

**Instructor:** Francis Bello

**Course:** CSD3354, Summer 2022

**Assignment:** 5 (student registration web app)

**Total points: 50**

You are to build a web app that will ask simple student details for registration.

Use classes, JavaScript, data annotations, forms, Bootstrap, routing, other functionalities, as shown in the sample screens below.

No database is required.

Follow the screen prototypes/samples below.

**Instructions:**

Follow these instructions carefully.

1. Create a new project in Visual Studio.
2. Choose ASP.Net Web Application C# for the project template.
	1. If you are using VS2022 and cannot find this project template:
		1. Start -> Visual Studio Installer
		2. For Visual Studio 2022, click Modify button
		3. Click Individual Components tab
		4. Search/check .Net Framework Project and Item Templates
		5. Click Install While Downloading
3. Supply name as **Register\_*<your group number>***
	1. E.g.: Register\_Group1
	2. See list of groupings on Moodle
	3. Not following this step, assignment will not be checked
4. On the Create a new ASP.Net Web Application window, select MVC
5. Add a controller
	1. Controller name: **RegisterController.cs**
6. Add a registration page that asks for:
	1. Firstname – required, between 3 to 25 characters
	2. Lastname – required, between 3 to 25 characters
	3. Address – required, between 10 to 50 characters
	4. Program – required, between 3 to 25 characters
	5. Academic Year – required, integer, between 2022 to 2025
7. The registration page also displays:
	1. Register button – will ask the user whether to proceed registration or not via message box. If proceed, the user is brought to the confirmation page.
	2. Reset button – clears out the entries in the fields
8. Add a confirmation page that displays:
	1. Student ID = first character entered for the firstname + the lastname + random number between 1 to 10; all upper case
	2. Firstname
	3. Lastname
	4. Address
	5. Program
	6. Academic year
9. The confirmation page also displays a link (use HTML helper, not direct <a> tag) to go back to the registration page.

**Screens:**

1. Registration page

Use case 1: default load



Ask for: firstname, lastname, address, program, academic year (instruction #3)

Display: Register, Reset buttons (instruction #4)

No need to type /register or /home in the URL

Use case 2: clicking Register button, empty fields



Prompt the user to proceed the registration or cancel via message box.

Click Cancel and the message box disappears.

Click OK with empty fields will trigger validation, will not perform any action. See the next screen below.



Use case 3: entering invalid field values



See individual field specs above (instruction #3)

Use case 4: entering valid values, clicking Register button



No validation will trigger.

Will prompt the user to proceed or cancel via message box.



Clicking Cancel will not perform any action, message box disappears.

Clicking OK will bring the user to the confirmation page.

1. Confirmation page

Use case: display information



Display all values from the registration page (firstname, lastname, address, program, academic year)

Display student ID field. See formula for this field (instruction #5-a)

Display ‘Back to Registration’ link (instruction #6). Click this link will bring the user to default landing page (use case 1)

**Submission requirement**.



1. Clean your solution first (Build -> Clean Solution)
2. Zip the Register folder and the Register Visual Studio Solution file
3. Upload this zipped file to Moodle (email will not be accepted)
4. **Do not** include the packages folder
5. If your zip file is greater than 5Mb (max upload for Moodle), remove the \bin folder from your zip file.
6. Put your student numbers as the filename.

**Scoring Matrix**

|  |  |  |
| --- | --- | --- |
| **Item** | **Points** | **Earned Points** |
| Correct project and zip name | 1 |  |
| Correct home/landing page URL | 5 |  |
| Correct fields on registration page | 5 |  |
| Correct buttons on registration page | 2 |  |
| Correct behavior (validation) of fields on registration page | 10 |  |
| Correct behavior of Register button (message box) | 2 |  |
| Correct behavior of Reset button (clears all fields) | 2 |  |
| Correct behavior of message box | 2 |  |
| Correct behavior of registration page (redirects user) | 2 |  |
| Correct fields on confirmation page | 7 |  |
| Correct behavior of link on confirmation page | 2 |  |
| Program compiles & runs without error | 10 |  |
| **Total score** | 50 |  |
| **Percent** |  |