 **TRƯỜNG ĐẠI HỌC SƯ PHẠM KĨ THUẬT**

**TP. HỒ CHÍ MINH**

 **------ 🙠🕮🙢 ------**

**ĐỒ ÁN 1**

**ĐỀ TÀI: VIẾT CHƯƠNG TRÌNH CÀI ĐẶT CƠ CHẾ KIỂM SOÁT TRUY CẬP CHO MỘT HỆ THỐNG TẬP TIN GIẢ ĐỊNH**

**GVHD: Nguyễn Đăng Quang**

**Nguyễn Thị Xuân Mai 17110184**

**Võ Nguyễn Minh Hiền 17110134**

**TP. Hồ Chí Minh, tháng 12 năm 2019**

**MỤC LỤC**

[**DANH MỤC CÁC BẢNG** 1](#_Toc27643653)

[**DANH MỤC CÁC HÌNH** 2](#_Toc27643654)

[**I.** **Đặc tả** 3](#_Toc27643655)

[***1.*** ***Phần mềm dùng làm gì?*** 3](#_Toc27643656)

[***2.*** ***Hướng giải quyết vấn đề*** 3](#_Toc27643657)

[**2.1.** **Java Swing** 3](#_Toc27643658)

[**2.1.1.** **Swing là gì?** 3](#_Toc27643659)

[**2.1.2.** **Tính năng của Swing** 3](#_Toc27643660)

[**2.2.** **Mô hình MVC** 4](#_Toc27643661)

[**2.2.1.** **Mô hình MVC là gì?** 4](#_Toc27643662)

[**2.2.2.** **Ưu điểm và nhược điểm của MVC** 4](#_Toc27643663)

[*2.2.2.1.* *Ưu điểm* 4](#_Toc27643664)

[*2.2.2.2.* *Nhược điểm* 4](#_Toc27643665)

[**2.2.3.** **Tại sao phải sử dụng mô hình MVC?** 4](#_Toc27643666)

[**2.3.** **Access Control List** 4](#_Toc27643667)

[**2.3.1.** **ACLs là gì?** 4](#_Toc27643668)

[**2.3.2.** **Mục đích sử dụng ACLs** 5](#_Toc27643669)

[**2.3.3.** **Nguyên tắc hoạt động của ACLs** 5](#_Toc27643670)

[**2.3.4.** **Ưu điểm và nhược điểm của ACLs** 5](#_Toc27643671)

[*2.3.4.1.* *Ưu điểm* 5](#_Toc27643672)

[*2.3.4.2.* *Nhược điểm* 5](#_Toc27643673)

[**2.4.** **SQL Server** 5](#_Toc27643674)

[**2.4.1.** **SQL Server là gì?** 5](#_Toc27643675)

[**2.4.2.** **Mục đích sử dụng SQL Server** 6](#_Toc27643676)

[***3.*** ***Dữ liệu, thông tin đầu vào*** 6](#_Toc27643677)

[**3.1.** **Cây thư mục** 6](#_Toc27643678)

[**3.2.** **SQL Server** 6](#_Toc27643679)

[**II.** **Phân công công việc** 6](#_Toc27643680)

[**III.** **Thiết kế** 7](#_Toc27643681)

[***1.*** ***Thiết kế lớp*** 7](#_Toc27643682)

[**1.1.** **Các lớp trong package Connection** 7](#_Toc27643683)

[**1.2.** **Các lớp trong package Model** 7](#_Toc27643684)

[**1.3.** **Các form trong package Mypack** 7](#_Toc27643685)

[***2.*** ***Thuật toán*** 8](#_Toc27643686)

[***3.*** ***Thiết kế cơ sở dữ liệu (SQL Server)*** 9](#_Toc27643687)

[***4.*** ***Thiết kế giao diện*** 9](#_Toc27643688)

[**4.1.** **Login** 9](#_Toc27643689)

[**4.2.** **AddUsers** 10](#_Toc27643690)

[**4.3.** **Menu** 11](#_Toc27643691)

[**4.4.** **Account** 11](#_Toc27643692)

[**4.5.** **Folder** 12](#_Toc27643693)

[**4.6.** **ChangePass** 12](#_Toc27643694)

[**IV.** **Cài đặt và kiểm thử** 13](#_Toc27643695)

[***1.*** ***Cài đặt*** 13](#_Toc27643696)

[***2.*** ***Kiểm thử*** 13](#_Toc27643697)

[**2.1.** **Tính năng đăng nhập** 13](#_Toc27643698)

[**2.2.** **Tính năng thêm tài khoản truy cập** 13](#_Toc27643699)

[**2.3.** **Tính năng phân quyền thư mục cho người dùng** 14](#_Toc27643700)

[**2.3.1.** **Nếu tài khoản đăng nhập là root** 14](#_Toc27643701)

[**2.3.2.** **Nếu tài khoản đăng nhập là user** 15](#_Toc27643702)

[**2.4.** **Tính năng phân quyền thư mục, tập tin** 15](#_Toc27643703)

[**2.4.1.** **Thư mục của tài khoản đăng nhập** 16](#_Toc27643704)

[**2.4.2.** **Phân quyền thư mục cho các tài khoản** 16](#_Toc27643705)

[**2.5.** **Tính năng thay đổi mật khẩu** 17](#_Toc27643706)

[**V.** **Kết luận** 18](#_Toc27643707)

[***1.*** ***Đánh giá chung*** 19](#_Toc27643708)

[***2.*** ***Đánh giá ưu nhược điểm*** 19](#_Toc27643709)

[**2.1.** **Ưu điểm** 19](#_Toc27643710)

[**2.2.** **Nhược điểm** 19](#_Toc27643711)

[***3.*** ***Hướng phát triển*** 19](#_Toc27643712)

# **DANH MỤC CÁC BẢNG**

Bảng 1.1 Phân công công việc

Bảng 1.2 Liệt kê các lớp có trong package Connection

Bảng 1.3 Liệt kê các lớp có trong package Model

Bảng 1.4 Liệt kê các lớp có trong package Mypack

Bảng 1.5 Mô tả các phương thức trong lớp Connection

Bảng 1.6 Thiết kế cơ sở dữ liệu

# **DANH MỤC CÁC HÌNH**

Hình 1.1 Giao diện đăng nhập

Hình 1.2 Giao diện thêm tài khoản

Hình 1.3 Giao diện chính

Hình 1.4 Giao diện hiển thị user để phân quyền

Hình 1.5 Giao diện cây thư mục

Hình 1.6 Giao diện đổi mật khẩu

Hình 1.7 Đăng nhập không thành công – tài khoản hoặc mật khẩu sai

Hình 1.8 Thêm tài khoản người dùng thành công

Hình 1.9 Tính năng kiểm tra user đã tồn tại

Hình 2.1 Hiển thị tất cả tài khoản của các user còn lại

Hình 2.2 Hiển thị tài khoản các user còn lại ngoại trừ user đang đăng nhập và root

Hình 2.3 Thư mục của root luôn là Full – Control

Hình 2.4 User không có quyền truy cập vào các thư mục nó chưa được phân quyền

Hình 2.5 Phân quyền thư mục thuộc sở hữu cho các user khác

Hình 2.6 Mật khẩu bị trùng với mật khẩu cũ

Hình 2.7 Mật khẩu mới nhập lại khác với mật khẩu mới

Hình 2.8 Thay đổi mật khẩu thành công

1. **Đặc tả**
2. ***Phần mềm dùng làm gì?***

Phần mềm giúp kiểm soát quyền truy cập vào hệ thống tập tin, thư mục giả định của các chủ thể (root, user) bằng phương pháp xây dựng cây thư mục có kết hợp với kiểm soát truy cập gồm các quyền: owner, read, write, execute – là tính năng của bảo mật giúp kiểm soát người dùng, tương tác giữa hệ thống và các tài nguyên khác. Các phương pháp xác định ai có thể truy cập gì đến mức độ nào do chủ sở hữu tập tin, thư mục phân quyền, hệ thống đã có cơ chế chứng thực người dùng.

1. ***Hướng giải quyết vấn đề***

* Sử dụng ngôn ngữ Java để xây dựng phần mềm – cụ thể là Java Swing để: xây dựng giao diện đăng nhập, thêm tài khoản, sửa mật khẩu, tạo cây thư mục, tập tin.
* Xây dựng mô hình MVC (Model – View – Controller)
* Sử dụng ACL (Access – Control List) để phân quyền truy cập các hệ thống tập tin đối với đối tượng người dùng.
* Sử dụng SQL Server để lưu Access – Control quyền của các thư mục, tập tin, tài khoản người dùng.
  1. **Java Swing**
     1. **Swing là gì?**
* Java Swing là một phần của Java Foundation Classes (JFC) được sử dụng để tạo các ứng dụng window – based. Nó được xây dựng trên API AWT (Abstract Windowing Toolkit) và được viết hoàn toàn bằng Java.
* Không giống như AWT, Java Swing cung cấp các thành phần không phụ thuộc vào nền tảng và nhẹ hơn.
* Gói javax.swing cung cấp các lớp cho java swing API như Jbutton, JtextField, JtextArea, JradioButton, Jcheckbox,....
  + 1. **Tính năng của Swing**
* Rich Controls – Swing cung cấp một bộ điều khiển nâng cao phong phú như Tree, TabbedPane, thanh trượt, colorpicker và điều khiển bảng.
* Tùy biến cao – các điều khiển xoay có thể được tùy chỉnh theo một cách rất dễ dàng và độc lập với biểu diễn bên trong.
* Pluggable look – and – feel – Swing dựa nhìn GUI Application và có thể thay đổi thời gian chạy, dựa trên các giá trị có sẵn.
  1. **Mô hình MVC**
     1. **Mô hình MVC là gì?**

MVC là viết tắt của Model – View – Controller. Là một kiến trúc phần mềm hay mô hình thiết kế được sử dụng trong kỹ thuật phần mềm. Cụ thể nó là mô hình phân bố source code thành ba phần, mỗi thành phần có một nhiệm vụ riêng biệt và độc lập với các thành phần khác.

* + 1. **Ưu điểm và nhược điểm của MVC**
       1. *Ưu điểm*

Thể hiện tính chuyên nghiệp trong lập trình, phân tích thiết kế. Do được chia thành các thành phần độc lập nên giúp phát triển ứng dụng nhanh, đơn giản, dễ nâng cấp, bảo trì...

* + - 1. *Nhược điểm*

Đối với dự án nhỏ việc áp dụng mô hình MVC gây cồng kềnh, tốn thời gian trong quá trình phát triển. Tốn thời gian trung chuyển dữ liệu của các thành phần.

* + 1. **Tại sao phải sử dụng mô hình MVC?**

Để lập trình chuyên nghiệp, làm việc trong một nhóm nhiều người, việc áp dụng mô hình trong thiết kế là điều bắt buộc. MVC là một mô hình khá đơn giản và thích hợp cho những người chưa nhiều kinh nghiệm.

* 1. **Access Control List**
     1. **ACLs là gì?**

ACLs (Access control lists) hay còn gọi là access lists, là một danh sách các quyền gắn liền với một đối tượng. ACL chỉ định người dùng hoặc quy định hệ thống nào được cấp quyền truy cập vào đối tượng, cũng như những hoạt động nào được phép trên các đối tượng nhất định. Mỗi mục ACL điển hình chỉ định một đối tượng và thao tác.

* + 1. **Mục đích sử dụng ACLs**
* Lưu lại danh sách các quyền mà tài khoản được phân quyền như: owner, read, write, execute.
* Khả năng điểu khiển truy cập người dùng
  + 1. **Nguyên tắc hoạt động của ACLs**
* Khi được truy xuất, ACL sẽ được đọc và thi hành từng dòng một từ trên xuống dưới, dòng nào chứa thông tin khớp với thông tin của gói tin đang được xem xét, dòng ấy sẽ được thi hành ngay và các dòng còn lại sẽ được bỏ qua.
* Một nguyên tắc nữa cần lưu ý là các dòng mới được khai báo sẽ được tự động thêm vào cuối ACL, tuy nhiên dòng cuối cùng thực sự của một ACL luôn là một dòng ngầm định “deny” tất cả, có nghĩa là nếu gói tin không match bất cứ dòng nào đã khai báo của ACL, nó sẽ bị deny.
  + 1. **Ưu điểm và nhược điểm của ACLs**
       1. *Ưu điểm*
* Tốc độ xử lý gói tin nhanh
* Mềm dẽo trong triển khai chính sách bảo mật
  + - 1. *Nhược điểm*
* Cấu hình phức tạp
* Không thể ngăn chặn tấn công ở tầng ứng dụng
* Giới hạn khả năng ghi lại sự kiện: chúng không ghi lại nội dung thực sự của gói tin để xác định ai cố gắng làm gì.
  1. **SQL Server** 
     1. **SQL Server là gì?**
* SQL Server là ngôn ngữ phi thủ tục, không yêu cầu cách thức truy cập cơ sở dữ liệu như thế nào. Tất cả các thông báo của SQL rất dễ dàng sử dụng và ít mắc lỗi.
* SQL Server cung cấp các tập lệnh phong phú cho các công việc hỏi đáp dữ liệu như:
* Chèn, xóa và cập nhật các hàng trong 1 quan hệ
* Thêm, xóa và sửa đổi các đối tượng của cở sở dữ liệu
* Điều khiển việc truy cấp tới cơ sở dữ liệu và các đối tượng của cơ sở dữ liệu để đảm bảo tính bảo mật, tính nhất quán và sự ràng buộc của cở sở dữ liệu.
  + 1. **Mục đích sử dụng SQL Server**
* Quản trị người dùng: tiện ích trực quan giúp quản lý người dùng. Tiện ích cho phép quản trị viên cơ sở dữ liệu dễ dàng thêm mới và xóa người dùng hiện tại khi cần; cấp, hủy phân quyền cũng như xem hồ sơ người dùng.
* Quản lý thư mục, tập tin: tiện ích trực quan giúp lưu đường dẫn của thư mục tập tin. Tiện ích cho phép quản trị viên cơ sở dữ liệu dễ dàng thêm mới và xóa thư mục tập tin thuộc quyền sở hữu của người dùng; cấp, hủy quyền của người dùng khác đối với thư mục, tập tin thuộc quyền sở hữu của mình.

1. ***Dữ liệu, thông tin đầu vào***
   1. **Cây thư mục**

* Xây dựng giao diện Tree để insert cây thư mục vào
* Viết code để lấy cây thư mục từ một ổ đĩa hay thư mục bất kì nào có sẵn trong máy đẩy lên form
  1. **SQL Server**
* Tạo các table để chứa thông tin:
  + Table Account: chứa thông tin các tài khoản
  + Table Folder: chứa đường dẫn và tên của thư mục
  + Table Managementfolder: chứa quyền các thư mục của user đó
* Sử dụng StoreProcedure để thực thi các việc cần thiết như insert, update quyền…

1. **Phân công công việc**

Bảng 1.1 Phân công công việc

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Công việc** | **Nguyễn Thị Xuân Mai** | **Võ Nguyễn Minh Hiền** |
| 1 | Phân tích và tìm hiểu đề tài | x | x |
| 2 | Thiết kế giao diện | x |  |
| 3 | Thiết kế database |  | x |
| 4 | Viết các chương trình chính | x | x |
| 5 | Kiểm tra và chỉnh sửa |  | x |
| 6 | Viết báo cáo | x |  |

1. **Thiết kế**
2. ***Thiết kế lớp***
   1. **Các lớp trong package Connection**

Bảng 1.2 Liệt kê các lớp có trong package Connection

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên lớp** | **Mục đích chính của lớp trong chương trình** |
| 1 | Connect | Kết nối chương trình với cơ sở dữ liệu (ở đây là SQL Server – dùng Driver), giúp ta có thể lấy dữ liệu từ cơ sở dữ liệu lên, và đổ dữ liệu từ chương trình xuống. |

* 1. **Các lớp trong package Model**

Bảng 1.3 Liệt kê các lớp có trong package Model

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên lớp** | **Mục đích chính của lớp trong chương trình** |
| 1 | Account | Mô tả thông tin đối tượng (userid, username, pass, name, role), tương tác với database |
| 2 | Permission | Mô tả thông tin đối tượng (path, userid, per), tương tác với database |

* 1. **Các form trong package Mypack**

Bảng 1.4 Liệt kê các lớp có trong package Mypack

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên lớp** | **Mục đích chính của form trong chương trình** |
| 1 | fAccount | Chứa tài khoản của các user cần được phân quyền |
| 2 | fAddusers | Thêm tài khoản người dùng |
| 3 | fChangePass | Đổi mật khẩu của người dùng |
| 4 | FileBrowser | LoadFolder từ laptop lên và phân quyền truy cập |
| 5 | fLogin | Lấy dữ liệu từ database lên và đăng nhập vào hệ thống với quyền đã được phân |
| 6 | fMenu | Chứa các nút để thực thi các chức năng của hệ thống |

1. ***Thuật toán***

Bảng 1.5 Mô tả các phương thức trong lớp Connection

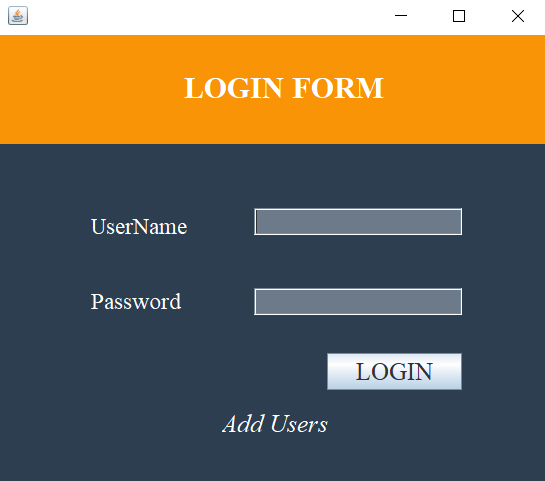
|  |  |  |
| --- | --- | --- |
| **STT** | **Tên phương thức** | **Mục đích** |
| 1 | public static void WriteFolder (String path, String name) throws SQLException | Ghi đường dẫn của những thư mục chưa có xuống database |
| 2 | public static void InsertPermission (String path, int id, String per) throws SQLException | Nếu user đó đã có quyền trong table managementfolder thì update lại quyền mới được phân quyền, ngược lại thì thêm mới |
| 3 | public static void InitPermission(int id) throws SQLException | Thêm quyền cho những tài khoản user mới tạo - ở đây mặc định là 0000 |
| 4 | public static List<Permission> GetPer (int userid) throws SQLException | Lấy danh sách các quyền của user theo userid để đổ vào một list |
| 5 | public static Account Login (String username, String pass) throws SQLException | Dùng để đăng nhập và kiểm tra tài khoản hay mật khẩu có khớp không |
| 6 | public static Account GetUser (String username) throws SQLException | Lấy thông tin của user theo username để add quyền vào list |
| 7 | public static List<Account> GetUsers() throws SQLException | Lấy danh sách user có trong table Account |

1. ***Thiết kế cơ sở dữ liệu (SQL Server)***

Bảng 1.6 Thiết kế cơ sở dữ liệu

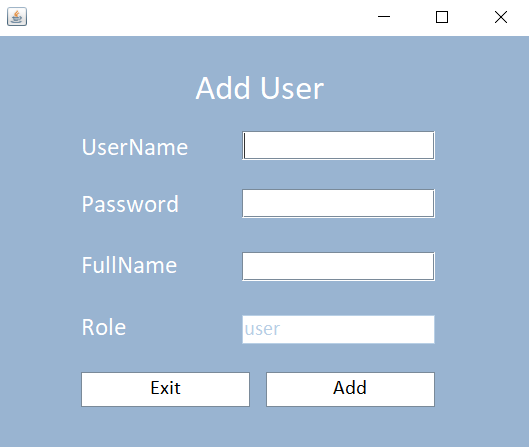
|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Bảng** | **Thuộc tính và kiểu dữ liệu trong bảng** | **Ý nghĩa thuộc tính** |
| 1 | account | id int | ID của người dùng |
| username varchar(255) | Tên đăng nhập |
| password varchar(255) | Mật khẩu của người dùng |
| fullname varchar(255) | Tên người dùng |
| role varchar(255) | Chức vụ của người dùng |
| 2 | folder | path varchar(255) | ID – đường dẫn của thư mục, tập tin |
| foldername varchar(255) | Tên của thư mục, tập tin |
| 3 | managementfolder | userid int | ID của người dùng |
| folderid varchar(255) | ID của thư mục, tập tin |
| permission varchar(4) | Phân quyền thư mục, tập tin |

1. ***Thiết kế giao diện***
   1. **Login**



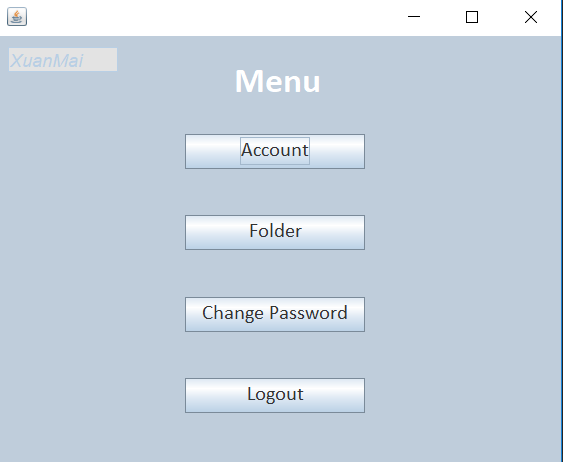
Hình 1.1 Giao diện đăng nhập

* 1. **AddUsers**



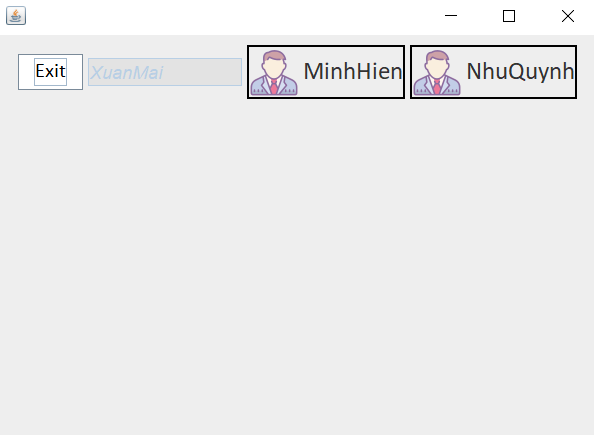
Hình 1.2 Giao diện thêm tài khoản

* 1. **Menu**



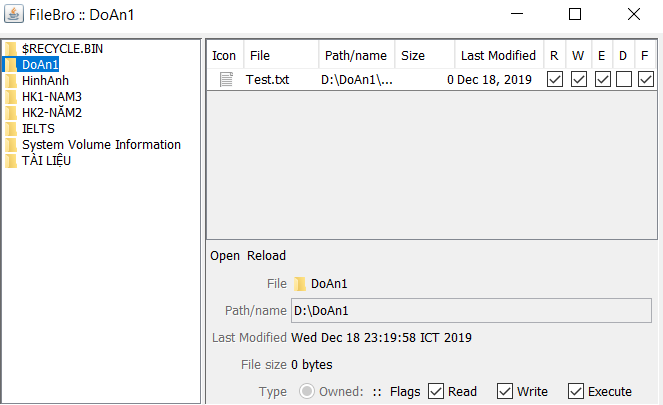
Hình 1.3 Giao diện chính

* 1. **Account**



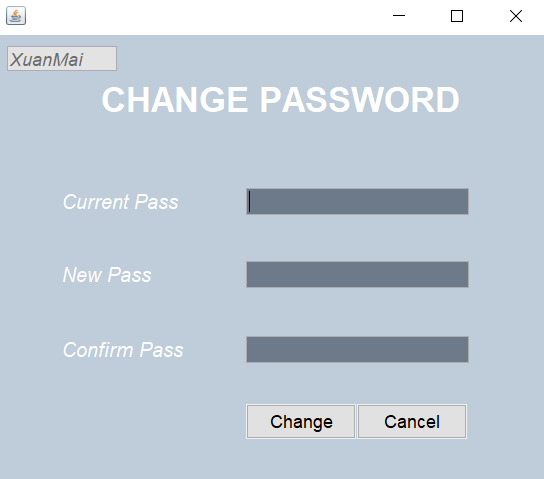
Hình 1.4 Giao diện hiển thị user để phân quyền

* 1. **Folder**



Hình 1.5 Giao diện cây thư mục

* 1. **ChangePass**

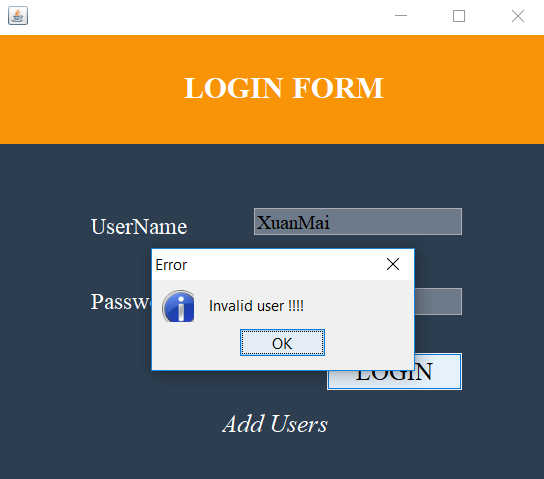


Hình 1.6 Giao diện đổi mật khẩu

1. **Cài đặt và kiểm thử**
2. ***Cài đặt***

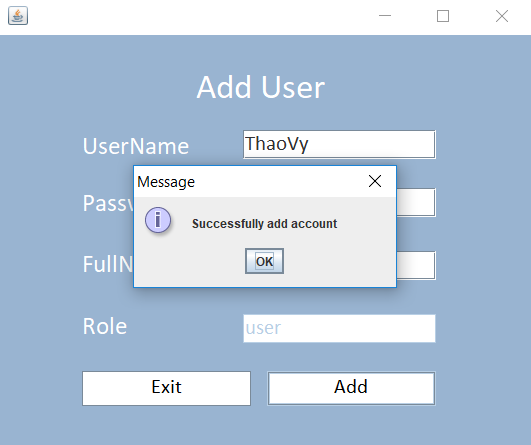
* Java Platform SE – JDK version 8.0.2310.11
* SQL Server version 18.2
* WindowBuilder Pro version 1.9.2
* SQLServer – Connector
* Eclipse EE version 4.13

1. ***Kiểm thử***
   1. **Tính năng đăng nhập**

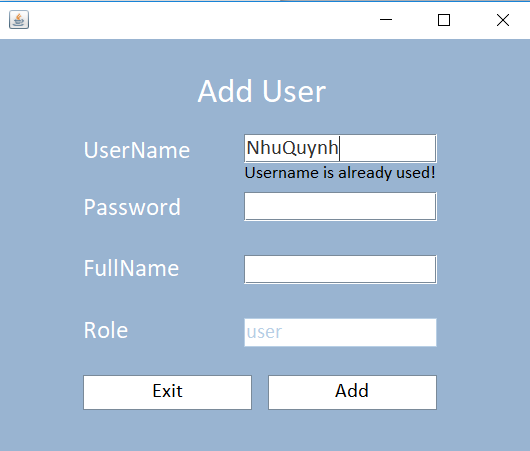


Hình 1.7 Đăng nhập không thành công – tài khoản hoặc mật khẩu sai

* 1. **Tính năng thêm tài khoản truy cập**

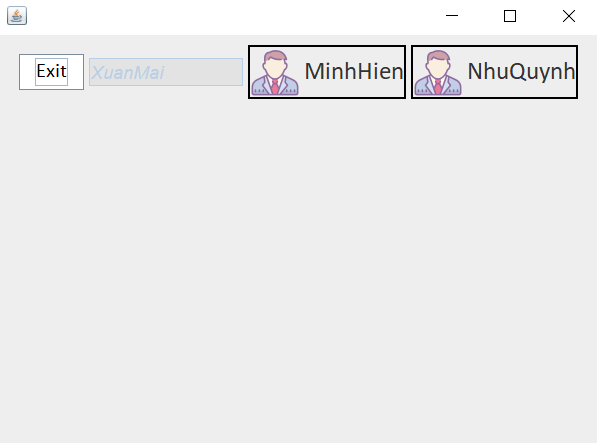


Hình 1.8 Thêm tài khoản người dùng thành công



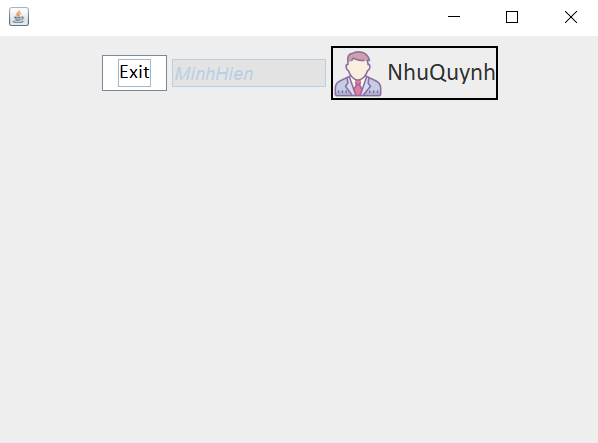
Hình 1.9 Tính năng kiểm tra user đã tồn tại

* 1. **Tính năng phân quyền thư mục cho người dùng**
     1. **Nếu tài khoản đăng nhập là root**



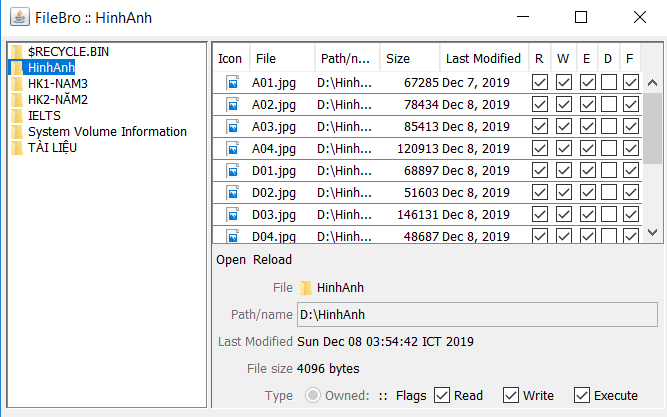
Hình 2.1 Hiển thị tất cả tài khoản của các user còn lại

* + 1. **Nếu tài khoản đăng nhập là user**

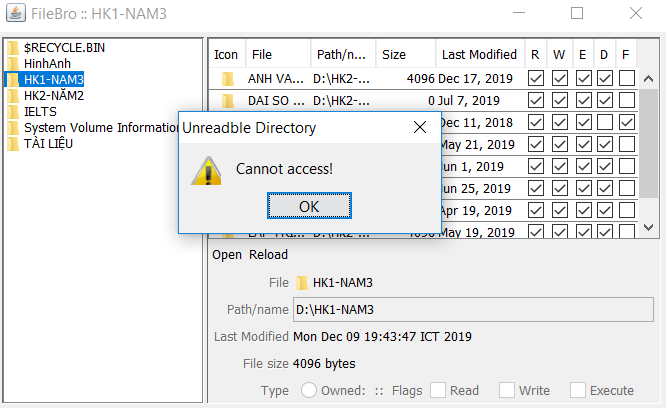


Hình 2.2 Hiển thị tài khoản các user còn lại ngoại trừ user đang đăng nhập và root

* 1. **Tính năng phân quyền thư mục, tập tin**
     1. **Thư mục của tài khoản đăng nhập**

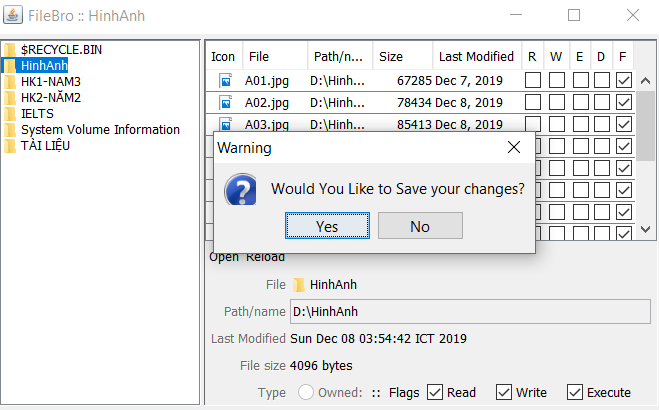


Hình 2.3 Thư mục của root luôn là Full – Control



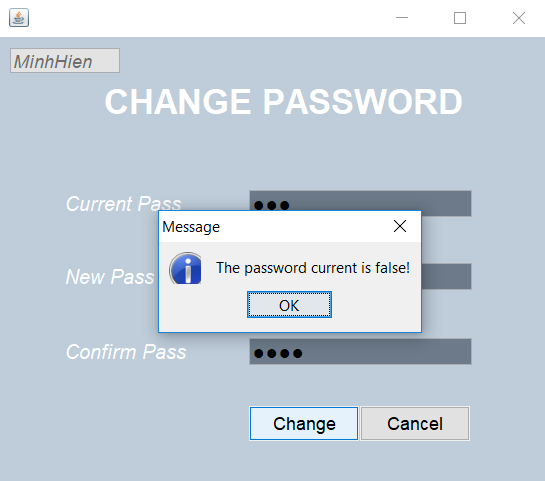
Hình 2.4 User không có quyền truy cập vào các thư mục nó chưa được phân quyền

* + 1. **Phân quyền thư mục cho các tài khoản**

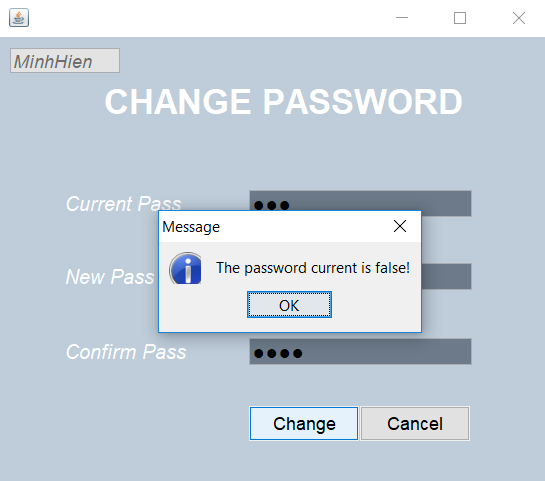


Hình 2.5 Phân quyền thư mục thuộc sở hữu cho các user khác

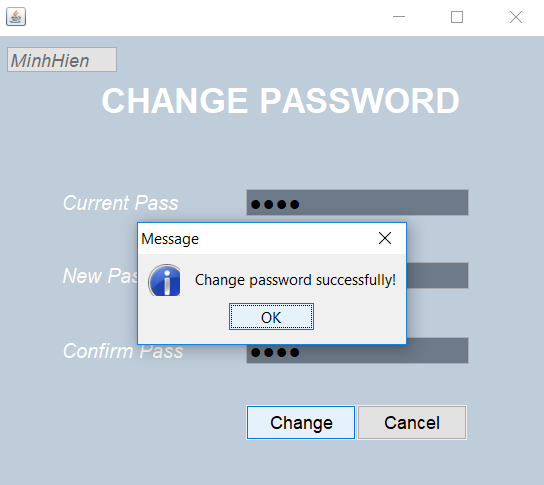
* 1. **Tính năng thay đổi mật khẩu**



Hình 2.6 Mật khẩu bị trùng với mật khẩu cũ



Hình 2.7 Mật khẩu mới nhập lại khác với mật khẩu mới



Hình 2.8 Thay đổi mật khẩu thành công

1. **Kết luận**
2. ***Đánh giá chung***

Đồ án hoàn thành được 70% mục tiêu đề ra

1. ***Đánh giá ưu nhược điểm***
   1. **Ưu điểm**

* Giao diện đơn giản, thân thiện với người dùng
* Phân quyền được tài khoản đăng nhập
* Kiểm soát được quyền truy cập của người dùng đối với các thư mục, tập tin
  1. **Nhược điểm**
* Giao diện còn sơ sài
* Chưa cập nhật được hết quyền thư mục con theo thư mục cha nếu load ổ có thư mục quá nhiều
* Code chưa chặt chẽ làm cho khi thao tác thì có nhiều lúc thư mục được phân quyền không như mong muốn

1. ***Hướng phát triển***

* Xây dựng giao diện bắt mắt hơn
* Chỉnh sửa lại code cho logic và sạch
* Thêm chủ thể groupuser