

Individual MVC Application

[Submit Assignment](#)

Due Jul 16 by 9am **Points** 34 **Submitting** a website url
Available Jun 24 at 9am - Jul 18 at 9am 24 days

Submission Guidelines: The final product will be deployed to the web using Microsoft's Azure Web Services and can be linked to in the projects section of your Portfolio.

Overview: Your assignment is to build a .NET Framework MVC 5 Web Application using n-tier architecture covering a topic of your choice.

Requirements:

- Come up with an idea for an application that will use a minimum of 3 custom data tables with a minimum of one foreign key relationship
- Create a wireframe of the application and review it with an instructor
- Build out an n-tier structured MVC (think ElevenNote) with the following tiers
 - **Data:** This layer houses your classes that relate to the DB (POCOs, Identity, etc)
 - **Models:** This layer houses the reusable models for the rest of the layers
 - **Services:** This layer is where you'll have most of your heavy backend logic
 - **WebMVC:** This is where the front end (Controllers, Views, Styles, etc) will be
- Deploy to Azure
- Link to deployed site on your Portfolio
- Repository has a README.md associated with the project
- Submitted by Due Date

.NET Red Badge Project Rubric

Criteria	Ratings					Pts
Full Stack WebMVC (Front End)	5.0 to >4.0 pts Advanced Student application does not resemble the pre-built ASP.NET MVC. in any fashion. Student utilized HTML and CSS effectively, incorporated Javascript and/or JQUERY on more than one page. Site is well designed, polished	4.0 to >3.0 pts Proficient Application views have been greatly modified, but may still have some features similar to the pre-built ASP.NET MVC, such as the navigation bar, table styling, login page, etc..	3.0 to >2.0 pts Basic Student has completed some styling but views resemble the pre-built ASP.NET MVC.	2.0 to >0.0 pts In Progress Minimal changes implemented in the views.	0.0 pts See Instructor	5.0 pts
Full Stack WebMVC (Back End)	5.0 to >4.0 pts Advanced Student created three or more tables and established three or more different types of data relationships. Student utilized interfaces and dependency injection. Student was able to explain the architecture of the application with no correction or assistance.	4.0 to >3.0 pts Proficient Student created three or more tables and established at least two different types of data relationships. Student able to speak to the architecture with minimal help or correction.	3.0 to >2.0 pts Basic Student met the requirement of three tables and one relational table. Student was able to talk to the architecture of the application with some prompts.	2.0 to >0.0 pts In Progress Student application does not meet requirements or submitted after due date. unable to speak to the architecture of the application.	0.0 pts See Instructor	5.0 pts
Database Structure	5.0 to >4.0 pts Advanced Student created three or more tables and established three or more different types of data relationships.	4.0 to >3.0 pts Proficient Student created three or more tables and established at least two different types of data relationships.	3.0 to >2.0 pts Basic Student met the requirement of three tables and one relational table.	2.0 to >0.0 pts In Progress Student application does not meet requirements or submitted after due date.	0.0 pts See Instructor Students needs to speak with instructors about meeting this criteria.	5.0 pts

Criteria	Ratings					Pts
Controller Functionality	5.0 to >4.0 pts Advanced Student created full CRUD functionality for three or more tables.	4.0 to >3.0 pts Proficient Student created full CRUD functionality for two or more tables.	3.0 to >2.0 pts Basic Student created some CRUD functionality for two or more tables	2.0 to >0.0 pts In Progress Student created some or all CRUD functionality for one or table.	0.0 pts See Instructor Students needs to speak with instructors about meeting this criteria.	5.0 pts
Implementation of Software Development Principles	5.0 to >4.0 pts Advanced Student utilized multiple interfaces and properly implemented dependency injection across their entire project as well as practiced Single Responsibility with all classes.	4.0 to >3.0 pts Proficient Student utilized interfaces and dependency injection in one or more layers in their project and had clear classes that served single responsibilities.	3.0 to >2.0 pts Basic Student created and implemented interfaces in parts of their project as well as had clear defined classes.	2.0 to >0.0 pts In Progress Student defined clear classes but did not implement interfaces in their classes.	0.0 pts Not Completed	5.0 pts
Wireframe Completed	1.0 pts Completed		0.0 pts Not Completed			1.0 pts
Planning Document Completed	1.0 pts Completed		0.0 pts Not Completed			1.0 pts
Data Layer	1.0 pts Completed		0.0 pts Not Completed			1.0 pts
Models Layer	1.0 pts Completed		0.0 pts Not Completed			1.0 pts
Service Layer with Separated Services	1.0 pts Completed		0.0 pts Not Completed			1.0 pts

Criteria	Ratings		Pts
MVC Layer with Separated Controllers	1.0 pts Completed	0.0 pts Not Completed	1.0 pts
Deploy to Azure	1.0 pts Completed	0.0 pts Not Completed	1.0 pts
Link to Github repository or deployed site on Portfolio	1.0 pts Completed	0.0 pts Not Completed	1.0 pts
Submitted by Due Date	1.0 pts Completed	0.0 pts Not Completed	1.0 pts
			Total Points: 34.0