Project 1 –Extended Instructions

Objective:

-We have provided sample data which we expect to see in the database. You can embellish it accordingly to fit your database schema.

-You can test your design using your own data. But at the time of evaluation we expect to have only the sample data provided by us in your database.

-Section one gives a flow that your menu-driven program should follow.

-Section two gives the sample data.

Table of contents

1. Program Flow Table

The menu driven program should be able to do the following.

Note: The following describes only some of the actions. It is expected that your program provides the other functionalities too.

1. Start:

Step	Input Display	User Input	Processing & Output
1.	1. Login 2. Create User 3. Exit	Enter Option	-validate choice and print "invalid choice" and go back if necessary

2. Login

Step	Input display	User Input	Processing & Output
2.1	Enter Uid followed by password	Enter uid and password	-Recognize if user Is a student, professor, TA -Print "Login Incorrect" and go back

3. Logged in as Student (general example)

Step	Input Display	User Input	Processing & Output
3.1	1. Select Course 2. Add Course 3. Back	Enter choice	-validate choice and print "invalid choice" and go back if necessary
3.2	Select Course: 1. CSC440-Database Management Systems 2. Back	Enter choice	-validate choice and print "invalid choice" and go back if necessary
3.3	Add course: Enter Course token	Enter token	-Validate course token. -If token belongs to a course after due date display: "Course Over, Cannot Register" -If, invalid id, then print "Invalid ID" and go back -Else update tables and print "enrolled" and print course options

3a. Course options for student (CSC440)

Step	Input Display	User Input	Processing & Output
3a.0	Course options: 1. View Scores 2. Attempt Homework 3. View Past Submission	Enter choice	-display options -Print "Invalid Option" and go back -Go back login

	4. Back		
3a.1	View Scores: Homework_number | Attempt_number 1. HW1 First attempt 2. HW1 Second attempt 3. HW2 First attempt 4. Back	Enter choice	-Display score -Wait for next input (enter '4' to go back)
3a.2	Attempt Homework: -Display list of open home works i.e. 1. HW 2	Enter choice	-If number of attempts for selected homework are exhausted-print "No more attempts" and go back -If not, display questions & options
3a.2.1	HW1 questions: Q1. …. a) …. b) …. c) …. d) …. Your answer?	Enter choice	-check whether answer entered is correct or incorrect, and record points. Display next question -print "Invalid Option" and display Q1 again.
3a.2.2	Q2. …… a) …. b) … c) … d) … Your answer?	Enter choice	-check whether answer entered is correct or incorrect, and record points. Display "homework completed." -print "Invalid Option" and display Q2 again.
3a.2.3	1. Back	Enter choice	Go back
3a.4	View Past Submission -Display homework attempts which are past and within due date Homework past Due Date: 1. HW1 First attempt 2. HW1 Second attempt Homework within due date: 3. HW2 First attempt 4. Back	Enter choice	-Process choice and display options accordingly -Print "Invalid option" and display options again -Go back
3a.4.1	HW1 1: *(For every question)* Q1. … a) … b) … c) … d) … Your answer: … Correct: Yes/No Explanation: … 1. Back	Enter choice	-Go back
3a.4.2	HW2 1: *(For every question)* Q1. … a) …	Enter choice	-Go back

	b) … c) … d) … Your Answer: .. Correct : Yes/No Hint: … 1. Back		

4. Logged in as Professor (login as Ms. Kemafor Ogan)

Step	Input Display	User Input	Processing & Output
4.1	4. Select Course 5. Add Course 6. Back	Enter choice	-validate choice and print "invalid choice" and go back if necessary
4.2	Select Course: 3. CSC440-Database Management Systems 4. Back	Enter choice	-validate choice and print "invalid choice" and go back if necessary

4a. Course options for professor (CSC440)

4a.0	For CSC440: 1. Add homework 2. Edit Homework 3. Add question 4. Add answer 5. Reports 6. Back	Enter choice	-validate choice and print "invalid choice" and go back if necessary
4a.1	Add Homework: One by one take input for: -start date -end date -number of attempts -score selection scheme -question numbers -Correct answer points -Incorrect answer points	Enter the specified info separated by enter	-Take in each input and update the database
4a.2	Edit Homework: Display the list of homeworks 1. Hw1 First attempt 2. Hw1 Second attempt 3. Hw2 First attempt 4. back	Enter choice	-validate choice and print "invalid choice" and go back if necessary
4a.2.1	Edit HW2 1: Choose what to update: 1. Start date 2. End date 3. Number of attempts 4. Score selection 5. Question numbers	Enter choice	-validate choice and print "invalid choice" and go back if necessary

	6. Correct answer points 7. Incorrect answer points 8. Back		
4a.2.2	Eg, End date Display "enter new end date"	Enter date	-Validate date and update tables and go back. -If not validated, print error message and bo back
4a.3	Add question: Display a list of sections: 1. Database Fundamentals 2. ER Design 3. Security and Authorization 4. Back	Enter choice	-validate choice and print "invalid choice" and go back if necessary
4a.3.1	Enter question, correct answer , 1st incorrect answer, 2nd incorrect answer, 3rd incorrect answer separated by newline	Enter the specified information	-Validate that all 5 types of information are entered and update tables. Go back.
4a.4	Add answer: Select question: 1. Q1 ….. 2. Q2 ….. 3. Q3 ….. 4. Back	Enter choice	-validate choice and print "invalid choice" and go back if necessary
4a.4.1	Eg, Q1: Enter answer type 1. Correct 2. Incorrect	Enter choice	-validate choice and print "invalid choice" and go back if necessary
4a.4.2	Print " Enter answer"	Enter Answer	-validate answer and update tables.
4a.5	Reports: Either provide user with options to select to the type of reporting query or prompt the user to enter the entire query.	Select options/ Enter query	Display query results

2. Sample Data

Student data:

Id	Password
Ssbudha	123bud
Sskanit	123kan
Agholak	123hol
Rjoseph	123jos
Tbirajd	123bir

Faculty data:

Id	Password
Kogan	123kogan

Score Selection:

- first attempt
- last attempt
- maximum of all attempts
- average of all attempts

Course data:

- Course id :- CSC440
- Token: CSC440SPR13
- Course name: Database Systems
- Course start date: 1st January 2013
- Course end date: 10th May 2013
- Professor: Kemafor Ogan
- Teaching Assistant: Aishwarya Neelakantan
- Topics: 1) Database Fundamentals, 2) Security and Authorization 3) ER Design and other topics
- Students: ssbudha, sskanit, agholak,rjoseph

- Course id :- CSC541
- Token : CSC541FLL11
- Course name: Advanced Data Structures
- Course start date: 1st Augutst 2011 the course
- Course end date: 15th December 2011
- Professor: Rada Chirkova
- Teaching Assistant: Jitendra Harlalka
- Topics: 1) Binary search trees and Btrees, 2) Hashing 3) Files and indexing and other topics
- Set of students
- **Note that the course is over its due date

- Course id :- CSC501
- Token: CSC501SPR12
- Course name: Operating Systems
- Course start date: 1st January 2012
- Course end date: 10th May 2012
- Professor: Dr. R. Mueller
- Teaching Assistant: Pamela Hart
- Topics: 1) Processes and Threads 2)Memory Organization 3) Deadlocks and other topics
- Set of students: agholak

Assessment Data:

- Homework 1
- based on Database Fundamentals
- 2 retries
- start date: 12th February 2013
- end date: 1st March 2013
- number of points per correct question: 3
- number of points deducted per question: 1
- Score selection: first attempt
- ssbudha did not attempted the homework, sskanit attempted twice, agholak attempted once.

- Homework 2
- based on ER design
- 1 retries
- start date: 12th March 2013
- end date: 1st April 2013
- number of points per correct question: 5
- number of points deducted per question: 2
- score selection: average
- questions: Q1, Q2
- sscudha has attempted, sskanit and agholak are yet to attempt.

Question and Answer data:

id	**Content**
Q1	Question 1
Ans1	Correct 1.1
Ans2	Correct 1.2
Ans3	Correct 1.3
Ans4	Incorrect 1.1
Ans5	Incorrect 1.2
Ans6	Incorrect 1.3

Ans7	Incorrect 1.4
Ans8	Incorrect 1.5
Ans9	Incorrect 1.6
Ans10	Incorrect 1.7
Hint1	Hint 1.1
Hint2	Hint 1.2
Hint3	Hint 1.3
Exp1	Explanation 1.1
Exp2	Explanation 1.2
Exp3	Explanation 1.3

id	**Content**
Q2	Question 2
Ans11	Correct 2.1
Ans12	Correct 2.2
Ans13	Correct 2.3
Ans14	Incorrect 2.1
Ans15	Incorrect 2.2
Ans16	Incorrect 2.3
Ans17	Incorrect 2.4
Ans18	Incorrect 2.5
Ans19	Incorrect 2.6
Ans20	Incorrect 2.7
Hint4	Hint 2.1
Hint5	Hint 2.2
Hint6	Hint 2.3
Exp4	Explanation 2.1
Exp5	Explanation 2.2
Exp6	Explanation 2.3

id	**Content**
Q3	Question 3
Ans21	Correct 3.1
Ans22	Correct 3.2
Ans23	Correct 3.3
Ans24	Incorrect 3.1
Ans25	Incorrect 3.2
Ans26	Incorrect 3.3
Ans27	Incorrect 3.4
Ans28	Incorrect 3.5
Ans29	Incorrect 3.6
Ans30	Incorrect 3.7
Hint7	Hint 3.1
Hint8	Hint 3.2
Hint9	Hint 3.3
Exp7	Explanation 3.1
Exp8	Explanation 3.2
Exp9	Explanation 3.3

3. Submission:

The following files should be submitted as a zip file named unityid1-unityid2-540S13-1 by 11.59pm eastern time 13th March, 2013.

- The source code of your program.
- ER diagram.
- Readme text which may describe the constraints not captured by your program.
- A text file of create table statements capturing the E-R diagram in SQL and a short paragraph describing what constraints are captured and how e.g. using CHECK constraints or triggers or privileges etc.
- A text file including the table population statements that you used i.e. INSERT statements to add data to your tables.
- A text file including the sample SQL queries to the four query descriptions that will be provided to you.
- You will be presenting a project demo after the submission. The date for the demo will be announced later. You should pre populate your database with the given sample data.
- We will run a few queries based on the sample queries, test homework generation and also test if you have captured other constraints.

4. Grading Scheme:

E-R model – 25 pts
Create table (SQL) statements –25 pts
SQL queries – 30 pts
Application specifications – 20 pts
Scenarios tested in Demo – 25 pts
*Bonus pts – 20 pts

5. Appendix

The following gives a brief description of the table in words.

-Login or create user. Validate Login. Need to recognize whether the logged in entity is a student/professor/TA. Anybody can create a user account.

-**Student:** Display the student activities: Select Course, Enroll for course. Activities for each course: View score, attempt homework.

-**View score**: Display list of home works. Select a home work. For the selected homework list the attempt and the score on that homework. Also display the final score. **Attempt Homework**: Display a list of all home works which are valid in the given period. Select homework. Check whether the number of attempts for selected homework has been exhausted. If not, display each question along with options. Record the answer for that question.

You will be evaluated on the randomization of questions between homework attempts of the same user and different users for the same homework.

-**Professor:** Display activities: Select Course. For each course, Add homework, Edit homework, Add question, Add answer.

-**Add Homework** should take in details about the time period, no of attempts, score selected scheme, question numbers etc. **Edit Homework** should allow editing the above details. **Add Question** should ensure that every question is associated with a valid topic of the course. Also every new question that is added should be followed by 3 incorrect and 1 correct options. **Add Answer** should associate every additional answer with a valid question and topic in the course.