

Báo cáo Quản lý chất lượng của Bài tập lớn Library Management System dựa trên Functional Testing

Link github: <https://github.com/Dainn98/LibraryManagement>

Link đề bài: https://drive.google.com/file/d/1RvSRAJzLm1_BLev62MX6nATW2a4T5Sws/view

Sản phẩm có 4 lỗi, trên tổng số 63 testcase. Nên mật độ lỗi: Error density ~ **0.26 KLOC** (=4/15.325)

Bảng báo cáo tổng hợp lỗi của unit test

	Function	Số testcase	Pass	Fail	Runtime (ms)
1	DocManagementTest	13	13	0	912
2	DocTest	10	9	1	209
3	ManageLoanTest	6	4	2	606
4	PendingApprovalsTest	14	14	0	433
5	SignInTest	5	4	1	552
6	StatisticsTest	15	15	0	650
Sum		63	59	4	3362

Bảng báo cáo tổng hợp line of code (LOC)

STT	Language	Files	Blank	Comment	Code
1	Java	68	1288	3332	8602
2	CSS	7	408	144	3288
3	FXML	13	70	14	3178
4	SQL	1	38	7	257
SUM		89	1804	3497	15325

Bảng báo Use case bị fail

STT	Use case	Mô tả kịch bản kiểm thử	Điều kiện tiên quyết	Kết quả mong muốn	Kết quả thực tế
1	testLoanExistsInDatabaseAfterAdd	Khi thêm một loan mới vào cơ sở dữ liệu, hệ thống phải lưu trữ đầy đủ thông tin loan và cho phép truy xuất lại đúng loan đó bằng mã loan	<ul style="list-style-type: none"> - Tài liệu tồn tại và còn ít nhất một bản để mượn - Người dùng tồn tại trong hệ thống - LoanDAO hoạt động đúng với các hàm thêm và truy xuất loan 	<ul style="list-style-type: none"> - Truy xuất lại được loan vừa thêm (loan không phải giá trị null) - Số lượng sách mà loan lưu trữ phải trùng với số lượng người dùng đã mượn - Mã tài liệu trong loan phải trùng với mã tài liệu đã dùng để tạo loan 	Loan không tồn tại trong Database

2	testBorrowMoreThanAvailable	Người dùng cố gắng mượn số lượng sách lớn hơn số lượng sách hiện có trong hệ thống	<ul style="list-style-type: none"> - Tài liệu tồn tại và có đúng ba bản còn lại - Người dùng hợp lệ và có thể thực hiện thao tác mượn sách - LoanDAO có kiểm tra điều kiện số lượng trước khi thêm loan 	<ul style="list-style-type: none"> - Hệ thống từ chối tạo loan - Hàm thêm loan trả về giá trị báo lỗi (không tạo loan mới) 	Hệ thống vẫn cho phép mượn vượt quá số lượng
3	testEqualsAndHashCode	Kiểm tra chức năng so sánh hai tài liệu: hai tài liệu có toàn bộ thông tin giống nhau phải được xem là bằng nhau, còn tài liệu có thông tin khác thì phải được xem là khác	<ul style="list-style-type: none"> - Hai đối tượng tài liệu có mọi trường dữ liệu giống nhau hoàn toàn - Một đối tượng tài liệu khác có dữ liệu khác 	<ul style="list-style-type: none"> - Hệ thống xác nhận hai tài liệu giống nhau là bằng nhau - Hệ thống tạo mã băm giống nhau cho hai tài liệu giống nhau - Hệ thống xác nhận tài liệu khác là không bằng tài liệu gốc 	Hai tài liệu giống nhau nhưng hashCode không giống nhau
4	testLoginAsUserSuccess	Người dùng nhập đúng tên đăng nhập và mật khẩu, hệ thống thực hiện xác thực thành công và trả về thông tin của người dùng	<ul style="list-style-type: none"> - Tài khoản có tên đăng nhập "test05" tồn tại trong cơ sở dữ liệu - Mật khẩu "test" là mật khẩu đúng của tài khoản này 	<ul style="list-style-type: none"> - Hệ thống xác thực thành công - Hàm đăng nhập trả về đối tượng thuộc loại User, thể hiện rằng người dùng đăng nhập hợp lệ 	Hệ thống không cho phép đăng nhập