STARS Web App

CMPT 561 - Homework 1

This is an individual assignment worth 5%. Due date is by 11pm Saturday 24th October 2015~~5pm Wend 22nd October 2015~~. Push your work to Github as you make progress.

1. Requirements

You are required to design and develop Web forms for a Support & Tracking of At-Risk Students (**STARS**).

The aim of STARS is to manage follow-up actions to help students who are at-risk and improve their success rates.

STARS allows the user to keep track of the actions and activities done the course Instructors, the Program Coordinators and the Academic Advisers. In homework 1 you are required to design and produce the html pages for STARS User Interface (UI).

Key entities of this system are:

Student	StudentId, FirstName, LastName, Gender, ProgramId, GPA, AtRiskCategory, Semester (e.g., Fall 2015) Program could be (ProgramId is the value between parentheses): • Architecture and Urban Planning (AUP) • Chemical Engineering (CHE) • Civil Engineering (CVE) • Computer Science (CS) • Computer Engineering (CE) • Electrical Engineering (EE) • Mechanical Engineering (ME) • Industrial and Systems Engineering (ISE) AtRiskCategory could be: • Probation 1 • Probation 2 • Probation 3 - Final Probation • Reinstated • Failing a Course Twice • Warning, Last Term GPA < 2.0
Course	CRN, CourseCode, Name, Semester, InstructorId
StudentCourse	StudentId, CRN
FollowupActions	ActionId, Date, Semester, ActionType, Title, Description, ByWhom, CourseCRN, Attachments - ActionType: Meeting, Referral, Email, Letter - CourseCRN is only applicable if the action is done by the instructor - ByWhom is set the AdviserId who entered the action - Attachments: a user can attach one or more attachments
FollowupActionStudent	ActionId, StudentId

Adviser	AdviserId, LoginName, Firstname, Lastname, Type, Department~~,~~ ~~Program~~ The Adviser Type could be: Instructor, Coordinator, ~~CENG~~ College Adviser
AdviserProgram	AdviserId, ProgramId

Assume that the data related to all entities (except FollowupActions~~follow-up actions~~) will be provided to you ~~from~~ by the Student Retention department.

You will focus on 3 use cases:

- **List of students at risk** under the care of the current user. Upon login,
  - The Coordinator and the ~~CENG~~ College Adviser by default get all the students belonging to the~~ir~~ programs they manage. But they can filter the list of students by program or they can return to the default view.
  - The Instructor by default gets the list of students in the courses that the instructor is teaching in the current semester. But they can filter the list of students by course or they can return to the default view. Note that a student may appear multiple times in the list if they are taking multiple courses with the instructor. Hence, you should include the CourseCode when displaying the list to the instructor.

  From this list, the user can either add a follow-up action or view the list of existing actions. The user can add a follow-up action either for a particular student or for list of selected students.
- **Add Follow-up Action** to allow the user to select a Student or a list of students ~~from the list above~~, enter the Action date (by default this should be set to Today's date) and ~~an~~ other Action details as described in the table above. When the action is entered by the instructor, the action CourseCRN should be set the CourseCRN of the selected student.
- **List Actions** to display actions for a particular student by default done by the current user. The user can request getting actions ~~or~~ done by a particular adviser type or get all the actions. The user can then get the action details and attachments. They can also add an action.

  From this view, the user can navigate backward and forward through the list of students to get their follow-up actions or they can get the follow-up actions for a particular student without leaving this view.

The home page of the STARS should display the list of students at risk under the care of the current user.

STARS web pages should use HTML 5 and CCS. The pages should be connected and have **dummy data** and behave like a complete application (without server side processing). The pages should comply with Web user interface design best practices. Also remember that 'there is elegance in simplicity'.

2. Grading rubric

Criteria	%	Rubric Below will be used
Complete and correct design and implementation of the requirements:		
• Home page with menu	12	

• Students List	20	
• Add Action	20	
• Actions List	25	
• View Action	10	
• Navigation between pages	5	
Testing Word documentation with evidence of correct implementation using snapshots illustrating the results of testing.	8	
Total	100	
Copying and/or plagiarism or not being able to explain or answer questions about the implementation	-100%	

Detailed rubric that will be used to evaluate your website and your pages

Criteria	Exceeds Expectations (1)	Meets Expectations (0.8)	Needs Improvement (0.6)	Unacceptable (0.5 to 0)
Navigation	Navigation buttons/links easy to locate and follow. Logical and intuitive sequence	Navigation buttons/links somewhat confusing to use or locate. Logical sequence.	Buttons/links confusing to use or locate. Confusing sequence.	Buttons/links missing from page. No sequence.
Principles of Design	All principles applied consistently throughout web site.	Principles of design applied in most, but not all of the web site.	Limited application of design principles.	Use of design principles not evident.
Professional Look and Feel	The Web site has an exceptionally attractive, eye-catching and usable layout. It is easy to locate all important elements. White space, graphic elements and/or alignment are used effectively to organize material.	The Web pages have an attractive and usable layout. It is easy to locate all important elements.	The Web pages have a usable layout, but may appear busy or boring. It is easy to locate most of the important elements.	Novice: Design and development effort is inconsistent. The Web pages are cluttered looking or confusing. It is often difficult to locate important elements. Design is dull and has sloppy appearance.
Use of Cascading Style Sheets	Evidence of advanced use of styles sheet design. CSS utilized throughout web site.	Good use of CSS utilized in most of the web site.	Minimal use of CSS.	No evidence of CSS.
Quality of the implementation	Clean HTML and CSS using appropriate features.	Bit complex HTML and CSS and few inappropriate features used.	Unnecessary complex HTML and CSS and some inappropriate features used.	Too complex HTML and CSS. Many inappropriate features used.

Spelling and Grammar	There are no errors in spelling, punctuation or grammar.	There are 1-3 errors in spelling, punctuation or grammar.	There are 4-5 errors in spelling, punctuation or grammar.	There are more than 5 errors in spelling, punctuation or grammar.

Ground Rules

- All assignments **must be your own original work**, not based on the work of other students, online examples/tutorials, or any other material from any other source. Any assignments found to be based on work other than your own will automatically be given a **grade of zero**, and may lead to further disciplinary action as per QU policy.
- All assignments must be submitted electronically to Github. You should push your work to Github as you make progress. Late submission policy: 10 points deduction for each late day and 0 after 3 days.

Appendix A – At Risk Policy Matrix