**KẾ HOẠCH KIỂM THỬ HỆ THỐNG QUẢN LÝ KÝ TÚC XÁ**

**1. GIỚI THIỆU**

**Mục đích:**  
Kế hoạch kiểm thử này được xây dựng nhằm xác định phạm vi, chiến lược và quy trình kiểm thử cho hệ thống quản lý ký túc xá. Mục tiêu là đảm bảo chất lượng phần mềm thông qua việc phát hiện sớm lỗi, kiểm tra tính năng, hiệu suất và bảo mật trước khi triển khai sản phẩm.

**Đối tượng sử dụng:**

* Lập trình viên
* Kiểm thử viên
* Quản lý dự án
* Nhân viên hỗ trợ kỹ thuật
* Các bên liên quan khác

**2. PHẠM VI KIỂM THỬ**

* **Kiểm thử chức năng:**
  + Quản lý tài khoản (tạo, sửa, vô hiệu, tìm kiếm tài khoản).
  + Quản lý thông tin cá nhân của người dùng.
  + Quản lý sinh viên: xếp phòng, tìm kiếm sinh viên, cập nhật thông tin.
  + Quản lý hợp đồng: tạo, ký kết, tìm kiếm hợp đồng.
  + Quản lý hóa đơn: tạo, tìm kiếm, in hóa đơn và thanh toán.
  + Quản lý dịch vụ và phòng: thêm/sửa/xóa thông tin phòng, xếp phòng cho sinh viên, phân công trực ban.
  + Thanh toán và báo cáo thống kê (số phòng trống, số sinh viên, số tiền thu – chi).
  + Xem tổng quan hệ thống (dành cho khách).
* **Kiểm thử giao diện người dùng:**
  + Đánh giá tính thân thiện, trực quan của giao diện web.
  + Đảm bảo tương thích trên các trình duyệt (Chrome, Firefox, Edge).
* **Kiểm thử hiệu suất:**
  + Đo lường thời gian phản hồi của hệ thống khi có nhiều người dùng đồng thời.
  + Xác định giới hạn tải của hệ thống.
* **Kiểm thử bảo mật:**
  + Kiểm tra chống tấn công SQL Injection, XSS, và các lỗ hổng bảo mật khác.
  + Đảm bảo dữ liệu người dùng được mã hóa và bảo vệ đúng cách.
* **Kiểm thử khả năng chịu tải:**
  + Xác định hiệu năng của hệ thống dưới điều kiện tải cao
* . **Kiểm thử tĩnh** 
  + Kiểm thử các tài liệu liên quan của dự án (SDD, SRS, Test Case, Github readme, Test report)

**3. PHƯƠNG PHÁP KIỂM THỬ**

**3.1. Loại Kiểm Thử**

* **Kiểm thử chức năng:**
  + Xác minh các tính năng hoạt động đúng như yêu cầu trong SRS.
* **Kiểm thử giao diện người dùng:**
  + Đánh giá trải nghiệm người dùng, bố cục, khả năng tương tác và tính nhất quán của giao diện.
* **Kiểm thử hiệu suất:**
  + Sử dụng các công cụ đo lường thời gian phản hồi và khả năng xử lý khi có nhiều truy cập đồng thời.
* **Kiểm thử bảo mật:**
  + Kiểm tra các điểm yếu bảo mật và lỗ hổng thông qua các công cụ tự động và kiểm tra thủ công.
* **Kiểm thử khả năng chịu tải:**
  + Mô phỏng nhiều người dùng để xác định giới hạn của hệ thống.
* **Kiểm thử tĩnh:**
  + Đọc và kiểm tra phát hiện các lỗi trong tài liệu ( lỗi chính tả, lỗi ngữ nghĩa, các hình vẽ, tính hoàn thiện của tài liệu)
  + Lên checklist để xác định những yếu tố cần kiểm thử
  + Sử dụng các phần mềm report lỗi

**3.2. Công Cụ Kiểm Thử**

* **Kiểm thử chức năng:** Sử dụng Selenium cho kiểm thử tự động.
* **Kiểm thử hiệu suất và chịu tải:** Sử dụng JMeter để mô phỏng tải và đo lường hiệu suất.
* **Kiểm thử bảo mật:** Sử dụng OWASP ZAP để quét lỗ hổng bảo mật.
* **Kiểm thử tĩnh:** Sử dụng Github issue trên trang github chính của dự án để báo cáo lỗi.

**4. MÔI TRƯỜNG KIỂM THỬ**

* **Hệ điều hành:**
  + Windows, Linux (tùy theo môi trường triển khai và máy kiểm thử).
* **Trình duyệt:**
  + Google Chrome, Mozilla Firefox, Microsoft Edge (các phiên bản mới nhất).
* **Cơ sở dữ liệu:**
  + MySQL (theo cấu hình triển khai).
* **Máy chủ ứng dụng:**
  + Apache Tomcat hoặc Nginx (tùy thuộc vào môi trường sản xuất).
* **Môi trường mạng:**
  + Mạng nội bộ và truy cập qua Internet (có thể kiểm thử từ xa qua VPN nếu cần).

**5. TIÊU CHÍ CHẤP NHẬN**

* Tất cả các trường hợp kiểm thử quan trọng phải đạt được kết quả mong đợi.
* Không có lỗi nghiêm trọng (Critical) hoặc lỗi ảnh hưởng đến trải nghiệm người dùng.
* Hiệu suất hệ thống đáp ứng yêu cầu đã đề ra dưới tải cao.
* Bảo mật hệ thống không có lỗ hổng nghiêm trọng và dữ liệu được bảo vệ an toàn.
* Các tài liệu kiểm thử phải có tiêu chí rõ ràng để đánh giá mức độ đầy đủ và chính xác:
  + Độ bao phủ yêu cầu: Mọi yêu cầu trong SRS phải có ít nhất một test case tương ứng.
  + Tính rõ ràng: Test case không được gây hiểu nhầm hoặc mơ hồ.
  + Khả năng tái sử dụng: Các test case nên được viết theo hướng có thể tái sử dụng trong tương lai.

**6. LỊCH TRÌNH KIỂM THỬ**

|  |  |  |
| --- | --- | --- |
| **Giai đoạn** | **Thời gian** | **Nội dung** |
| Phân tích yêu cầu | 21/02 - 23/02 | Xác định các yêu cầu kiểm thử từ SRS và SDD |
| Thiết kế test case | 24/02 - 28/02 | Viết và thiết kế các test case chi tiết |
| Chuẩn bị môi trường | 01/03 - 02/03 | Cài đặt, cấu hình môi trường kiểm thử, cài đặt công cụ kiểm thử |
| Thực hiện kiểm thử tĩnh | 03/03 – 10/03 | Lên checklist, đọc các tài liệu liên quan để tìm lỗi, ghi nhận lỗi và báo cáo |
| Thực hiện kiểm thử động | 11/03 - 20/03 | Chạy test case, ghi nhận lỗi và báo cáo kết quả |
| Kiểm thử hồi quy | 21/03 - 23/03 | Kiểm thử lại sau khi sửa lỗi, đảm bảo không tái xuất lỗi |
| Báo cáo kết quả | 24/03 - 25/03 | Tổng hợp, đánh giá kết quả kiểm thử và báo cáo cho nhóm phát triển |

**7. RỦI RO VÀ GIẢI PHÁP**

* **Thiếu tài nguyên kiểm thử:**
  + Giải pháp: Phân bổ nhân sự và ưu tiên kiểm thử các chức năng quan trọng trước.
* **Lỗi nghiêm trọng phát sinh:**
  + Giải pháp: Xác định nguyên nhân và lập kế hoạch sửa lỗi. Sau đó, triển khai quy trình xử lý lỗi nhanh chóng và kiểm thử lại hệ thống để đảm bảo lỗi đã được khắc phục triệt để.
* **Hiệu suất không đạt yêu cầu dưới tải cao:**
  + Giải pháp: Tối ưu hệ thống sớm, chạy kiểm thử hiệu suất định kỳ và áp dụng các biện pháp tối ưu (cache, tối ưu truy vấn CSDL, …).
* **Vấn đề bảo mật:**
  + Giải pháp: Sử dụng các công cụ quét bảo mật tự động và thực hiện kiểm tra bảo mật định kỳ, cập nhật các bản vá bảo mật mới nhất.

**8. Đánh giá và báo cáo**

**8.1. Tổng hợp kết quả**

* Số lượng lỗi phát hiện trong tài liệu và mã nguồn.
* Mức độ nghiêm trọng của từng lỗi (critical, major, minor).
* Thống kê tỷ lệ lỗi theo loại tài liệu.

**8.2. Đánh giá chất lượng**

* Mức độ tuân thủ tiêu chuẩn của tài liệu và mã nguồn.
* So sánh tài liệu với best practices trong ngành.

**8.3. Báo cáo**

* Báo cáo chi tiết liệt kê các lỗi, nguyên nhân, và đề xuất sửa chữa.
* Gửi báo cáo cho nhóm phát triển vào ngày 12/03/2025.
* Cung cấp bảng phân tích lỗi để theo dõi tiến trình sửa chữa.

**9. Vai trò và trách nhiệm**

|  |  |  |
| --- | --- | --- |
| **Thành viên** | **Vai trò** | **Trách nhiệm** |
| Nguyễn Đình Việt | Người kiểm thử chính | Đọc và đánh giá tài liệu |
| Trần Mai Trung | Xác minh yêu cầu | Đảm bảo tài liệu phù hợp với thực tế |
| Nguyễn Văn Thiện | Hỗ trợ phân tích lỗi | Sửa lỗi trong tài liệu nếu cần |
| Nguyễn Đình Việt | Phê duyệt kết quả kiểm thử | Đảm bảo tiến độ kiểm thử |
| Nguyễn Sơn Tùng | Người đánh giá phản hồi | Thu thập và xử lý phản hồi tài liệu |

**10. Theo dõi và cải tiến**

* Theo dõi tiến độ sửa chữa các lỗi đã phát hiện.
* Cập nhật test plan dựa trên phản hồi từ nhóm phát triển.
* Ghi nhận bài học kinh nghiệm để cải tiến quy trình kiểm thử tĩnh cho các dự án sau.
* Áp dụng bài học từ các dự án trước để nâng cao hiệu quả kiểm thử.