**TRƯỜNG CAO ĐẲNG THỰC HÀNH FPT**



**SOF304 – KIỂM THỬ NÂNG CAO**

**KIỂM THỬ DỰ ÁN**

**App Quản lý Coffe**

*Lớp:* **SD18307**

*Nhóm:* **3**

*Thực hiện:* **Nguyễn Văn Quyết (Nhóm Trưởng)**

**Hoàng Tấn Thiên**

**Phạm Thị Kim Ngân**

**Nguyễn Thị Thảo Vy**

**Bùi Minh Đức**

***TP. Đà Nẵng, tháng 4 năm 2024***

**NỘI DUNG CHÍNH CỦA DỰ ÁN**

**Giới Thiệu Dự Án**

* Phần mềm quản lý quán cà phê là một ứng dụng hoặc hệ thống phần mềm được thiết kế để giúp quản lý và vận hành hiệu quả các hoạt động của một quán cà phê. Giúp người quản trị quản lý được các nhân viên, xem những thống kê doanh thu hàng năm , lưu trữ các dữ liệu mà khách hàng và xem lại những lịch sử hóa đơn đã thanh toán

**Cấu Trúc Trang Web**

**Trang Chủ:**

Home-Trang chủ (chọn bàn đặt, chọn thêm món ).

**Login :**

Đăng nhập và đổi mật khẩu

**Quản lý Nhân viên:**

**Thông tin gồm:**

* Mã nhân viên
* Mật khẩu
* Họ và tên
* Admin
* Địa chỉ
* Ngày sinh
* Các chức năng cơ bản Thêm, Sửa, Xóa.

**Trang chi tiết nhân viên gồm đầy đủ thông tin người dùng, có thể chỉnh sửa chi tiết tại trang này.**

A screenshot of a video chat

Description automatically generated

**Sơ Đồ Quan Hệ Thực Thể**

A diagram of a computer network

Description automatically generated

**Các Yêu Cầu Chức Năng**

**Đăng nhập và Đăng xuất**

* + Người dùng nhập Username và Password để đăng nhập vào hệ thống
  + Nếu người dùng quên mật khẩu, họ có thể nhấp vào liên kết 'Quên mật khẩu' trên trang Đăng nhập. Màn hình sẽ hiển thị cho phép người dùng nhập Tên đăng nhập hoặc Email để lấy lại mật khẩu. Nếu Tên đăng nhập hoặc Email không tồn tại sẽ có thông báo lỗi, ngược lại mật khẩu mới sẽ gửi vào email tương ứng với username trong cơ sở dữ liệu.

**Chức năng cho admin đăng nhập:**

* Chức năng xem danh sách người dùng
* Chức năng thêm người dùng mới
* Chức năng cập nhật người dùng
* Chức năng xóa người dùng

**Chức năng cho user đăng nhập:**

* Chức năng xem danh sách thống kê
* Chức năng xem sản phẩm

**CÁC CÔNG CỤ MÃ NGUỒN MỞ**

**Giới thiệu về Junit**

Trong Java, để thực hiện viết code cho Unit Test chúng ta có thể sử dụng một trong hai Framework: [**JUnit**](https://junit.org/)và [**TestNG**](https://testng.org/)**.**

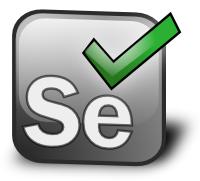
**JUnit** là một framework mã nguồn mở, miễn phí, đơn giản dùng để unit test cho ngôn ngữ lập trình Java. Trong Java, chúng ta thường sẽ sử dụng method để làm unit test.

Chúng ta có thể sử dụng JUnit để viết code test cho cả unit testing và integration testing.

**Các tính năng của Junit**

* JUnit là một framework mã nguồn mở, sử dụng để viết và chạy kiểm thử.
* Cung cấp các annotation để định nghĩa các phương thức kiểm thử.
* Cung cấp các Assertion để kiểm tra kết quả mong đợi.
* Cung cấp các test runner để thực thi các test script.
* Test case JUnit có thể được chạy tự động.
* Test case JUnit có thể được tổ chức thành các test suite.
* JUnit cho thấy kết quả test một cách trực quan: pass (không có lỗi) là màu xanh và fail (có lỗi) là màu đỏ.

**Giới Thiệu Vê Selenium**

**Selenium là gì?**

Selenium là bộ kiểm thử tự động miễn phí (mã nguồn mở) dành cho các ứng dụng web trên các trình duyệt và nền tảng khác nhau. Nó khá là giống với HP Quick Test Pro (QTP bây giờ là UFT) chỉ khác là Selenium thì tập trung vào việc tự động hoá các ứng dụng dựa trên nền tảng web.

Kiểm thử được thực hiện bằng cách sử dụng công cụ Selenium thường được gọi là Kiểm thử Selenium. Selenium không chỉ là 1 công cụ độc lập mà là 1 bộ công cụ của phần mềm, mỗi bộ đều đáp ứng được nh u cầu kiểm thử khác nhau của 1 tổ chức. Nó có 4 thành phần.

* Selenium Integrated Development Environment (IDE)
* Selenium Remote Control (RC)(s1)
* WebDriver
* Selenium Grid

Hiện tại, Selenium RC và WebDriver được hợp nhất thành một framework duy nhất để tạo ra Selenium 2. Còn Selenium 1 thì tham chiếu đến Selenium RC.

**Những điều cần lưu ý khi sử dụng Selenium**

Tests nên luôn có 1 starting point đã biết. Với Selenium, nó có nghĩa là mở 1 page để bắt đầu 1 workflow.

Tests không nên phải phụ thuộc vào bất kỳ 1 test nào khác để chạy. Nếu 1 test dự định add 1 cái gì đó thì phải đảm bảo rằng không có 1 test khác sẽ delete cái định add đó. Điều này đảm bảo rằng nếu có gì đó bị sai trong 1 test, thì nó cũng không phải là unnecessary failures.

Tests chỉ nên test 1 thứ tại 1 thời điểm.

Tests nên làm sạch sau khi chạy xong.

Nếu không thỏa mãn các rule này khi chạy test tự động Selenium thì có thể sẽ gặp phải nhiều issues sau đó và nếu như có rất nhiều TH cần phải test thì những vấn đề nhỏ này sẽ dẫn tới có rất nhiều phần của 1 test suite sẽ bị ra kết quả thất bại.

**Test Case**

**A screenshot of a computer

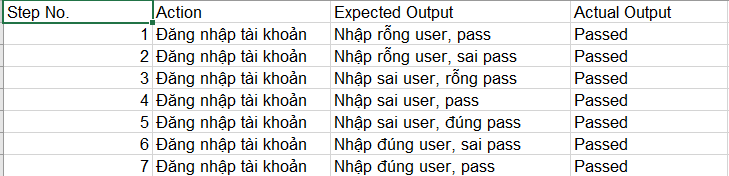
Description automatically generated**

**A white rectangular box with black text

Description automatically generated with medium confidence**

**SỬ DỤNG SELINIUM ĐỂ TEST MỘT SỐ CHỨC NĂNG**

**Chức Năng Đăng Nhập – Quản lý người dùng**



**Code Test**

**Test Login**

**A screenshot of a computer program

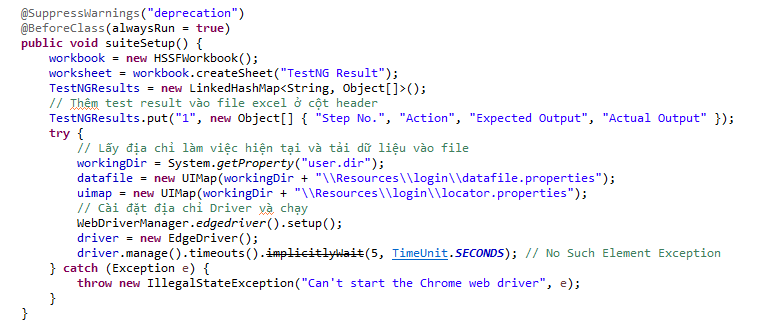
Description automatically generated**

**A screenshot of a computer code

Description automatically generated**

**Khởi Tạo Webdriver Chạy Đầu Tiên Và Sử Dụng Testng Để Kiểm Thử.**

**Webdriver luôn chạy trước class**

****

**A computer code with text

Description automatically generated**

**Test delete nhân viên**

A screenshot of a computer program

Description automatically generated

**Test insert nhân viên**

A screenshot of a computer program

Description automatically generated

A computer screen shot of a computer code

Description automatically generated