

# Assignment # 3

(Date Assigned: Dec 28, 2020)

(Due Date: January 10, 2020 Time: 11:59 PM)

AOA Everyone! The main aim of this assignment is to:

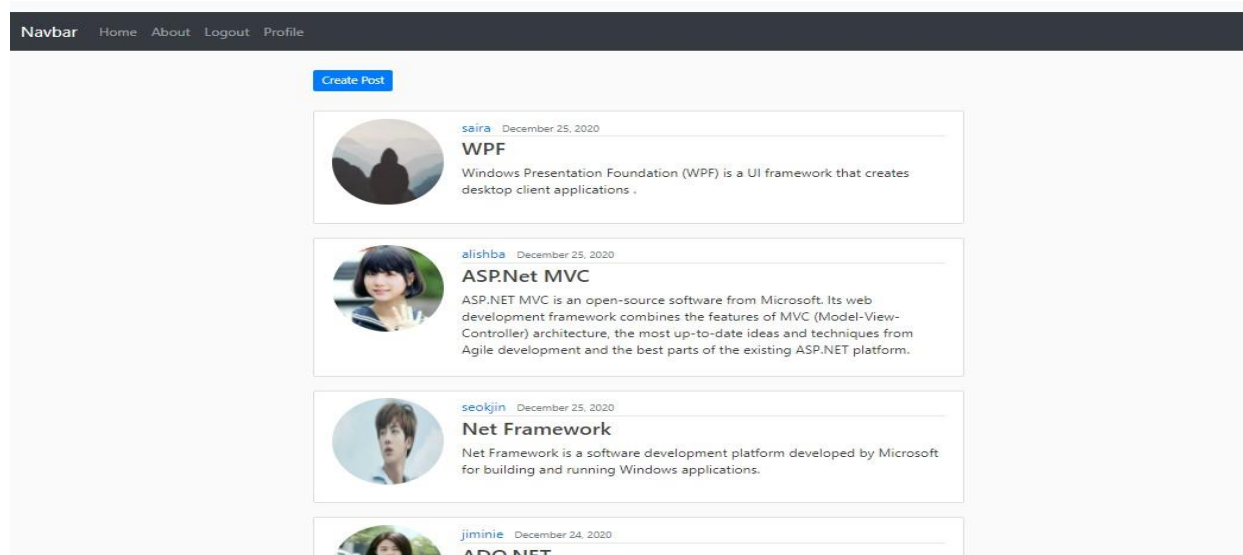
- Familiarize you with basic CRUD operations in ASP.net MVC.
- To offer as helpful practice for using Database.
- To apply design pattern (MVC).
- And yes, to help us judge how much you have learnt in the past weeks.

## Assignment Statement

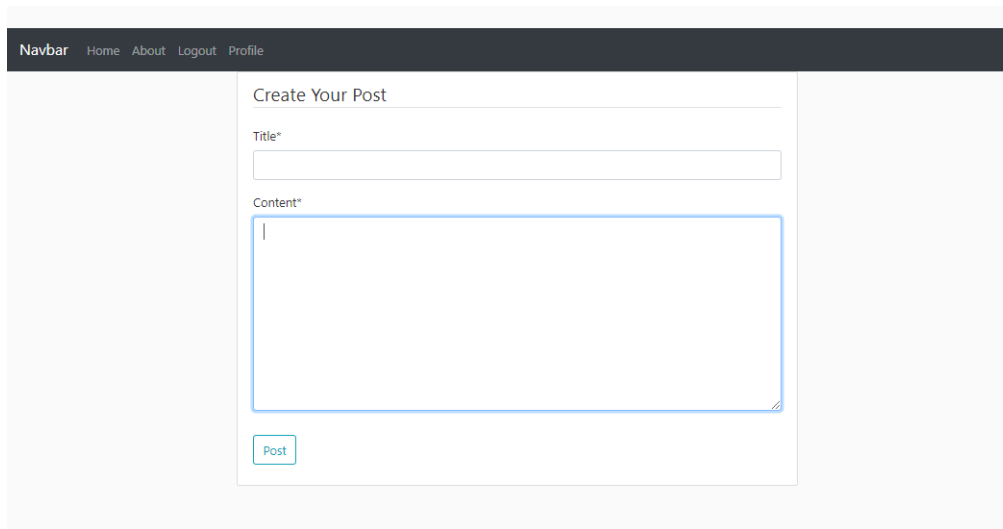
In this assignment, you are required to create a simple Blog Application. There are two types of users: admin and regular users. Admin can add, delete, and update all the users. Simple users can only view others posts while can create an account and can edit/update/delete posts written by themselves (like we all do on facebook/Instagram). They can also update their profile.

## Content / Requirement

When your program starts, it should provide users with options of Login and Signup. When a user is successfully logged in Home page should be shown to him. All the posts written by all the users in your database should be displayed on Home page as shown in the figure:

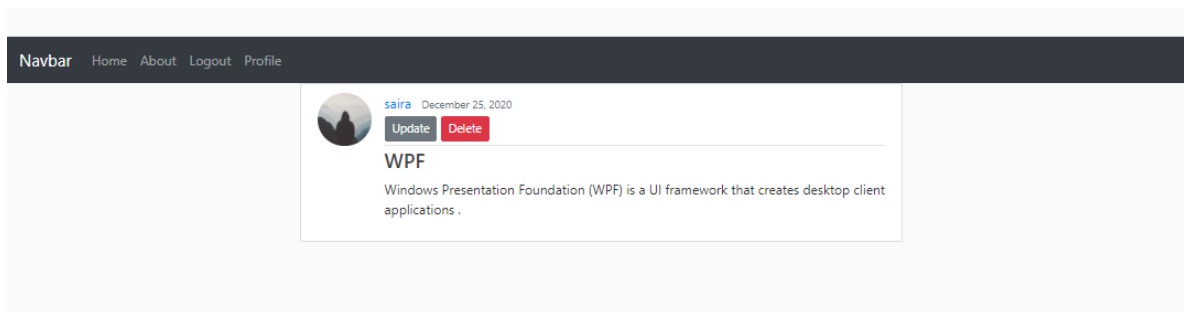


Note that On Home screen there is a button “Create Post” on the top of posts. When User clicks on it the following screen should be displayed:



The screenshot shows a web application interface with a dark blue navigation bar at the top containing the text "Navbar Home About Logout Profile". Below the navigation bar, the main content area is titled "Create Your Post". It features a form with two input fields: "Title\*" and "Content\*". The "Content\*" field is a larger text area with a blue border. Below the text area is a small blue button labeled "Post".

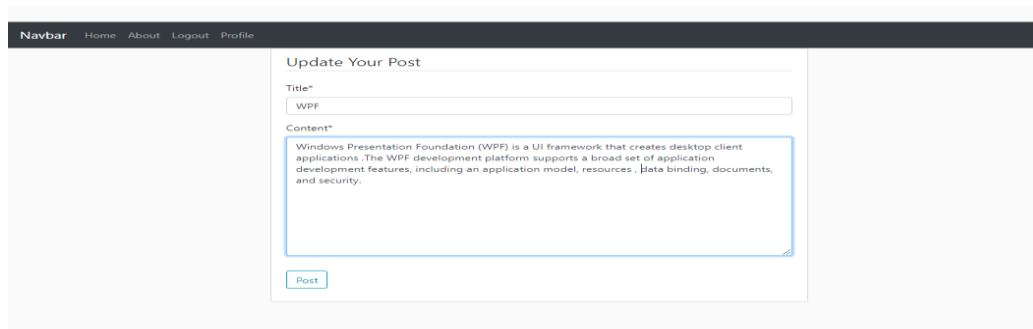
After Creating a Post when user clicks on “Post” button , The data should be saved in the database and then redirect user to home screen where new post written by you along with all the previous posts are displayed. In the current Example I am logged in as Saira. Now if Saira wants to edit her post. She’ll click on her post Title and the following screen should be displayed.



The screenshot shows a web application interface with a dark blue navigation bar at the top containing the text "Navbar Home About Logout Profile". Below the navigation bar, the main content area displays a post by a user named "saira" on "December 25, 2020". The post has a profile picture and two buttons: "Update" and "Delete". The post title is "WPF" and the content is "Windows Presentation Foundation (WPF) is a UI framework that creates desktop client applications .".

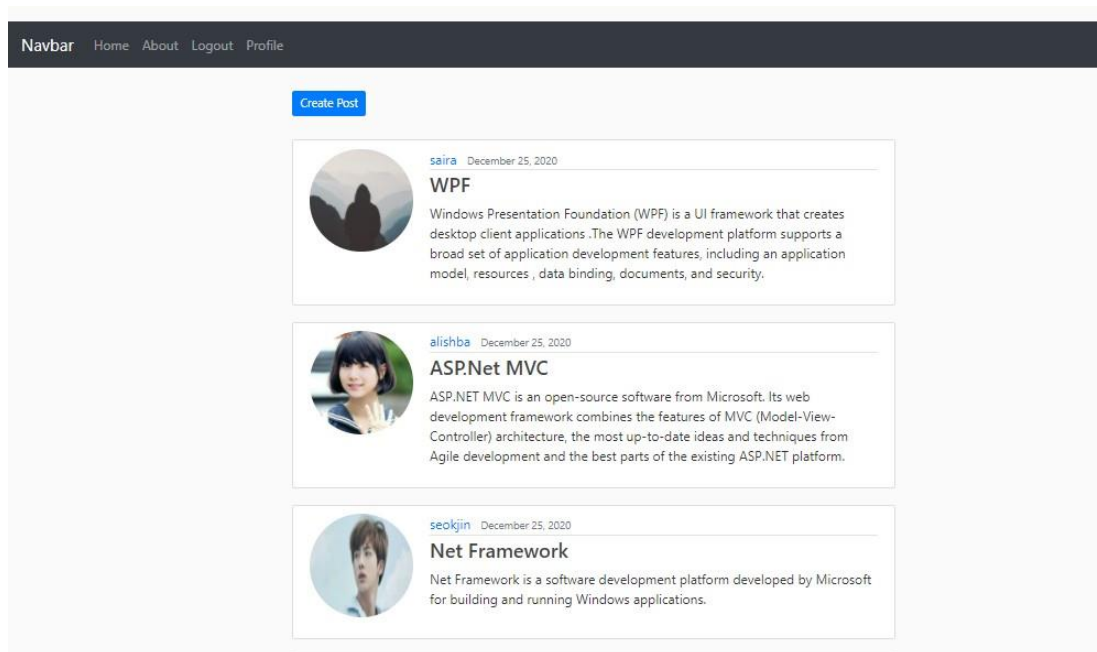
When she clicks on delete button, the post should be deleted.

When she clicks on Update, following screen should be displayed. After making changes when clicks on “Post” button, the post should be updated.



The screenshot shows a web application interface with a dark blue navigation bar at the top containing the text "Navbar Home About Logout Profile". Below the navigation bar, the main content area is titled "Update Your Post". It features a form with two input fields: "Title\*" and "Content\*". The "Content\*" field is a larger text area with a blue border. Below the text area is a small blue button labeled "Post".

And the home screen will be displayed and the post should be updated there as well.



If she clicks on other people's Posts, she will be provided with a screen where she can only read their post in detail but cannot edit/delete it as shown below. Only the author of the post should delete the post.



Now look at the navigation bar, if she Clicks on “Profile”, following screen should be displayed:

The screenshot shows a web application interface. At the top is a dark navigation bar with links: 'Navbar', 'Home', 'About', 'Logout', and 'Profile'. Below this is a profile page for a user named 'saira'. The profile section includes a circular profile picture placeholder, the name 'saira', and the title 'Profile'. Below the title are several form fields: 'Username\*' with the value 'saira' and a note 'Required. 150 characters or fewer. Letters, digits and @/./+/-/\_ only.'; 'Email\*' with the value 'abc@gmail.com'; 'Image\*' with a note 'Currently: default.jpg' and a 'Change:' section containing a 'Choose File' button and the text 'No file chosen'; 'Old Password\*' with a masked input field; and 'New Password\*' with a masked input field. At the bottom of the form is an 'UPDATE' button.

She can update her profile and when she clicks on “update” button, her profile should be updated and she should be redirected to home screen.

Now from navigation bar when she clicks on ‘about’, you can simply show her some dummy data (doesn’t matter)

When she clicks on logout, she should be logged out and login screen should be displayed.

## Design Cues and Requirements

- Use Database to store all data.
- Code intelligently.
- Exception handling should be done, if any.
- **MVC pattern should not be violated at any cost.**
- **Use of Bootstrap is compulsory.**
- All data entry points should have proper error checks and error messages.
- You can change interface designs, above pictures are added only for reference.
- There must be proper commenting throughout your code.
- Use Function and Variable names intelligently. The harder your code is to check, the higher are the chances of you getting lower marks. If your program crashes for some reason during checking, I will be just scrolling through your code and give you marks on the quality of the code. So, it’s purely in your interest to use nice functions and variables names

## **Marking Scheme**

Correct Validations	10%
Comments	5%
Proper Sign Up and Login	10%
Database Structure	5%
MVC Pattern not violated	15%
Profile Functionality	25%
Post Functionality	25%
Bootstrap	5%

***“I’m like a surfer, first you just paddle and fall off the board but as time goes by you can stand up on the bigger waves.”***

**~ Kim Namjoon (BTS)**

**Good Luck 😊**