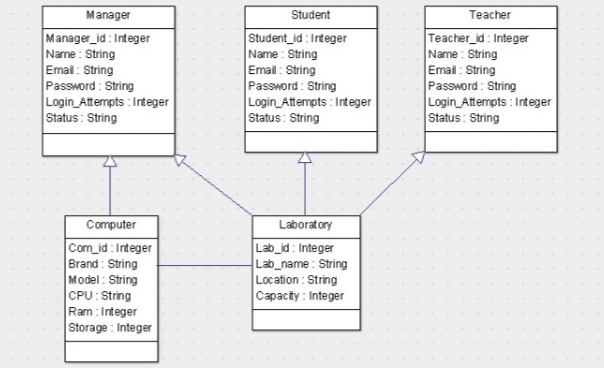


|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | | |
|  | | Domain Model & Sequence Diagram | | | | |  | |
|  |  | | | | | | |  |
|  | | | |  |  | | | |
|  | | | | Zohaib Zafar (Lead) Muhammad Talha Abdul Rehman |  | | | |
|  | | | | Date17 March 2025Course titleSoftware ConstructionTeacher’s NameDr. Onaiza Maqbool |  | | | |
|  | | |  | | |  | | |

**Domain Model Description**

The domain model provided represents the core entities and their attributes in the Laboratory Management System. This system is designed to manage laboratories, computers, and users (Manager, Student, and Teacher) efficiently.



**Domain Model Overview:**The domain model consists of the following key entities:

1. Manager
2. Student
3. Teacher
4. Computer
5. Laboratory

Each entity has specific attributes that define its properties and relationships within the system. The model focuses on managing user roles, laboratory resources, and computer hardware, ensuring a structured and organized approach to system operations.

**Entity Descriptions:**

**1. Manager**

Attributes:

Manager\_id (Integer): A unique identifier for the manager.

Name (String): The full name of the manager.

Email (String): The email address of the manager.

Password (String): The password for the manager's account.

Login\_Attempts (Integer): Tracks the number of failed login attempts.

Status (String): Indicates the account status (e.g., Active, Blocked).

Role:

The manager oversees the laboratories, manages computer resources, and handles user requests (e.g., account unblocking, password resets).

**2. Student**

Attributes:

Student\_id (Integer): A unique identifier for the student.

Name (String): The full name of the student.

Email (String): The email address of the student.

Password (String): The password for the student's account.

Login\_Attempts (Integer): Tracks the number of failed login attempts.

Status (String): Indicates the account status (e.g., Active, Blocked).

Role:

Students can log in to the system, access laboratory resources, and submit complaints or requests.

**3. Teacher**

Attributes:

Teacher\_id (Integer): A unique identifier for the teacher.

Name (String): The full name of the teacher.

Email (String): The email address of the teacher.

Password (String): The password for the teacher's account.

Login\_Attempts (Integer): Tracks the number of failed login attempts.

Status (String): Indicates the account status (e.g., Active, Blocked).

Role:

Teachers can log in to the system, access laboratory resources, and submit complaints or requests.

**4. Computer**

Attributes:

Com\_id (Integer): A unique identifier for the computer.

Brand (String): The brand of the computer (e.g., Dell, HP).

Model (String): The model of the computer.

CPU (String): The processor type of the computer.

Ram (Integer): The amount of RAM in the computer (in GB).

Storage (Integer): The storage capacity of the computer (in GB).

Role:

Represents a physical computer in a laboratory. Each computer has specific hardware specifications and can be associated with software installations and complaints.

**5. Laboratory**

Attributes:

Lab\_id (Integer): A unique identifier for the laboratory.

Lab\_name (String): The name of the laboratory.

Location (String): The physical location of the laboratory.

Capacity (Integer): The maximum number of computers the laboratory can accommodate.

Role:

Represents a physical lab containing multiple computers. Laboratories are managed by the manager and used by students and teachers.

**Relationships Between Entities**

Manager:

Manages Laboratory (1-to-Many).

Handles account-related requests (e.g., unblocking accounts, resetting passwords).

Student and Teacher:

Interact with Computer (Many-to-Many).

Submit complaints or requests related to computers.

Laboratory:

Contains Computer (1-to-Many).

Computer:

Belongs to one Laboratory (Many-to-1).

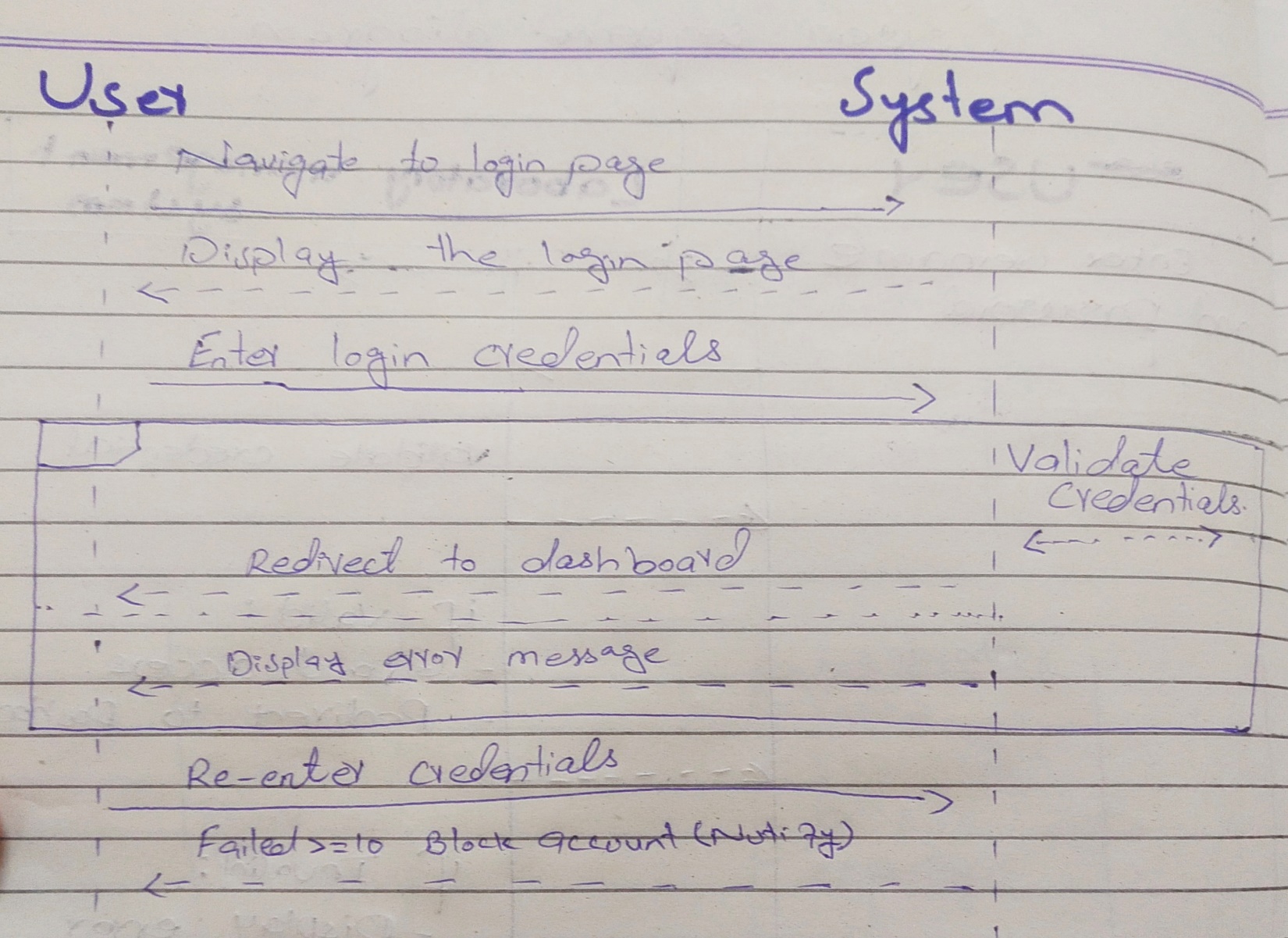
Can be associated with multiple users (Students and Teachers).

**Name: Zohaib Zafar**

**Reg. No: 04072312049**

**System Sequence Diagram (Login Use Case)**

The sequence diagram describes the steps involved in the Login Use Case, starting from the user navigating to the login page and ending with either successful access to the dashboard or an error message due to invalid credentials. The diagram also includes a mechanism to block the account after multiple failed login attempts.



**Steps in the Login Use Case:**

**Navigate to Login Page:**

The user navigates to the login page of the system.

The system displays the login interface, prompting the user to enter their credentials.

**Enter Login Credentials:**

The user enters their username and password in the provided fields.

The user clicks the Login button to submit the credentials.

**Validate Credentials:**

The system validates the entered credentials against the stored data in the database.

**If the credentials are valid:**

The system grants access to the user.

The user is redirected to their respective dashboard (based on their role: Manager, Student, or Teacher).

**If the credentials are invalid:**

The system displays an error message (e.g., "Invalid username or password").

The user is prompted to re-enter their credentials.

**Failed Login Attempts:**

If the user exceeds a predefined number of failed login attempts (e.g., 3 or 5 attempts):

The system blocks the account for security reasons.

The user is notified that their account has been blocked.

The user must contact the Manager to unblock the account or reset the password.

**Key Components of the Sequence Diagram:**

**User:**

Initiates the login process by navigating to the login page and entering credentials.

Receives feedback from the system (e.g., access granted, error message, or account blocked).

**System:**

Displays the login page and handles the validation of credentials.

Redirects the user to the dashboard upon successful login.

Displays error messages for invalid credentials.

Tracks failed login attempts and blocks the account if necessary.

**Alternate Scenarios:**

**Forgot Password:**

If the user forgets their password, they can click a Forgot Password link on the login page.

The system prompts the user to enter their email address.

The system sends a password reset link to the user's email.

**Account Blocked:**

If the account is blocked, the user must contact the Manager to unblock it.

The Manager can reset the password or unblock the account manually.

**User Authentication:**

Ensures that only authorized users (Manager, Student, Teacher) can access the system.

**Security Mechanisms:**

Tracks failed login attempts to prevent brute-force attacks.

Blocks accounts after multiple failed attempts to enhance security.

**User Feedback:**

Provides clear feedback to the user (e.g., successful login, invalid credentials, account blocked).

**Role-Based Access:**

Redirects users to their respective dashboards based on their role (Manager, Student, Teacher).

**Name: Abdul Rehman**

**Reg. No: 04072312041**

**System Sequence Diagram ("Manage Hardware Specification" Use Case)**

The Manage Hardware Specification use case involves the Manager updating or viewing the hardware specifications of computers in the laboratory. Below is the sequence diagram and a detailed description of the interactions and steps involved in this use case.

**Actors:**

Manager: The user responsible for managing hardware specifications.

System: The Laboratory Management System.

**Steps in the Sequence Diagram:**

Manager navigates to the Manage Hardware Specification page.

System displays the list of computers and their current hardware specifications.

Manager selects a computer to update its hardware specifications.

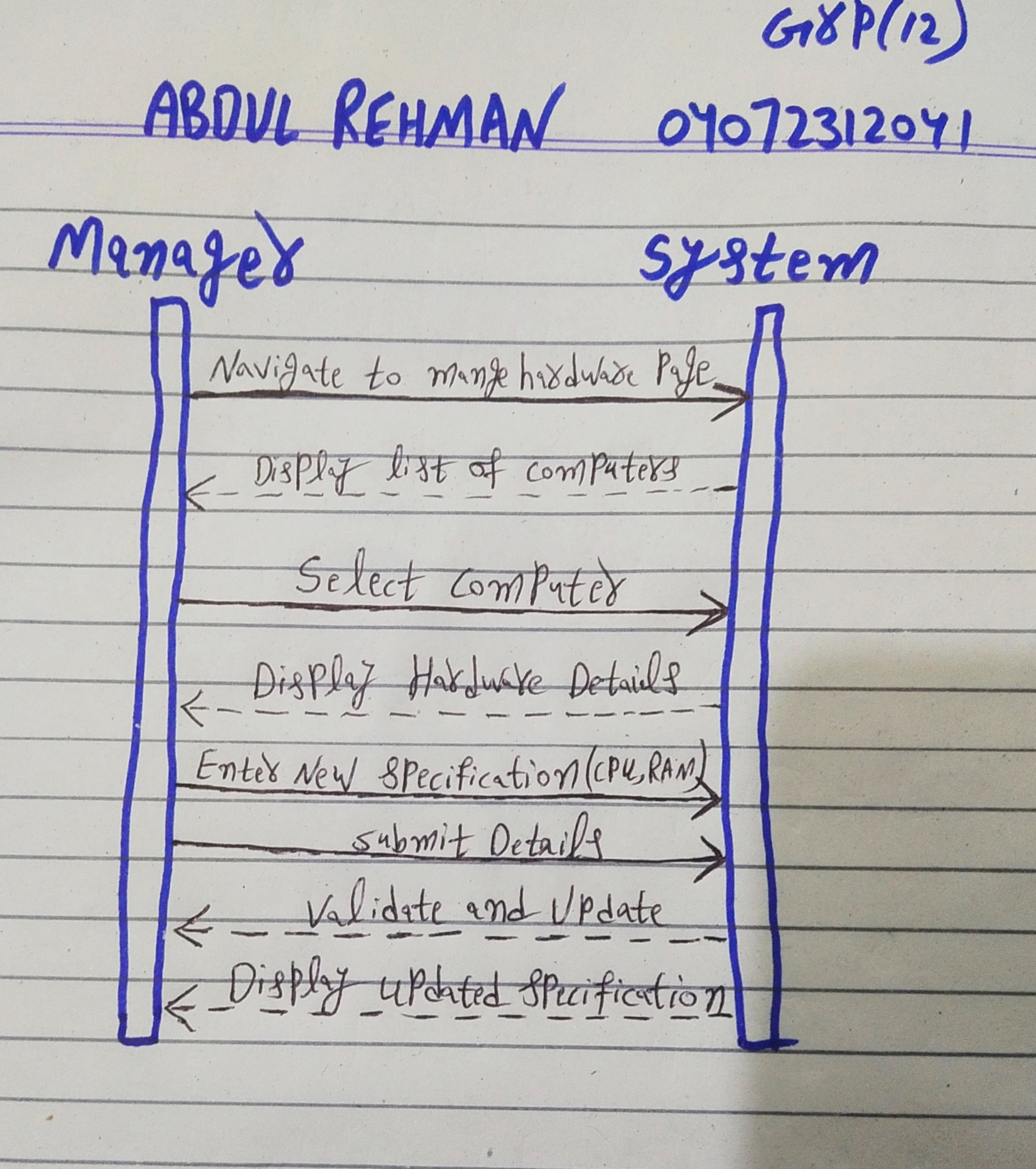
System displays the current hardware details of the selected computer.

Manager enters new hardware specifications (e.g., CPU, RAM, Storage).

Manager submits the updated specifications.

System validates the input and updates the hardware specifications in the database.

System confirms the update and displays the updated hardware specifications.

****

1. **Manager Navigates to Manage Hardware Specification Page:**

The Manager accesses the system and navigates to the Manage Hardware Specification page.

The System displays a list of all computers in the laboratory along with their current hardware specifications.

1. **Manager Selects a Computer:**

The Manager selects a specific computer from the list to update its hardware specifications.

The System displays the current hardware details of the selected computer (CPU, RAM, Storage).

1. **Manager Enters New Hardware Specifications:**

The Manager enters the updated hardware specifications for the selected computer:

CPU: The new processor type.

RAM: The new amount of RAM (in GB).

Storage: The new storage capacity (in GB).

The Manager submits the updated specifications.

1. **System Validates and Updates the Database:**

The System validates the input to ensure it meets the required format and constraints.

If the input is valid, the System updates the hardware specifications in the database.

If the input is invalid, the System displays an error message and prompts the Manager to correct the input.

1. **System Confirms the Update:**

The System confirms that the hardware specifications have been updated successfully.

The updated hardware specifications are displayed to the Manager.

**Preconditions:**

Manager Authentication:

The Manager must be logged into the system.

Computer Existence:

The computer whose hardware specifications are being managed must exist in the system.

Access Privileges:

The Manager must have the necessary permissions to manage hardware specifications.

**Postconditions:**

Successful Update:

The hardware specifications of the selected computer are updated in the database.

The Manager receives a confirmation message indicating the successful update.

Failed Update:

If the validation fails (e.g., invalid input), the system displays an error message.

The hardware specifications remain unchanged.

**Alternate Scenarios:**

**Invalid Input:**

If the Manager enters invalid data (e.g., non-numeric values for RAM or Storage):

The System displays an error message (e.g., "Invalid input. Please enter numeric values.").

The Manager is prompted to re-enter the specifications.

**Computer Not Found:**

If the selected computer is not found in the database:

The System displays an error message (e.g., "Computer not found.").

The Manager is redirected to the list of computers.

**Name: Muhammad Talha**

**Reg. No: 04072312004**

**Use Case Text ("Submit Complaints" Use Case)**

Submit complaints is the use case in which user (student and teacher) will be allowed to submit complaints about the computer’s hardware and software issues and any updation.

**Actors:**

User : Students and teachers will be the user of this system and they will be able to submit complaints by using this system.

**Pre-Conditions:**

1. User must have an account in Laboratory Management System.
2. User must be logged in into the system.
3. System must be functional properly.
4. Computer must be connected to the internet.

#### Post-Conditions:

1. **Successful Submission:**
   * The complaint is successfully stored in the system's database.
   * The user receives a confirmation.
2. **Failed Submission**:
   * The complaint is not stored in the database.
   * The user receives an error.
   * The user is requested to correct any invalid input and resubmit the complaint.

### **Success Scenario**

### **Steps in the Success Scenario:**

1. **User Navigates to Submit Complaint Page**:

The user (Student or Teacher) logs into the system and navigates to the **Submit Complaint** page.

1. **System Displays Complaint Form:**

The system displays a form for entering complaint details, including:

* + - Description: A detailed description of the issue.
    - User ID: The ID of the user associated with the complaint.

1. **User Fills in Complaint Details:**
   * The user enters the required details in the form.
2. **User Submits the Complaint:**
   * The user clicks the **Submit** button to send the complaint to the system.
3. **System Validates the Complaint:**
   * The system checks if all required fields are filled.
4. **System Stores the Complaint:**
   * The system stores the complaint in the database with the sender ID.
5. **System Confirms Submission:**
   * The system displays a success message (e.g., "Your complaint has been submitted successfully.").
6. **User Receives Confirmation:**
   * The user sees the confirmation message.

**Alternate Scenario**

**Steps In Alternate Scenario:**

#### ****User Doesn’t Have an Account****

* The user tries to access the **Submit Complaint** page but does not have an account.
* The system displays a message: "You do not have an account. Please request the manager to create an account for you."
* The user contacts the **Manager** to request account creation.
* The **Manager** creates an account for the user.
* The user logs in with the new account and proceeds to submit the complaint.

#### ****User Is Not Logged In****

* The user tries to access the Submit Complaint page without logging in.
* The system detects that the user is not logged in and redirects them to the Login Page.
* The user logs in with their credentials.
* After successful login, the system redirects the user to the Submit Complaint page.
* The user proceeds to submit the complaint.

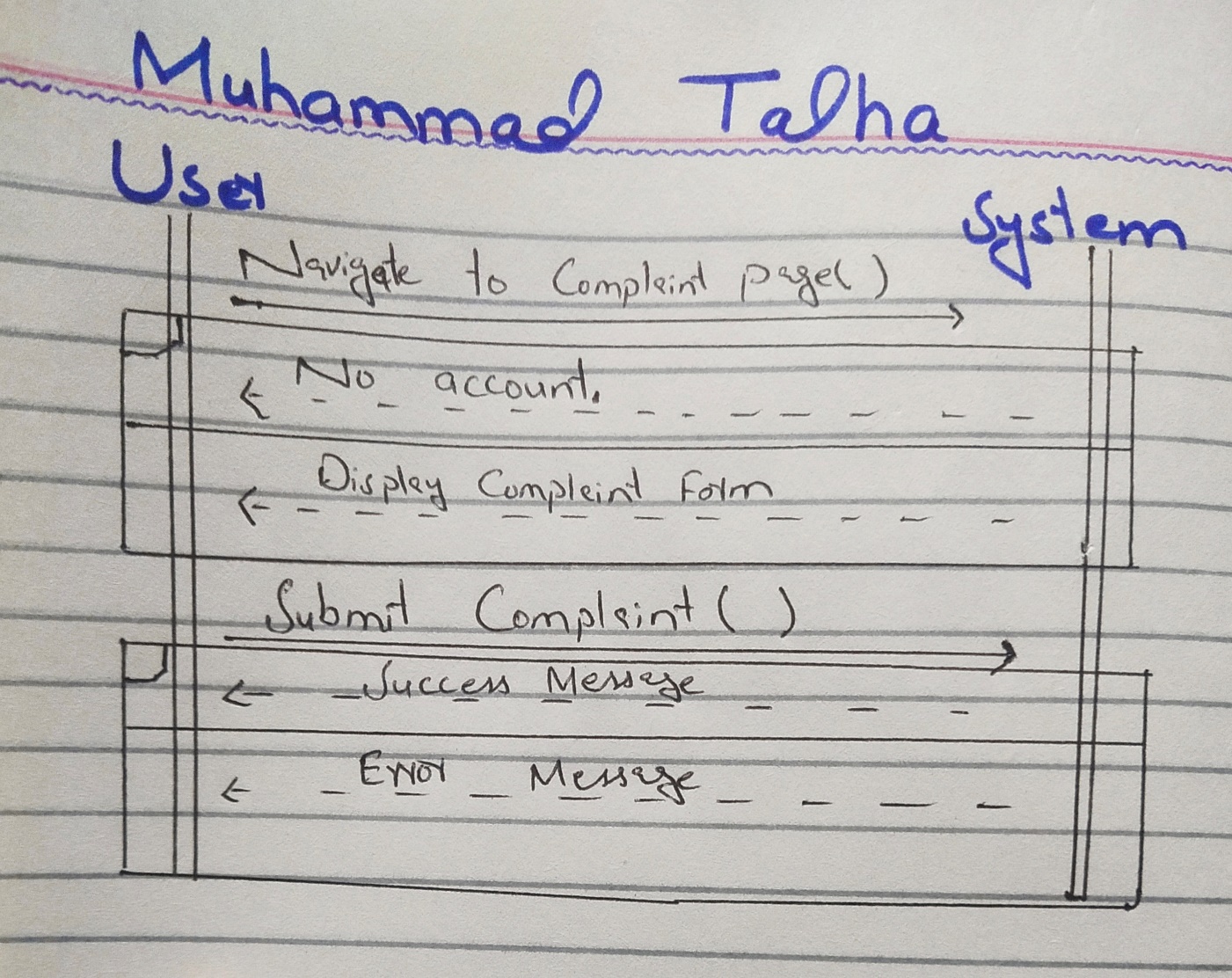
#### ****Internet Not Connected****

* The user tries to submit a complaint, but the computer is not connected to the internet.
* The system detects no internet connection and displays an error message: "No internet connection. Please check your network and try again."
* The user ensures the computer is connected to the internet.
* The user retries submitting the complaint.

#### ****Invalid Input****

* The user fills in the complaint form but enters an invalid Computer ID or leaves a required field empty.
* The system validates the input and detects the error.
* The system displays an error message:
* If the Computer ID is invalid: "Invalid Computer ID. Please enter a valid ID."
* If a required field is empty: "Please fill in all required fields."
* The user corrects the input and resubmits the complaint.
* The system validates the corrected input and proceeds with the submission.

**System Sequence Diagram**

****