**Employee Login**

**System**

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **SI.No** | **Content** | **Page No.** |
| **1** | Introduction  **1.1** Problem Statement  **1.2** Description | **3** |
| **2** | Requirements - Technical specification | **4** |
| **3** | Test Plan | **5** |
| **4** | Test Cases | **5** |
| **5** | Expected results | **5** |
| **6** | Conclusion | **6** |

**1.** **INTRODUCTION**

**1.1 Problem Statement:**

A login is a set of credentials used to authenticate a user. Most often, these consist of a [username](https://techterms.com/definition/username) and [password](https://techterms.com/definition/password). However, a login may include other information, such as a PIN number, [passcode](https://techterms.com/definition/passcode), or [passphrase](https://techterms.com/definition/passphrase). Some logins require a [biometric](https://techterms.com/definition/biometrics) identifier, such as a fingerprint or retina scan. Logins are used by [websites](https://techterms.com/definition/website), computer [applications](https://techterms.com/definition/application), and mobile [apps](https://techterms.com/definition/app). They are a security measure designed to prevent unauthorized access to confidential [data](https://techterms.com/definition/data). When a login fails (i.e, the username and password combination does not match a user account), the user is disallowed access.

Many systems block users from even trying to log in after multiple failed login attempts.

**1.2 Description:**

At a basic level, logins make user accounts possible. Most systems require unique usernames, which ensures every user's login is different. On a more advanced level, logins provide a security layer between unsecured and secured activity. Once a user logs in to a secure website, for example, all data transfers are typically [encrypted](https://techterms.com/definition/encryption). This prevents other systems from viewing or recording the data transferred from the server.

**2.** **REQUIREMENTS**

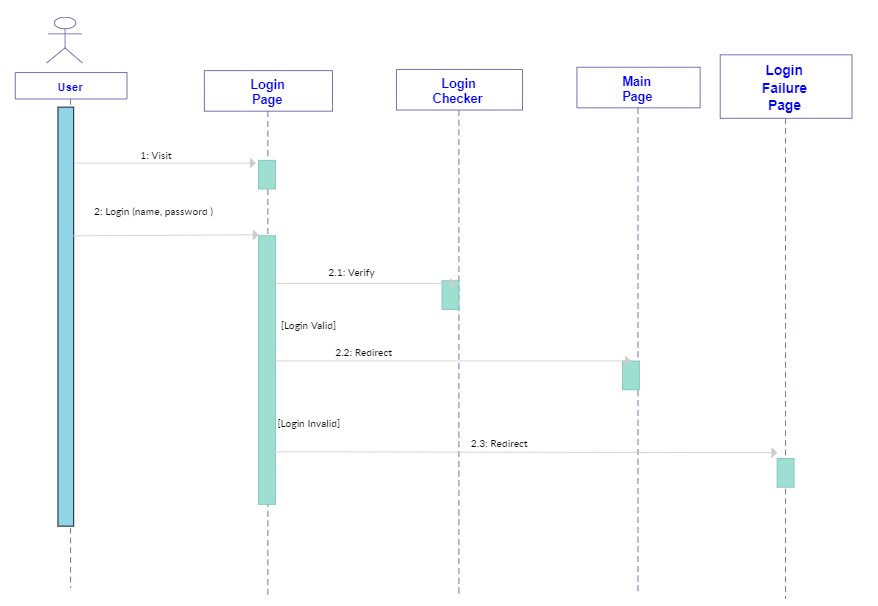
**TECHNICAL SPECIFICATION:**

**1.** Code blocks version 17.12

**2.** GNU GCC compiler

**3.** Windows

**UML DIAGRAM:**

****

**TEST PLAN AND TEST CASES**

**TEST PLAN:**

|  |  |
| --- | --- |
| **SL .No** | **Test Objective** |
| **1** | Check if the program menu displays all options correctly. |
| **2** | Check if the admin login authenticates the user to enter a valid name. |
| **3** | Check if the admin login authenticates the user to enter a valid password. |
| **4** | Check if the user enters invalid information. |
| **5** | Check if the user exceeded maximum no.of Login attempts. |

**TEST CASES:**

|  |  |
| --- | --- |
| **Test Case** | **Expected Output** |
| Displays all options | PASS |
| Valid name | PASS |
| Valid password | PASS |
| Exceeded maximum no.of Login attempts | PASS |

**CONCLUSION**

Understanding user authentication is crucial because it’s a key step in the process that keeps unauthorized users from gaining access to sensitive information. A strengthened authentication process ensures that User A only has access to the information they need and can’t see the sensitive information of User B. That’s why it’s so important that your organization is not the next on that list of victims. In order to prevent such a situation, it’s a good idea to invest in high-quality authentication tools to help you secure your website and protect it from potential breaches.