# Test Plan: [Blazor WEB Project Users and Security]

I. Introduction

1. Purpose of the Test Plan.

* The purpose to this test plan is to make sure that it is cross-platform compatible and the login component uses 2-factor when logged in and is safe for cross-site scripting (XSS).

1. Scope – [what this test plan covers].

* The scope of this test plan covers making my blazor cross-platform compatible and registration of new user, login with 2-factor authentication and save for cross-site scripting.

1. Features to be Tested 1. [feature to be tested], 2. [another feature to be tested], and so on…

* Cross-site scripting
* Register new user
* Login component
* 2-factor authentication

II. Test Environment

1. [Unix]
2. Enabled/Install WSL (might not work on Home Editions)
3. Ubuntu
4. .NET 8.0
5. Mock Database (SQLite)
6. [Windows]
7. Should be able to run out of the bag.

III. Test Cases *(can be* ***security test****,* ***compatibility test****,* ***performance test*** *and so on, choose from sprint backlog what can be testet…)*

1. Cross-site scripting (**Security test**)

To test if an input field is safe for cross-site scripting:

1. In the input field try to write this line : <script>alert('XSS');</script>
2. Click on submit button.
3. Verify that the script you wrote in the input field is not executed. It should be rendered as plain text.
4. **Expected Result**: &lt;script&gt;alert('XSS');&lt;/script&gt;
5. Register a new user (**Functional test**)

To test if registration process of a new user is working correctly.

1. Navigate to Register page
2. Enter your email and password and click submit.
3. Scan QR with authenticator app to register 2-factor login for user.
4. Click on confirm registration of user.
5. **Expected Result**: Successful registration for valid email and password.   
   It should also show error messages for invalid input as incorrect email format or weak password.
6. 2-Factor authentication (**Security test**)

To test if 2-factor authentication is working properly after successfully registration of new user:

1. Navigate to Login page
2. Log in with valid email and password.
3. After logging in, verify that it promps with a 2-factor.
4. Enter an invalid 2-factor code and verify the error message.
5. Enter the valid 2-factor code from the authenticator app and user is redirected to Home Page that tells you ‘You are logged in!’
6. **Expected Result**: User is prompted 2-factor authentication after logging in.

IV. Conclusion

All functional and security test were completed. The login component were triggered a 2-factor authentication as expected and user input in input field is safe for XSS attacks.

**Project Backlog for Blazor Web App**

1. **Create Blazor Web App**: Set up a new project using Blazor to develop a user-friendly web application.
2. **Set Up Database(CodeBefore)**: Use Entity Framework to create and manage a database for storing user information and other necessary data.
3. **User Registration**: Implement a feature that allows users to sign up and create an account.
4. **User Authentication**: Enable users to log in securely, ensuring that only registered users can access certain parts of the app.
5. **Two-Factor Authentication**: Add an extra layer of security by implementing QR code-based two-factor authentication for users during login.