# SAMS Project

## *Test Plan*

*Thomas Travers*
*Grantham University*
*CS406*

**SAMS Project**

**Thomas Travers**

Test Plan

---

*1. Select one use case in the SAMS project, and follow steps on pages 519 -521 in the textbook to generate test cases.*

*i. Submit:*

*·1 The expanded use case, like figure 20.12*

*·2 A table identifying input values for use case testing, like figure 20.13*

*·3 Test case generation table, like figure 20.14*

*·4 Use case based test data, like figure 20.15*

*2. Follow guidance on pages 526-532 to create a test plan for the SAMS website.*

*i. A test plan should include:*

*·1 Test objectives*

*·2 Types of tests*

*·3 Test methods and techniques*

*·4 Test cases—Include just the test case created in 1*

*·5 Test coverage criteria*

*·6 Documents needed*

*·7 Required resources*

*·8 Effort estimation and schedule*

***1. Use Case:***

UC04. Login (Actor: Student, System: SAMS) This use case allows a registered student to login to the system.

UC04. Login | 34 hours | I chose 34 because this task requires us to design a database for our

project. The database must have carefully designed tables to prevent data corruption. It might be beneficial to also add LDAP as the authentication mechanism as this standard can be used for other applications on campus as well including campus email, student portal access, thus limiting account creations. If this system is already implemented, we can *probably save a few hours by utilizing the existing LDAP system.*

### *1.1 Expanded Use Case*

| **Actor: Existing User** | **System: SAMS Web Login** |
|---|---|
| | 0. System displays homepage with login / register |
| 1. User clicks login link. | 2. System displays login form |
| 3. User fills in login ID & password, then clicks login | 4. System verifies the login ID, password and<br>4.1 displays student portal page or<br>4.2 displays error message and stays on login page. |

### *1.2 Identify Input Values*

| **Input Element** | **Type** | **Value Specification** | **Valid** | **Invalid** | **Exceptional Cases** |
|---|---|---|---|---|---|
| Login ID | String | Length must be between 8 to 20 characters | Login ID meets input validation criteria and is a registered LDAP user | Login ID does not meet acceptable criteria or is not an LDAP user | Length of 0 |
| Password | Password | Length must be between 8 to 20 characters | Password meets input validation criteria and its hash is validated with associated user | Login password does not meet input criteria or is not associated with the user | Length of 0 |

### *1.3 Test Case Generation Table*

| **Test Case** | **Login ID** | **Password** | **Expected Outcome** |
|---|---|---|---|
| 1 | Valid | Valid | display student portal |
| 2 | Valid | Invalid | display error and stay on login page |
| 3 | Valid | Exceptional | display error and stay on login page |
| 4 | Invalid | Valid | display error and stay on login page |
| 5 | Exceptional | Valid | display error and stay on login page |
| 6 | Exceptional | Exceptional | display error and stay on login page |

---

### 1.4 Use case based test data

| Test Case | Login ID | Password | Expected Outcome |
|---|---|---|---|
| 1 | johndoe@myuniversity.edu | j4h6PcZ9xd sSnx5e | display student portal |
| 2 | johndoe@myuniversity.edu | pass | display error and stay on login page |
| 3 | johndoe@myuniversity.edu | VHhQkt8Tu 9ZyYDJGD 7aUb | display error and stay on login page |
| 4 | johndoe | vBHxZ3gM a5hU3sAa | display error and stay on login page |
| 5 | VHhQkt8Tu9ZyYDJGD7aUbVHhQkt8Tu9ZyYDJGD7aUbVHh Qkt8Tu9ZyYDJGD7aUb@myuniversity.edu | hVpQj75K mUJWttxZ | display error and stay on login page |
| 6 | VHhQkt8Tu9ZyYDJGD7aUbVHhQkt8Tu9ZyYDJGD7aUbVHh Qkt8Tu9ZyYDJGD7aUb@myuniversity.edu | a | display error and stay on login page |

***2.1 Test Objectives***

*The objective of the tests are to validate user input so that we can ensure effective and secure login practices.*

***2.2 Types of tests***

*We will be using user validation input testing.*

***2.3 Test methods and techniques***

*For this, we can use HttpUnit to submit our tests. With this, we can automate most of the tasks. We know our inputs and our expected outputs. If our inputs and expected outputs don't match, we will know that we have a problem in our code.*

***2.4 Test cases***

| Test Case | Login ID | Password | Expected Outcome |
|---|---|---|---|
| 1 | Valid | Valid | display student portal |
| 2 | Valid | Invalid | display error and stay on login page |
| 3 | Valid | Exceptional | display error and stay on login page |

| | | | |
|---|---|---|---|
| 4 | Invalid | Valid | display error and stay on login page |
| 5 | Exceptional | Valid | display error and stay on login page |
| 6 | Exceptional | Exceptional | display error and stay on login page |

***2.5 Test coverage criteria***

*The tests will cover valid, invalid, and exceptional input of all user input fields. All fields are required to pass testing in order to meet requirements*

***2.6 Documents needed***

*We will require documents for use cases as well as UML models of the SAMS project. These will be useful for drafting our test case scenarios.*

***2.7 Required resources***

*We will require access to the current LDAP system employed on campus to create a functional dummy account to test our SAMS project.*

***2.8 Effort estimation***

*Since most of the testing can become automated and implemented into the other development phases of the project, I would estimate no more than an hour or two to write / implement tests for UC04.*