Assessment One: Django Assessment

Released: 01:30 PM (Perth) on 10th March 2024
Last Upload: 11:59 PM (Perth) on 24th March 2024

This is an **OPEN BOOK** assessment. There are ten (10) questions. Answer all of the questions. There are 40 marks available.

You must work alone on this assessment. You must not communicate with anyone other than your Lecturer or Unit Coordinator regarding any aspect of this Assessment.

All submissions may be subjected to plagiarism testing, collusion and other forms of academic misconduct testing. You must cite any and all code from any source, other than code adapted from the workshop slides. You are free to choose how you wish to cite this code, as long as it is cited in a consistent format.

Problem

You have been approached by a university looking to create a web application to manage the students in the school of computer science. The school offers several courses such as Master of Computing, Bachelor of Cyber Security, and Bachelor of Software Engineering. The web application will store information on the two things – Courses and Students.

Each Course has a course name, the name of course coordinator, and size (Maximum number of students). Each Student has a name, email address, student id, and date of birth.

A Student can enroll in only and exactly one Course. However, a Course can have more than one Student. The total number of students in a course is calculated by counting the total number of enrolled students in that course. *The data model should implement these characteristics.*

The task of this assignment is to develop a simple web application using Django to allow the viewing of a list of students, a list of courses and a detailed view of a course.

The Assignment (Your Work) **(40 marks)**

1. Create a Django Project and App with appropriate names for the purpose of this assignment. Ensure that the App is part of the project.
Total 2 marks: 1 mark for creation of Project and App, 1 mark for linking.

2. Develop the models required to store the information required in this system. *Total 8 marks: For each of two models, 1 mark for correct identification and naming of model, 1 mark for correct identification and naming of attributes, 1 mark for correct identification and use of data types. Considering both, 1 mark for relationship between models, 1 mark for the method in models.*

3. In a Word document, describe why you chose the attributes and data types that you chose and why you implemented the relationship(s) between the two models in the way that you did. You may wish to contrast your choices with alternatives and consider the functionality enabled by your choices.
Total 3 marks; 1 mark for attribute explanation, 1 mark for data type explanation and 1 mark for relationship explanation.

4. Create and apply the migrations for your model into the SQLite database.

Total 2 marks: 1 mark for creating the migrations and 1 mark for applying them.

5. Using both a Project URL file and App URL file, create the URL patterns for the list of Students, list of Courses and a single Course.
Total 3 marks: 1 mark for grouped URLs and 2 marks for individual Match.

6. Create the View functions for the list of Students and list of Courses. This will also require the creation of templates to display the lists. Ensure that you create valid HTML pages (containing and utilizing the "HTML", "Head" and "Body" HTML tags, as well as a "H2" tag for the heading of each page).
Total 8 marks: For each of two pages, 1 mark for retrieving data, 1 mark for successfully rendering the template with data, 1 mark for template structure and 1 mark for template logic.

7. Create the View function for the single Course. This view should list the Students in each Course, the information stored about a single Course and the total number of students enrolled in the Course.
Total 6 marks: 2 marks for retrieving data, 1 mark for successfully rendering the template with data, 1 mark for template data and 2 marks for template logic.

8. Modify your templates such that they inherit from a common base template. Provide links (using HTML "A" tags) for the List of Courses and List of Students in the base template at the top of the page. Modify the List of Courses such that each Course links to the single Course page for it.
Total 4 marks: 1 mark for navigation links, 1 mark for links from events, 1 mark for creating base template, 1 mark for utilizing base template.

9. In the Word document used for question three, detail which CRUD operations you would restrict to the course coordinator, students and website visitors respectively and justify your decisions.
Total 3 marks: 1 mark for each group and its operations justified.

10. Finally, in the same document, discuss two of the benefits of implementing this App within Django, with respect to either security, performance or ease of development. You may provide two benefits from the same category.
Total 1 mark: 0.5 marks for each benefit.

Please do note that partial marks may be awarded (at 0%, 25%, 50% and 75%), commensurate with the effort and correctness of the answer to the component of the question being answered.

You will need to investigate the Django documentation as to how to achieve the relationship detailed within "The Problem" as well as undertake some personal research for the HTML component of the assignment. You are also welcome to include additional logic as long as it does not detract from the above steps, however no additional marks will be awarded.

Submission

An assignment submission point is provided on Blackboard LMS for submission of this assignment by the time detailed at the top of the document. Please .zip(or .tar) together your project (ensuring you submit the entire folder containing manage.py and all of its subfolders), alongside the Word document and a Declaration of Originality and submit this as a single attachment.

Please direct any queries to **Sheik Mohammad Mostakim Fattah** (*sheik.fattah@curtin.edu.au*), as clarifications will be made available on the Assignments section of Blackboard for all to see.