SRS – Software Requirements Specification: WebApplication

BlankFactor

Contents

Document history

Ver.	Date	Resp.	Description
1.0	2020-12-12	MH	First baseline version

1 Introduction

This document outlines the requirements for the “WebApplication”, a system featuring log-in functionality on a web server solution. The intention is that the system should provide basic authentication functionality and later be extended to include more useful features, such as a time reporting system.

The BlankFactor dev team develops the system.

2 Reference documents

There are no reference documents in this version.

3 Background and goals

3.1 Main goals

The system's primary goal is to provide login and logout functionality on an IIS server, which can serve as a basis for developing more advanced systems, such as time-reporting systems.

3.2 Actors and their objectives

The following main actors use the system.

User: A user can log in to the system and log out of the system. The main objective is for users to log in and log out easily.

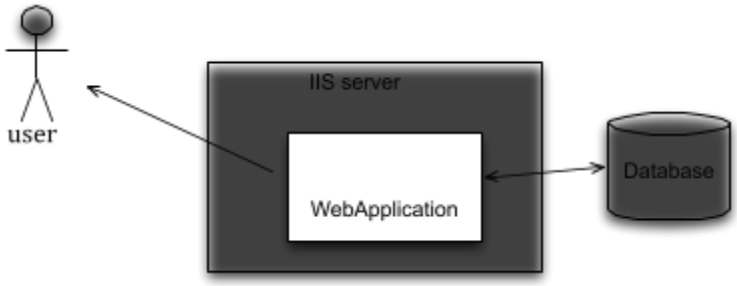

Figure 1: Context diagram

4 Terminology

Functionality selection page	A page with basic system functionality, visible only to logged-in users.
login-state	Information whether a user of the system is logged in or logged out.
password	A password used for authentication.
server session	ISS server entity for storing information in user sessions.
username	The name by which the user is known to the system.

5 Context diagram

A context diagram is displayed in Figure 1.

6 Functional requirements

6.1 Login and logout

Requirement 6.1.1. For one user, the login state can be either logged-in or not.

Requirement 6.1.2. The system shall store the login state in a server session.

Scenario 6.1.2.1. User login Precondition:
The user is not logged in.

1. The user accesses the system.
2. The user must provide a username and password on a login page.
3. The user provides a correct username and password.
4. The user is logged in, and a functionality page is displayed.

Requirement 6.1.3. The system should support scenario 6.1.2.1. **Scenario 6.1.3.1.** User logout
Precondition:
The user is logged in.

1. The user accesses the system.
2. The user is presented with a page that includes a logout link.
3. The user requests to be logged out.
4. The user is logged out and informed about this through the next page.

Requirement 6.1.4. The system should support scenario 6.1.2.1.

Scenario 6.1.4.1. Failed user login Precondition:
The user is not logged in.

1. The user accesses the system.
2. The user must provide a username and password on a login page.
3. The user provides a username and password not registered in the database.
4. The user is not logged in, and an error message is shown. The user is prompted to re-enter their username and password.

Requirement 6.1.5. The system should support scenario 6.1.4.1.

Requirement 6.1.6. When a user accesses the system and is not logged in, they should be prompted to enter a username and password. No other information should be provided to the user.

Requirement 6.1.7. When a user submits a username and password, they should be compared to the list of users. If the user is granted access to the system, the server state should be changed to "logged in," and the functionality page should be shown.

Requirement 6.1.8. All pages shown to a logged-in user should include a log-out functionality, e.g., a button for logging out of the system.

Requirement 6.1.9. If a logged-in user is inactive for more than 20 minutes, they should be logged out and must log in again before continuing to use the system.

Requirement 6.1.10. If a user clicks the 'Remember me' checkbox before logging in, they should be logged in. The next time the user attempts to log in, they should not need to enter their username and password.

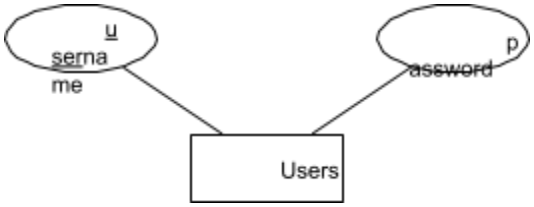

Figure 2: ER diagram

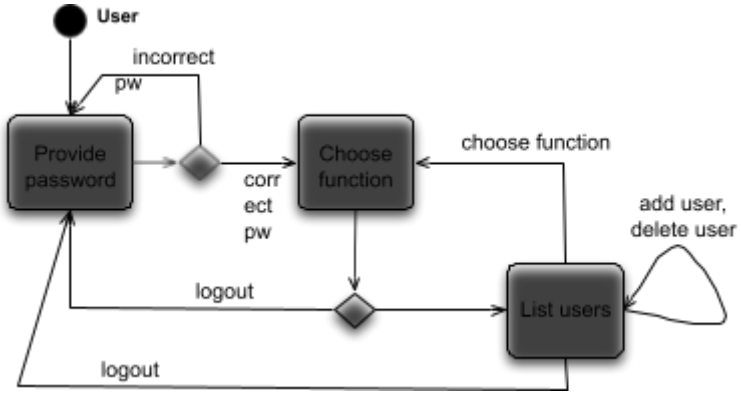

Figure 3: Activity diagram showing the most important sequences for an administrator

6.2 Data

Requirement 6.2.1. Usernames should consist of 5-10 characters, with ASCII (decimal) values 48-57, 65-90, and 97-122 allowed.

Requirement 6.2.2. Usernames should be unique for each user.

Requirement 6.2.3. Passwords should consist of 6 characters, with ASCII (decimal) values of 97-122 allowed.

Requirement 6.2.4. The data model for storing user information is displayed in Figure 2.

6.3 General requirements

Requirement 6.3.1. The system should be able to handle and detect incorrect input (based on Section 6.2). No incorrect input should be able to crash the system or corrupt the data.

7 Quality requirements

7.1 Maintainability

Requirement 7.1.1. Knowledge corresponding to the knowledge goals of EDA016, along with a basic understanding of SQL, should be sufficient to understand, maintain, and further develop the system.

7.2 Performance

Requirement 7.2.1. When the system is used, the response to any request should, in at least 95% of all cases, be given within 1.0 s.

8 Project requirements

8.1 Development environment

Requirement 8.1.1. The system should be developed for the IIS server.

Requirement 8.1.2. The system should be developed in C# .NET.

Requirement 8.1.3. An MSSQL database should be used to store data saved between sessions.