


TRƯỜNG ĐẠI HỌC TỰ THỰC QUỐC TẾ SÀI GÒN KHOA KỸ THUẬT & KHOA HỌC MÁY TÍNH PHÁT TRIỂN, VẬN HÀNH VÀ BẢO TRÌ PHẦN MỀM	<b>Lab 4</b> <b>Quản lý Thay đổi và Nâng Cấp          Phần Mềm</b>	 The Saigon International University
--	---	---

## Mục tiêu

1. Hiểu rõ quy trình quản lý thay đổi phần mềm, từ phân tích yêu cầu đến triển khai.
2. Sử dụng công cụ quản lý thay đổi (Jira/Trello) để lập kế hoạch và theo dõi tiến độ.
3. Thực hiện nâng cấp phần mềm với các tính năng mới và cải tiến.
4. Viết kiểm thử và thực hiện kiểm tra tính năng để đảm bảo tính ổn định.
5. Tích hợp và tối ưu hóa hệ thống.

## Tình huống mô tả

Công ty XYZ, chuyên về dịch vụ nhân sự, yêu cầu nâng cấp hệ thống quản lý thông tin nội bộ (MIS) với các tính năng sau:

1. **Chức năng Đăng nhập Nâng Cao**
  - Phân quyền Admin và User.
  - Khóa tài khoản sau 3 lần nhập sai.
  - Ghi log hoạt động đăng nhập.
2. **Chức năng Quản lý Nhân viên**
  - Admin: Thêm, sửa, xóa thông tin nhân viên.
  - User: Chỉ xem thông tin cá nhân.
3. **Chức năng Báo Cáo Hiệu Suất**
  - Admin: Xem báo cáo hiệu suất toàn bộ công ty hoặc theo phòng ban.
  - User: Chỉ xem được báo cáo cá nhân.
4. **Ghi Log Hệ Thống**
  - Lưu trữ tất cả các hoạt động như: đăng nhập, thêm/sửa/xóa nhân viên, xem báo cáo.
  - Phân tích log để phát hiện lỗi hoặc hành vi bất thường.

## Nội dung Thực hành

### Phần 1: Quản lý Yêu Cầu Thay Đổi

Bài tập 1: Lập kế hoạch trong Trello/Jira

1. Tạo **board Trello hoặc Jira** với các cột:
  - **To Do, In Progress, Testing, Done.**
2. Chia các yêu cầu thành các **tasks**, ví dụ:
  - Tạo tính năng đăng nhập nâng cao.

- Xây dựng module quản lý nhân viên.
  - Tạo báo cáo hiệu suất.
  - Tích hợp và kiểm thử.
3. **Câu hỏi:**
- Làm thế nào để phân chia công việc hiệu quả trong nhóm?
  - Nếu một nhiệm vụ gặp vấn đề trong kiểm thử, bạn sẽ xử lý thế nào?

## **Phần 2: Thực hiện Tính Năng**

### **Bài tập 2: Đăng Nhập Nâng Cao**

#### **Mục tiêu:**

1. Xây dựng tính năng đăng nhập với logic rõ ràng.
2. Phân biệt quyền hạn giữa Admin và User.
3. Bảo mật với tính năng khóa tài khoản.
4. Ghi log toàn bộ hoạt động đăng nhập.

#### **Yêu cầu chi tiết:**

1. **Quản lý tài khoản:**
  - Dùng một cấu trúc dữ liệu (HashMap) để lưu thông tin tài khoản:
    - Tài khoản mẫu:
      - admin : Mật khẩu admin123
      - user1 : Mật khẩu user123
  - Lưu trữ số lần đăng nhập sai trong một HashMap khác:
    - Khóa tài khoản sau 3 lần nhập sai.
2. **Phân quyền:**
  - Admin: Được phép truy cập mọi chức năng trong hệ thống.
  - User: Chỉ được phép truy cập thông tin cá nhân hoặc báo cáo cá nhân.
3. **Bảo mật:**
  - Tài khoản bị khóa khi nhập sai mật khẩu 3 lần liên tiếp.
  - Hiển thị thông báo: "Account locked!" nếu tài khoản bị khóa.
4. **Ghi log:**
  - Lưu các hoạt động đăng nhập:
    - Thành công: "User [username] logged in successfully."
    - Thất bại: "Failed login attempt for [username]."
    - Khóa tài khoản: "User [username] account locked after 3 failed attempts."

#### **Hướng dẫn thực hiện:**

1. **Xây dựng logic đăng nhập cơ bản:**
  - Sinh viên viết chương trình với các tính năng:
    - Yêu cầu nhập username và password.
    - Kiểm tra thông tin đăng nhập:
      - Thành công: Hiển thị "Login successful!".
      - Thất bại: Hiển thị "Invalid credentials!".
  - Tự kiểm tra bằng cách thêm tài khoản mẫu vào HashMap.
2. **Thêm logic phân quyền:**

- Nếu là admin, hiển thị: "Welcome Admin! You have full access."
  - Nếu là user, hiển thị: "Welcome User! You have limited access."
3. **Thêm tính năng khóa tài khoản:**
- Kiểm tra số lần nhập sai của từng tài khoản.
  - Khóa tài khoản sau 3 lần nhập sai:

```
if (loginAttempts.get(username) >= 3) {  
    System.out.println("Account locked!");  
    continue;  
}
```

4. **Ghi log vào console:**

- Ghi log trạng thái đăng nhập thành công/thất bại.

**Bài tập dành cho sinh viên:**

1. Viết chương trình hoàn chỉnh cho chức năng đăng nhập nâng cao theo yêu cầu trên.
2. Ghi log vào file thay vì chỉ hiển thị trên console.
3. Kiểm tra:
  - Đăng nhập thành công.
  - Đăng nhập thất bại.
  - Tài khoản bị khóa sau 3 lần nhập sai.

**Bài tập 3: Quản lý Nhân Viên**

**Yêu cầu:**

1. Xây dựng module quản lý nhân viên:
  - Admin: Thêm, sửa, xóa nhân viên.
  - User: Chỉ được xem thông tin cá nhân.
2. Ghi log hoạt động quản lý nhân viên:
  - Ví dụ: "Admin admin added employee John Doe."

**Câu hỏi:**

1. Làm thế nào để phân quyền hiệu quả giữa Admin và User?
2. Nếu cần lưu trữ thông tin nhân viên vào file, bạn sẽ thực hiện như thế nào?

**Phần 3: Kiểm Thử và Đánh Giá**

**Yêu cầu cho sinh viên**

**Bài tập 4 : Sinh viên thực hiện các bước sau để kiểm thử các chức năng trong hệ thống:**

**1. Viết kiểm thử đơn vị (unit test):**

- Chức năng đăng nhập:
  - Đăng nhập thành công với thông tin đúng.
  - Đăng nhập thất bại khi nhập sai mật khẩu.
  - Tài khoản bị khóa sau 3 lần nhập sai.
  - Đăng nhập với tài khoản không tồn tại.
- Chức năng quản lý nhân viên:
  - Thêm nhân viên mới.
  - Sửa thông tin nhân viên.
  - Xóa nhân viên.
  - Kiểm tra phân quyền, chỉ Admin có quyền quản lý.
- Ghi log hoạt động:
  - Log hoạt động đăng nhập thành công và thất bại.
  - Log khi tài khoản bị khóa.
  - Log sau khi thêm, sửa, hoặc xóa nhân viên.

**2. Lập kế hoạch kiểm thử:**

- Liệt kê các trường hợp kiểm thử trong bảng kiểm thử.
- Mô tả dữ liệu đầu vào, kết quả mong đợi, và trạng thái Pass/Fail.

**3. Thực hiện kiểm thử:**

- Chạy mã nguồn và nhập các giá trị từ kế hoạch kiểm thử.
- Quan sát kết quả thực tế và ghi lại trong bảng kiểm thử.

**4. Phân tích và cải tiến:**

- Ghi nhận các lỗi nếu xảy ra.
- Đề xuất cải tiến hoặc sửa lỗi để hoàn thiện hệ thống.

**Bảng kiểm thử mẫu (Ví dụ tham khảo)**

STT	Chức năng	Trường hợp kiểm thử	Dữ liệu đầu vào	Kết quả mong đợi	Trạng thái
1	Đăng nhập	Đăng nhập thành công với Admin	Username: admin, Password: admin123	"Login successful!"	Pass
2	Đăng nhập	Đăng nhập thất bại (sai mật khẩu)	Username: user1, Password: wrong	"Invalid credentials!"	Pass
3	Đăng nhập	Tài khoản bị khóa sau 3 lần nhập sai	Username: user1, Password: wrong (x3)	"Account locked!"	Pass
4	Đăng nhập	Đăng nhập với tài khoản không tồn tại	Username: unknown, Password: 1234	"Invalid username!"	Pass
5	Quản lý nhân viên	Thêm nhân viên mới	Tên: John Doe, ID: E001	"Employee added successfully."	Pass
6	Quản lý nhân viên	Sửa thông tin nhân viên	ID: E001, Email: john.doe@example.com	"Employee updated successfully."	Pass

7	Quản lý nhân viên	Xóa nhân viên	ID: E001	"Employee deleted successfully."	Pass
8	Quản lý nhân viên	User thử xóa nhân viên	Username: user1, ID: E001	"Access Denied!"	Pass
9	Ghi log	Kiểm tra log sau khi đăng nhập thành công	Username: admin, Password: admin123	Log: "User admin logged in successfully."	Pass

### **Yêu cầu nộp bài**

**1. Bảng kiểm thử chi tiết:**

- Bao gồm các trường hợp kiểm thử, dữ liệu đầu vào, kết quả mong đợi, kết quả thực tế, và trạng thái Pass/Fail.
- Sinh viên tự bổ sung thêm các trường hợp kiểm thử cho hệ thống của mình.

**2. Phân tích lỗi:**

- Nếu có lỗi xảy ra, cần mô tả nguyên nhân và đề xuất cách sửa lỗi.

**3. Log hoạt động:**

- Nộp file log ghi lại toàn bộ hoạt động hệ thống trong quá trình kiểm thử.

**4. Mã nguồn hoàn chỉnh:**

- Bao gồm mã chính và mã kiểm thử.

## **Phần 4: Đo Lường và Tối Ưu**

### **Bài tập 5: Đo Lường Hiệu Năng**

- Đo thời gian xử lý khi:
  - Đăng nhập với 10, 100, và 1,000 tài khoản.
  - Xử lý báo cáo với 100, 500, và 1,000 nhân viên.
- Ghi nhận kết quả và đề xuất cải tiến.

### **Yêu Cầu Nộp Bài**

**1. Báo cáo:**

- Chụp màn hình Trello/Jira (nhiệm vụ và trạng thái).
- Mã nguồn trước và sau khi thay đổi.
- Kết quả kiểm thử (Pass/Fail).
- Log hệ thống.

**2. Mã nguồn hoàn chỉnh:**

- Bao gồm mã chính và mã kiểm thử.