Web Application Evaluation

CS348 Coursework 3

Swansea University

Author:
Ethan Davies
828633

December 2016

1 Understanding of the MVC 5 framework

In this section the MVC 5 framework will be broken down and evaluated, whilst making relating the discussion to the implementation of my project.

1.1 Applying the framework

For the implementation of the web application the MVC 5 framework for ASP .NET was used. The web application was developed in Visual Studio 2015. The MVC 5 framework is an incredibly powerful framework, "ASP.NET MVC 5 is a framework for building scalable, standards-based web applications using well-established design patterns" [1]. The MVC framework provides the developer with a range of features such as ASP .NET Web API, Display Modes, Enhancements to default project templates and binding and minification [3]. The MVC 5 framework was used in order to implement the designed application in the first coursework,

1.2 Models, Views and Controllers

MVC stands for Model View Controller, and in Visual Studio ASP .NET MVC 5 projects are split up into just that, Models, Views and Controllers. The models in MVC are used to represent the data itself, for example; in my application and example of a model would be the model of an announcement on the application. This model will then contain the values that a typical announcement would have, such as the body, title and so forth. Models can also be used for a concept called model views, this means that views can have specific models attached to them. This means that any data you wish to display on a view can be taken from a view model, this allows the programmer to create specific models for each views which in turn allows them to choose what kind of data and which attributes they wish to access when dealing with the model in the controller for a specific view. "Models are the classes that represent the domains" [3] that a user is interested in, the data in these classes can then be accessed in ASP .NET MVC using the Entity Framework [3]

Views are pages that the user will be displayed once the application is deployed. As mentioned previously, views can have specific models attached to them, these views are then passed the models by the controller. Views can be used for displaying information and also allow user input. In ASP .NET MVC 5 programmers have access to the Razor engine when developing their application, "Razor is an ASP.NET programming syntax used to create dynamic web pages with the C# or Visual Basic .NET programming languages" [2]. The main focus of the Razor View Engine is to provide "code focused templating for HTML generation" [3]. The Razor syntax in one that is very easy to read, and when compared to the Web Forms view engine's syntax it is very simplistic [3]. The main advantage that the Razor engine and its syntax brings is that it does not require for the programmer to have learn a new language. The Razor syntax can be initiated with @, after this the engine knows that from that point on to start reading in C# programming language. Razor is very powerful, and a developer will likely already know HTML and C#, and another one of its main advantages is that "it is so lightweight and HTML-focused, you're free to use the editor of your choice." [3], this means that even though the editor in Visual Studio that includes IntelliSense features is very good, the code can also work with any other text editor. [3]. Along with the Razor engine, Views can be used effectively to provide a good user interface for the user, where input can be validated and sent to controllers in order to update a database existing value or say add a new value all together, features such as the ViewBag can also be used in order to pass data back and forth from the controller to the view and so forth, this was applied in my project when the user would select a type of sorting they required or to validate a search value they entered.

Controllers are "special classes that manage the relationship between the view and the controller" [3]. The controller is responsible for responding and dealing with any user input that is received from the relevant view [3]. It does this by taking the input, talking to the relevant model then deciding which view to render

based on the input that was received from the current view [3]. Any data flowing in and out of the application generally has to go through the controller, the controllers purpose is then to decide what to do with this data based on the methods that are in the controller. This could be changing a value or adding an element to a database. When a new ASP .NET MVC 5 project is created in Visual Studio, the application will generated a framework for you to work from. This will include the ability to create user account and manage them. In the project a folder called controllers will be generated and inside will be the default home, manage and account controllers that are responsible for dealing with account data when a new account is created or a user tries to log in and the manage controller will be responsible for any change in the user data that is stored such as passwords. These controllers are connected to views and in turn to models that the controller can use in order to change data based on the definition of a model and its attributes.

1.3 Relation to the Implementation

Having discussed models, views and controllers I can now discuss how this framework relates to the implementation of my project. The task at hand was to develop a fragment of a Virtual Learning Environment where lecturers are able to post announcements that students can view and comment on. MVC was the perfect framework in order to implement something like this. In my implementation a view model was created that included all the models required to be able to build an announcement for the application, this model was titled view models. This included models such as courses for each announcement, the announcement itself as well as the comments that would be posted on each announcement. Inside this view model were also other view models, such as PostViewModel and CommentViewModel. These view models were used in order to create specific models for specific views, view specific models. These could then be accessed by the view and the controller in order to display information on the view and also accept user input and update the model itself. The post view model includes many different attributes that all relate to an announcement post, these include the post itself, the comments and the courses that the post was created for. By sending this model the view, the view is able to display the data contained within that model. The view in question is the post view, this view would be responsible for displaying the data contained within the model like the title or the post, its body and its comments and categories etc. The model is sent to the post view via the controller.

The example above was repeated in views throughout the project, which resulted in many more view specific models for the controller to deal with. In my project however, the controller did not deal directly with the changing of items in the database. In order to achieve this a repository was set up that contained access methods to the database that would change data, whenever an announcement, course or comment were created, deleted or updated these methods were called by creating a repository object in the controller in order to access the methods.

1.4 Comparing to Other Frameworks

There are many different frameworks out there that are used by different web technologies. One of the most popular is MVC but there are also others that are popular and becoming increasingly popular. The decision to select each of these model frameworks is dependent on the task that needs to be completed. “The MVP (Model View Presenter) design pattern also comprises of three components - the model, the view and the presenter” [7]. In this design the controller from the MVC framework is replaced by the Presented [7], the presenter works by referring back to the “view due to which the mock of the view is easier and unit testing of applications that leverage the MVP design patter over the MVC design pattern are much easier.” [7]. The MVP framework is generally preferred when the application that you are designing needs to provide support for multiple user interface technologies [7]. MVP isn’t the only alternative however. MVVM or Model View ViewModel is essentially a refinement of the popular MVC design [7]. The ViewModel in MVVM is used to “facilitation Presentation Seperation” [7]. In MVVM the logic of the system is stored in the presenter and the view is completely isolated from the model and while the presenter isn’t aware of the view, the view is aware of the presenter and the presenter is used to represent an abstract view of the user interface [7]. “The viewmodel in MVVM is responsible for

presentation separation and exposes methods and commands to manage the state of a view and manipulate the model" [7].

2 Evaluating ASP .NET MVC 5

When initially working with ASP .NET MVC 5 I was face with much confusion to what was going on and to understanding the relationships between the model, view and controller. In order to further my understanding I searched for many tutorials and blog posts online to help me with my understanding. In order to better my understanding I created many different small projects before attempting to create the project for this coursework, I followed the course to do list tutorial as well as others showing how to deal with databases and view specific models as well as tutorials on how to upload images and create blog posts. As I completed these tutorials and small projects I began to realise what could be done with the framework, especially when learning about CSS styling and JavaScript. All this knowledge and the use of these tutorials was used when creating the project which will be evaluated later. As I became more comfortable with the framework I began to enjoy completing the work and became passionate about what was being made and was in awe many times of how it all worked seamlessly.

2.1 Alternate Web Technologies

As good as ASP .NET may be it isn't the only technology, there are many other web technologies and frameworks out there such as Angular [4]. "Angular is a complete JavaScript-based open-source front-end web application framework mainly maintained by Google and by a community of individuals and corporations to address many of the challenges encountered in developing single-page applications" [4]. The way angular works is quite different from the ASP .NET MVC 5 framework. Angular works by first reading the HTML page that has custom tag attributes embedded and then interprets the attributes as directives to bind input or output parts of a page to a model [4]. Angular does not interpret requirements in the system based on the models at hand, that is angular does not require the system to have observable patterns in order to make use of the framework [5]. Angular is used in order to develop single page applications, which provides the user with a user experience similar to that of using a desktop application. This is achieved through retrieving all necessary code such as HTML, JavaScript and CSS in one single page load [6]. Angular isn't the only competitor however, there are others available such as ASP .NET Web Forms and most notably PHP. PHP comes with any different benefits with the main one being that PHP will run on any server for free, an advantage that other web technologies do not possess [8]. PHP is a very well documented online web technology and there is no shortage of support available online. But one of PHP's main disadvantages compared to ASP .NET web technologies is the inability to create a GUI that can rival that of ASP .NET MVC 5 [8]. All the technologies above could have easily been used for the project to complete the implementation of the announcement system.

2.2 ASP .NET MVC 5 Effectiveness

There are many different reason why ASP .NET MVC 5 is deemed as effective, one of the main reasons is down to its MVC framework with the ASP .NET web technology providing a very good implementation of this framework. There are many other benefits to using the framework that make it an effective tool to provide a solution to a problem, one of the reason that it is so effective is that is gives the user full control over the rendered HTML. One of the main attractions to using this framework in ASP .NET is that is provides and enables the user to engage with a test driven development process. ASP .NET MVC 5 boasts very easy integration with different JavaScript frameworks which can then be implemented in the views with ease. From a technical standpoint the organization in MVC is very effective and very clean. MVC manages to separate all your concerns by making it organized and granular, it promotes great design.

3 Security

In the web application security was addressed in the models, views and configuration files. One of the main risks when developing the application was the vulnerability to cross site scripting attacks, this risk was primarily mitigated by using the HTML encoding feature that the Razor engine provides as part of the framework. By encoding the values from the models that are being passed to and from the controller, the view helps mitigate any malicious code that users may try to implement when input is requested. The main rule of thumb that was adhered to when implementing was to treat all the input from the users as malicious and not to trust it, therefor by encoding model view attributes, when they are saved in the database and entered into the view, no code Is executed and no injection will occur. Other areas where steps were taken to improve the security of the application was in the web configuration file in the project. Steps were taken to prevent cookie theft and also reduce the security risks that error posting presents by allowing custom errors for when the application is deployed. The issue of cross site request forgery was mitigated by using anti forgery tokens in the views and validating them accordingly. The risk of over posting attacks was also mitigated by using view specific models in the project.

Even though ASP .NET MVC 5 automatically encodes data for you there are still many other risks out there that are not accounted for that have been mentioned above. ASP .NET MVC 5 does not automatically prevent you again cross site request forgery, this is when a malicious user exploits the trust of the website on other users, and this issue was prevented by manually entering the anti-forgery tokens in the relevant views. The same can be said for cookie theft and error posting issues, these had to be altered in the web configuration file. ASP .NET does not automatically prevent open redirection security issues, in order to solve this you must make sure in the controller that all redirecting URL's are local to the application and do not take the user to some external site.

Overall I believe that I have taken the steps necessary to help prevent any attacks on the system, whilst I am sure there are some advanced vulnerabilities that the application does not account for, I believe that using the practices stated above means that the security of the application can be deemed as sufficient at a basic level.

4 Evaluation

During the development of my web application I encountered many different challenges, with the primary one being learning the framework and the languages required in order to build a web application. Previously I had no experience with HTML or JavaScript so I found these quick difficult to pick up, but after enough practice a following tutorials I began to learn quickly. C# wasn't much of an issue as it shares many similarities with other OOP languages such as Java which I have been learning since first year. Building the application itself was a very satisfying process, and I am proud of the resulting application. The project contained many different subtle features that complement the basic requirements very nicely. Features such as sorting a list of posts by data and title were added, this feature improves user experience greatly and is a feature that is incredibly useful whilst also being very relevant to the coursework specification. Users can also sort by courses and also search by text for a specific post. One of the most impressive features that the application includes is its comments system. Users are able to comment on different posts, these comments can then be replied to by other users and so forth, thus displaying the replies and child replies of the comments, all these comments can then be deleted or edited by their respective users. The application manages to deliver the minimal specification requirements of posting announcements, viewing announcements, commenting on announcements and also being able to view which users have not seen certain announcements. As well as this each announcement can have a course/module attached to it which allows users to then filter announcements based on course.

I believe the implementation of my application can be judged as a success not only from a technical standpoint but also from a visual standpoint. A CSS file has been used in order to undertake all the styling of each view in the application. The colour palette has slightly changed from the original design to incorporate a theme of orange and light sea green colours that accent each other well. The CSS files have

also been used to improve the navigation bar to make it feel more responsive by highlighting menu items as they are selected and added a border around them. Borders have been a common theme throughout the styling of my application and they have been used to enclose important and relevant data in specific areas to try and keep the attention of the user firmly on the task they are trying to complete. I am very happy with the styling of my application as I am not generally good at design and am proud to have produced a UI design that I think is good.

One thing I wanted to significantly improve when developing my application was the way that users would input data. Originally this was done with HTML default text boxes and text areas, but as a tutorial I was following progressed I was introduced into using an external API text editor called CKEDITOR. This text editor provided users with many different tools for when they were inputting data and provided them with a clean, sleek and easy to use interface for inputting text into the system. The text editor is used for when large bodies of text are required from the user such as comments and announcement bodies.

One of the aspects of my application that I would improve and wish to improve on in the future would be the inclusion of more JavaScript in the project. As it stands there is some JavaScript in the project when it comes to comments and the displaying of comments, but I would like to do more with JavaScript and validation, especially when it comes to the validation of the text editor's content. Going forward I would like to implement some Ajax functionality in the system, this would be incredibly useful for the comments sections where comments could be added using Ajax which would give the user the illusion that a comment has been added instantly to the page. If I were to do this coursework and application again I would like to add a lot more features for the lecturers and students of the system, features that are more in line with those originally designed in the initial report. The project I have created was achieved using many different tutorials but primarily a blog tutorial that was followed on YouTube in order to create the application. I believe I have learnt a great deal about this framework by using this tutorial and applying it the specification of the coursework, I believe that If I were to make another application I would be able to do so with much less assistance from the likes of tutorials online which is a great feeling.

Overall I believe that by comparing my application to the specification and to the original design that I have produced something that meets the criteria and a lot of the elements in the rubric. I am happy with the final product and plan to improve on it in the future and use it to help develop other applications that I will build.

References

1.) Microsoft (2016) *ASP.NET MVC 5*. Available at: https://www.asp.net/mvc/mvc5 (Accessed: 22 December 2016).

2.) NuGet (2016) 'ASP.NET razor', in *Wikipedia*. Available at: https://en.wikipedia.org/wiki/ASP.NET_Razor (Accessed: 22 December 2016).

3.) Galloway, J., Haack, P., Wilson, B. and Allen, S.K. (2012) *Professional ASP.NET MVC 4*. United States: Wrox Press.

4.) *AngularJS* (2016) in *Wikipedia*. Available at: https://en.wikipedia.org/wiki/AngularJS (Accessed: 22 December 2016).

5.) *What should I choose, angular.js or .NET MVC? Which language has better career prospects?* (no date) Available at: https://www.quora.com/What-should-I-choose-Angular-js-or-NET-MVC-Which-language-has-better-career-prospects (Accessed: 22 December 2016).

6.) *Single-page application* (2016) in *Wikipedia*. Available at: https://en.wikipedia.org/wiki/Single-page_application (Accessed: 22 December 2016).

7.) Kanjilal, J. (2015) *Exploring the MVC, MVP, and MVVM design patterns*. Available at: http://www.infoworld.com/article/2926003/microsoft-net/exploring-the-mvc-mvp-and-mvvm-design-patterns.html (Accessed: 22 December 2016).

8.) *What advantages does PHP have over ASP.NET?* (2016) Available at: http://stackoverflow.com/questions/3989352/what-advantages-does-php-have-over-asp-net (Accessed: 22 December 2016).