

Vasja Collaku's solution on the task given of Back End Developer

Hereby I will explain my approach to solving the task given. I will be sincere in my comments and I hope this will not be held against me.

As I have mentioned to Farjam, I have a very bad time schedule and today was the only day I had available to solve the task, so when I first read the description of the task and encountered the notations of RESTFUL and CRUD which I did not know before, I decided to go with creating a ASP.NET MVC, as that is what I had previous knowledge of and time was not on my side. However, I understand that that was not what you required of me.

I created an MVC project where:

- The Model is represented by the Banner class given in the task
- The View is represented by the html classes provided to the GUI. In the GUI of my project, In the home webpage, expect the initial template view, there is also the navigation bar with options of About page, Contact page and Banners page. About and Contact are represented in the Views/Home folder whereas the Banners view is represented in the Views/Home/Banners folder.
- The Controller is represented by the classes which handle the requests for the Home and Banners page.

The Banners page is where I have put the solution. In this page, there are displayed the banner entries of the (local) database (I created). On top of the page, there is a link to create a new banner entry. The user can enter a text (string) containing the HTML of the banner. A validation is done upon the input such as no plain text can be saved. After saving the entry, the creation time of it is saved as well as an attribute of the banner and it cannot be changed anymore. For each entry, there is the option of Editing (the HTML string, and the modification date is set/updated), Details, Deleting and Rendering (which opens another page where the banner HTML string is rendered properly).

In the end, when I was looking how to generate test units for my application, I came to understand the importance of completing the task as you had specified, with a web API instead of a MVC.