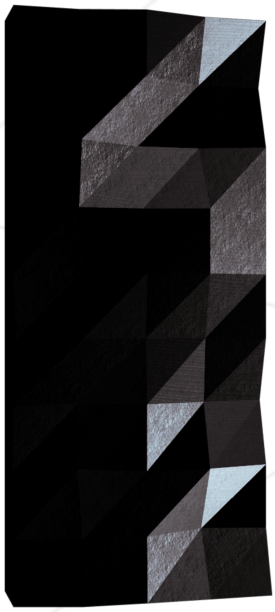

42_active_connect

Using OpenID and Microsoft Technologies

Summary: In this project, you will create a web application that utilizes OpenID for user authentication and authorization

Version: 1.00

Contents

Chapter I

Forewords

Test

Chapter II

Objectives

In this project, your goal is to develop a web application that incorporates OpenID for user authentication and authorization. OpenID is an open standard that allows users to log in to various websites using a single set of credentials. The application will be developed using Microsoft technologies and hosted on a Microsoft Windows-based server.

The main objectives of this project are as follows:

- Implement OpenID authentication: Users should be able to log in to your web application using their OpenID credentials from supported providers such as Google, Microsoft, or Facebook.
- User registration and profile management: Users should have the ability to create an account, update their profile information, and manage their account settings.
- Secure access control: Implement appropriate access control mechanisms to ensure that only authenticated users can access certain parts of the application.
- User roles and permissions: Implement role-based access control to differentiate between regular users and administrators. Administrators should have additional privileges, such as the ability to manage user accounts or moderate content.
- User-generated content: Provide users with the ability to create, edit, and delete their own content, such as articles, comments, or posts.
- Responsive and user-friendly interface: Design and develop a visually appealing and intuitive user interface for a seamless user experience.

The web application should comply with the General Data Protection Regulation (GDPR) in terms of user data management and privacy.

You are required to use the following Microsoft technologies for this project:

- Programming Language: C#.
- Framework: ASP.NET Core.
- Database: Microsoft SQL Server or Azure SQL Database

For the front end you are free to use the framework of your choice. For the back part it is mandatory to use a Microsoft specific technology. It can be a framework or an equivalent.

The web application must be hosted on a Microsoft Windows-based server. You can choose a hosting provider or set up your own server environment.

Additionally, you are required to propose the creation of a deployment automation script using your knowledge of a specific application (e.g., PowerShell, Azure DevOps, or Visual Studio).

Chapter III

Mandatory Part

Your task is to create a fully functional web application that incorporates the features mentioned in the introduction.

Here are the specific requirements:

- User Authentication with OpenID: Implement OpenID authentication in your web application. Users should be able to log in using their OpenID credentials from supported providers.
- User Registration and Profile Management: Provide users with the ability to create an account, update their profile information, and manage their account settings. Ensure that the user registration process complies with the GDPR.
- Secure Access Control: Implement access control mechanisms to ensure that only authenticated users can access certain parts of the application. Unauthenticated users should be redirected to the login page.
- User Roles and Permissions: Implement role-based access control to differentiate between regular users and administrators. Administrators should have additional privileges, such as the ability to manage user accounts or moderate content.
- User-Generated Content: Allow users to create, edit, and delete their own content within the application. This could include posts, articles, comments, or any other relevant content.
- Responsive and User-Friendly Interface: Design and develop a responsive and visually appealing user interface to provide a seamless user experience across different devices.
- GDPR Compliance: Ensure that user data is handled in compliance with the GDPR regulations. Implement mechanisms to obtain user consent, provide data transparency, and allow users to manage their data.

You are free to choose the specific features, design, and additional functionalities for your web application, as long as they meet the requirements mentioned above.

Provide any necessary script regarding the deployment environment, database setup, and dependencies.

III.1 General Data Protection Regulation (GDPR)

If you are not familiar with the General Data Protection Regulation (GDPR), it is essential to understand its principles and implications, especially regarding user data management and privacy. The GDPR is a regulation that aims to protect the personal data and privacy of individuals within the European Union (EU) and the European Economic Area (EEA). It sets out strict rules and guidelines for organizations on how they should handle and process personal data.

To gain a better understanding of the GDPR and its requirements, it is highly recommended to visit the official website of the European Commission on data protection[1]. This website provides comprehensive information about the GDPR, including its principles, objectives, and user rights. It also offers additional resources to delve deeper into the topic and ensure compliance with the regulation.

If you are unfamiliar with the GDPR, please take the time to visit the provided link and familiarize yourself with the regulations before proceeding with this project.

If you are not living in a country where GDPR rules apply, you may have similar data privacy laws. For convenience purpose on this project, you still need to stick with the GDPR rules. The idea in this project is to make you aware of such kind of topic as a developer on an international website using personal data.

[1] https://commission.europa.eu/law/law-topic/data-protection/data-protection-eu_en

Chapter IV

Submission and peer-evaluation

Submit your project by pushing it to your Git repository as usual. Only the contents of your repository will be evaluated during the defense. Make sure to double-check the names of your folders and files to ensure they are correct.

Additionally, propose the creation of a deployment automation script using your knowledge of a specific application (e.g., PowerShell, Azure DevOps, or Visual Studio).

IV.1 Project Structure (Part 1)

```
project/
└── app/
    ├── Controllers/
    │   ├── HomeController.cs
    │   ├── UserController.cs
    │   └── ...
    ├── Models/
    │   ├── User.cs
    │   ├── Post.cs
    │   └── ...
    ├── Views/
    │   ├── Home/
    │   │   ├── Index.cshtml
    │   │   ├── About.cshtml
    │   │   └── ...
    │   ├── User/
    │   │   ├── Profile.cshtml
    │   │   ├── Edit.cshtml
    │   │   └── ...
    │   └── Shared/
    │       ├── _Layout.cshtml
    │       ├── _Navbar.cshtml
    │       └── ...
    ├── Services/
    │   ├── AuthService.cs
    │   ├── UserService.cs
    │   └── ...
    ├── appsettings.json
    ├── Program.cs
    └── Startup.cs
```

IV.2 Project Structure (Part 2)

```
project/
├── migrations/
│   ├── 20220101000000_CreateTables.cs
│   ├── 20220101000001_AddFieldsToUser.cs
│   └── ...
├── scripts/
│   ├── deploy.ps1
│   └── ...
├── wwwroot/
│   ├── css/
│   │   ├── main.css
│   │   └── ...
│   ├── js/
│   │   ├── main.js
│   │   └── ...
│   └── images/
│       ├── logo.png
│       └── ...
├── Readme.md
└── project.sln
```