.

Create Account

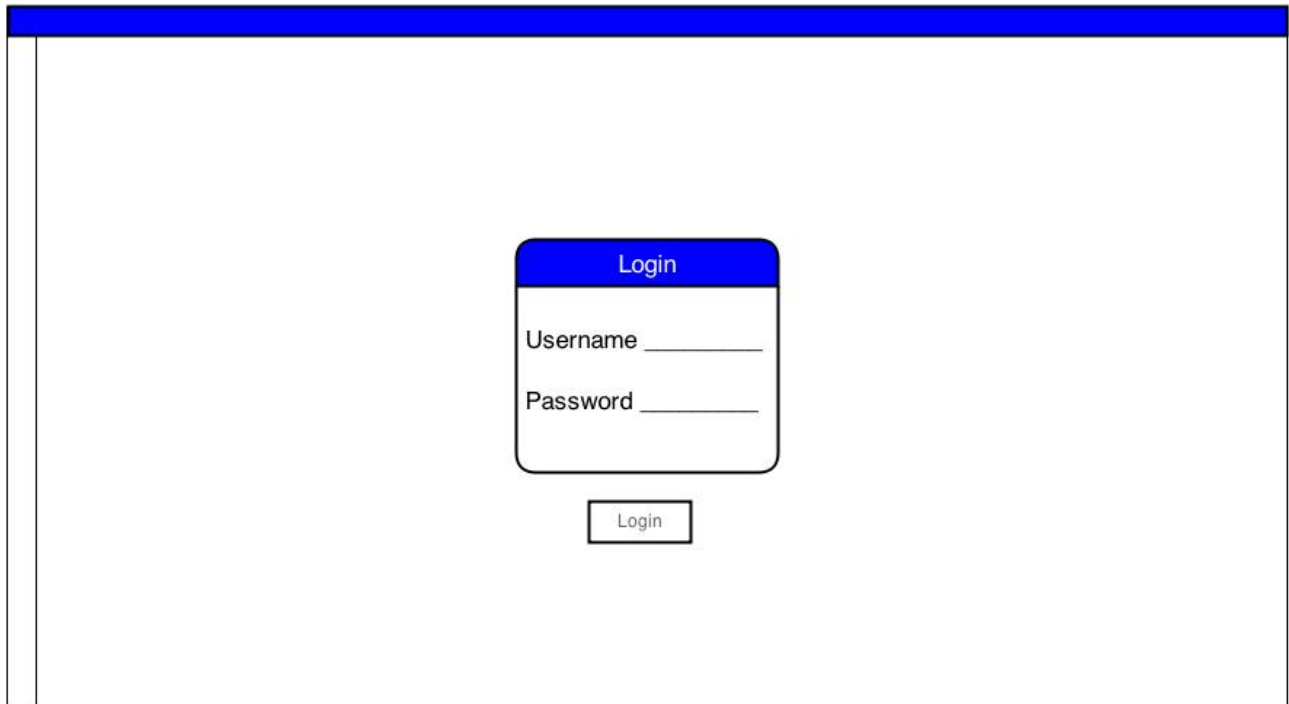
Username _____

Password _____

Create

Figure 4: User Interface to create question

11.7 Login



The diagram illustrates a login user interface within a window. The window has a blue title bar at the top. Inside the window, there is a central login form. The form has a blue header with the text "Login". Below the header, there are two input fields: "Username" and "Password", each followed by a horizontal line for text entry. Below these fields is a rectangular button labeled "Login".

Figure 5: Login User Interface

11.8 Student

Questions

Question 1

Question 2

Question 3

Question 4

4.a

4.b

4.c

Question 5

Logout

Question 1: Use the tools provided to construct a Chen style Diagram of the situation outlined in Chapter 1 Section 3a.

Submit

General

Text & Images

Connection

Advanced

Images

Figure 6: User Interface for the student