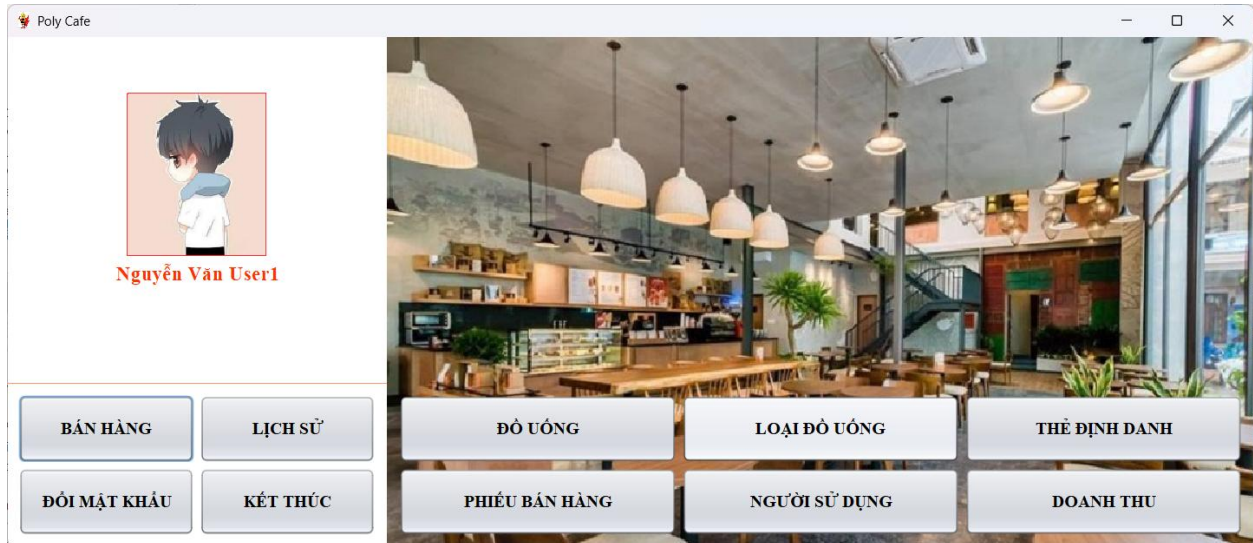
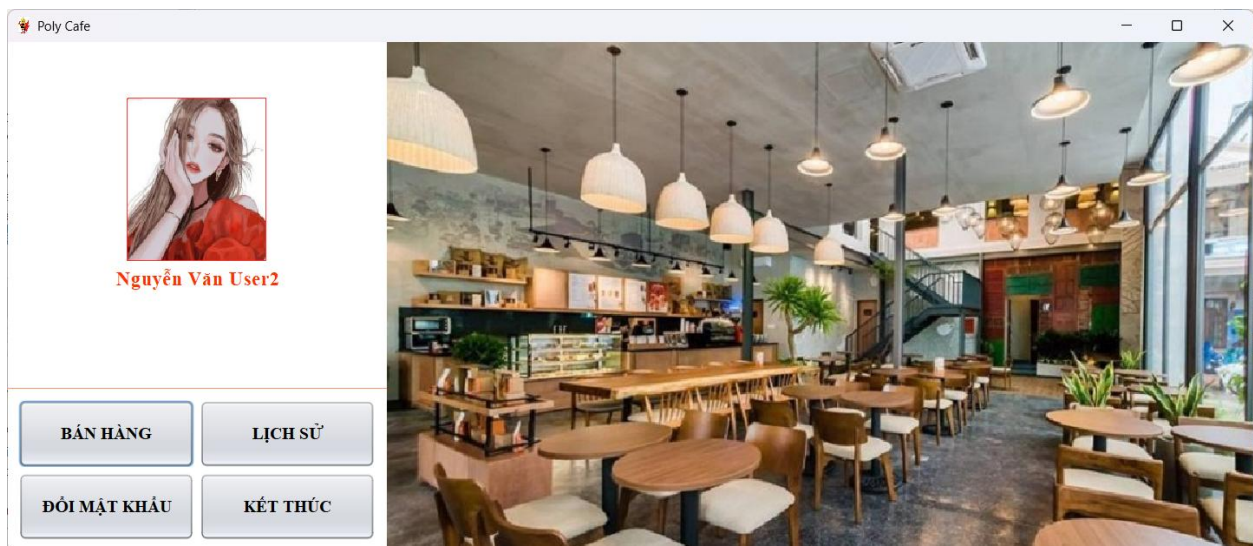


THIẾT KẾ GIAO DIỆN CHO CỬA SỔ CHÍNH POLYCAFEJFRAME



Hình 1: Đăng nhập với vai trò Quản lý



Hình 2: Đăng nhập với vai trò Nhân viên bán hàng

ĐẶT TÊN CÁC THÀNH PHẦN GIAO DIỆN

Text/Title	Control	Name
Poly Cafe	JFrame	PolyCafeJFrame
GIAO DIỆN DÙNG CHUNG		
Hình ảnh người đăng nhập	JLabel	lblPhoto
Họ và tên người đăng nhập	JLabel	lblFullname
BÁN HÀNG	JButton	btnSales

LỊCH SỬ	JButton	btnHistory
ĐỔI MẬT KHẨU	JButton	btnChangePassword
KẾT THÚC	JButton	btnExit
GIAO DIỆN QUẢN LÝ		
ĐỒ UỐNG	JButton	btnDrinkManager
LOẠI ĐỒ UỐNG	JButton	btnCategoryManager
THẺ ĐỊNH DANH	JButton	btnCardManager
PHIẾU BÁN HÀNG	JButton	btnBillManager
NGƯỜI SỬ DỤNG	JButton	btnUserManager
DOANH THU	JButton	btnRevenueManager

HOẠT ĐỘNG TƯƠNG TÁC

TƯƠNG TÁC	XỬ LÝ TƯƠNG TÁC
Window.init	Hiện thì cửa sổ chào Hiện thị cửa sổ đăng nhập Hiện thị họ tên và hình ảnh của người sử dụng Hiện thị icon của cửa sổ Di chuyển cửa sổ ra trung tâm màn hình
btnExit.Click	Kết thúc làm việc
btnChangePassword.Click	Mở cửa sổ đổi mật khẩu
btnSales.Click	Mở cửa sổ bán hàng
btnHistory.Click	Mở cửa sổ lịch sử bán hàng của người đăng nhập
btnDrinkManager.Click	Mở cửa sổ quản lý đồ uống
btnCategoryManager.Click	Mở cửa sổ quản lý loại đồ uống
btnCardManager.Click	Mở cửa sổ quản lý thẻ định danh
btnBillManager.Click	Mở cửa sổ quản lý phiếu bán hàng

btnUserManager.Click	Mở cửa sổ quản lý tài khoản người sử dụng
btnRevenueManager.Click	Mở cửa sổ quản lý doanh thu

HƯỚNG DẪN THỰC HIỆN:

- Tạo interface poly.cafe.ui.PolyCafeController khai báo các phương thức xử lý tương tác người sử dụng

```
public interface PolyCafeController {
    /**
     * Hiển thị cửa sổ chào
     * Hiển thị cửa sổ đăng nhập
     * Hiển thị thông tin user đăng nhập
     * Disable/Enable các thành phần tùy thuộc vào vai trò đăng nhập
     */
    void init();

    default void exit(){
        if(XDialog.confirm("Bạn muốn kết thúc?")){
            System.exit(0);
        }
    }
    default void showJDialog(JDialog dialog){
        dialog.setLocationRelativeTo(null);
        dialog.setVisible(true);
    }
    default void showWelcomeJDialog(JFrame frame){
        this.showJDialog(new WelcomeJDialog(frame, true));
    }
    default void showLoginJDialog(JFrame frame){
        this.showJDialog(new LoginJDialog(frame, true));
    }
    default void showChangePasswordJDialog(JFrame frame){
        this.showJDialog(new ChangePasswordJDialog(frame, true));
    }
    default void showSalesJDialog(JFrame frame){
        this.showJDialog(new SalesJDialog(frame, true));
    }
    default void showHistoryJDialog(JFrame frame){
        this.showJDialog(new HistoryJDialog(frame, true));
    }

    default void showDrinkManagerJDialog(JFrame frame){
        this.showJDialog(new DrinkManagerJDialog(frame, true));
    }
}
```

```

    }
    default void showCategoryManagerJDialog(JFrame frame){
        this.showJDialog(new CategoryManagerJDialog(frame, true));
    }
    default void showCardManagerJDialog(JFrame frame){
        this.showJDialog(new CardManagerJDialog(frame, true));
    }
    default void showBillManagerJDialog(JFrame frame){
        this.showJDialog(new BillManagerJDialog(frame, true));
    }
    default void showUserManagerJDialog(JFrame frame){
        this.showJDialog(new UserManagerJDialog(frame, true));
    }
    default void showRevenueManagerJDialog(JFrame frame){
        this.showJDialog(new RevenueManagerJDialog(frame, true));
    }
}

```

- Gắn kết Controller vào giao diện PolyCafeJFrame để xử lý tương tác

```

public final class PolyCafeJFrame extends javax.swing.JFrame implements
PolyCafeController

```

- Viết mã nguồn cho init()

```

@Override
public void init() {
    this.setIconImage(XIcon.getIcon("trump-small.png").getImage());
    this.setLocationRelativeTo(null);

    this.showWelcomeJDialog(this);
    this.showLoginJDialog(this);
}

```

- Kết nối các sự kiện với các phương thức điều khiển

```

public PolyCafeJFrame() {
    initComponents();
    this.init(); // được gọi khi cửa sổ chưa được mở ra
}

private void btnHistoryActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    this.showHistoryJDialog(this); // được gọi khi click vào nút btnHistory
}
....

```