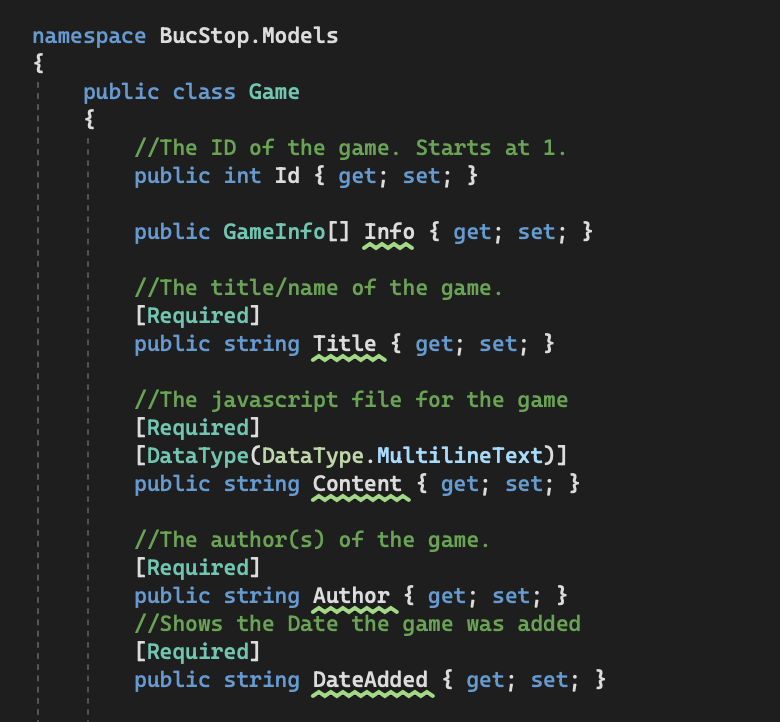
**MVC – Model View Controller**

**Models**

Akin to objects. Models the data of the website.

Game for instance holds information related to games:

* Title
* Author
* Date
* Play Count



Structured like an object, but has additional information, such as [Required] that is caught by ASP to flag what data is needed when the model is created and used.

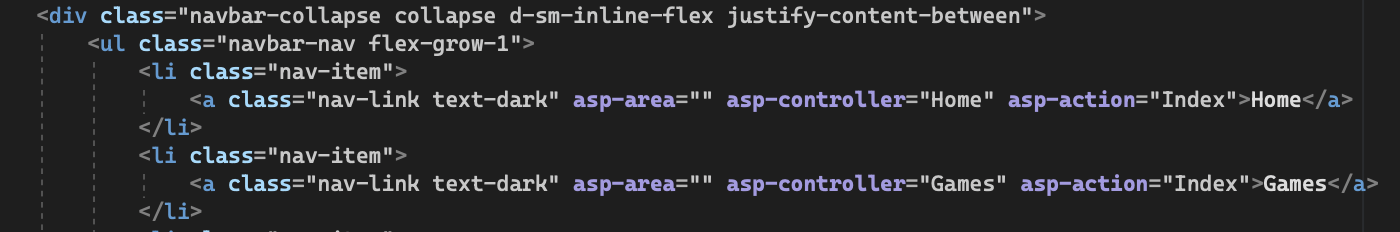
**Controllers**

Controllers manages the logic of the webapp and its views. It manages which views are used by the webpage (or layout) at a given time.

A screen shot of a computer program

Description automatically generated

Controllers hold functions that return views. These functions are then called in the webapp to dictate which view is used at a given moment. The return value of View() can take in additional parameters to specify which view will be served. If it is left blank, it will default to the method name – i.e. Admin would grab the Admin view.

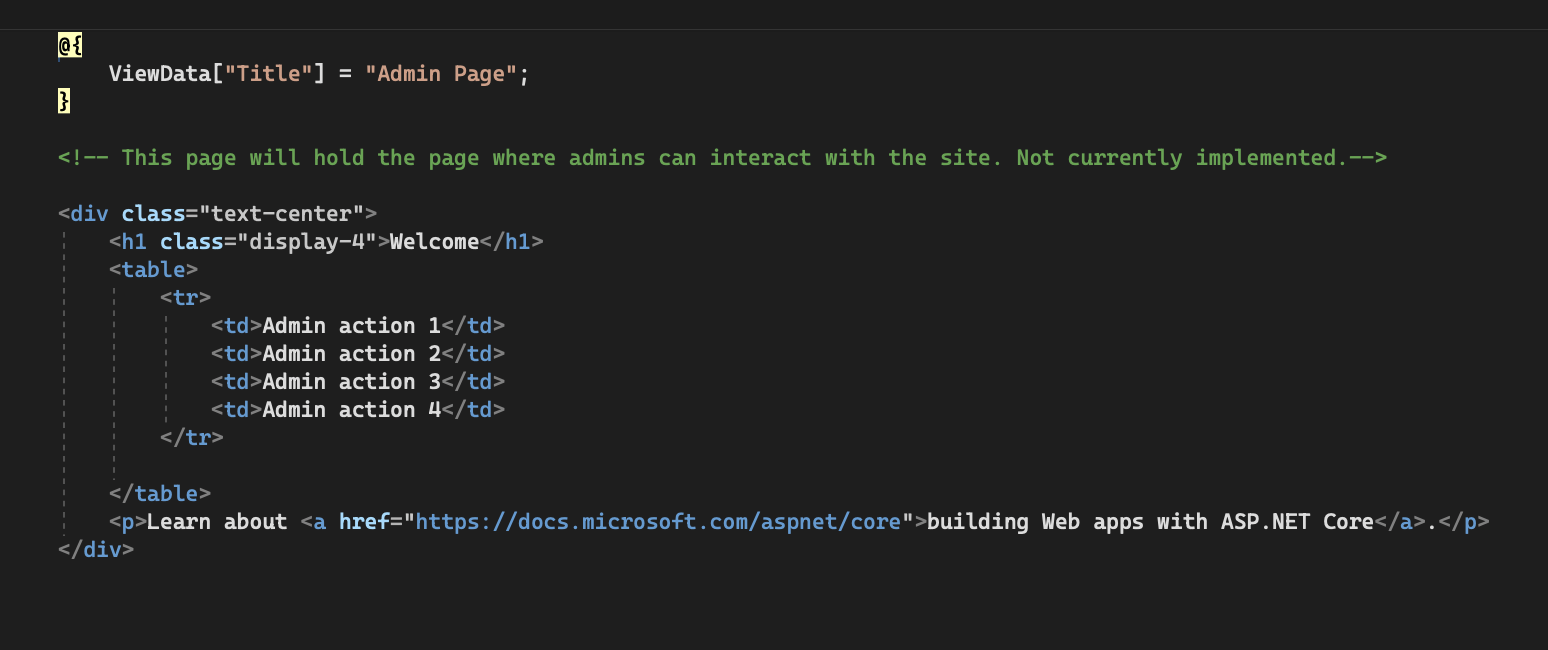


Controllers can be called using a links. These take in an asp-controller and asp-action. The action corresponds to the methods within the given controller.

**Views**

Views are html templates that are called by the controller and served to the user. They use the .cshtml file type.

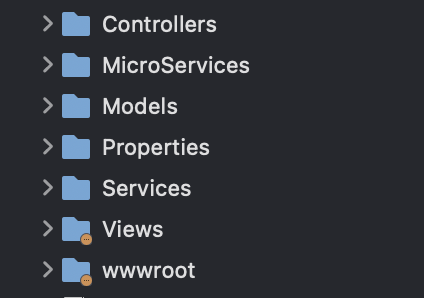
\*\*They are usually not full html pages\*\* i.e. no header and footers



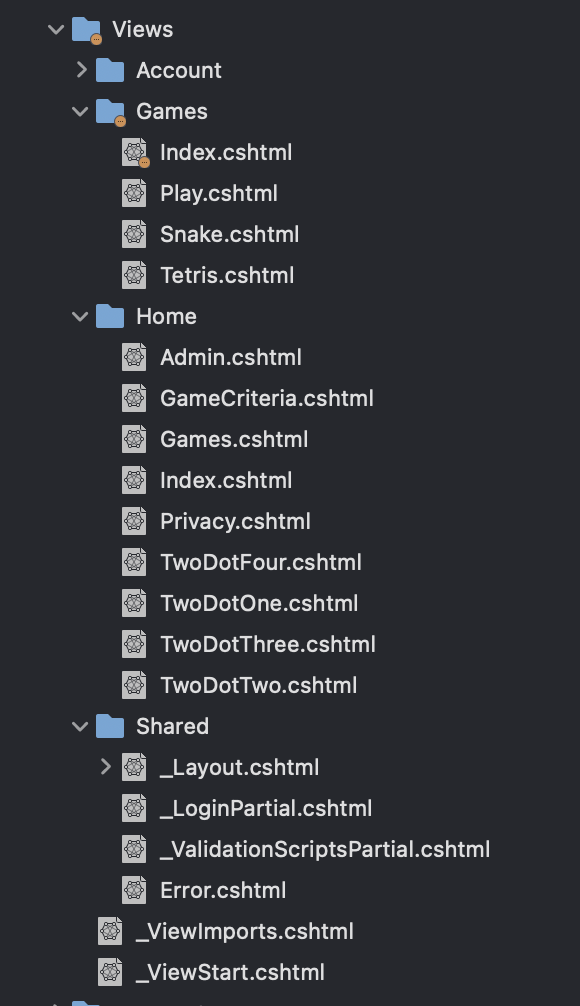
View data can be added to the top of the file

**File Structure**

File structure is very important. While pathnames can be used for views, Controllers will typically look in specific folders for views.



Here is a top-level view of what a ASP MVC file structure might look like. While most of the folders contain files, views require a bit more in-depth structure.



Here you can see the structure for views. We have multiple folders. Shared is a folder for views that will be used between all pages. Scroll down for more info.

The other folders (Account, Games, Home) all correlate to controllers. Underneath the folders are the specific views. ASP uses this folder structure and naming scheme to properly identify which views are meant to be rendered.

**Naming Scheme**

ASP MVC uses a specific naming scheme that helps it identify controllers and views.

Methods that return an empty view need to match the name of a view underneath the appropriate folder.

Controllers follow the convention of [Name]Controller.cs

The name is then used for ASP controller in views.

**Partial Layout**

Views are usually not whole webpages but rather portions of a webpage.

The rest of the webpage is usually served up via the Views/Shared/\_Layout.cshtml, although other layout names can be used as well if other partial views are required.

A black background with text

Description automatically generated

The layout typically contains a @RenderBody call that identifies where in the html a view should be added.