

# Fresher's Training – Final Test- .Net

## NOTES:

1. Please read the business story and all requirements carefully.
2. The mockups provided are just for reference, feel free to use your **creativity**.
3. Deliverables include \*.zip file with your source code and database scripts.  
Format <Name>\_<EmployeeId>\_FinalTest.zip
4. Also provide a “**How to setup**” guide to bring application to running condition.
5. Please go through the evaluation criteria provided at last.

**Duration: 60 - 70 hours**

**Maximum Marks: 100**

## Business Story

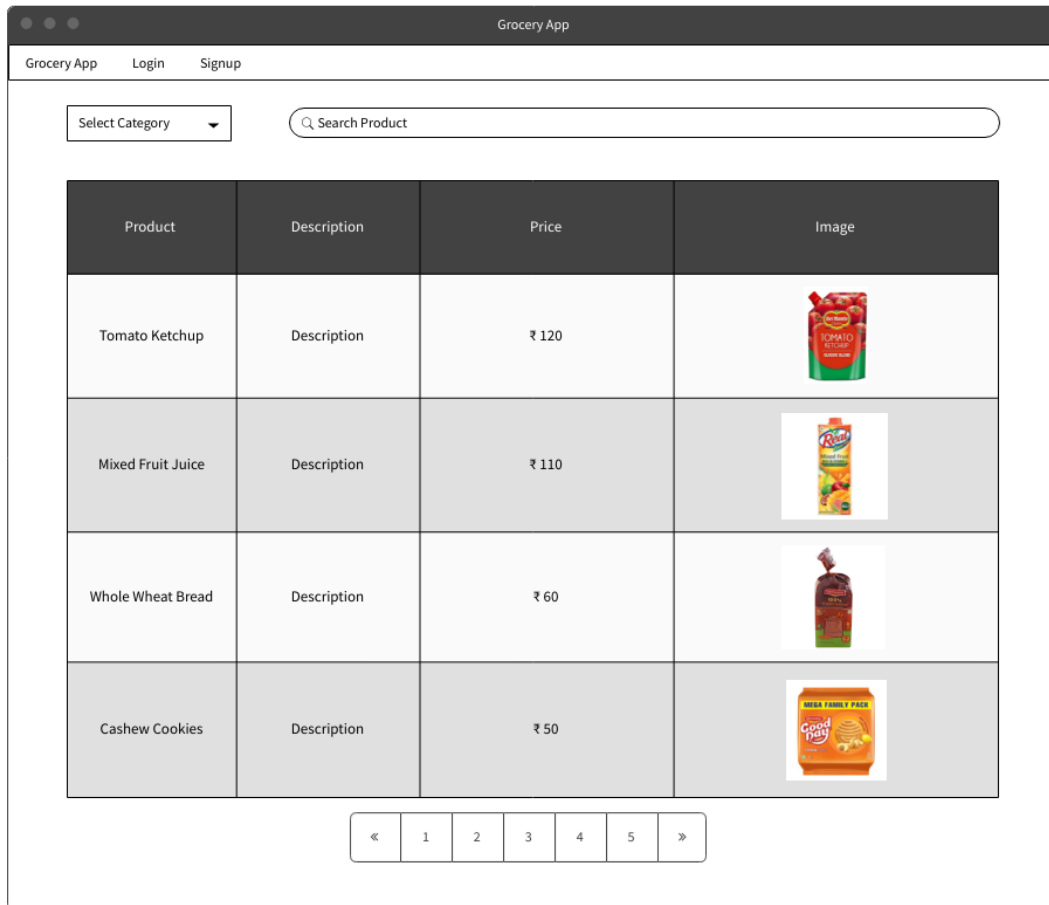
We have been hired to develop an e-commerce application for a small grocery business that sells products online. The business owner wants a simple yet efficient application to showcase the products and enable customers to purchase them. The application is expected to increase sales and improve customer satisfaction.

## Requirements

The application being created should fulfill the following requirements. The validations are mentioned wherever relevant. Please refer mockups for better understanding.

1. Dashboard: Anonymous user view
  - a. Users can view Login and Signup button in the application header.
  - b. Users can view the list of products.
  - c. Application should allow customers to view products by category and search for products based on product name or description.
  - d. Application should support pagination and sorting for list of products.
  - e. Clicking on any product should redirect to the product details page.





Dashboard – Anonymous User

## 2. Signup and Login:

### a. Fields on Signup screen:

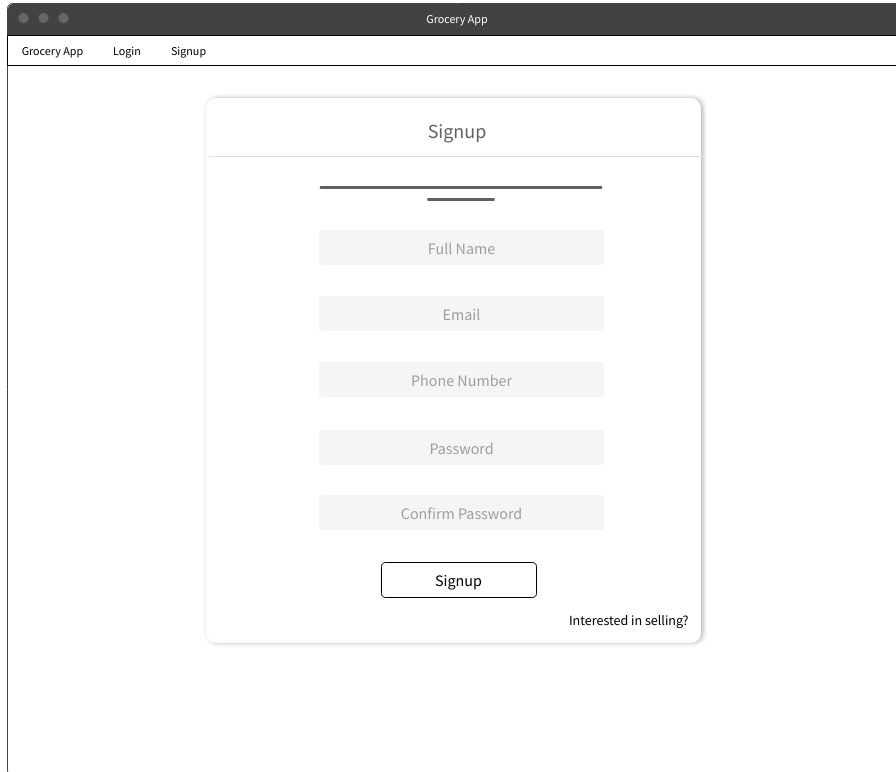
	Validation
Full Name	Max 50-character alphabets only
Email	Must be unique in the system and valid as per standard format.
Phone Number	Valid as per the standard 10 digits format.
Password	Minimum length is 8 and it must have at-least 1 special character, 1 number and 1 alphabet
Confirm Password	should match with password

Note:

- All fields are mandatory.
- Payment module is out of scope for this assignment.

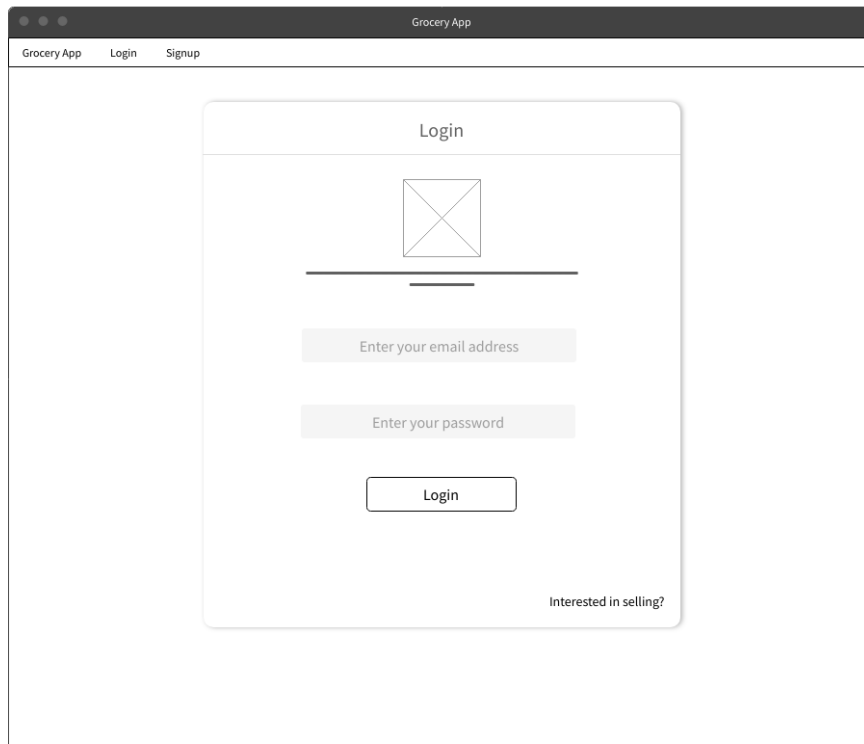
### b. Login: Login should work on providing a registered **email** and **password**. After successful login user lands on the dashboard page.





The screenshot shows a web browser window titled "Grocery App". The navigation bar contains links for "Grocery App", "Login", and "Signup". The main content area features a "Signup" form with the following fields: Full Name, Email, Phone Number, Password, and Confirm Password. A "Signup" button is located below these fields. At the bottom right of the form, there is a link that says "Interested in selling?".

### Signup







The screenshot shows a web browser window titled "Grocery App". The navigation bar contains links for "Grocery App", "Login", and "Signup". The main content area features a "Login" form with a placeholder image (a square with an 'X') above the input fields. The fields are labeled "Enter your email address" and "Enter your password". A "Login" button is located below these fields. At the bottom right of the form, there is a link that says "Interested in selling?".

### Login

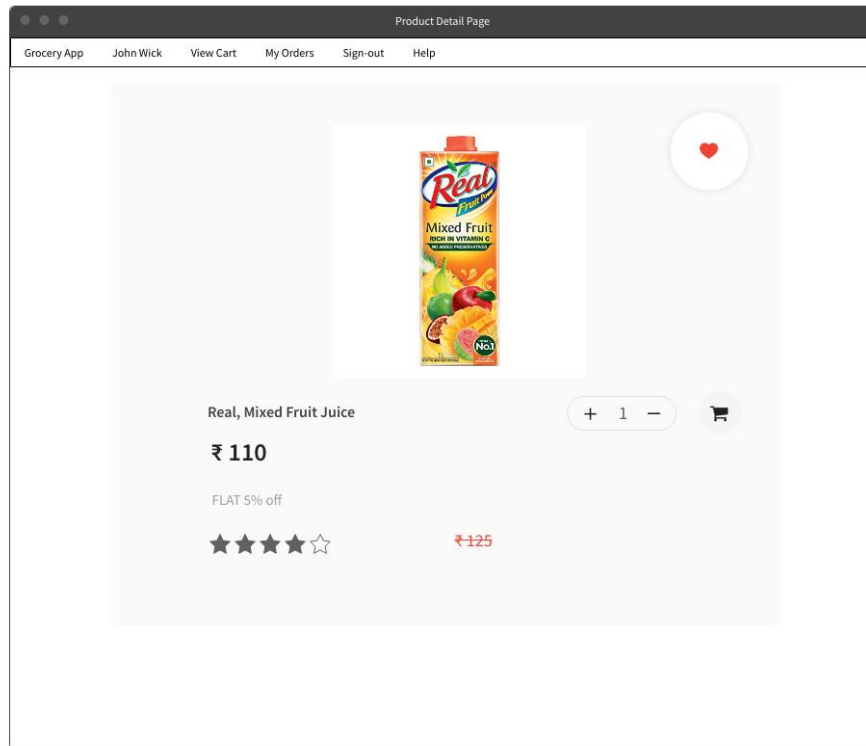


3. Dashboard for logged in user:

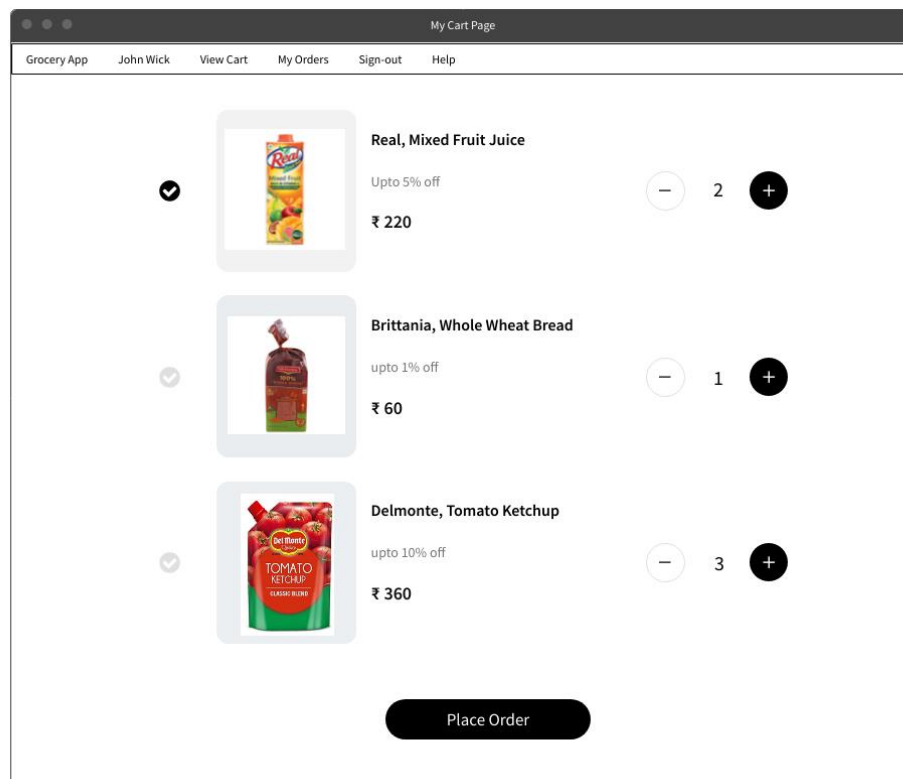
- Users will be able to see Full Name, View Cart, My Orders and Sign-out button in the application header. Login and Signup buttons will be hidden.
- Users can view all the options visible to anonymous users on the dashboard.
- Show Message '**No Items in Cart**' on cart page if the user cart is empty.
- Only logged in users can add products to their cart, view their cart, and remove products from their cart.
- Show **Add to cart** button and **Quantity** dropdown on product detail page. Users will need to choose quantity and click Add button to add the product to the cart.
- Show 'Out of Stock' instead of Add to Cart button if product available quantity is zero.
- Cart page will have a **Place Order** button. Upon clicking on the place order button, generate a unique order id and notify it to user and cart should be refreshed. Handling address and payment is not required. Subsequently update the quantity of the product(s) ordered.
- Users can view their orders using the **My Orders** Page.

Grocery App			
Grocery App	John Wick	View Cart	My Orders Sign-out Help
Select Category	Search Product		
Product	Description	Price	Image
Tomato Ketchup	Description	₹ 120	
Mixed Fruit Juice	Description	₹ 110	
Whole Wheat Bread	Description	₹ 60	
Cashew Cookies	Description	₹ 50	
« 1 2 3 4 5 »			

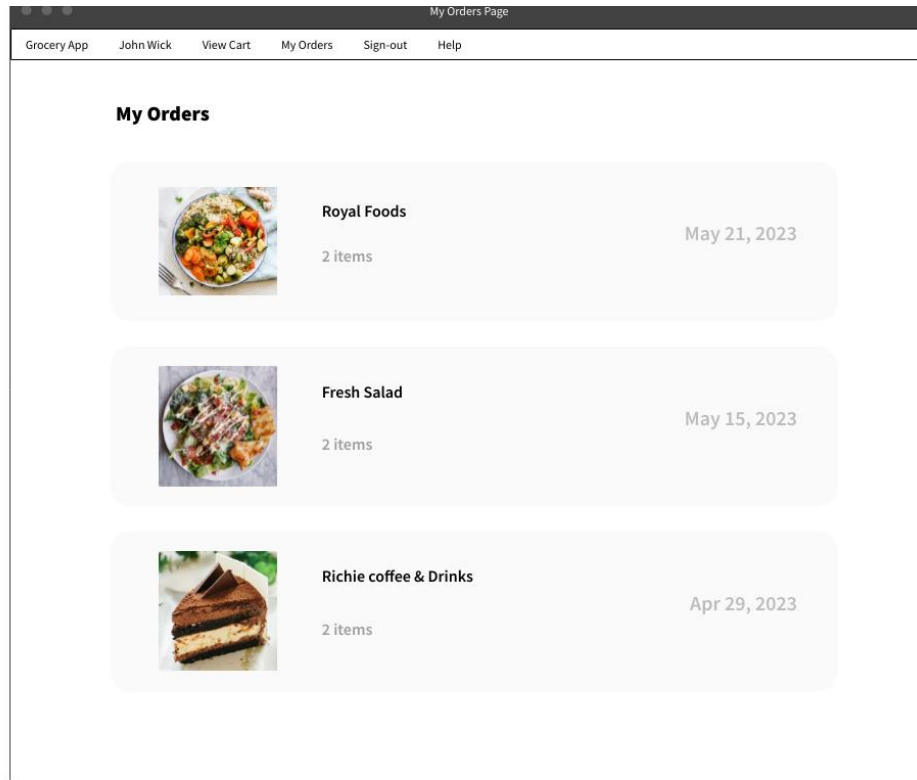
Dashboard – Logged-in User



Product Detail Page



Cart Page



### My Orders Page

#### 4. Dashboard for Admin Users:

- Login with admin credentials will redirect to admin view.
- Application should have support for multiple admins  
(Note: You can assume admin to be identified based on "isAdmin" flag in database)
- After successful login, the admin lands on admin-product-listing page with edit and delete buttons where they can edit/delete product.
- Admin can add new products using the Add Product button in the application header.
- Add new product should ask following inputs from the admin. Design UI accordingly.

Fields	Validation
Product Name	Max 100-character alphanumeric, required
Description	Max 255-character alphanumeric, required
Category	Max 100-character alphanumeric, required
Available Quantity	Numeric, required
Image	JPG or PNG format only, required
Price	Decimal, required
Discount	Decimal, optional
Specification (Weight, Size etc.)	Max 100-character alphanumeric, optional

- Edit should open the same page with all the fields prepopulated in the form and Save button.



Product	Description	Price	Image	Action
Tomato Ketchup	Description	₹ 120		<button>Edit</button> <button>Delete</button>
Mixed Fruit Juice	Description	₹ 110		<button>Edit</button> <button>Delete</button>
Whole Wheat Bread	Description	₹ 60		<button>Edit</button> <button>Delete</button>
Cashew Cookies	Description	₹ 50		<button>Edit</button> <button>Delete</button>

« 1 2 3 4 5 »

Dashboard – Admin User

**Note:**

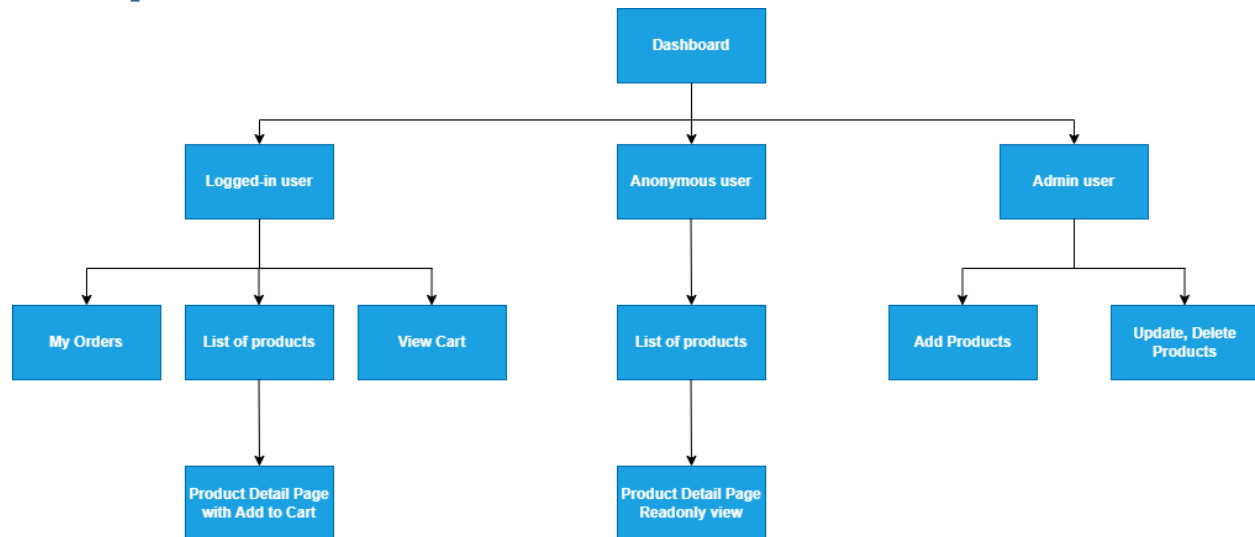
- No need to ask user address or payment details. It is out of scope for this assignment.
- Show toast notifications to users while performing any action.
- Single Image for a product is sufficient.
- Admin users can be added to the system via Seed Method.
- The application should be responsive and work well on desktop and mobile devices.
- The application should have a user-friendly interface that is easy to navigate.
- The application should be scalable and able to accommodate additional features in the future.
- Take relevant assumptions and mention it in the readme file, if any of the requirements are not clear or in case of ambiguity.
- Wireframes attached are just for reference, feel free to enhance User Experience as much as you can.

**Bonus Question**

1. Provide an option to Logged-in users to post reviews for the products. Display these reviews alongside product detail page.
2. Provide an option to Admin User to view five most ordered products in a particular month of a year. Admin should have the ability to choose year and month. Show data for current month and year by-default.



## Sitemap



## Question 1

Create an end-to-end functional application using n-tier architecture to achieve the business story discussed above. To aid your thinking process and to set expectations in terms of output required find listed below the various components that go into making such an application.

1. SQL database
  - a. Should contain tables (if you want procedures – not mandatory) required for the business case.
    - i. Consider good design – Normalization, Constraints etc.
    - ii. Map complete information
2. Data Access Layer
  - a. Should contain classes to help interact with the database.
  - b. Use Entity Framework for database interaction.
  - c. Feel free to use procedures (if you created them above) or use entities alone to complete DAC operations.
3. Business Layer
  - a. Should contain classes to represent business logic to cover all the requirements mentioned in Requirements section.

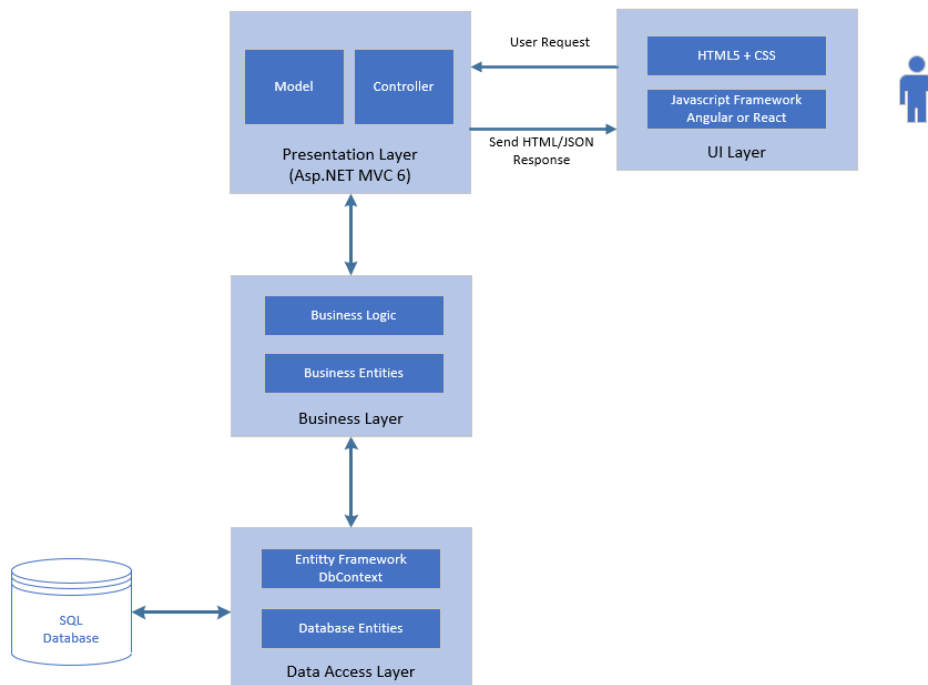
Note - Any common code – DTOs, constants etc., which can be used by any other layer can be made part of Shared Layer (optional).

4. Presentation Layer: Highlight use of Asp.Net MVC 6 And/or .NET Web APIs.
5. UI presentation layer using **Angular or React**, HTML, CSS, JS.
  - a. This should take care of UI validations.
  - b. Make RESTful calls to API controller for data.





The following diagram shows a high-level structure and data flow for the application.



## Technology Stack

Following technologies should be used to create the application:

- UI Layer – HTML5, JS, CSS3, Angular/React
- Presentation Layer– Asp.Net MVC 6, Asp.Net Web API
- Business and Data Layer – .NET, Entity Framework, SQL Server

## Evaluation Criteria

The application will be evaluated against following criteria:

1. Basic application design and data structures usage.
2. Appropriate usage of OOP's and SOLID principles.
3. Functional completeness.
4. **Make sure your application runs with sharing code on other's machine and not just on your machine.**
5. Adhering to coding standards and guidelines. Follow clean coding practices.
6. Extra marks for attempting the Bonus Question.

