

**TRƯỜNG ĐẠI HỌC**

**KHOA CÔNG NGHỆ THÔNG TIN**

**BÁO CÁO BÀI TẬP LỚN**

**CÔNG NGHỆ PHẦN MỀM**

**ĐỀ TÀI:**

**QUẢN LÝ BÁN HÀNG CỦA SIÊU THỊ**

|  |  |
| --- | --- |
| Giảng viên hướng dẫn: |  |
| Sinh viên thực hiện: |  |
| Thành viên nhóm: |  |
| Lớp: |  |

Hà Nội, năm 2024

**BÁO CÁO KIỂM THỬ KẾ TOÁN**

|  |  |
| --- | --- |
| **Tên tài liệu:** Hệ thống quản lý bán hàng của siêu thị | **Mã tài liệu:** QLKHDN |
| **Tên module:** Kiểm thử kế toán | **Mã module:** KT4 |
| **Tên người làm:** | **Mã người làm:** N1 |
| **Thời hạn:** Ngày bắt đầu: 18/4/2024                    Ngày kết thúc: 24/4/2024 | |
| Ngày nộp lần 1: 18/4/2024 | |
| Phiên bản/ File: YC1-20240418.docx | |

**NGƯỜI LÀM KÝ XÁC NHẬN**

**MỤC LỤC**

I. MÔ TẢ 3

II. MÔI TRƯỜNG KIỂM THỬ 4

III. PHẠM VI KIỂM THỬ 5

1. Tính toán Tổng Số Lượng Đã Bán Theo Ngày 5

2. Tính toán Tổng Doanh Thu Theo Ngày 5

3. Tính toán Doanh Thu Theo Sản Phẩm 5

4. Lấy Tất Cả Báo Cáo Lương 5

5. Lấy Tất Cả Báo Cáo Doanh Thu 5

6. Xóa Báo Cáo Doanh Thu 5

IV. KẾ HOẠCH KIỂM THỬ 6

1. Kiểm thử các Phương thức Tính Toán 6

2. Kiểm thử Lấy Danh sách Báo Cáo 6

3. Kiểm thử Xóa Báo Cáo Doanh Thu 6

V. KẾT QUẢ KIỂM THỬ 7

VI. KẾT LUẬN 8

# I. MÔ TẢ

Báo cáo này là kết quả của quá trình kiểm thử các phương thức và chức năng của lớp SalaryReportController trong một ứng dụng Java Swing.

Lớp SalaryReportController đóng vai trò quan trọng trong hệ thống, điều khiển và xử lý các tác vụ liên quan đến việc tạo, quản lý và truy vấn báo cáo lương và báo cáo doanh thu.

Lớp này là một phần quan trọng của hệ thống quản lý, cung cấp các phương thức để tương tác với cơ sở dữ liệu và truy xuất thông tin cần thiết cho việc hiển thị và báo cáo. Bằng cách sử dụng SalaryReportController, ứng dụng có thể hiển thị thông tin lương của nhân viên, thống kê doanh thu theo ngày, tháng hoặc năm, và nắm bắt thông tin về sản phẩm bán chạy nhất.

Qua báo cáo này, chúng ta sẽ đi sâu vào quá trình kiểm thử các phương thức của SalaryReportController để đảm bảo tính đúng đắn, hiệu suất và độ tin cậy của chúng trong việc xử lý dữ liệu. Đồng thời, báo cáo cũng sẽ phản ánh các biện pháp kiểm thử, kết quả thu được và những khuyến nghị cải thiện cho lớp này.

# II. MÔI TRƯỜNG KIỂM THỬ

Trong quá trình phát triển ứng dụng, việc kiểm thử đóng một vai trò quan trọng để đảm bảo tính ổn định và chất lượng của phần mềm. Để thực hiện quá trình kiểm thử một cách hiệu quả, chúng tôi đã thiết lập một môi trường kiểm thử đáng tin cậy và chi tiết như sau:

Hệ điều hành được sử dụng là Windows 10, một trong những hệ điều hành phổ biến và được sử dụng rộng rãi trên nền tảng máy tính cá nhân. Ngôn ngữ lập trình chính được sử dụng trong ứng dụng là Java, cùng với Framework kiểm thử JUnit 5, được tích hợp sâu vào quá trình phát triển để viết và chạy các test case.

Môi trường phát triển chính là IntelliJ IDEA, phiên bản Ultimate 2021.2, đã được cài đặt và cấu hình đầy đủ để phát triển ứng dụng Java Swing và JUnit. Đồng thời, công cụ quản lý phiên bản Git được sử dụng để quản lý mã nguồn và theo dõi các thay đổi.

Máy chủ MySQL Server 8.0.26 được sử dụng để lưu trữ và truy xuất dữ liệu cho các báo cáo lương và doanh thu trong ứng dụng. Môi trường kiểm thử được thiết lập với cấu hình tương tự như môi trường phát triển để đảm bảo tính nhất quán và tin cậy của các kết quả kiểm thử.

Thời gian kiểm thử được lên kế hoạch trong vòng 2 tuần, bao gồm việc viết test case, thực hiện kiểm thử và báo cáo kết quả. Tất cả những công việc này sẽ do nhóm phát triển chịu trách nhiệm, bao gồm cả các nhà phát triển và kiểm thử viên.

# III. PHẠM VI KIỂM THỬ

Kiểm thử sẽ tập trung vào việc đảm bảo rằng các phương thức của SalaryReportController hoạt động đúng đắn và trả về kết quả chính xác khi truy cập và xử lý dữ liệu từ cơ sở dữ liệu.

## 1. Tính toán Tổng Số Lượng Đã Bán Theo Ngày

Phương thức: getTotalQuantitySoldByDate(Date startDate, Date endDate)

Mô tả: Tính toán và trả về tổng số lượng hàng đã bán trong một khoảng thời gian nhất định.

## 2. Tính toán Tổng Doanh Thu Theo Ngày

Phương thức: getTotalRevenueByDate(Date startDate, Date endDate)

Mô tả: Tính toán và trả về tổng doanh thu thu được từ các giao dịch bán hàng trong một khoảng thời gian nhất định.

## 3. Tính toán Doanh Thu Theo Sản Phẩm

Phương thức: getRevenueByProduct()

Mô tả: Tính toán và trả về doanh thu theo từng sản phẩm, được nhóm lại dựa trên ID của sản phẩm.

## 4. Lấy Tất Cả Báo Cáo Lương

Phương thức: getAllSalaryReport()

Mô tả: Truy vấn và lấy tất cả thông tin báo cáo lương từ cơ sở dữ liệu, bao gồm thông tin về nhân viên, ngày báo cáo, kỳ công, tổng lương, số lần đi làm, vv.

## 5. Lấy Tất Cả Báo Cáo Doanh Thu

Phương thức: getAllDoanhThu()

Mô tả: Truy vấn và lấy tất cả thông tin báo cáo doanh thu từ cơ sở dữ liệu, bao gồm thông tin về giao dịch, ngày bán, tổng số tiền, tên khách hàng, phương thức thanh toán, thông tin nhân viên, sản phẩm bán được và số lượng bán.

## 6. Xóa Báo Cáo Doanh Thu

Phương thức: deleteSalesReport(int salesReceiptID)

Mô tả: Xóa một báo cáo doanh thu khỏi cơ sở dữ liệu dựa trên ID của giao dịch.

Các chức năng này được kiểm thử để đảm bảo tính đúng đắn và hoạt động chính xác của hệ thống trong việc quản lý và xử lý thông tin liên quan đến lương và doanh thu.

# IV. KẾ HOẠCH KIỂM THỬ

## 1. Kiểm thử các Phương thức Tính Toán

Mục tiêu: Xác nhận tính đúng đắn và hiệu suất của các phương thức tính toán tổng số lượng đã bán và tổng doanh thu theo ngày, tháng và năm.

Chiến lược: Sử dụng dữ liệu mẫu đã biết trước và tạo các trường hợp kiểm thử để đảm bảo tính chính xác và độ bao phủ của phương thức.

Kỳ vọng: Các phương thức sẽ trả về kết quả chính xác và hoạt động hiệu quả trong mọi điều kiện và trường hợp.

## 2. Kiểm thử Lấy Danh sách Báo Cáo

Mục tiêu: Đảm bảo rằng việc lấy danh sách báo cáo lương và doanh thu từ cơ sở dữ liệu hoạt động chính xác và đầy đủ.

Chiến lược: Sử dụng dữ liệu mẫu và kiểm tra tính chính xác của danh sách báo cáo trả về so với dữ liệu trong cơ sở dữ liệu.

Kỳ vọng: Danh sách báo cáo sẽ được trả về một cách đầy đủ và chính xác, bao gồm tất cả các thông tin cần thiết.

## 3. Kiểm thử Xóa Báo Cáo Doanh Thu

Mục tiêu: Xác nhận tính đúng đắn và an toàn của việc xóa báo cáo doanh thu và các chi tiết liên quan từ cơ sở dữ liệu.

Chiến lược: Tạo các tình huống kiểm thử để xác định xem phương thức xóa hoạt động chính xác và không gây ra sự cố hoặc mất mát dữ liệu.

Kỳ vọng: Báo cáo doanh thu và các chi tiết liên quan sẽ được xóa một cách an toàn và không ảnh hưởng đến dữ liệu khác trong cơ sở dữ liệu.

Đồng thời, trong quá trình kiểm thử, sẽ sử dụng Framework JUnit 5 để tự động hóa quá trình kiểm thử và đảm bảo tính liên tục và lặp lại của quá trình kiểm thử.

# V. KẾT QUẢ KIỂM THỬ

Phương thức getTotalQuantitySoldByDate(Date startDate, Date endDate):

Đầu Vào: Hai ngày bắt đầu và kết thúc.

Đầu Ra: Số nguyên, tổng số lượng sản phẩm được bán trong khoảng thời gian chỉ định.

Kết Quả: Phương thức trả về số nguyên dương hoặc 0 nếu không có dữ liệu phù hợp.

Phương thức getTotalRevenueByDate(Date startDate, Date endDate):

Đầu Vào: Hai ngày bắt đầu và kết thúc.

Đầu Ra: Số thực, tổng doanh thu được tạo ra trong khoảng thời gian chỉ định.

Kết Quả: Phương thức trả về một số thực dương hoặc 0.0 nếu không có dữ liệu phù hợp.

Phương thức getAllSalaryReport() và getAllDoanhThu():

Đầu Ra: Một DefaultTableModel chứa thông tin về các báo cáo lương và doanh thu.

Kết Quả: Đảm bảo rằng DefaultTableModel không null và chứa dữ liệu đúng đắn.

Phương thức deleteSalesReport(int salesReceiptID):

Đầu Vào: ID của báo cáo doanh thu cần xóa.

Đầu Ra: True nếu việc xóa thành công, False nếu không thành công.

Kết Quả: Đảm bảo rằng việc xóa báo cáo doanh thu và các chi tiết liên quan diễn ra đúng đắn.

# VI. KẾT LUẬN

Kiểm thử đã chứng minh rằng SalaryReportController hoạt động như mong đợi và đảm bảo tính đúng đắn và tin cậy của các phương thức và chức năng của nó trong việc quản lý báo cáo lương và doanh thu trong ứng dụng của chúng ta.

Tuy nhiên, để duy trì và nâng cao chất lượng của hệ thống, việc tiếp tục tập trung vào các biện pháp kiểm thử là hết sức quan trọng. Cải thiện hiệu suất và độ ổn định của hệ thống trong quá trình vận hành là một mục tiêu không ngừng được hướng đến.

Trong quá trình kiểm thử hiện tại, các phương pháp như kiểm thử tích hợp, kiểm thử hiệu suất và kiểm thử bảo mật đã được thực hiện. Tuy nhiên, vẫn cần tiếp tục tối ưu hóa các quy trình và tiêu chuẩn kiểm thử để đảm bảo rằng hệ thống không chỉ đáp ứng được yêu cầu kỹ thuật mà còn đáp ứng được mong đợi của người dùng.

Bên cạnh đó, việc xây dựng các kịch bản kiểm thử tự động và việc triển khai kiểm thử liên tục có thể giúp giảm thiểu rủi ro và tăng cường khả năng phát hiện lỗi sớm. Điều này sẽ giúp giảm thiểu thời gian và chi phí kiểm thử trong quá trình phát triển và duy trì hệ thống.

Trong kết luận, việc tiếp tục cải thiện và mở rộng phạm vi kiểm thử sẽ giúp đảm bảo rằng hệ thống không chỉ đáp ứng được nhu cầu hiện tại mà còn sẵn sàng đáp ứng các thách thức trong tương lai.