Blazor User Form

Web Application

This web application is designed for managing user records on a website. The application includes functionality for creating, reading, updating, and deleting user information. Below is an overview of the key components and features of the application.

User Entity

The `User` entity is defined in the root project within the "User.cs" file. The entity includes properties such as ID, First Name, Middle Name, Last Name, Gender, City, and Occupation.

Database Configuration

The application uses Entity Framework Core for database operations. The `ApplicationDbContext` class, located in the `BlazorAppUserPortal.DataContext` namespace, establishes a connection to a SQL Server database and specifies a table named "Users" for storing user entities. The database connection string is configured in the `appsettings.json` file.

```
"ConnectionStrings":
{
 "DefaultConnection":
"Server=localhost\\SQLEXPRESS;Database=BlazorApp;Trusted_Connection=True;MultipleActiveResultSets=true;TrustServerCertificate=True"
}
```

Dependency Injection and DbContext Registration

The `UserService` class is responsible for CRUD operations on the `User` entity within the Blazor application. It is registered as a transient service for dependency injection. Additionally, the `ApplicationDbContext` is registered with Entity Framework Core to connect to the SQL Server database.

```
// Register the UserService as a transient service
builder.Services.AddTransient<UserService>();

// Register the ApplicationDbContext with Entity Framework Core
builder.Services.AddDbContext<ApplicationDbContext>(item =>
item.UseSqlServer(builder.Configuration.GetConnectionString("DefaultConnection")))
```

User List Component

The `UsersList` component displays a list of users, providing options to add, edit, and delete users. It utilizes the `UserService` for data operations.In the Following, screenshots of each operation are attached.

This screenshot shows the webpage for adding a new user to the system. Users can fill out the form with user details and click the "Save" button.
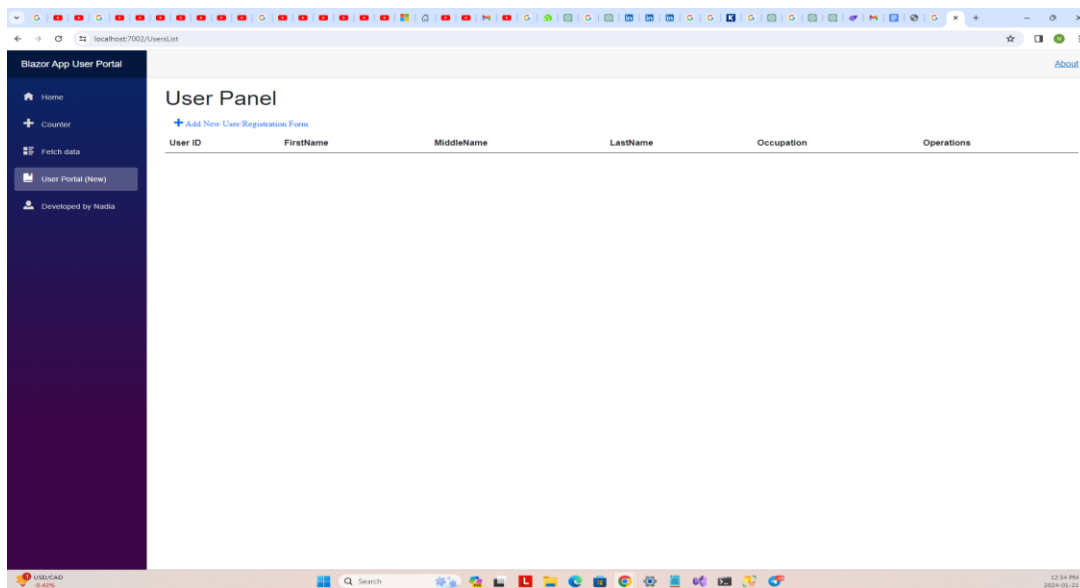

Figure 0-1: The List of User

Add User Page

The `AddUser` page presents a user registration form with fields for First Name, Middle Name, Last Name, Gender, City, and Occupation. The form allows users to save a new user or cancel the operation. The `CreateUser` method triggers the insertion of a new user, while the `Cancel` method navigates back to the user list without saving changes.

Figure 0-2 Add New User to List / Form

Figure 0-3 the List of Users

Delete User Page

The `DeleteUser` page is designed for deleting a user record. It displays user details and a confirmation message. Users can choose to proceed with the deletion or cancel the operation. Upon deletion, the user is navigated back to the user list.
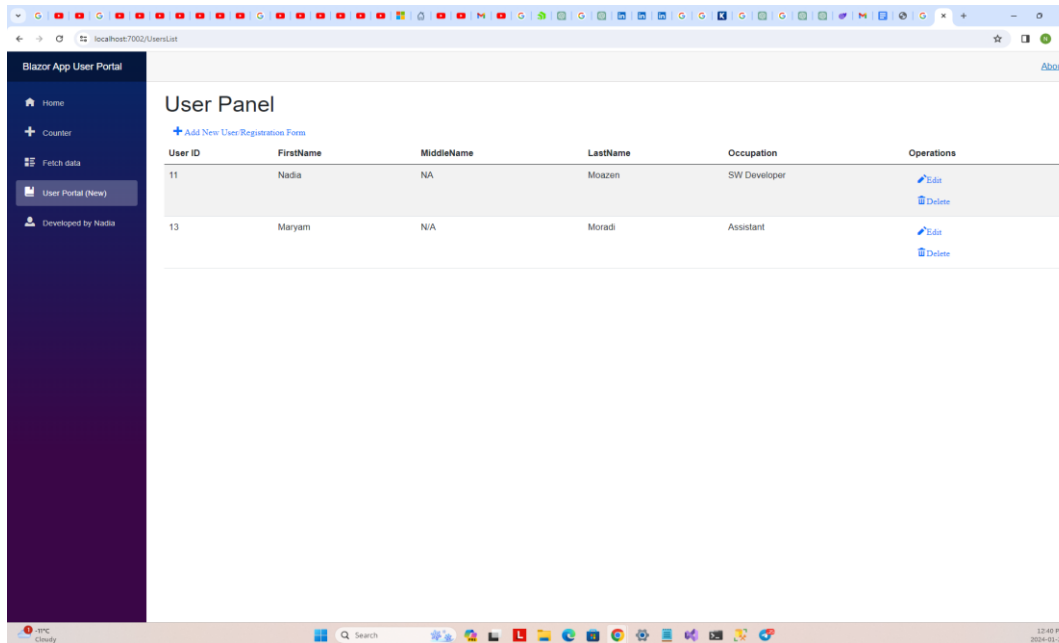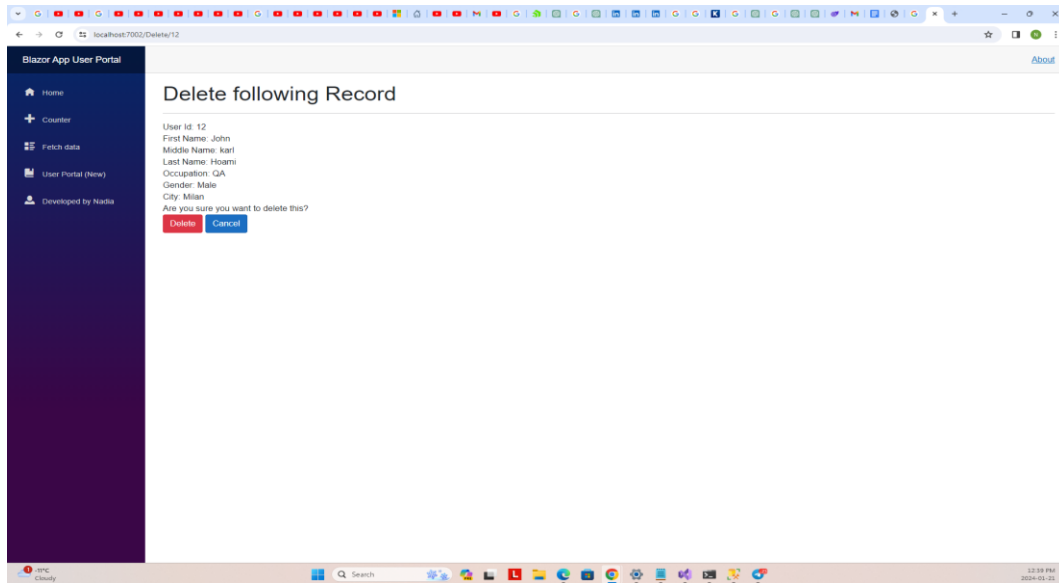
Edit User Page

The `EditUser` component serves as an interface for editing user details. It fetches user details based on the provided ID, allowing users to make changes. The component provides options to update the user information or cancel the operation.

Figure 0-4 The Modification Form/Page

Figure 0-5 The List of User after a Modification operation

Getting Started

Ensure the database connection string in `appsettings.json` is configured correctly.

Run the Entity Framework Core migration to create the database schema.

- Open the Package Manager Console and run `AAdd-Migration InitialMigration` and `Update-Database`.

Build and run the Blazor application.