Assignment 5 – Membership Web Application

Name: Jaskirat Singh

1. Introduction

This project demonstrates the required components for Assignment 5 of the course. The goal was to build a functional ASP.NET Web Forms application that communicates with a WCF service, processes data stored in an XML file, and includes local components such as state management and DLL functionality. The application simulates a membership system where the user can enter a username and check if it already exists in the system.

2. Application Overview

This web application includes:

- A username validation interface built using **ASP.NET Web Forms**
- A **WCF service** that checks whether the username exists in Members.xml
- A **DLL component** implementing SHA-256 hashing functionality
- Use of **cookies** and **Global.asax session logic**
- XML-based storage for usernames

The application provides a simple "Try-It" interface where the user can test the service and system components.

3. Components Summary Table

Component	**Type**	**Purpose**
Default.aspx	ASPX Page	User interface for username validation and Try-It testing
MembershipService.svc	WCF Service	Reads the XML file and checks username availability
PhoenixSecurity.dll	DLL Library	Implements secure hashing using SHA-256
Members.xml	XML File	Stores existing usernames
Global.asax	Application File	Tracks visits using Application state
Cookies	State Storage	Saves and retrieves user information across sessions

4. Testing Instructions

To verify the assignment functionality:

1. Run the project using IIS Express.
2. Open Default.aspx in the browser.
3. Enter a username into the input field.
4. Click the "Check Availability" button.
5. The application will display one of the following messages:
    - "✓ Username is available."
    - "❌ Username already exists."
6. Verify Global state and cookie functionality if included.

5. Result Summary

The system successfully integrates:

- WCF communication
- XML-based persistent storage
- Local component hashing module
- ASP.NET GUI design
- State behavior using cookie and application events

All required components for Assignment 5 function as expected.

6. Conclusion

This assignment demonstrates the integration of a service-oriented architecture with ASP.NET Web Forms. The application successfully validates user input through a backend service and maintains local and persistent data storage.

The submitted system meets the Assignment 5 requirements and provides a foundation for extending the functionality in Assignment 6.