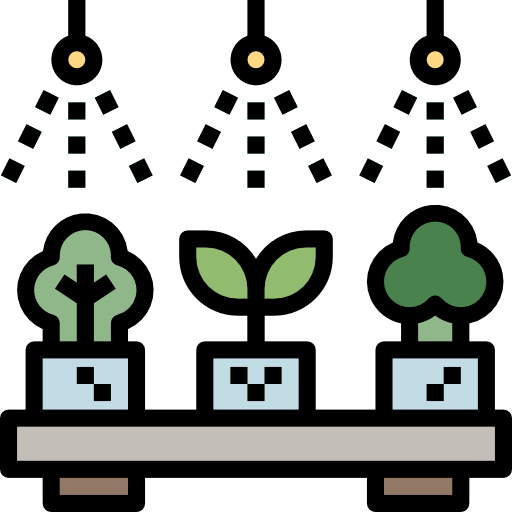
**FarmSys FPoly**

Development by ProTeam

**Document Phần Mềm  
Quản Lý Nông Trại Thủy Canh**

Ngày 6 tháng 11 năm 2021

# Tổng quan về sản phẩm

FarmSys là phầm mềm quản lý nông trại 4.0. Phần mềm giúp chủ trang trại dễ dàng quản lý tình trạng hoạt động nông trại của mình một cách dễ dàng. Bên cạnh đó, còn giúp cho người tiêu dùng cuối có thể đễ dàng truy xuất nguồn gốc và quy trình chăm sóc nông sản.

## Development Team:

* Leader: Vương Nguyên Trung
* Member: Nguyễn Huỳnh Đông Triều
* Member: Đinh Quang Tuấn
* Member: Trần Tiến Dũng
* Member: Nguyễn Diệp Thế Tri
* Member: Giáp Văn Hải

Trung Disign, Triều Coder, Tuấn Data, Dũng Slide, Tri Conan - Hải ….

### Teacher:

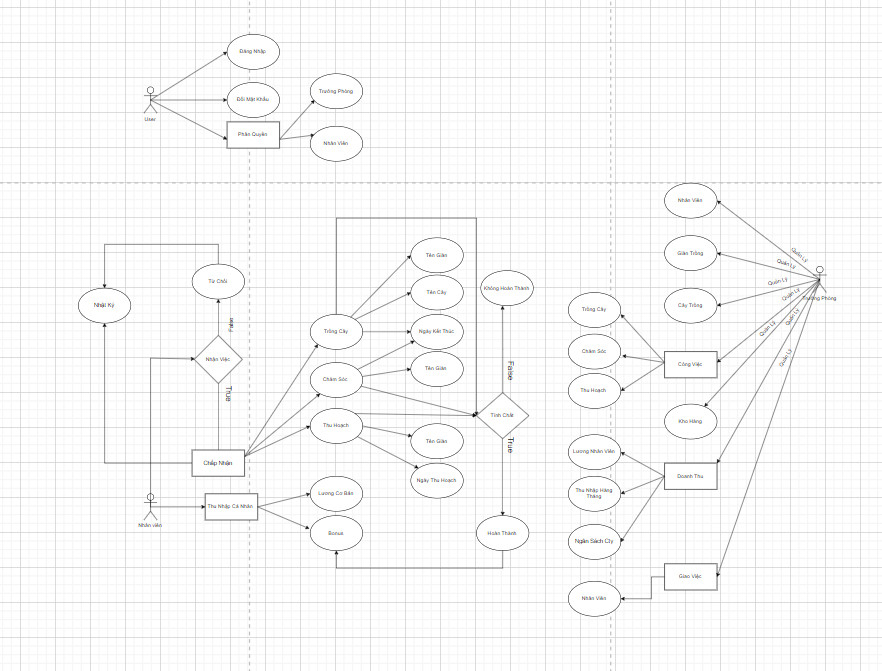
* Thầy: Thầy Nguyễn Đỗ Anh Khoa

# 

1. **PHÂN TÍCH**
   1. **Hiện trạng**

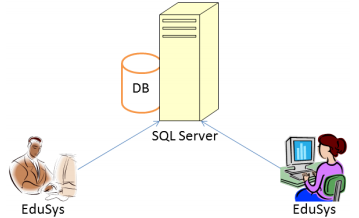
* Các nông trại thủy canh ở Việt Nam đang hoạt động sản xuất còn lạc hậu, chủ yếu theo kinh nghiệm cảm tính cá nhân. Các chủ nông trại thiếu công cụ quản lý nên việc quản lý trang trại trở nên khó khăn nếu tăng quy mô hoạt động hoặc các hoạt động đa dạng hóa nông sản.
* Người tiêu dùng gặp khó khăn trong quá trình chọn lựa nông sản sạch của bữa ăn hằng ngày vì không truy xuất nguồn gốc nông sản.
* Thấy được nhu cầu của người dùng ProTeam (Nhóm 1) đã xây dựng thành công chương trình Quản lý nông trại thủy canh FarmSys.
  1. **Yêu cầu hệ thống**
     1. **Yêu cầu chức năng nghiệp vụ**
* Quản lý nhân viên
* Quản lý cây trồng
* Quản lý dàn trồng
* Quản lý công việc của nhân viên
* Xuất báo cáo, nhật ký công việc
* Xuất thông tin sản phẩm
* Xem hình ảnh nông trại
  + 1. **Yêu cầu vể bảo mật**
* Tất cả mọi thành viên phải đăng nhập mới sử dụng được.
* Quản lý được phép thực hiện tất cả các chức năng.
* Nhân viên không được phép xóa dữ liệu và chỉ sử dụng được các chức năng mà người quản lý nông trại cấp phép.
  + 1. **Yêu cầu về môi trường công nghệ**
* Ứng dụng phải được thực với công nghệ Swing và JDBC chạy trên mọi hệ điều hành với môi trường JDK tối thiểu 1.8
* Hệ quản trị CSDL SQL Server 2008 trở lên
* OpenCV 2.4.9
  1. **Sơ đồ USE CASE**

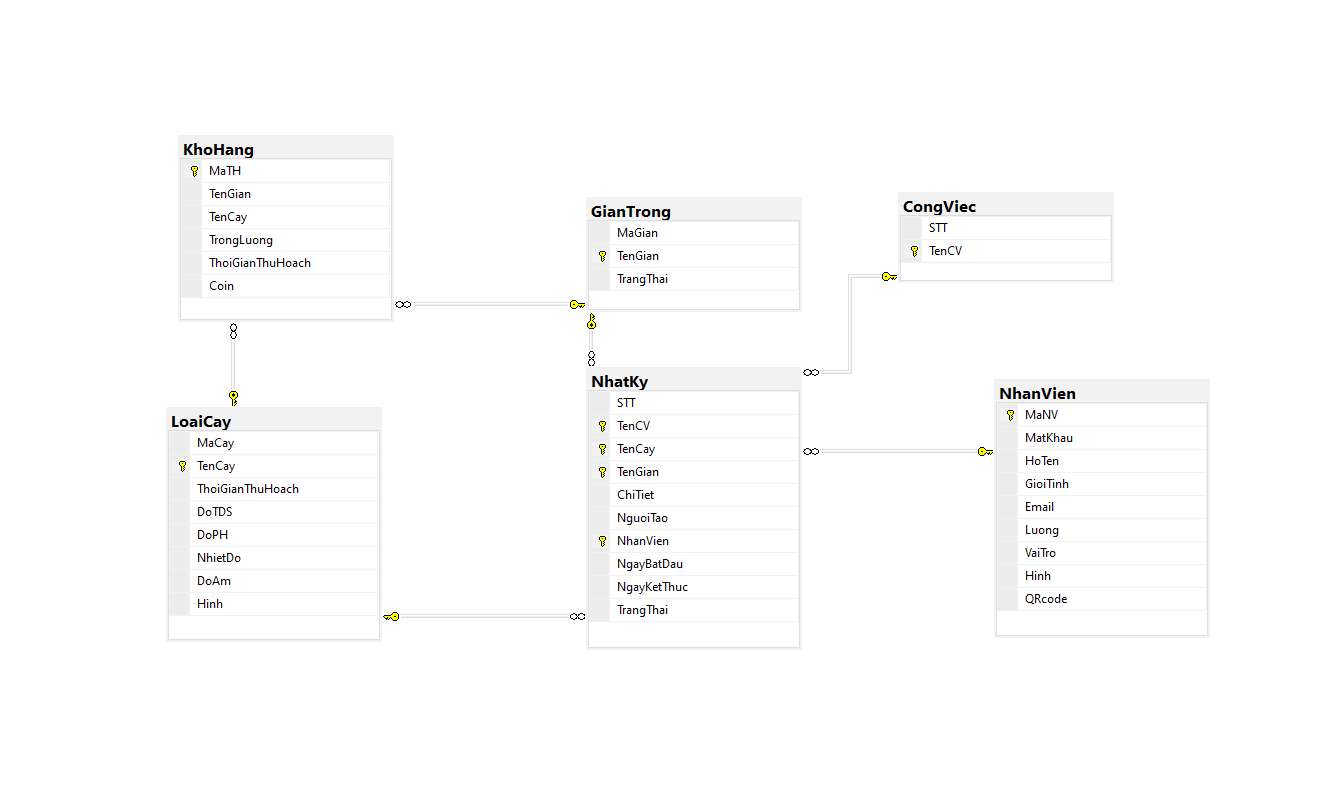
Use case là sơ đồ tổng quan về mặt chức năng và phân vai trò người sử dụng. Dựa vào yêu cầu hệ thống của khách hàng, chúng ta có thể phác thảo sơ đồ use case như sau.



* **CHI TIẾT CÁC CHỨC NĂNG:**
  + Mỗi chức năng quản lý bao gồm các chức năng con
    - Chức năng tổng hợp thống kê số liệu
      * Bảng điểm theo khóa học phải có cấu trúc là:
        + Mã người học
        + Họ và tên
        + Điểm
      * Tổng hợp điểm từng chuyên đề
        + Chuyên đề
        + Số học viên
        + Điểm cao nhất
        + Điểm thấp nhất
        + Điểm trung bình
      * Doanh thu từng chuyên đề theo năm phải có cấu trúc:
        + Chuyên đề
        + Số khóa
        + Số học viên
        + Doanh thu
        + Học phí thấp nhất
        + Học phí cao nhất
      * Thống kê số người học từng năm phải có cấu trúc:
        + Năm
        + Số học viên
        + Ngày đăng ký sớm nhất
        + Ngày đăng ký muộn nhất
      * Chức năng đăng nhập và đăng xuất
        + Đăng nhập: để sử dụng phần mềm, bắt buộc phải đăng nhập
        + Đăng xuất: đăng nhập lại với người dùng khác hoặc dừng lại đi đâu đó

1. **THIẾT KẾ**
   1. **Mô hình triển khai**

* Ứng dụng phần mềm được xây dựng để phục vụ cho nhiều người dùng nhưng cơ sở dữ liệu thì lưu trữ tập trung
* 
* Cần một máy cài SQL Server 2008+. Máy này cần hệ điều hành window xp trở lên
* Các máy nhân viên phòng đào tạo cài phần mềm FarmSys. Các máy nhân viên cần JDK 1.8+ với hệ điều hành bất ký
  1. **Thiết kế cơ sở dữ liệu**
     1. **Sơ đồ quan hệ thực thể**
        + Trong hệ thống đã được mô tả trên chúng ta thấy có các thực thể sau
          - Cây trồng: quản lý thông tin cây trồng
          - Giàn trồng: quản lý thông tin giàn trồng
          - Công việc: quản lý thông tin công việc
          - Nhật ký: quản lý lịch sử hoạt động của từng công việc do chủ nông trại đưa ra.
          - Kho hàng: quản lý số lượng nông sản thu hoạch được và tình trạng doanh thu bán hàng.
          - Nhân viên: quản lý thông tin tài khoản chủ trang trại và nhân viên.
        + Sau khi phân tích chúng ta có thể thấy các thực thể có mối quan hệ với nhau như sơ đồ ERD sau



* + - * Yêu cầu ràng buộc quan hệ:
        + Tất cả các relationship đều là CASCADE (cập nhật dây chuyền)
        + Tất cả các relationship đều là NO ACTION ngoại trừ KhoaHoc-HocVien là CASCADE
    1. **Sơ đồ quan hệ thực thể**
       1. **Nhân viên**

| **Tên cột** | **Kiểu** | **Ràng buộc** | **Ghi chú** |
| --- | --- | --- | --- |
| MaNV | varchar(15) | PK, NOT NULL | Mã nhân viên |
| MatKhau | varchar(16) | NOT NULL | Mật khẩu |
| HoTen | nvarchar(50) | NOT NULL | Họ tên |
| GioiTinh | bit |  | Giới tính |
| Email | nvarchar(30) | NOT NULL | Email |
| Luong | int |  | Lương |
| VaiTro | bit | NOT NULL | Vai trò |
| Hinh | varchar(30) |  | Hình |
| QRcode | varchar(40) |  | Mã QR Code mã hóa |

* + - 1. **Cây trồng**

| **Tên cột** | **Kiểu** | **Ràng buộc** | **Ghi chú** |
| --- | --- | --- | --- |
| MaCay | int | NOT NULL | Tự tăng |
| TenCay | nvarchar(30) | PK, NOT NULL | Tên cây |
| ThoiGianThuHoach | int | NOT NULL | Thời gian thu hoạch |
| DoTDS | float |  | Nồng độ dinh dưỡng |
| DoPH | float |  | Độ PH |
| NhietDo | float |  | Nhiệt độ |
| DoAm | float |  | Độ ẩm |
| Hinh | varchar(30) | NOT NULL | Hình |

* + - 1. **Giàn trồng**

| **Tên cột** | **Kiểu** | **Ràng buộc** | **Ghi chú** |
| --- | --- | --- | --- |
| MaGian | int |  | Mã giàn trồng |
| TenGian | nvarchar(30) | PK | Tên giàn trồng |
| TrangThai | bit |  | Trạng thái |

* + - 1. **Công việc**

| **Tên cột** | **Kiểu** | **Ràng buộc** | **Ghi chú** |
| --- | --- | --- | --- |
| STT | int |  | Số thứ tự |
| TenCV | nvarchar(30) | PK | Tên công việc |

* + - 1. **NhatKy**

| **Tên cột** | **Kiểu** | **Ràng buộc** | **Ghi chú** |
| --- | --- | --- | --- |
| STT | int |  | Số thứ tự |
| TenCV | nvarchar(30) | PK | Tên công việc |
| TenCay | nvarchar(30) | PK | Tên cây |
| TenGian | nvarchar(30) | PK | Tên giàn trồng |
| ChiTiet | nvarchar(200) |  | Chi tiết |
| NguoiTao | varchar(15) |  | Người tạo |
| NhanVien | varchar(15) | PK | Nhân viên |
| NgayBatDau | date |  | Ngày bắt đầu |
| NgayKetThuc | date |  | Ngày kết thúc |
| TrangThai | int |  | Trạng thái |

* + - 1. **KhoHang**

| **Tên cột** | **Kiểu** | **Ràng buộc** | **Ghi chú** |
| --- | --- | --- | --- |
| MaTH | int | PK | Mã thu hoạch |
| TenGian | nvarchar(30) | FK | Tên giàn |
| TenCay | nvarchar(30) | FK | Tên cây |
| TrongLuong | float |  | Trọng lượng |
| ThoiGianThuHoach | date |  | Thời gian thu hoạch |
| Coin | float |  | Thành tiền |

* 1. **Thiết kế giao diện**
     1. **Sơ đồ tổ chức giao diện**

Giao diện chính là một cửa sổ chứa menu và form quản lý tổng thể. Qua đó người sử dụng chương trình có thể biết được những thông tin một cách tổng quan và là cửa sổ chính để truy cập các cửa sổ con.

Mỗi người phải đăng nhập để xác định chức vụ trong trang trại nên cửa sổ đăng nhập sẽ hiện lên trước khi truy cập chương trình.

Ngoài ra, chương trình khởi động cần một khoảng thời gian để lấy dữ liệu và database nên sẽ có màn hình chào và thanh tiến trình để người dùng có cảm giác chương trình đang hoạt động.

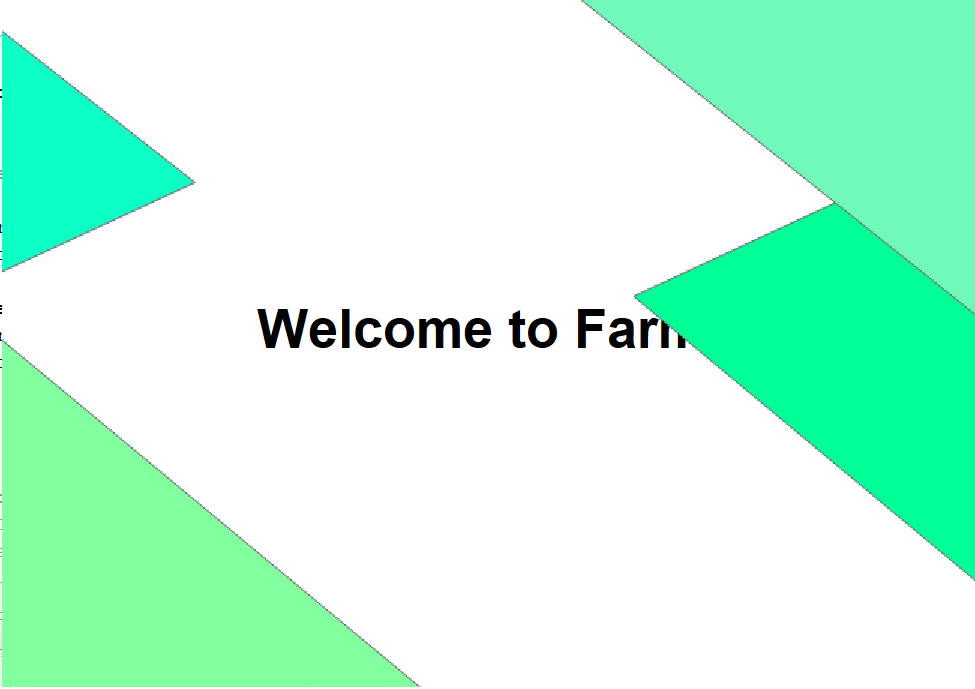
* + 1. **Thiết kế giao diện và các chức năng nghiệp vụ**

Trong phần thiết kế giao diện sử dụng hình ảnh chủ đề màu xanh của **NỀN NÔNG NGHIỆP XANH** làm màu chủ đạo của chương trình. Màu xanh giúp dịu mắt người dùng sử dụng trong thời gian lâu, giúp người dùng không gặp tình trạng mỏi mắt. Ngoài ra màu xanh còn là màu tượng trưng cho lá phổi xanh của Trái Đất.

Quy ước đặt tên chữ cái đầu:

| **Control** | **Prefix** | **Mô tả** |
| --- | --- | --- |
| JLabel | lbl | Nhãn |
| JButton | btn | Nút |
| JTextField, JTextArea, JPassword, JTextPane | txt | Các loại ô nhập |
| JRadioButton | rdo | Hộp chọn |
| JCheckBox | chk | Hộp kiểm |
| ButtonGroup | bgr | Nhóm các button |
| JComboBox | cbo | Hộp chọn xổ xuống |
| JPanel | pnl | Khung chứa |
| JTabbedPane | tab |  |
| JTable | tbl |  |
| …. |  |  |

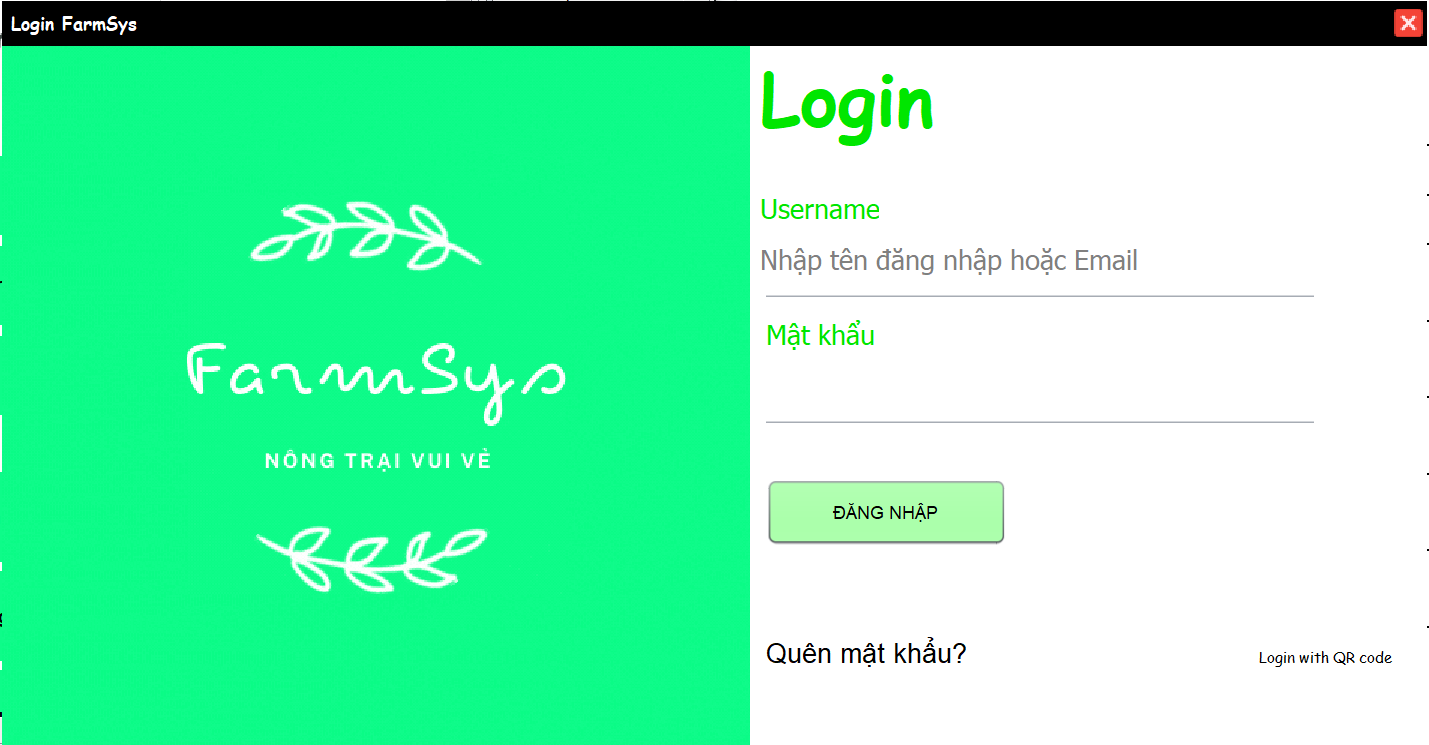
* + - 1. **Thiết kế màn hình chào**

****

*Chi tiết như bản bên dưới:*

| **Control** | **Prefix** | **Description** |
| --- | --- | --- |
| JPanel Form | ChaoJPanel | Set Layout: AbsoluteLayout |
| Label | lblHieuUng1 | text:  icon: Path 1.png |
| Label | lblHieuUng2 | text:  icon: Path 1.png |
| Label | lblHieuUng3 | text:  icon: Path 1.png |
| Label | lblHieuUng4 | text:  icon: Path 1.png |
| Label | lblTitle | text: Welcome to FarmSys  font: Forte 36 Bold |

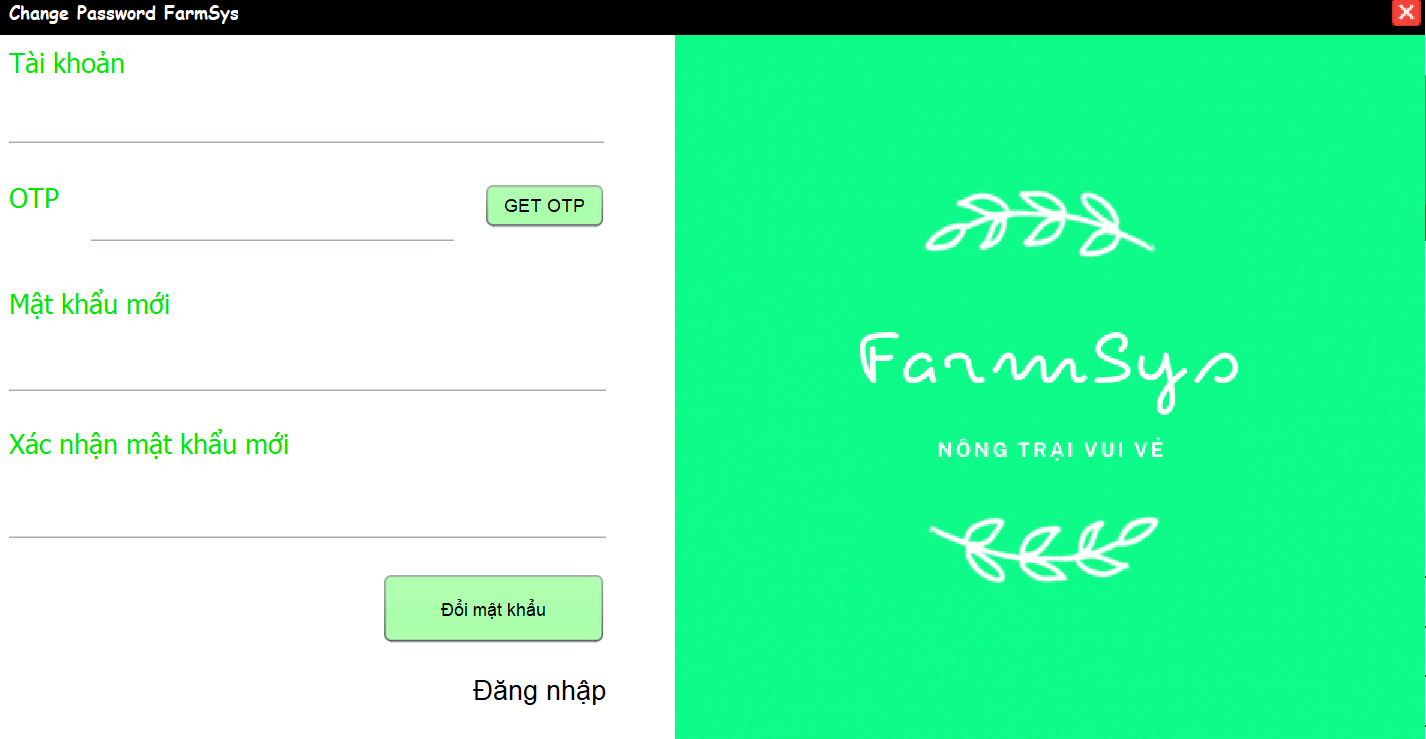
* + - 1. **Thiết kế form đăng nhập**

****

*Chi tiết như bảng bên dưới*

| **Control** | **Prefix** | **Mô tả** |
| --- | --- | --- |
| JPanel | pnlDieuKhien | Đặt Panel Điều Khiển |
| Jlabel | lblExit | - font: Tahoma 11 plain  - icon: close\_window\_16px.png |
| Jlabel | lblTieuDe | - font: Comic Sans  - foreground: [255,255,255] |
| JPanel | pnlTong | Đặt Panel Tổng |
| JPanel | pnlNhap | Đặt Panel Nhập |
| Jlabel | lblLogin | - font: Comic Sans  - text: Login |
| Jlabel | lblUserName | - font: Comic Sans  - text: User Name |
| JTextField | txtMaNhanVien | - font: Tahoma 18 Plain  - editable: true |
| Jlabel | lblMatKhau | - font: Comic Sans  - text: Mật Khẩu |
| JPasswordFiled | txtMatKhau | - font: Tahoma 18 Plain  - editable: true |
| JSeparator | SeparatorA | text: null |
| JSeparator | SeparatorB | text: null |
| JButton | btnDangNhap | - font: Tahoma 8 plain  - text: Đăng Nhập  - button Đăng Nhập  - background:[153,255,153] |
| Jlabel | lblQuenMatKhau | - font: Comic Sans  - text:Quên Mật Khẩu ? |
| Jlabel | lblLoginWithQR | - font: Comic Sans  - text: Login with QR code |
| Jlabel | lblLogo | - font: Comic Sans  - icon: logofarmSys.gif |
| JTextField | txtQRCode | - font: Tahoma 18 Plain  - editable: true |

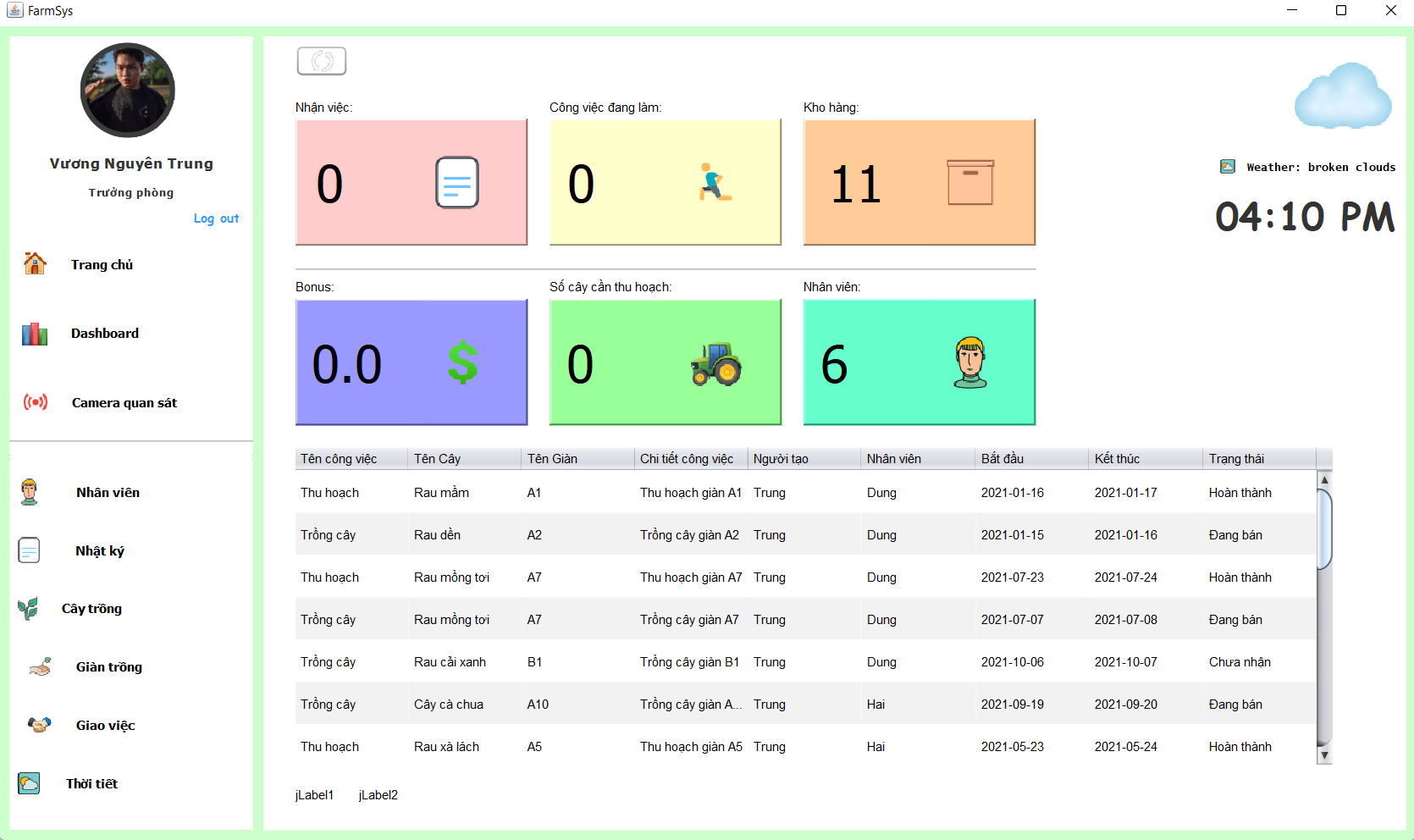
* + - 1. **Thiết kế form Quên mật khẩu**

****

*Chi tiết như bảng bên dưới*

| Control | Prefix | Mô tả |
| --- | --- | --- |
| JTextField | txttoaikhoanlaymk | - font: Tahoma 18 Plain  - editable: true |
| JTextField | txtmknew | - font: Tahoma 18 Plain  - editable: true |
| JButton | btndoimk | - font: Tahoma 8 plain  - text: Đổi mật khẩu  - button Đổi mật khẩu |
| Jlabel | jLabel5 | text: null |
| Jlabel | jLabel10 | text: null |
| JSeparator | jSeparator1 | text: null |
| JSeparator | jSeparator10 | text: null |
| JLabel | jLabel13 | text: null |
| JLabel | lblquaylaidangnhap | - font: Comic Sans  - text:Đăng nhập ? |
| JSeparator | jSeparator5 | text: null |
| JTextField | txtOTP | - font: Comic Sans |
| JLabel | jLabel5 | text: null |
| JButton | btnGetOTP | - font: Tahoma 8 plain  - text: GET OTP  - button:GET OTP |
| JSeparator | jSeparator6 | text: null |
| JTextfield | txtxnmknew | - font: Comic Sans |
| JLabel | jLabel6 | text: null |
| JLabel | lblIcon | text: null |

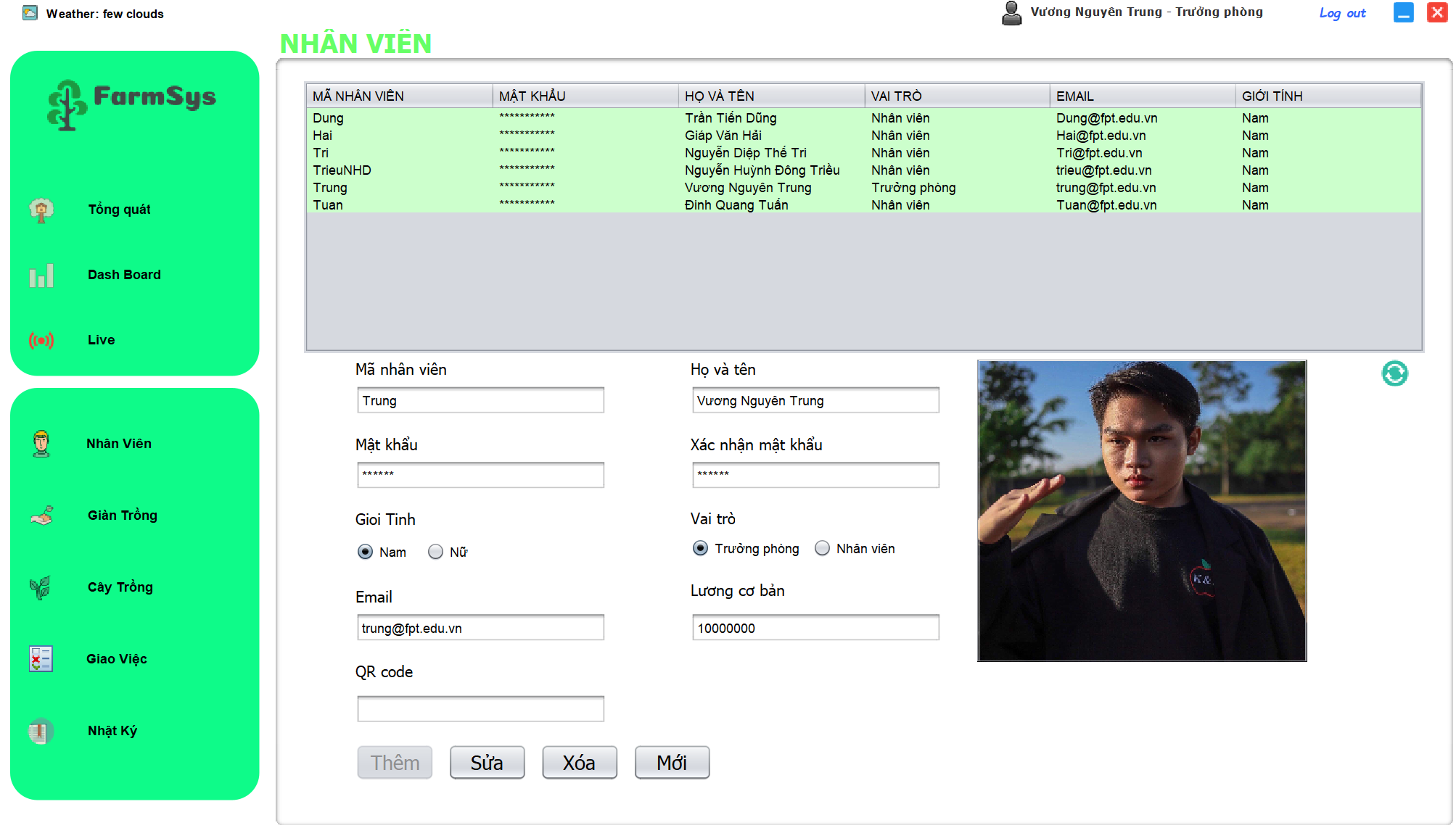
* + - 1. **Thiết kế form main chính**

****

*Chi tiết như bảng bên dưới*

| Control | Prefix | Mô tả |
| --- | --- | --- |
| JLabel | lblDongHo | - font: Tahoma 11 plain  - icon: sun\_100px.png |
| JLabel | lblsunrain | - font: Tahoma 11 plain  - icon: close\_window\_16px.png |
| JPanel | pnlkhohang | text: null |
| JLabel | lblDemKhoHang | - font: Tahoma 11 plain  - text: null |
| JLabel | lblLogoKhoHang | - font: Tahoma 11 plain  - text: null |
| JPanel | pnltodo | - font: Tahoma 11 plain  - text: null |
| JLabel | lblDemNhanVien | - font: Tahoma 11 plain  - text: null |
| JLabel | logoNhanViec | - font: Tahoma 11 plain  - icon: nhanviec\_20px.png |
| JLabel | lblcongvieccanlam | - font: Tahoma 11 plain  - text: Công việc cần làm |
| JLabel | lblkhohang | - font: Tahoma 11 plain  - text: Kho hàng |
| JLabel | lblnhanviec | - font: Tahoma 11 plain  - text: Nhận việc |
| JLabel | lblbonus | - font: Tahoma 11 plain  - text: Bonus |
| JPanel | pnlBonous | - font: Tahoma 11 plain  - text: null |
| JLabel | lblDemDoneByMonth | - font: Tahoma 11 plain  - text: null |
| JLabel | logoDollar | - font: Tahoma 11 plain  - icon: dollar\_20px.png |
| JLabel | lblcaycanTH | - font: Tahoma 11 plain  - text: Cây cần thu hoạch |
| JPanel | pnlCaycanTH | - font: Tahoma 11 plain  - text: null |
| JLabel | lblDemSoCayTH | - font: Tahoma 11 plain  - text: null |
| JLabel | logocaycanTH | - font: Tahoma 11 plain  - icon: cayTH\_20px.png |
| JSeparator | jSeparator1 | - text: null |
| JScrollPane | jScrollPane1 | - text: null |
| JTable | tblNhatKy | - Font : Tahoma 11 Plain  - Foreground : [0,0,0]  - Model : “STT,Tên công việc ,Tên Cây , Tên Gian,Chi tiết công việc,Người tạo, Nhân viên, Bắt đầu,Kết thúc, Trạng thái“  - SelectionBackground: [0,120,215]  - RowHeight : 16 |
| JButton | jButton1 | - font: Tahoma 11 plain  - icon: refesh\_25px.png |
| JLabel | lblweather | - font: Tahoma 11 plain  - text: null |
| JPanel | pnlhienthiNV | - font: Tahoma 11 plain  - text: null |
| JLabel | lbldemNV | - font: Tahoma 11 plain  - text: null |
| JLabel | logocaycanTH1 | - font: Tahoma 11 plain  - icon: caycanTH\_25px.png |
| JLabel | lblNhanVien | - font: Tahoma 11 plain  - text: Nhân viên |
| JPanel | pnlavatar | - font: Tahoma 11 plain  - text: null |
| JLabel | lblhoten | - font: Tahoma 11 plain  - text: Vương Nguyên Trung |
| JLabel | lblvaitro | - font: Tahoma 11 plain  - text: Trưởng phòng |
| ImageAvatar | imageAvatar | - font: Tahoma 11 plain  - text: null |
| JLabel | lblLogout | - font: Tahoma 11 plain  - text: Log out |
| JPanel | pnlmenu | - font: Tahoma 11 plain  - text: null |

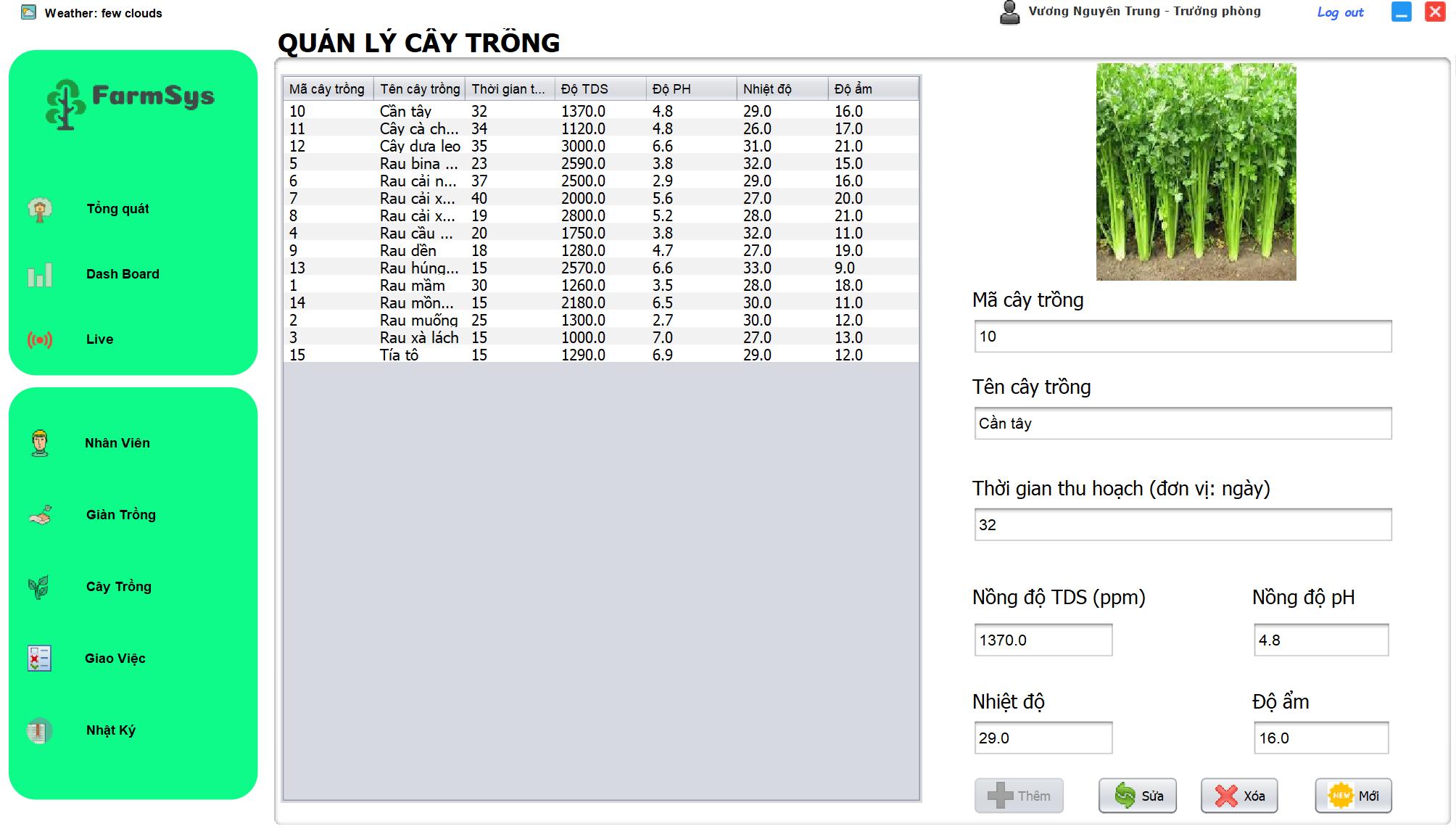
* + - 1. **Thiết kế form Quản lý nhân viên**

****

*Chi tiết như bảng bên dưới*

| Control | Prefix | Mô tả |
| --- | --- | --- |
| JPanel | pnlTong | Đặt Panel Tổng |
| JPanel | pnlList | - font: Tahoma 11  - background: [240,240,240] |
| JTable | tblNhanVien | - Font : Tahoma 11 Plain  - Foreground : [0,0,0]  - Model : “Mã nhân viên , Mật Khẩu , Họ và tên, Vai Trò, Email, Giới Tính “  - SelectionBackground: [0,120,215]  - RowHeight : 16 |
| JPanel | pnlEdit | - font: Tahoma 11  - background: [255,255,255] |
| Jlabel | lblHoTen | - font: Tahoma 14 plain  - text: Họ và tên |
| Jlabel | lblMatKhau | - font: Tahoma 14 plain  - text: Mật Khẩu |
| Jlabel | lblXacNhanMK | - font: Tahoma 14 plain  - text: Xác Nhận Mật Khẩu |
| Jlabel | lblGioiTinh | - font: Tahoma 14 plain  - text: Giới Tính |
| Jlabel | lblVaiTro | - font: Tahoma 14 plain  - text: Vai Trò |
| Jlabel | lblMaNhanVien | - font: Tahoma 14 plain  - text: Mã Nhân Viên |
| Jlabel | lblEmail | - font: Tahoma 14 plain  - text: Email |
| Jlabel | lblLuongCoBan | - font: Tahoma 14 plain  - text: Lương Cơ Bản |
| Jlabel | lblQRCode | - font: Tahoma 14 plain  - text: QR Code |
| Jlabel | lblHinh | - font: Tahoma 14 plain  - icon: avatar.png |
| Jlabel | lblReset | - font: Tahoma 14 plain  - icon: icons8\_synchronize\_25px.png |
| JTextField | txtMaNhanVien | - font: Tahoma 11 Plain  - editable: true |
| JTextField | txtHoTen | - font: Tahoma 11 Plain  - editable: true |
| JTextField | txtMatKhau | - font: Tahoma 11 Plain  - editable: true |
| JTextField | txtXacNhanMK | - font: Tahoma 11 Plain  - editable: true |
| JTextField | txtVaiTro | - font: Tahoma 11 Plain  - editable: true |
| JTextField | txtEmail | - font: Tahoma 11 Plain  - editable: true |
| JTextField | txtLuongCoBan | - font: Tahoma 11 Plain  - editable: true |
| JTextField | txtQRCode | - font: Tahoma 11 Plain  - editable: false |
| JButton | btnThem | - font: Tahoma 18 plain  - text: Thêm  - button Thêm  - background:[240,240,240] |
| JButton | btnSua | - font: Tahoma 18 plain  - text: Sửa  - button Sửa  - background:[240,240,240] |
| JButton | btnXoa | - font: Tahoma 18 plain  - text: Xóa  - button Xóa  - background:[240,240,240] |
| JButton | btnMoi | - font: Tahoma 18 plain  - text: Mới  - button Mới  - background:[240,240,240] |
| JRadioButton | rdoNam | - text: Nam  - selected: true  - buttonGroup: buttonGroup1 |
| JRadioButton | rdoNu | - text: Nữ  - selected: true  - buttonGroup: buttonGroup1 |
| JRadioButton | rdoTruongPhong | - text: Trưởng Phòng  - selected: true  - buttonGroup: buttonGroup2 |
| JRadioButton | rdoNhanVien | - text: Nhân Viên  - selected: true  - buttonGroup: buttonGroup2 |

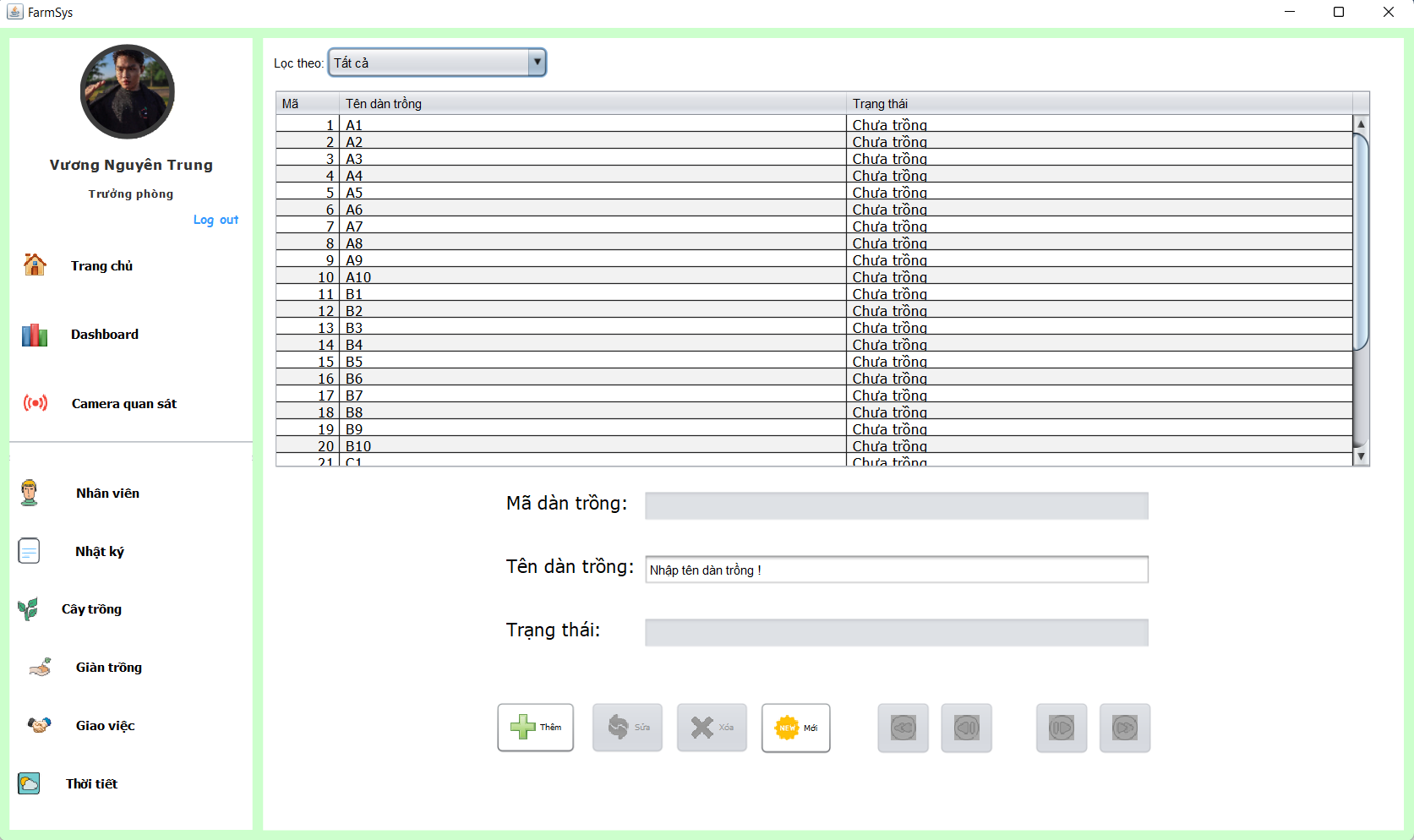
* + - 1. **Thiết kế form Quản lý cây trồng**

****

*Chi tiết như bảng bên dưới*

| **Control** | **Prefix** | **Mô tả** |
| --- | --- | --- |
| JTable | tblCayTrong | - Font : Tahoma 11 Plain  - Foreground : [0,0,0]  - Model : “Mã cây trồng ,Tên Cây , Thời gian thu hoạch,Độ TDS,Độ ẩm, Nhiệt độ, Độ PH“  - SelectionBackground: [0,120,215]  - RowHeight : 16 |
| JLabel | jLalbel1 | - font: Tahoma 14 plain  - text:Mã cây trồng |
| JTextfield | txtMaCay | - font: Tahoma 14 plain  - editable: false |
| JLabel | jLalbel2 | - font: Tahoma 14 plain  - text: Tên cây trồng |
| JTextfield | txtTenCay | - font: Tahoma 14 plain  - editable: true |
| JLabel | jLalbel3 | - font: Tahoma 14 plain  - text: Thời gian thu hoạch |
| JTextfield | txtThoiGianThuHoach | - font: Tahoma 14 plain  - editable: true |
| JLabel | jLalbel4 | - font: Tahoma 14 plain  - text: Độ TDS |
| JTextfield | txtDoTDS | - font: Tahoma 14 plain  - editable: true |
| JLabel | jLalbel5 | - font: Tahoma 14 plain  - text: Độ PH |
| JTextfield | txtDoPH | - font: Tahoma 14 plain  - editable: true |
| JLabel | jLalbel6 | - font: Tahoma 14 plain  - text: Nhiệt độ |
| JTextfield | txtNhietDo | - font: Tahoma 14 plain  - editable: true |
| JLabel | jLalbel7 | - font: Tahoma 14 plain  - text: Dộ ẩm |
| JTextfield | txtDoAm | - font: Tahoma 14 plain  - editable: true |
| JButton | btnAđd | - font: Tahoma 18 plain  - text: Thêm  - button Thêm  - background:[240,240,240] |
| JButton | btnUpdate | - font: Tahoma 18 plain  - text: Sửa  - button Sửa  - background:[240,240,240] |
| JButton | btnDelete | - font: Tahoma 18 plain  - text: Xóa  - button Xóa  - background:[240,240,240] |
| JButton | btnNew | - font: Tahoma 18 plain  - text: Mới  - button Mới  - background:[240,240,240] |
| JLalbel | lblHinh | text : null |

* + - 1. **Thiết kế form Quản lý giàn trồng**

****

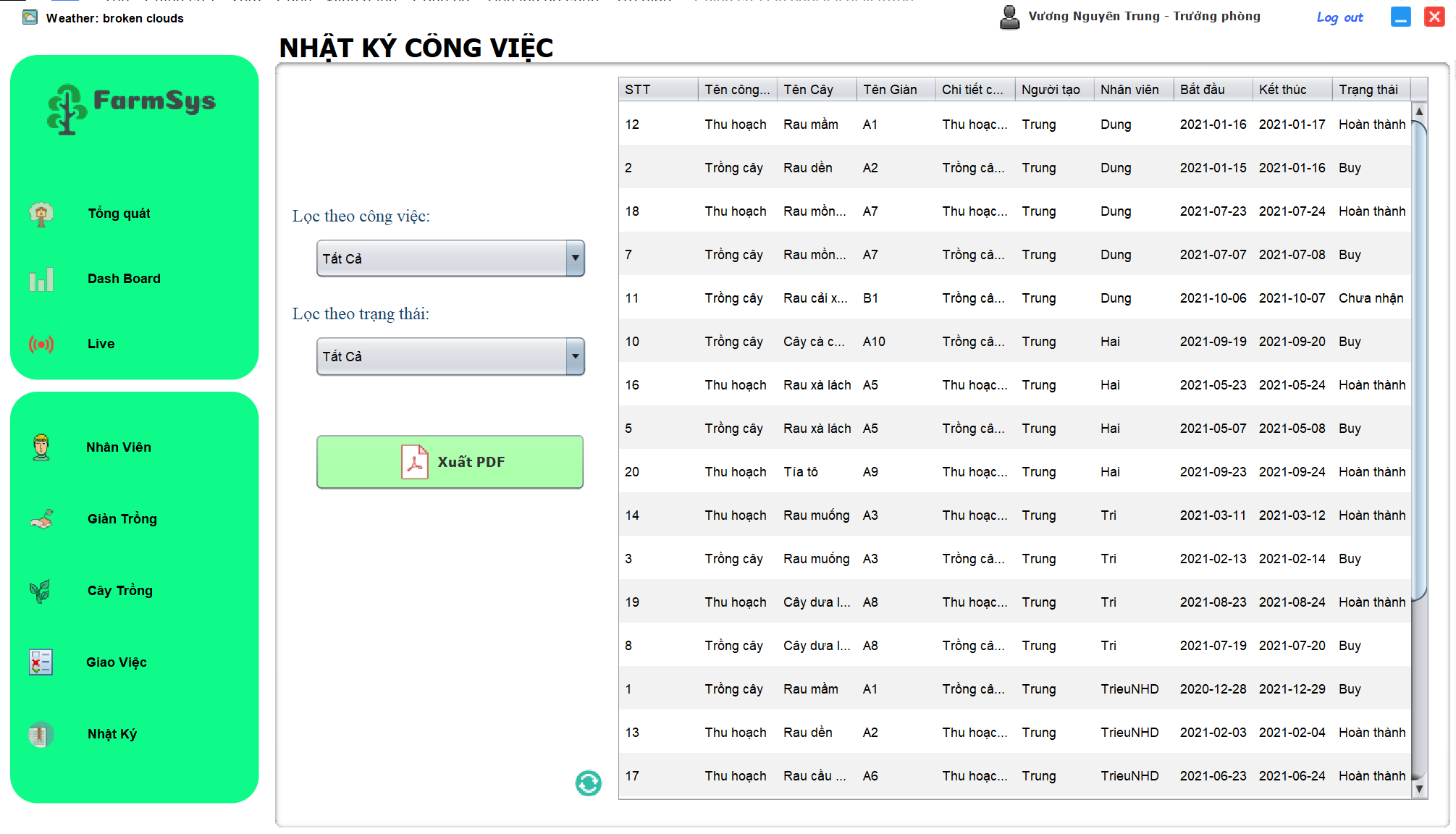
| Control | Prefix | Mô tả |
| --- | --- | --- |
| JPanel | pnlTong | Đặt Panel Tổng |
| JTabbedPane | tabs | - width : 1070  - height : 720 |
| JPanel | pnlEdit | - font: Tahoma 11  - background: [255,255,255] |
| Jlabel | lblMaDanTrong | - font: Tahoma 18 plain  - text: Mã dàn trồng |
| JTextField | txtMaDanTrong | - font: Tahoma 11  - editable: false |
| Jlabel | lblTenDanTrong | - font: Tahoma 18 plain  - text: Tên dàn trồng |
| JTextField | txtTenDanTrong | - font: Tahoma 11  - editable: true |
| Jlabel | lblTrangThai | - font: Tahoma 18 plain  - text: Trạng Thái |
| JTextField | txtTrangThai | - font: Tahoma 11  - editable: false |
| JButton | btnThem | - font: Tahoma 8 plain  - text: Thêm  - button Thêm  - icon: add.png |
| JButton | btnSua | - font: Tahoma 8 plain  - text: Sửa  - button Sửa  - icon: refresh.png |
| JButton | btnXoa | - font: Tahoma 8 plain  - text: Xóa  - button Xóa  - icon: delete.png |
| JButton | btnMoi | - font: Tahoma 8 plain  - text: Mới  - button Mới  - icon: new.png |
| JButton | btnFirst | - font: Tahoma 8 plain  - button đầu tiên  - icon: first.png |
| JButton | btnPrev | - font: Tahoma 8 plain  - button Trước  - icon: prev.png |
| JButton | btnNext | - font: Tahoma 8 plain  - button Sau  - icon: next.png |
| JButton | btnLast | - font: Tahoma 8 plain  - button cuối cùng  - icon: last.png |
| JPanel | pnlList | - font: Tahoma 11  - background: [255,255,255] |
| Jlabel | lblLoc | - font: Tahoma 18 plain  - text: Lọc theo |
| JComboBox | cboLoc | - font: Tahoma 11 plain  - model: Tất cả, Đang hoạt động,  Chưa hoạt động |
| JTable | tblDanTrong | - Font : Tahoma 11 Plain  - Foreground : [0,0,0]  - Model : “Mã giàn trồng , Tên giàn trồng , Trạng Thái “  - SelectionBackground: [0,120,215]  - RowHeight : 16 |

* + - 1. **Thiết kế form Giao việc**



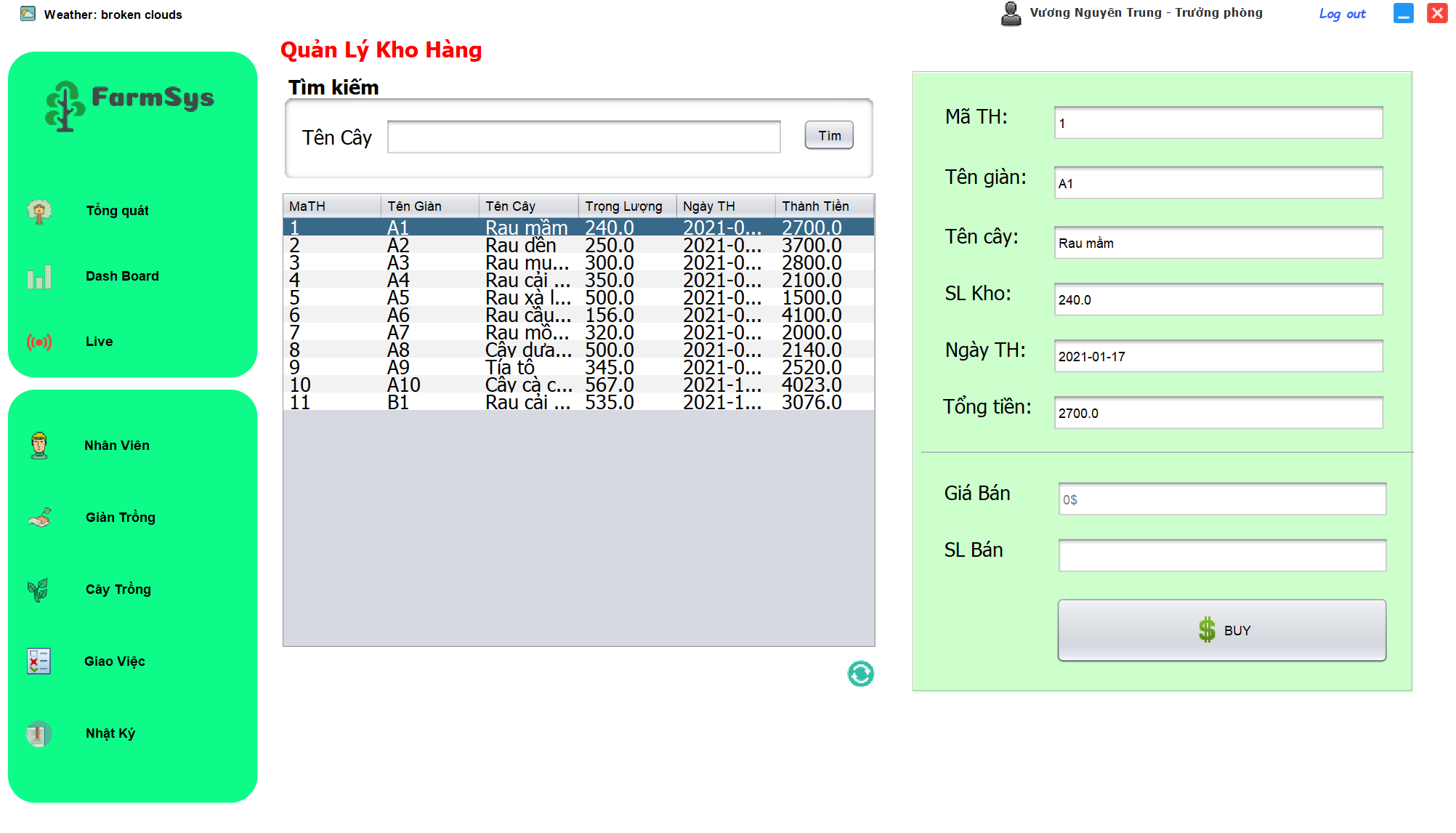
| **Control** | **Prefix** | **Mô tả** |
| --- | --- | --- |
| JPanel | pnlTong | Đặt Panel Tổng |
| JPanel | pnlGiaoViec | - Border Title: “Giao việc”  - layout : AbsoluteLayout |
| Jlabel | lblTenCongViec | - font: Tahoma 18 plain  - text: Tên công việc |
| JComboBox | cboCongViec | - font: Tahoma 11 plain  - model: Combo Box Model Editor |
| Jlabel | lblCayTrong | - font: Tahoma 18 plain  - text: Cây trồng |
| JComboBox | cboCayTrong | - font: Tahoma 11 plain  - model: Combo Box Model Editor |
| Jlabel | lblGianCay | - font: Tahoma 18 plain  - text: Giàn Cây |
| JComboBox | cboGianCay | - font: Tahoma 11 plain  - model: Combo Box Model Editor |
| Jlabel | lblGhiChu | - font: Tahoma 18 plain  - text: Ghi chú |
| JTextArea | txtGhiChu | - font: Tahoma 14 palin  - editable: true |
| Jlabel | lblNguoiTao | - font: Tahoma 18 plain  - text: Người Tạo |
| JTextField | txtNguoiTao | - font: Tahoma 11  - editable: true |
| Jlabel | lblNgayBatDau | - font: Tahoma 18 plain  - text: Ngày Bắt Đầu |
| JDateChooser | dtcNgayBatDau | - enable: true  - foreground: [0,0,0] |
| Jlabel | lblNgayKiemTra | - font: Tahoma 18 plain  - text: Ngày Kiểm Tra |
| JDateChooser | dtcNgayKiemTra | - enable: true  - foreground: [0,0,0] |
| Jlabel | lblNguoiNhanViec | - font: Tahoma 18 plain  - text: Người Nhận Việc |
| JComboBox | cboNhanViec | - font: Tahoma 11 plain  - model: Combo Box Model Editor |
| JButton | btnGiaoViec | - font: Tahoma 15 bold  - text: Gao Việc  - button Giao Việc  - icon: icons8\_checkmark\_35px.png |

* + - 1. **Thiết kế form Nhật ký**

****

| **Control** | **Prefix** | **Mô tả** |
| --- | --- | --- |
| JPanel | pnlNhatKy | - Border : Title “NHẬT KÝ CÔNG VIỆC”  - layout : AbsoluteLayout |
| JLabel | lblTimKiem | - Font : Tahoma : Bold: 16  - Foreground : [0,0,0]  - Text : “Tìm kiếm:” |
| JTextField | txtTimKiem | - Font : Tahoma : 11 |
| JLabel | lbltungay | - Font : Tahoma : 11  - Foreground : [0,0,0]  - Text : “Từ Ngày:” |
| JDateChooser | Datengaybatdau |  |
| JLabel | lblden | - Font : time new roman : 16  - Foreground : [25,69,107]  - Text : “ Lọc theo trạng thái” |
| JComboBox 2 | cboLocTrangThai | - width : 251  - heigh : 38  - model : “Tất Cả , Chưa nhận ,  Đang làm , Từ chối , Hoàn thành , Hoàn thành muộn ” |
| JButton | btnXuatPDF | - Background : [153,255,153]  - Font : Tahoma 13 Blod  - Foreground : [51,51,51]  - icon: icons8\_pdf\_35px\_1.png  - Text : Xuất PDF  - Width 250  - Height 53 |
| JTable | tblNhatKy | - Font : Tahoma 11 Plain  - Foreground : [0,0,0]  - Model : “ STT , Tên công việc ,  Tên cây , Tên Giàn , Chi tiết công việc , Người tạo , Nhân viên , Bắt đầu ,  Kết thúc , Trạng Thái “  - SelectionBackground: [0,120,215]  - RowHeight : 40 |

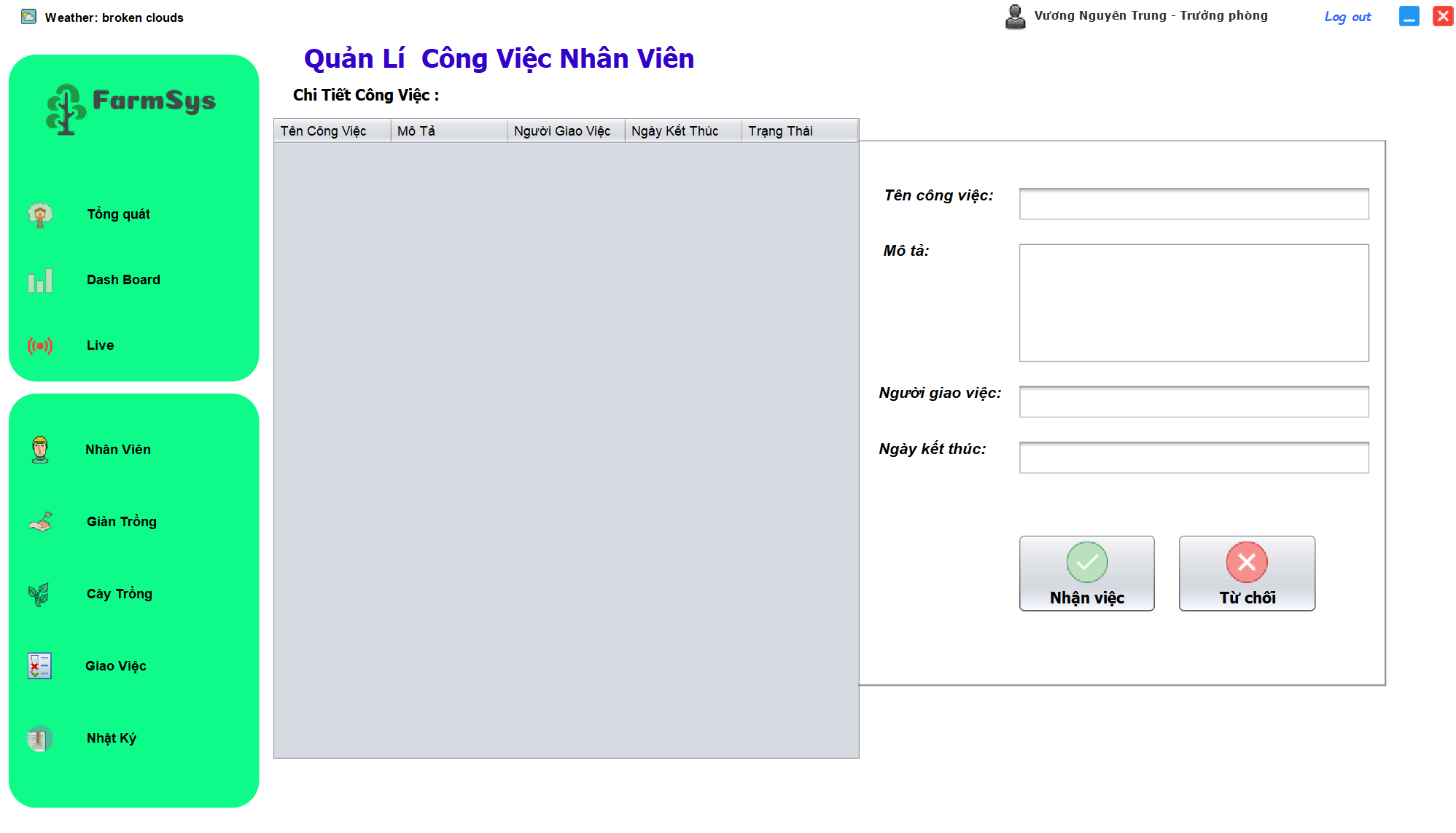
* + - 1. **Thiết kế form Kho hàng**

****

*Chi tiết như bảng bên dưới*

| **Control** | **Prefix** | **Mô tả** |
| --- | --- | --- |
| JPanel | pnlTong | Đặt Panel Tổng |
| JPanel | pnlTimKiem | - Border Title: “Tìm Kiếm”  - layout : AbsoluteLayout |
| Jlabel | lblQuanLyKhoHang | - font: Tahoma 20 Bold  - text: Quản Lý Kho Hàng |
| Jlabel | lblTenCay | - font: Tahoma 18 Plain  - text: Tên Cây |
| JTextField | txtTenCay | - font: Tahoma 18 Plain  - editable: true |
| JButton | btnTim | - font: Tahoma 11 Plain  - text: Tìm  - button Tìm  - background:[240,240,240] |
| Jlabel | lblReset | - font: Tahoma 11 Plain  - icon: icons8\_synchronize\_25px.png |
| JTable | tblKhoHang | - Font : Tahoma 18 Plain  - Foreground : [0,0,0]  - Model : “Mã TH, Tên giàn, Tên Cây ,Trọng Lượng, Ngày TH, Thành tiền “  - SelectionBackground: [0,120,215]  - RowHeight : 16 |
| JPanel | pnlEdit | - background: [204,255,204]  - font: Tahoma 11 |
| Jlabel | lblMaTH | - font: Tahoma 18 Plain  - text: Mã TH |
| Jlabel | lblTenGian | - font: Tahoma 18 Plain  - text: Tên Giàn |
| Jlabel | lblTenCay | - font: Tahoma 18 Plain  - text: Tên Cây |
| Jlabel | lblSLKho | - font: Tahoma 18 Plain  - text: SL Kho |
| Jlabel | lblNgayTH | - font: Tahoma 18 Plain  - text: Ngày TH |
| Jlabel | lblTongTien | - font: Tahoma 18 Plain  - text: Tổng Tiền |
| Jlabel | lblGiaBan | - font: Tahoma 18 Plain  - text: Giá Bán |
| Jlabel | lblSLBan | - font: Tahoma 18 Plain  - text: SL Bán |
| JTextField | txtMaTH | - font: Tahoma 11 Plain  - editable: false |
| JTextField | txtTenGian | - font: Tahoma 11 Plain  - editable: false |
| JTextField | txtTenCay | - font: Tahoma 11 Plain  - editable: false |
| JTextField | txtSLKho | - font: Tahoma 11 Plain  - editable: false |
| JTextField | txtNgayTH | - font: Tahoma 11 Plain  - editable: false |
| JTextField | txtTongTiem | - font: Tahoma 11 Plain  - editable: false |
| JTextField | txtGiaBan | - font: Tahoma 11 Plain  - editable: true |
| JTextField | txtSLBan | - font: Tahoma 11 Plain  - editable: true |
| JButton | btnBuy | - font: Tahoma 11 Plain  - text: Buy  - button Buy  - icon: Dollar.png |
| JSeparator | SeparatorA |  |

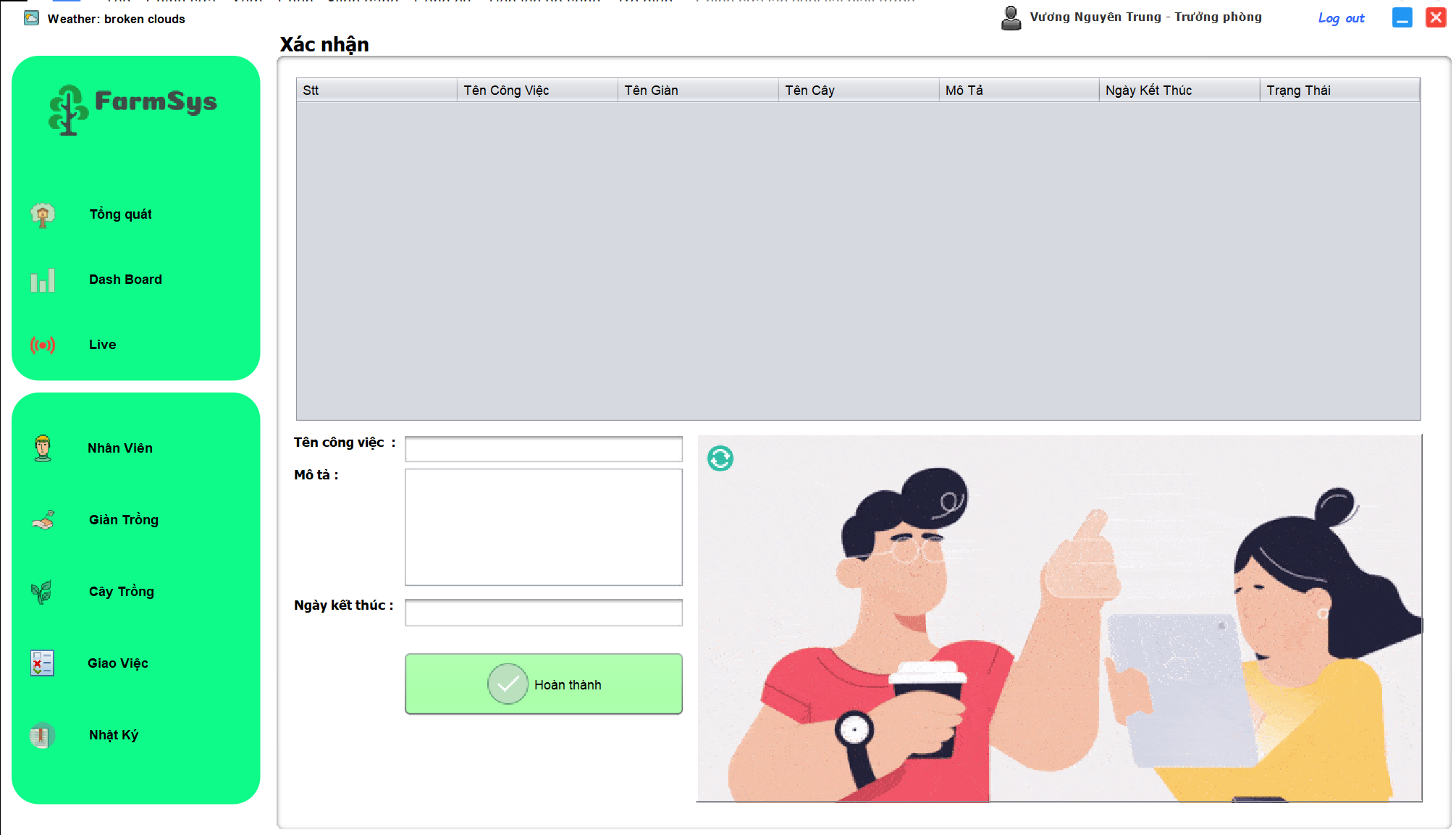
* + - 1. **Thiết kế form Quản lý công việc chờ nhận**

****

*Chi tiết như bảng bên dưới*

| **Control** | **Prefix** | **Mô tả** |
| --- | --- | --- |
| JPanel | pnlTong | Đặt Panel Tổng |
| Jlabel | lblQuanLyCVNhanVien | - font: Tahoma 18 Plain  - text: Quản Lý Công Việc Nhân Viên |
| JPanel | pnlTable | Đặt Panel Table |
| Jlabel | lblChiTietCongViec | - font: Tahoma 14 Bold  - text: Chi Tiết Công Việc |
| JTable | tblCongViec | - Font : Tahoma 11 Plain  - Foreground : [0,0,0]  - Model : “Tên Công Việc, Mô tả, Người Giao Việc, Ngày Kết Thúc, Trạng Thái “  - SelectionBackground: [0,120,215]  - RowHeight : 16 |
| Jlabel | lblReset | - font: Tahoma 11 Plain  - icon: icons8\_synchronize\_25px.png |
| JPanel | pnlThaoTac | - background: [255,255,255]  - border: BevelBorder |
| Jlabel | lblTenCongViec | - font: Arial 14 Bold Italic  - text: Tên Công Việc |
| Jlabel | lblMota | - font: Arial 14 Bold Italic  - text: Mô tả |
| Jlabel | lblNguoiGiaoViec | - font: Arial 14 Bold Italic  - text: Người Giao Việc |
| Jlabel | lblNgayKetThuc | - font: Arial 14 Bold Italic  - text: Ngày Kết Thúc |
| JTextArea | txtMota | - font: Tahoma 11 Plain  - editable: true |
| JTextField | txtTenCongViec | - font: Tahoma 11 Plain  - editable: false |
| JTextField | txtNguoiGiaoViec | - font: Tahoma 11 Plain  - editable: false |
| JTextField | txtNgayKetThuc | - font: Tahoma 11 Plain  - editable: false |
| JButton | btnNhanViec | - font: Tahoma 11 Plain  - text: Nhận Việc  - button Nhận Việc  - icon: icons8\_ok\_40px.png |
| JButton | btnTuChoi | - font: Tahoma 11 Plain  - text: Từ Chối  - button Từ Chối  - icon: icons8\_cancel\_40px.png |

* + - 1. **Thiết kế form Quản lý công việc đang làm**

****

*Chi tiết như ảnh bên dưới*

| **Control** | **Prefix** | **Mô tả** |
| --- | --- | --- |
| JPanel | pnlTong | Đặt Panel Tổng |
| JPanel | pnlDanhSach | - background: [255,255,255]  - border: TitledBorder  - text: Xác Nhận |
| JTable | tblCongViec | - Font : Tahoma 11 Plain  - Foreground : [0,0,0]  - Model : “STT, Tên Công Việc, Tên Giàn, Tên Cây,Mô tả, Ngày Kết Thúc, Trạng Thái “  - SelectionBackground: [0,120,215]  - RowHeight : 16 |
| Jlabel | lblReset | - font: Tahoma 11 Plain  - icon: icons8\_synchronize\_25px.png |
| Jlabel | lblHinh | - font: Tahoma 11 Plain  - icon: hoanthanh.gif |
| Jlabel | lblTenCongViec | - font: Tahoma 11 Bold  - text: Tên Công Việc |
| Jlabel | lblMota | - font: Tahoma 11 Bold  - text: Mô tả |
| Jlabel | lblNgayKetThuc | - font: Tahoma 11 Bold  - text: Ngày Kết Thúc |
| JTextArea | txtMota | - font: Tahoma 11 Plain  - editable: true |
| JTextField | txtTenCongViec | - font: Tahoma 11 Plain  - editable: false |
| JTextField | txtNgayKetThuc | - font: Tahoma 11 Plain  - editable: false |
| JButton | btnHoanThanh | - font: Tahoma 11 Plain  - text: Hoàn Thành  - button Hoàn Thành  - icon: icons8\_ok\_40px.png |

1. **VIẾT MÃ**
   1. **Viết mã tạo CSDL**
      1. **Tạo CSDL**

| use master  go  if exists (select \* from sys.databases where name ='FarmSys')  drop DATABASE FarmSys  go  create database FarmSys  go  use FarmSys  go  Create table NhanVien (  MaNV varchar(15) primary key not null,  MatKhau varchar(16) not null,  HoTen nvarchar(50) not null,  GioiTinh bit,  Email nvarchar(30)not null,  Luong int,  VaiTro bit not null,  Hinh varchar(30) ,  QRcode varchar(40)  );  go  Create table GianTrong (  MaGian int IDENTITY(1,1),  TenGian nvarchar(30),  TrangThai bit,  /\* "0: chưa trồng  1: trồng" \*/  primary key(TenGian)  );  go  Create table LoaiCay (  MaCay int IDENTITY(1,1) not null,  TenCay nvarchar(30) primary key not null,  ThoiGianThuHoach int not null,  DoTDS float,  DoPH float,  NhietDo float,  DoAm float,  Hinh varchar(30) not null  );  go  Create table CongViec(  STT int IDENTITY(1,1),  TenCV nvarchar(30) primary key  );  Create table NhatKy (  STT int IDENTITY(1,1),  TenCV nvarchar(30),  TenCay nvarchar(30),  TenGian nvarchar(30),  ChiTiet nvarchar(200),  NguoiTao varchar(15),  NhanVien varchar(15),  NgayBatDau date,  NgayKetThuc date,  TrangThai int  /\* 0: to do  1: doing  2: từ chối  3: hoàn thành  4: hoàn thành + trể  5: đang bán \*/    Primary key (NhanVien,TenGian,TenCay,TenCV),    foreign key (NhanVien) references NhanVien(MaNV),  foreign key (TenGian) references GianTrong(TenGian),  foreign key (TenCay) references LoaiCay(TenCay),  foreign key (TenCV) references CongViec(TenCV)    );  go  Create table KhoHang(  MaTH int IDENTITY(1,1),  TenGian nvarchar(30),  TenCay nvarchar(30),  TrongLuong float,  ThoiGianThuHoach date,  Coin float,  Primary key (TrongLuong,MaTH),  foreign key (TenGian) references GianTrong(TenGian),  foreign key (TenCay) references LoaiCay(TenCay)  );  go  Create table HoaDon(  MaHD int IDENTITY(1,1),  TenCay nvarchar(30),  TrongLuong float,  ThoiGian date,  ThanhTien float,  Primary key (MaHD),  foreign key (TenCay) references LoaiCay(TenCay)  );  go  insert into NhanVien  values  ('Trung','240102',N'Vương Nguyên Trung',1,'trung@fpt.edu.vn',10000000,1,'a.png',null),  ('TrieuNHD','123456',N'Nguyễn Huỳnh Đông Triều',1,'trieu@fpt.edu.vn',10000000,0,'b.png',null),  ('Tri','123456',N'Nguyễn Diệp Thế Tri',1,'Tri@fpt.edu.vn',10000000,0,'c.png',null),  ('Dung','123456',N'Trần Tiến Dũng',1,'Dung@fpt.edu.vn',2000000,0,'d.png',null),  ('Hai','123456',N'Giáp Văn Hải',1,'Hai@fpt.edu.vn',10000000,0,'e.png',null),  ('Tuan','123456',N'Đinh Quang Tuấn',1,'Tuan@fpt.edu.vn',10000000,0,'f.png',null)  go  insert into GianTrong (TenGian,TrangThai)  values  ('A1','0'),  ('A2','0'),  ('A3','0'),  ('A4','0'),  ('A5','0'),  ('A6','0'),  ('A7','0'),  ('A8','0'),  ('A9','0'),  ('A10','0'),  ('B1','0'),  ('B2','0'),  ('B3','0'),  ('B4','0'),  ('B5','0'),  ('B6','0'),  ('B7','0'),  ('B8','0'),  ('B9','0'),  ('B10','0'),  ('C1','0'),  ('C2','0'),  ('C3','0'),  ('C4','0'),  ('C5','0'),  ('C6','0'),  ('C7','0'),  ('C8','0'),  ('C9','0'),  ('C10','0')  go    INSERT INTO LoaiCay  VALUES  (N'Rau mầm','30','1260','3.5','28','18','Hinh1.jpg'),  (N'Rau muống','25','1300','2.7','30','12','Hinh2.jpg'),  (N'Rau xà lách','15','1000','7.0','27','13','Hinh3.jpg'),  (N'Rau cầu vòng','20','1750','3.8','32','11','Hinh4.jpg'),  (N'Rau bina (Cải bó xôi)','23','2590','3.8','32','15','Hinh5.jpg'),  (N'Rau cải ngọt','37','2500','2.9','29','16','Hinh6.jpg'),  (N'Rau cải xanh','40','2000','5.6','27','20','Hinh7.jpg'),  (N'Rau cải xoan','19','2800','5.2','28','21','Hinh8.jpg'),  (N'Rau dền','18','1280','4.7','27','19','Hinh9.jpg'),  (N'Cần tây','32','1370','4.8','29','16','Hinh10.jpg'),  (N'Cây cà chua','34','1120','4.8','26','17','Hinh11.jpg'),  (N'Cây dưa leo','35','3000','6.6','31','21','Hinh12.jpg'),  (N'Rau húng quế','15','2570','6.6','33','09','Hinh13.jpg'),  (N'Rau mồng tơi','15','2180','6.5','30','11','Hinh14.jpg'),  (N'Tía tô','15','1290','6.9','29','12','Hinh15.jpg')  go  insert into CongViec (TenCV)  values  (N'Trồng cây'),  (N'Chăm sóc'),  (N'Thu hoạch')  go  insert into NhatKy (TenCV,TenCay,TenGian,ChiTiet,NguoiTao,NhanVien,NgayBatDau,NgayKetThuc,TrangThai)  values  (N'Trồng cây',N'Rau mầm','A1',N'Trồng cây giàn A1','Trung','TrieuNHD','2020-12-28','2021-12-29','5'),  (N'Trồng cây',N'Rau dền','A2',N'Trồng cây giàn A2','Trung','Dung','2021-01-15','2021-01-16','5'),  (N'Trồng cây',N'Rau muống','A3',N'Trồng cây giàn A3','Trung','Tri','2021-02-13','2021-02-14','5'),  (N'Trồng cây',N'Rau cải ngọt','A4',N'Trồng cây giàn A4','Trung','Tuan','2021-02-26','2021-02-27','5'),  (N'Trồng cây',N'Rau xà lách','A5',N'Trồng cây giàn A5','Trung','Hai','2021-05-07','2021-05-08','5'),  (N'Trồng cây',N'Rau cầu vòng','A6',N'Trồng cây giàn A6','Trung','TrieuNHD','2021-06-03','2021-06-04','5'),  (N'Trồng cây',N'Rau mồng tơi','A7',N'Trồng cây giàn A7','Trung','Dung','2021-07-07','2021-07-08','5'),  (N'Trồng cây',N'Cây dưa leo','A8',N'Trồng cây giàn A8','Trung','Tri','2021-07-19','2021-07-20','5'),  (N'Trồng cây',N'Tía tô','A9',N'Trồng cây giàn A9','Trung','Tuan','2021-09-08','2021-09-09','5'),  (N'Trồng cây',N'Cây cà chua','A10',N'Trồng cây giàn A10','Trung','Hai','2021-09-19','2021-09-20','5'),  (N'Trồng cây',N'Rau cải xanh','B1',N'Trồng cây giàn B1','Trung','Dung','2021-10-06','2021-10-07',''),  (N'Thu hoạch',N'Rau mầm','A1',N'Thu hoạch giàn A1','Trung','Dung','2021-01-16','2021-01-17','3'),  (N'Thu hoạch',N'Rau dền','A2',N'Thu hoạch giàn A2','Trung','TrieuNHD','2021-02-03','2021-02-04','3'),  (N'Thu hoạch',N'Rau muống','A3',N'Thu hoạch giàn A3','Trung','Tri','2021-03-11','2021-03-12','3'),  (N'Thu hoạch',N'Rau cải ngọt','A4',N'Thu hoạch giàn A4','Trung','Tuan','2021-04-04','2021-04-05','3'),  (N'Thu hoạch',N'Rau xà lách','A5',N'Thu hoạch giàn A5','Trung','Hai','2021-05-23','2021-05-24','3'),  (N'Thu hoạch',N'Rau cầu vòng','A6',N'Thu hoạch giàn A6','Trung','TrieuNHD','2021-06-23','2021-06-24','3'),  (N'Thu hoạch',N'Rau mồng tơi','A7',N'Thu hoạch giàn A7','Trung','Dung','2021-07-23','2021-07-24','3'),  (N'Thu hoạch',N'Cây dưa leo','A8',N'Thu hoạch giàn A8','Trung','Tri','2021-08-23','2021-08-24','3'),  (N'Thu hoạch',N'Tía tô','A9',N'Thu hoạch giàn A9','Trung','Hai','2021-09-23','2021-09-24','3'),  (N'Thu hoạch',N'Cây cà chua','A10',N'Thu hoạch giàn A10','Trung','Tuan','2021-10-23','2021-10-24','3'),  (N'Thu hoạch',N'Rau cải xanh','B1',N'Thu hoạch giàn A10','Trung','TrieuNHD','2021-11-15','2021-11-16','3')  go  /\*  select NhatKy.TenGian,LoaiCay.TenCay,(SELECT DATEADD(day, +(LoaiCay.ThoiGianThuHoach) , NhatKy.NgayKetThuc)) as 'NgayTH' from LoaiCay inner join NhatKy on LoaiCay.TenCay = NhatKy.TenCay where TenCV like N'Trồng cây' and GETDATE() >= (SELECT DATEADD(day, +(LoaiCay.ThoiGianThuHoach) , NhatKy.NgayKetThuc))  select \* from NhatKy  select \*from NhatKy where TrangThai = 3 and NhanVien like N'TrieuNHD' and NgayKetThuc between (select CONVERT(varchar,dateadd(d,-(day(getdate()-1)),getdate()),106)) and (select CONVERT(varchar,dateadd(d,-(day(dateadd(m,1,getdate()))),dateadd(m,1,getdate())),106))  select NhatKy.TenGian,LoaiCay.TenCay,(SELECT DATEADD(day, +(LoaiCay.ThoiGianThuHoach) , NhatKy.NgayKetThuc)) as 'NgayTH' from LoaiCay inner join NhatKy on LoaiCay.TenCay = NhatKy.TenCay where TenCV like N'Trồng cây' and GETDATE() >= (SELECT DATEADD(day, +(LoaiCay.ThoiGianThuHoach) , NhatKy.NgayKetThuc)) and TrangThai = 2  \*/  insert into KhoHang (TenGian,TenCay,TrongLuong,ThoiGianThuHoach,Coin)  values  ('A1',N'Rau mầm','240','2021-01-17','2700'),  ('A2',N'Rau dền','250','2021-02-04','3700'),  ('A3',N'Rau muống','300','2021-03-12','2800'),  ('A4',N'Rau cải ngọt','350','2021-04-05','2100'),  ('A5',N'Rau xà lách','500','2021-05-24','1500'),  ('A6',N'Rau cầu vòng','156','2021-06-24','4100'),  ('A7',N'Rau mồng tơi','320','2021-07-24','2000'),  ('A8',N'Cây dưa leo','500','2021-08-24','2140'),  ('A9',N'Tía tô','345','2021-09-24','2520'),  ('A10',N'Cây cà chua','567','2021-10-24','4023'),  ('B1',N'Rau cải xanh','535','2021-11-16','3076')  go  select distinct (select SUM(coin) from KhoHang where ThoiGianThuHoach between '2021/01/01' and '2021/01/31') as 'TongTien01',  (select SUM(coin) from KhoHang where ThoiGianThuHoach between '2021/02/01' and '2021/02/28')as 'TongTien02',  (select SUM(coin) from KhoHang where ThoiGianThuHoach between '2021/03/01' and '2021/03/31')as 'TongTien03',  (select SUM(coin) from KhoHang where ThoiGianThuHoach between '2021/04/01' and '2021/04/30')as 'TongTien04',  (select SUM(coin) from KhoHang where ThoiGianThuHoach between '2021/05/01' and '2021/05/31')as 'TongTien05',  (select SUM(coin) from KhoHang where ThoiGianThuHoach between '2021/06/01' and '2021/06/30')as 'TongTien06',  (select SUM(coin) from KhoHang where ThoiGianThuHoach between '2021/07/01' and '2021/07/31')as 'TongTien07',  (select SUM(coin) from KhoHang where ThoiGianThuHoach between '2021/08/01' and '2021/08/31')as 'TongTien08',  (select SUM(coin) from KhoHang where ThoiGianThuHoach between '2021/09/01' and '2021/09/30')as 'TongTien09',  (select SUM(coin) from KhoHang where ThoiGianThuHoach between '2021/10/01' and '2021/10/31')as 'TongTien10',  (select SUM(coin) from KhoHang where ThoiGianThuHoach between '2021/11/01' and '2021/11/30')as 'TongTien11',  (select SUM(coin) from KhoHang where ThoiGianThuHoach between '2021/12/01' and '2021/12/31')as 'TongTien12'  from KhoHang |
| --- |

* + 1. **SQL truy vấn và thao tác**
       1. **NhanVien**

| String INSERT\_SQL = "INSERT INTO NhanVien(MaNV, MatKhau, HoTen, GioiTinh, Email, Luong, VaiTro, Hinh, QRcode) VALUES(?,?,?,?,?,?,?,?,?)";  String UPDATE\_SQL = "UPDATE NhanVien SET MatKhau=?,HoTen=?,GioiTinh=?,Email=?, Luong=?, VaiTro=?, Hinh=?,QRcode=? WHERE MaNV=?";  String UPDATEQRcode\_SQL = "UPDATE NhanVien SET QRcode = ? WHERE QRcode = ?";  String DELETE\_SQL = "DELETE FROM NhanVien WHERE MaNV=?";  String SELECT\_ALL\_SQL = "SELECT \*FROM NhanVien";  String SELECT\_BY\_ID\_SQL = "SELECT \* FROM NhanVien WHERE MaNV=?";  String SELECT\_BY\_QR\_SQL = "SELECT \* FROM NhanVien WHERE QRcode=?";  String SELECT\_BY\_Email\_SQL = "SELECT \* FROM NhanVien WHERE Email=?";  String RESET\_PASS\_SQL = "UPDATE NhanVien SET MatKhau=? WHERE MaNV=?";  String SELECT\_NhanVien\_SQL = "SELECT \*FROM NhanVien WHERE VaiTro= 0";  String SELECT\_NhanVienQRcode\_SQL = "SELECT \*FROM NhanVien WHERE QRcode = ?";  String SELECT\_BY\_img\_SQL = "SELECT \* FROM NhanVien WHERE HoTen=?"; |
| --- |

* + - 1. **CayTrong**

| String INSERT\_SQL = "INSERT INTO LoaiCay(TenCay,ThoiGianThuHoach,DoTDS,DoPH,NhietDo,DoAm,Hinh) VALUES(?,?,?,?,?,?,?)";  String UPDATE\_SQL = "UPDATE LoaiCay SET TenCay=?, ThoiGianThuHoach=?, DoTDS=?, DoPH=?, NhietDo=?, DoAm=?, Hinh=? WHERE MaCay=?";  String DELETE\_SQL = "DELETE FROM LoaiCay WHERE MaCay=?";  String SELECT\_ALL\_SQL = "SELECT \* FROM LoaiCay";  String SELECT\_BY\_ID\_SQL = "SELECT \* FROM LoaiCay WHERE MaCay=?";  String SELECT\_BY\_ID\_TenCay = "SELECT \* FROM LoaiCay WHERE TenCay =?"; |
| --- |

* + - 1. **GianTrong**

| String INSERT\_SQL = "INSERT INTO GianTrong (TenGian, TrangThai) VALUES(?,?)";  String UPDATE\_SQL = "UPDATE GianTrong SET TenGian =?, TrangThai =? WHERE MaGian =?";  String DELETE\_SQL = "DELETE FROM GianTrong WHERE MaGian =?";  String SELECT\_ALL\_SQL = "SELECT \* FROM GianTrong";  String SELECT\_BY\_ID\_SQL = "SELECT \* FROM GianTrong WHERE MaGian =?";  String SELECT\_BY\_ID\_TT = "SELECT \* FROM GianTrong WHERE TrangThai = ?";  String SELECT\_BY\_ID\_TenGian = "SELECT \* FROM GianTrong WHERE TenGian =?";  String UPDATE\_TrangThai\_SQL = "UPDATE GianTrong SET TrangThai = 1 WHERE TenGian = ?";  String UPDATE\_TraVeCbo\_SQL = "UPDATE GianTrong SET TrangThai = 0 WHERE TenGian = ?"; |
| --- |

* + - 1. **NhatKy**

| String INSERT\_SQL = "INSERT INTO NhatKy (TenCV, TenCay, TenGian, ChiTiet, NguoiTao, Nhanvien, NgayBatDau, NgayKetThuc, TrangThai) VALUES(?,?,?,?,?,?,?,?,?)";  String select\_all\_sql = "select\*from NhatKy";  String select\_by\_ten\_sql = "select \* from NhatKy where TenCV = ?";  String select\_by\_trangthai\_sql = "select \* from NhatKy where TrangThai = ?";  String select\_by\_trangthaivaten\_sql = "select \* from NhatKy where TrangThai = ? and NhanVien like ?";  String select\_formtodoanddoing\_sql = "select \* from NhatKy where STT = ?";  String UPDATE\_Done\_SQL = "UPDATE NhatKy SET TrangThai = 3 where STT = ?";  String UPDATE\_SQL = "UPDATE NhatKy SET TrangThai = 1 where STT = ?";  String UPDATE\_TuChoi\_SQL = "UPDATE NhatKy SET TrangThai = 2 where STT = ?";  String select\_by\_Done\_Month\_SQL = "select \*from NhatKy where TrangThai = 3 and NhanVien like ? and NgayKetThuc between (select CONVERT(varchar,dateadd(d,-(day(getdate()-1)),getdate()),106)) and (select CONVERT(varchar,dateadd(d,-(day(dateadd(m,1,getdate()))),dateadd(m,1,getdate())),106))";  String UPDATE\_TrangThaiTH\_SQL = "UPDATE NhatKy SET TrangThai = 5 where TenGian = ? "; |
| --- |

* + - 1. **KhoHang**

| String INSERT\_SQL = "INSERT INTO KhoHang (TenGian, TenCay, TrongLuong, ThoiGianThuHoach) VALUES(?,?,?,?)";  String SELECT\_ALL\_SQL = "select\*from KhoHang where TrongLuong > 0";  String SELECT\_BY\_TEN\_SQL = "select \* from KhoHang where TenGian = ?";  String DELETE\_SQL = "DELETE FROM KhoHang where TenDan = ?";  String UPDATE\_SQL = "UPDATE KhoHang SET TrongLuong =? WHERE MaTH =?";  String SELECT\_SLBan\_SQL = "UPDATE KhoHang set TrongLuong = ? , Coin = ? where MaTH = ?"; |
| --- |

* 1. **Lập trình JDBC và các lớp hỗ trợ**
     1. **Lớp hỗ trợ**
        1. **Lớp tiện ích DateHelper**

| package com.farmsys.Helper;  import java.text.ParseException;  import java.text.SimpleDateFormat;  import java.util.Date;  public class DateHelper {  static final SimpleDateFormat DATE\_FORMATER = new SimpleDateFormat("MM/dd/yyyy");  /\*\*  \* Chuyển đổi String sang Date  \*  \* @param date là String cần chuyển  \* @param pattern là định dạng thời gian  \* @return Date kết quả  \*/  public static Date toDate(String date, String... pattern) {  try {  if (pattern.length > 0) {  DATE\_FORMATER.applyPattern(pattern[0]);  }  if (date == null) {  return DateHelper.now();  }  return DATE\_FORMATER.parse(date);  } catch (ParseException ex) {  throw new RuntimeException(ex);  }  }  /\*\*  \* Chuyển đổi từ Date sang String  \*  \* @param date là Date cần chuyển đổi  \* @param pattern là định dạng thời gian  \* @return String kết quả  \*/  public static String toString(Date date, String... pattern) {  if (pattern.length > 0) {  DATE\_FORMATER.applyPattern(pattern[0]);  }  if (date == null) {  date = DateHelper.now();  }  return DATE\_FORMATER.format(date);  }  /\*\*  \* Lấy thời gian hiện tại  \*  \* @return Date kết quả  \*/  public static Date now() {  return new Date();  }  /\*\*  \* Bổ sung số ngày vào thời gian  public static Date addDays(Date date, int days) {  date.setTime(date.getTime() + days \* 24 \* 60 \* 60 \* 1000);  return date;  }  public static Date add(int days) {  Date now = DateHelper.now();  now.setTime(now.getTime() + days \* 24 \* 60 \* 60 \* 1000);  return now;  }  } |
| --- |

* + - 1. **Lớp tiện ích JdbcHelper**

| package com.farmsys.Helper;  import java.sql.Connection;  import java.sql.DriverManager;  import java.sql.PreparedStatement;  import java.sql.ResultSet;  import java.sql.SQLException;  public class JdbcHelper {  static String driver = "com.microsoft.sqlserver.jdbc.SQLServerDriver";  static String dburl = "jdbc:sqlserver://localhost:1433;database=FarmSys";  static String user = "sa";  static String password = "Password123@jkl#";  static {  try {  Class.forName(driver);  } catch (ClassNotFoundException e) {  throw new RuntimeException(e);  }  }  public static void ConnectDAO() {  try {  Connection conn = DriverManager.getConnection(dburl, user, password);  } catch (SQLException ex) {  throw new RuntimeException(ex);  }  }  public static PreparedStatement getStmt(String sql, Object... args) throws SQLException {  Connection conn = DriverManager.getConnection(dburl, user, password);  PreparedStatement stmt;  if (sql.trim().startsWith("{")) {  stmt = conn.prepareCall(sql); //PROC  } else {  stmt = conn.prepareStatement(sql); // SQL  }  for (int i = 0; i < args.length; i++) {  stmt.setObject(i + 1, args[i]);  }  return stmt;  }  public static int update(String sql, Object... args) throws SQLException {  try {  PreparedStatement stmt = JdbcHelper.getStmt(sql, args);  try {  return stmt.executeUpdate();  } finally {  stmt.getConnection().close();  }  } catch (Exception e) {  throw new RuntimeException(e);  }  }  public static ResultSet query(String sql, Object... args) throws SQLException {  PreparedStatement stmt = JdbcHelper.getStmt(sql, args);  return stmt.executeQuery();  }  public static Object value(String sql, Object... args) {  try {  ResultSet rs = JdbcHelper.query(sql, args);  if (rs.next()) {  return rs.getObject(0);  }  rs.getStatement().getConnection().close();  return null;  } catch (SQLException e) {  throw new RuntimeException(e);  }  }  } |
| --- |

* + - 1. **Lớp tiện ích Auth**

| package com.farmsys.Helper;  import com.farmsys.Entity.NhanVien;  public class Auth {  public static NhanVien user = null;  public static void clear(){  Auth.user = null;  }  public static boolean isLogin(){  return Auth.user != null;  }  public static boolean isManager(){  return Auth.isLogin() && user.isVaiTro();  }  } |
| --- |

* + - 1. **Lớp tiện ích MsgBox**

| package com.farmsys.Helper;  import java.awt.Component;  import javax.swing.JOptionPane;  public class MsgBox {  public static void alert(Component parentComponent, String message){  JOptionPane.showMessageDialog(parentComponent, message, "FarmSys said", JOptionPane.INFORMATION\_MESSAGE);  }  public static boolean confirm(Component parent, String message){  int result = JOptionPane.showConfirmDialog(parent, message, "FarmSys said", JOptionPane.YES\_NO\_OPTION, JOptionPane.QUESTION\_MESSAGE);  return result == JOptionPane.YES\_OPTION;  }  public static String prompt(Component parent, String message){  return JOptionPane.showInputDialog(parent, message, "FarmSys said", JOptionPane.INFORMATION\_MESSAGE);  }  } |
| --- |

* + - 1. **Lớp tiện ích XImage**

| package com.farmsys.Helper;  import java.awt.Image;  import java.io.File;  import java.io.IOException;  import java.net.URL;  import java.nio.file.Files;  import java.nio.file.Path;  import java.nio.file.Paths;  import java.nio.file.StandardCopyOption;  import javax.swing.ImageIcon;  public class XImage {  public static Image getAppIcon() {  URL url = XImage.class.getResource("com/farmsys/icons/avatar.png");  return new ImageIcon(url).getImage();  //Lấy địa chỉ hình ảnh và tạo ảnh --> return  }  public static void save(File src) {  File dst = new File("logos", src.getName());  if (!dst.getParentFile().exists()) {  dst.getParentFile().mkdir();  }  try {  Path from = Paths.get(src.getAbsolutePath());  Path to = Paths.get(src.getAbsolutePath());  Files.copy(from, to, StandardCopyOption.REPLACE\_EXISTING);  } catch (IOException e) {  throw new RuntimeException(e);  }  }    public static boolean saveImage(File file) {  File dir = new File("logos");  // Tạo thư mục nếu chưa tồn tại  if (!dir.exists()) {  dir.mkdirs();  }  File newFile = new File(dir, file.getName());  try {  // Copy vào thư mục logos (đè nếu đã tồn tại)  Path source = Paths.get(file.getAbsolutePath());  Path destination = Paths.get(newFile.getAbsolutePath());  Files.copy(source, destination, StandardCopyOption.REPLACE\_EXISTING);  return true;  } catch (Exception ex) {  return false;  }  }  public static ImageIcon read(String fileName) {  File path = new File("logos", fileName);  return new ImageIcon(path.getAbsolutePath());  }  } |
| --- |

* + - 1. **HandleAPI**

| package com.farmsys.Helper;  import com.google.gson.Gson;  import com.farmsys.data.JsonResult;  import com.google.gson.JsonSyntaxException;  import java.io.IOException;  import okhttp3.OkHttpClient;  import okhttp3.Request;  import okhttp3.Response;  import okhttp3.ResponseBody;  public class HandleAPI {  public static JsonResult getJsonData(String city) {  OkHttpClient client = new OkHttpClient();  Gson gson = new Gson();  JsonResult data = null;  Request res = new Request.Builder().url("http://api.openweathermap.org/data/2.5/forecast?q=Ho Chi Minh city&APPID=bffca17bcb552b8c8e4f3b82f64cccd2&units=metric").build();  try {  Response response = client.newCall(res).execute();  ResponseBody body = response.body();  data = gson.fromJson(body.string(), JsonResult.class);  } catch (JsonSyntaxException | IOException e) {  }  return data;  }  } |
| --- |

* + - 1. **MailHelper**

| package com.farmsys.Helper;  import java.io.File;  import java.io.IOException;  import java.util.Date;  import java.util.Properties;  import javax.mail.Message;  import javax.mail.MessagingException;  import javax.mail.PasswordAuthentication;  import javax.mail.Session;  import javax.mail.Transport;  import javax.mail.internet.InternetAddress;  import javax.mail.internet.MimeBodyPart;  import javax.mail.internet.MimeMessage;  import javax.mail.internet.MimeMultipart;  /\*\*  \*  \* @author trieu Hướng dẫn sử dụng: Gửi mail văn bản: MailHelper.sendText("địa  \* chỉ mail", "tiêu đề", "nội dung"); Gửi mail chứa file:  \* MailHelper.sendFile("địa chỉ mail", "tiêu đề", "đường dẫn file");  \*/  public class MailHelper {  static String accountName = "farmsys.contact@gmail.com";  static String accountPassword = "FarmSys@123456";  public static void sendText(String email, String subject, String body) {  try {  Properties p = new Properties();  p.put("mail.smtp.auth", "true");  p.put("mail.smtp.starttls.enable", "true");  p.put("mail.smtp.host", "smtp.gmail.com");  p.put("mail.smtp.port", 587);  Session s = Session.getInstance(p,  new javax.mail.Authenticator() {  @Override  protected PasswordAuthentication getPasswordAuthentication() {  return new PasswordAuthentication(accountName, accountPassword);  }  });  String from = accountName;  String to = email;  Message msg = new MimeMessage(s);  msg.setFrom(new InternetAddress(from));  msg.setRecipients(Message.RecipientType.TO, InternetAddress.parse(to));  msg.setSubject(subject);  msg.setText(body);  msg.setSentDate(new Date());  Transport.send(msg);  } catch (MessagingException ex) {  System.out.println(ex);  }  }  public static void sendFile(String email, String subject, String body, String file) throws IOException {  try {  Properties p = new Properties();  p.put("mail.smtp.auth", "true");  p.put("mail.smtp.starttls.enable", "true");  p.put("mail.smtp.host", "smtp.gmail.com");  p.put("mail.smtp.port", 587);  Session s = Session.getInstance(p,  new javax.mail.Authenticator() {  @Override  protected PasswordAuthentication getPasswordAuthentication() {  return new PasswordAuthentication(accountName, accountPassword);  }  });  String from = accountName;  String to = email;  Message msg = new MimeMessage(s);  msg.setFrom(new InternetAddress(from));  msg.setRecipients(Message.RecipientType.TO, InternetAddress.parse(to));  msg.setSubject(subject);  msg.setText(body);  msg.setSentDate(new Date());  MimeMultipart multipart = new MimeMultipart();  MimeBodyPart messageBodyPart = new MimeBodyPart();  messageBodyPart.setText(body);  MimeBodyPart attachment = new MimeBodyPart();  attachment.attachFile(new File(file));  multipart.addBodyPart(messageBodyPart);  multipart.addBodyPart(attachment);  msg.setContent(multipart);  Transport.send(msg);  } catch (MessagingException ex) {  System.out.println(ex);  }  }  } |
| --- |

* + 1. **Model Class - Các lớp mô trả dữ liệu**
       1. **NhanVien**

| package com.farmsys.Entity;  public class NhanVien {  private String maNV;  private String matKhau;  private String hoTen;  private boolean gioiTinh;  private String email;  private int luong;  private String hinh;  private boolean vaiTro;  private String QRcodeString;  public NhanVien() {  }  public NhanVien(String maNV, String matKhau, String hoTen, boolean gioiTinh, String email, int luong, String hinh, boolean vaiTro, String QRcodeString) {  this.maNV = maNV;  this.matKhau = matKhau;  this.hoTen = hoTen;  this.gioiTinh = gioiTinh;  this.email = email;  this.luong = luong;  this.hinh = hinh;  this.vaiTro = vaiTro;  this.QRcodeString = QRcodeString;  }  public String getMaNV() {  return maNV;  }  public void setMaNV(String maNV) {  this.maNV = maNV;  }  public String getMatKhau() {  return matKhau;  }  public void setMatKhau(String matKhau) {  this.matKhau = matKhau;  }  public String getHoTen() {  return hoTen;  }  public void setHoTen(String hoTen) {  this.hoTen = hoTen;  }  public boolean isGioiTinh() {  return gioiTinh;  }  public void setGioiTinh(boolean gioiTinh) {  this.gioiTinh = gioiTinh;  }  public String getEmail() {  return email;  }  public void setEmail(String email) {  this.email = email;  }  public int getLuong() {  return luong;  }  public void setLuong(int luong) {  this.luong = luong;  }  public String getHinh() {  return hinh;  }  public void setHinh(String hinh) {  this.hinh = hinh;  }  public boolean isVaiTro() {  return vaiTro;  }  public void setVaiTro(boolean vaiTro) {  this.vaiTro = vaiTro;  }  public String getQRcodeString() {  return QRcodeString;  }  public void setQRcodeString(String QRcodeString) {  this.QRcodeString = QRcodeString;  }  @Override  public String toString() {  return maNV ;  }  } |
| --- |

* + - 1. **CayTrong**

| package com.farmsys.Entity;  public class CayTrong {  private int maCay;  private String tenCay;  private int thoiGianThuHoach;  private float doTDS;  private float doPH;  private float nhietDo;  private float doAm;  private String hinh;  public CayTrong(int maCay, String tenCay, int thoiGianThuHoach, float doTDS, float doPH, float nhietDo, float doAm, String hinh) {  this.maCay = maCay;  this.tenCay = tenCay;  this.thoiGianThuHoach = thoiGianThuHoach;  this.doTDS = doTDS;  this.doPH = doPH;  this.nhietDo = nhietDo;  this.doAm = doAm;  this.hinh = hinh;  }  public CayTrong() {  }  public int getMaCay() {  return maCay;  }  public void setMaCay(int maCay) {  this.maCay = maCay;  }  public String getTenCay() {  return tenCay;  }  public void setTenCay(String tenCay) {  this.tenCay = tenCay;  }  public int getThoiGianThuHoach() {  return thoiGianThuHoach;  }  public void setThoiGianThuHoach(int thoiGianThuHoach) {  this.thoiGianThuHoach = thoiGianThuHoach;  }  public float getDoTDS() {  return doTDS;  }  public void setDoTDS(float doTDS) {  this.doTDS = doTDS;  }  public float getDoPH() {  return doPH;  }  public void setDoPH(float doPH) {  this.doPH = doPH;  }  public float getNhietDo() {  return nhietDo;  }  public void setNhietDo(float nhietDo) {  this.nhietDo = nhietDo;  }  public float getDoAm() {  return doAm;  }  public void setDoAm(float doAm) {  this.doAm = doAm;  }  public String getHinh() {  return hinh;  }  public void setHinh(String hinh) {  this.hinh = hinh;  }  @Override  public String toString() {  return tenCay;  }  } |
| --- |

* + - 1. **GianTrong**

| package com.farmsys.Entity;  public class GianTrong {  private int maDan;  private String tenDan;  private boolean trangThai;  public GianTrong() {  }  public GianTrong(int maDan, String tenDan, boolean trangThai) {  this.maDan = maDan;  this.tenDan = tenDan;  this.trangThai = trangThai;  }  public int getMaDan() {  return maDan;  }  public void setMaDan(int maDan) {  this.maDan = maDan;  }  public String getTenDan() {  return tenDan;  }  public void setTenDan(String tenDan) {  this.tenDan = tenDan;  }  public boolean isTrangThai() {  return trangThai;  }  public void setTrangThai(boolean trangThai) {  this.trangThai = trangThai;  }  @Override  public String toString() {  return tenDan;  }  } |
| --- |

* + - 1. **CongViec**

| package com.farmsys.Entity;  public class CongViec {  int MaCV;  String TenCV;  public CongViec(int MaCV, String TenCV) {  this.MaCV = MaCV;  this.TenCV = TenCV;  }  public CongViec() {  }  public int getMaCV() {  return MaCV;  }  public void setMaCV(int MaCV) {  this.MaCV = MaCV;  }  public String getTenCV() {  return TenCV;  }  public void setTenCV(String TenCV) {  this.TenCV = TenCV;  }  @Override  public String toString() {  return TenCV ;  }    } |
| --- |

* + - 1. **NhatKy**

| package com.farmsys.Entity;  import java.util.Date;  public class NhatKy {  private int stt;  private String tenCV;  private String tenCay;  private String tenGian;  private String chiTiet;  private String nguoiTao;  private String nhanVien;  private Date ngayBatDau;  private Date ngayKetThuc;  private int trangThai; // 0: to do , 1:doing , 2:Từ chối , 3:Hoàn thành , 4:Hoàn thành + trễ  public NhatKy() {  }  public NhatKy(int stt, String tenCV, String tenCay, String tenGian, String chiTiet, String nguoiTao, String nhanVien, Date ngayBatDau, Date ngayKetThuc, int trangThai) {  this.stt = stt;  this.tenCV = tenCV;  this.tenCay = tenCay;  this.tenGian = tenGian;  this.chiTiet = chiTiet;  this.nguoiTao = nguoiTao;  this.nhanVien = nhanVien;  this.ngayBatDau = ngayBatDau;  this.ngayKetThuc = ngayKetThuc;  this.trangThai = trangThai;  }  public int getStt() {  return stt;  }  public void setStt(int stt) {  this.stt = stt;  }  public String getTenCV() {  return tenCV;  }  public void setTenCV(String tenCV) {  this.tenCV = tenCV;  }  public String getTenCay() {  return tenCay;  }  public void setTenCay(String tenCay) {  this.tenCay = tenCay;  }  public String getTenGian() {  return tenGian;  }  public void setTenGian(String tenGian) {  this.tenGian = tenGian;  }  public String getChiTiet() {  return chiTiet;  }  public void setChiTiet(String chiTiet) {  this.chiTiet = chiTiet;  }  public String getNguoiTao() {  return nguoiTao;  }  public void setNguoiTao(String nguoiTao) {  this.nguoiTao = nguoiTao;  }  public String getNhanVien() {  return nhanVien;  }  public void setNhanVien(String nhanVien) {  this.nhanVien = nhanVien;  }  public Date getNgayBatDau() {  return ngayBatDau;  }  public void setNgayBatDau(Date ngayBatDau) {  this.ngayBatDau = ngayBatDau;  }  public Date getNgayKetThuc() {  return ngayKetThuc;  }  public void setNgayKetThuc(Date ngayKetThuc) {  this.ngayKetThuc = ngayKetThuc;  }  public int getTrangThai() {  return trangThai;  }  public void setTrangThai(int trangThai) {  this.trangThai = trangThai;  }    } |
| --- |

* + - 1. **ThuHoach**

| package com.farmsys.Entity;  import java.util.Date;  public class ThuHoach {  private String TenGian;  private String TenCay;  private Date NgayTH;    public ThuHoach (){    }  public ThuHoach(String TenGian, String TenCay, Date NgayTH) {  this.TenGian = TenGian;  this.TenCay = TenCay;  this.NgayTH = NgayTH;  }  public String getTenGian() {  return TenGian;  }  public void setTenGian(String TenGian) {  this.TenGian = TenGian;  }  public String getTenCay() {  return TenCay;  }  public void setTenCay(String TenCay) {  this.TenCay = TenCay;  }  public Date getNgayTH() {  return NgayTH;  }  public void setNgayTH(Date NgayTH) {  this.NgayTH = NgayTH;  }  @Override  public String toString() {  return TenGian ;  }    } |
| --- |

* + - 1. **KhoHang**

| package com.farmsys.Entity;  import java.util.Date;  public class KhoHang {  private int MaTH;  private String TenGian;  private String TenCay;  private Float TrongLuong;  private Date NgayTH;  private Double GiaThanh;  public KhoHang() {  }  public KhoHang(int MaTH, String TenDan, String TenCay, Float TrongLuong, Date NgayTH, Double GiaThanh) {  this.MaTH = MaTH;  this.TenGian = TenDan;  this.TenCay = TenCay;  this.TrongLuong = TrongLuong;  this.NgayTH = NgayTH;  this.GiaThanh = GiaThanh;  }  public int getMaTH() {  return MaTH;  }  public void setMaTH(int MaTH) {  this.MaTH = MaTH;  }  public String getTenGian() {  return TenGian;  }  public void setTenGian(String TenDan) {  this.TenGian = TenDan;  }  public String getTenCay() {  return TenCay;  }  public void setTenCay(String TenCay) {  this.TenCay = TenCay;  }  public Float getTrongLuong() {  return TrongLuong;  }  public void setTrongLuong(Float TrongLuong) {  this.TrongLuong = TrongLuong;  }  public Date getNgayTH() {  return NgayTH;  }  public void setNgayTH(Date NgayTH) {  this.NgayTH = NgayTH;  }  public Double getGiaThanh() {  return GiaThanh;  }  public void setGiaThanh(Double GiaThanh) {  this.GiaThanh = GiaThanh;  }  } |
| --- |

* + - 1. **Chart**

| package com.farmsys.Entity;  public class Chart {  private Double Thang1;  private Double Thang2;  private Double Thang3;  private Double Thang4;  private Double Thang5;  private Double Thang6;  private Double Thang7;  private Double Thang8;  private Double Thang9;  private Double Thang10;  private Double Thang11;  private Double Thang12;    public Chart() {  }  public Chart(Double Thang1, Double Thang2, Double Thang3, Double Thang4, Double Thang5, Double Thang6, Double Thang7, Double Thang8, Double Thang9, Double Thang10, Double Thang11, Double Thang12) {  this.Thang1 = Thang1;  this.Thang2 = Thang2;  this.Thang3 = Thang3;  this.Thang4 = Thang4;  this.Thang5 = Thang5;  this.Thang6 = Thang6;  this.Thang7 = Thang7;  this.Thang8 = Thang8;  this.Thang9 = Thang9;  this.Thang10 = Thang10;  this.Thang11 = Thang11;  this.Thang12 = Thang12;  }  public Double getThang1() {  return Thang1;  }  public void setThang1(Double Thang1) {  this.Thang1 = Thang1;  }  public Double getThang2() {  return Thang2;  }  public void setThang2(Double Thang2) {  this.Thang2 = Thang2;  }  public Double getThang3() {  return Thang3;  }  public void setThang3(Double Thang3) {  this.Thang3 = Thang3;  }  public Double getThang4() {  return Thang4;  }  public void setThang4(Double Thang4) {  this.Thang4 = Thang4;  }  public Double getThang5() {  return Thang5;  }  public void setThang5(Double Thang5) {  this.Thang5 = Thang5;  }  public Double getThang6() {  return Thang6;  }  public void setThang6(Double Thang6) {  this.Thang6 = Thang6;  }  public Double getThang7() {  return Thang7;  }  public void setThang7(Double Thang7) {  this.Thang7 = Thang7;  }  public Double getThang8() {  return Thang8;  }  public void setThang8(Double Thang8) {  this.Thang8 = Thang8;  }  public Double getThang9() {  return Thang9;  }  public void setThang9(Double Thang9) {  this.Thang9 = Thang9;  }  public Double getThang10() {  return Thang10;  }  public void setThang10(Double Thang10) {  this.Thang10 = Thang10;  }  public Double getThang11() {  return Thang11;  }  public void setThang11(Double Thang11) {  this.Thang11 = Thang11;  }  public Double getThang12() {  return Thang12;  }  public void setThang12(Double Thang12) {  this.Thang12 = Thang12;  }  } |
| --- |

* + 1. **Data Class - Các lớp JOIN dữ liệu thời tiết**
       1. **City**

| package com.farmsys.data;  public class City {  private long id;  private String name;  private Coord coord;  private String country;  private long population;  private int timezone;  private long sinrise;  private long sunset;  public City() {  }  public long getId() {  return id;  }  public String getName() {  return name;  }  public Coord getCoord() {  return coord;  }  public String getCountry() {  return country;  }  public long getPopulation() {  return population;  }  public int getTimezone() {  return timezone;  }  public long getSinrise() {  return sinrise;  }  public long getSunset() {  return sunset;  }    } |
| --- |

* + - 1. **Clouds**

| package com.farmsys.data;  public class Clouds {  private int all;  public Clouds() {  }  public int getAll() {  return all;  }  public void setAll(int all) {  this.all = all;  }    } |
| --- |

* + - 1. **Coord**

| package com.farmsys.data;  public class Coord {  private double lat;  private double lon;  public Coord() {  }  public double getLat() {  return lat;  }  public double getLon() {  return lon;  }    } |
| --- |

* + - 1. **DailyWeather**

| package com.farmsys.data;  public class DailyWeather {  private long dt;  private Main main;  private Weather[] weather;  private Clouds clouds;  private Wind wind;  private int visibility;  private double pop;  private Rain rain;  private Sys sys;  private String dt\_txt;  public DailyWeather() {  }  public long getDt() {  return dt;  }  public Main getMain() {  return main;  }  public Weather[] getWeather() {  return weather;  }  public Clouds getClouds() {  return clouds;  }  public Wind getWind() {  return wind;  }  public int getVisibility() {  return visibility;  }  public double getPop() {  return pop;  }  public Rain getRain() {  return rain;  }  public Sys getSys() {  return sys;  }  public String getDt\_txt() {  return dt\_txt;  }    } |
| --- |

* + - 1. **JsonResult**

| package com.farmsys.data;  public class JsonResult {  private String cod;  private int message;  private int cnt;  private DailyWeather[] list;  private City city;  public JsonResult() {  }  public String getCod() {  return cod;  }  public int getMessage() {  return message;  }  public int getCnt() {  return cnt;  }  public DailyWeather[] getList() {  return list;  }  public City getCity() {  return city;  }    } |
| --- |

* + - 1. **Main**

| package com.farmsys.data;  public class Main {  private double temp;  private double feels\_like;  private double temp\_min;  private double temp\_max;  private int pressure;  private int sea\_level;  private int grnd\_level;  private int humidity;  private double temp\_kf;    public Main() {  }  public double getTemp() {  return temp;  }  public double getFeels\_like() {  return feels\_like;  }  public double getTemp\_min() {  return temp\_min;  }  public double getTemp\_max() {  return temp\_max;  }  public int getPresure() {  return pressure;  }  public int getSea\_level() {  return sea\_level;  }  public int getGrnd\_level() {  return grnd\_level;  }  public int getHumidity() {  return humidity;  }  public double getTemp\_kf() {  return temp\_kf;  }    } |
| --- |

* + - 1. **Rain**

| package com.farmsys.data;  public class Rain {  private double threeH;  public Rain(double threeH) {  this.threeH = threeH;  }  public double getThreeH() {  return threeH;  }  public void setThreeH(double threeH) {  this.threeH = threeH;  }    } |
| --- |

* + - 1. **Sys**

| package com.farmsys.data;  public class Sys {  private String pod;  public Sys() {  }  public String getPod() {  return pod;  }  public void setPod(String pod) {  this.pod = pod;  }    } |
| --- |

* + - 1. **Weather**

| package com.farmsys.data;  public class Weather {  private int id;  private String main;  private String description;  private String icon;  public Weather() {  }  public int getId() {  return id;  }  public void setId(int id) {  this.id = id;  }  public String getMain() {  return main;  }  public void setMain(String main) {  this.main = main;  }  public String getDescription() {  return description;  }  public void setDescription(String description) {  this.description = description;  }  public String getIcon() {  return icon;  }  public void setIcon(String icon) {  this.icon = icon;  }    } |
| --- |

* + - 1. **Wind**

| package com.farmsys.data;  public class Wind {  private double speed;  private int deg;  public Wind() {  }  public double getSpeed() {  return speed;  }  public void setSpeed(double speed) {  this.speed = speed;  }  public int getDeg() {  return deg;  }  public void setDeg(int deg) {  this.deg = deg;  }    } |
| --- |

* + 1. **DAO Class - Các lớp truy xuất dữ liệu**
       1. **FarmSysDAO**

| package com.farmsys.dao;  import java.util.List;  public abstract class FarmSysDAO<E,K> {  abstract public void insert(E entity);  abstract public void update(E entity);  abstract public void delete(K key);  abstract public List<E> selectAll();  abstract public E selectById(K key);  abstract protected List<E> selectBySql(String sql, Object... args);  } |
| --- |

* + - 1. **NhanVienDao**

| package com.farmsys.dao;  import com.farmsys.Entity.NhanVien;  import com.farmsys.Helper.JdbcHelper;  import java.sql.ResultSet;  import java.sql.SQLException;  import java.util.ArrayList;  import java.util.List;  import java.util.logging.Level;  import java.util.logging.Logger;  public class NhanVienDAO extends FarmSysDAO<NhanVien, String> {  String INSERT\_SQL = "INSERT INTO NhanVien(MaNV, MatKhau, HoTen, GioiTinh, Email, Luong, VaiTro, Hinh, QRcode) VALUES(?,?,?,?,?,?,?,?,?)";  String UPDATE\_SQL = "UPDATE NhanVien SET MatKhau=?,HoTen=?,GioiTinh=?,Email=?, Luong=?, VaiTro=?, Hinh=? WHERE MaNV=?";  String UPDATEQRcode\_SQL = "UPDATE NhanVien SET QRcode = ? WHERE QRcode = ?";  String DELETE\_SQL = "DELETE FROM NhanVien WHERE MaNV=?";  String SELECT\_ALL\_SQL = "SELECT \*FROM NhanVien";  String SELECT\_BY\_ID\_SQL = "SELECT \* FROM NhanVien WHERE MaNV=?";  String SELECT\_BY\_QR\_SQL = "SELECT \* FROM NhanVien WHERE QRcode=?";  String SELECT\_BY\_Email\_SQL = "SELECT \* FROM NhanVien WHERE Email=?";  String RESET\_PASS\_SQL = "UPDATE NhanVien SET MatKhau=? WHERE MaNV=?";  String SELECT\_NhanVien\_SQL = "SELECT \*FROM NhanVien WHERE VaiTro= 0";  String SELECT\_NhanVienQRcode\_SQL = "SELECT \*FROM NhanVien WHERE QRcode = ?";  @Override  public void insert(NhanVien entity) {  try {  JdbcHelper.update(INSERT\_SQL, entity.getMaNV(), entity.getMatKhau(), entity.getHoTen(), entity.isGioiTinh(), entity.getEmail(), entity.getLuong(), entity.isVaiTro(), entity.getHinh(), entity.getQRcodeString());  } catch (SQLException ex) {  Logger.getLogger(NhanVienDAO.class.getName()).log(Level.SEVERE, null, ex);  }  }  @Override  public void update(NhanVien entity) {  try {  JdbcHelper.update(UPDATE\_SQL, entity.getMatKhau(), entity.getHoTen(), entity.isGioiTinh(), entity.getEmail(), entity.getLuong(), entity.isVaiTro(), entity.getHinh(), entity.getMaNV());  } catch (SQLException ex) {  Logger.getLogger(NhanVienDAO.class.getName()).log(Level.SEVERE, null, ex);  }  }  public void updateQRcode(String entity) {  try {  JdbcHelper.update(UPDATE\_SQL, entity);  } catch (SQLException ex) {  Logger.getLogger(NhanVienDAO.class.getName()).log(Level.SEVERE, null, ex);  }  }  @Override  public void delete(String key) {  try {  JdbcHelper.update(DELETE\_SQL, key);  } catch (SQLException ex) {  Logger.getLogger(NhanVienDAO.class.getName()).log(Level.SEVERE, null, ex);  }  }  @Override  public List<NhanVien> selectAll() {  return this.selectBySql(SELECT\_ALL\_SQL);  }  public List<NhanVien> selectNhanVien() {  return this.selectBySql(SELECT\_NhanVien\_SQL);  }  @Override  public NhanVien selectById(String key) {  List<NhanVien> list = this.selectBySql(SELECT\_BY\_ID\_SQL, key);  if (list.isEmpty()) {  return null;  }  return list.get(0);  }  public NhanVien selectByQR(String key) {  List<NhanVien> list = this.selectBySql(SELECT\_BY\_QR\_SQL, key);  if (list.isEmpty()) {  return null;  }  return list.get(0);  }  public NhanVien selectByEmail(String key) {  List<NhanVien> list = this.selectBySql(SELECT\_BY\_Email\_SQL, key);  if (list.isEmpty()) {  return null;  }  return list.get(0);  }  public NhanVien selectByQRcodeFormNV(String key) {  List<NhanVien> list = this.selectBySql(SELECT\_NhanVienQRcode\_SQL, key);  if (list.isEmpty()) {  return null;  }  return list.get(0);  }  @Override  protected List<NhanVien> selectBySql(String sql, Object... args) {  List<NhanVien> list = new ArrayList<NhanVien>();  try {  ResultSet rs = JdbcHelper.query(sql, args);  while (rs.next()) {  NhanVien entity = new NhanVien();  entity.setMaNV(rs.getString("MaNV"));  entity.setMatKhau(rs.getString("MatKhau"));  entity.setHoTen(rs.getString("HoTen"));  entity.setGioiTinh(rs.getBoolean("GioiTinh"));  entity.setEmail(rs.getString("Email"));  entity.setLuong(rs.getInt("Luong"));  entity.setVaiTro(rs.getBoolean("VaiTro"));  entity.setHinh(rs.getString("Hinh"));  entity.setQRcodeString(rs.getString("QRcode"));  list.add(entity);  }  rs.getStatement().getConnection().close();  return list;  } catch (SQLException e) {  throw new RuntimeException(e);  }  }  public void resetPass(NhanVien entity) {  try {  JdbcHelper.update(RESET\_PASS\_SQL, entity.getMatKhau(), entity.getMaNV());  } catch (SQLException e) {  System.out.println(e);  }  }  } |
| --- |

* + - 1. **CayTrongDAO**

| package com.farmsys.dao;  import com.farmsys.Entity.CayTrong;  import com.farmsys.Helper.JdbcHelper;  import java.sql.ResultSet;  import java.sql.SQLException;  import java.util.ArrayList;  import java.util.List;  import java.util.logging.Level;  import java.util.logging.Logger;  public class CayTrongDAO extends FarmSysDAO<CayTrong, String> {  String INSERT\_SQL = "INSERT INTO LoaiCay(TenCay,ThoiGianThuHoach,DoTDS,DoPH,NhietDo,DoAm,Hinh) VALUES(?,?,?,?,?,?,?)";  String UPDATE\_SQL = "UPDATE LoaiCay SET TenCay=?, ThoiGianThuHoach=?, DoTDS=?, DoPH=?, NhietDo=?, DoAm=?, Hinh=? WHERE MaCay=?";  String DELETE\_SQL = "DELETE FROM LoaiCay WHERE MaCay=?";  String SELECT\_ALL\_SQL = "SELECT \*FROM LoaiCay";  String SELECT\_BY\_ID\_SQL = "SELECT \* FROM LoaiCay WHERE MaCay=?";  @Override  public void insert(CayTrong entity) {  try {  JdbcHelper.update(INSERT\_SQL, entity.getTenCay(), entity.getThoiGianThuHoach(), entity.getDoTDS(), entity.getDoPH(), entity.getNhietDo(), entity.getDoAm(), entity.getHinh());  } catch (SQLException ex) {  Logger.getLogger(CayTrongDAO.class.getName()).log(Level.SEVERE, null, ex);  }  }  @Override  public void update(CayTrong entity) {  try {  JdbcHelper.update(UPDATE\_SQL, entity.getTenCay(), entity.getThoiGianThuHoach(), entity.getDoTDS(), entity.getDoPH(), entity.getNhietDo(), entity.getDoAm(), entity.getHinh(), entity.getMaCay());  } catch (SQLException ex) {  Logger.getLogger(CayTrongDAO.class.getName()).log(Level.SEVERE, null, ex);  }  }  @Override  public void delete(String key) {  try {  JdbcHelper.update(DELETE\_SQL, key);  } catch (SQLException ex) {  Logger.getLogger(CayTrongDAO.class.getName()).log(Level.SEVERE, null, ex);  }  }  @Override  public List<CayTrong> selectAll() {  return this.selectBySql(SELECT\_ALL\_SQL);  }  @Override  public CayTrong selectById(String key) {  List<CayTrong> list = this.selectBySql(SELECT\_BY\_ID\_SQL, key);  if (list.isEmpty()) {  return null;  }  return list.get(0);  }  @Override  protected List<CayTrong> selectBySql(String sql, Object... args) {  List<CayTrong> list = new ArrayList<CayTrong>();  try {  ResultSet rs = JdbcHelper.query(sql, args);  while (rs.next()) {  CayTrong entity = new CayTrong();  entity.setMaCay(rs.getInt("MaCay"));  entity.setTenCay(rs.getString("TenCay"));  entity.setThoiGianThuHoach(rs.getInt("ThoiGianThuHoach"));  entity.setDoTDS(rs.getFloat("DoTDS"));  entity.setDoPH(rs.getFloat("DoPH"));  entity.setNhietDo(rs.getInt("NhietDo"));  entity.setDoAm(rs.getFloat("DoAm"));  entity.setHinh(rs.getString("Hinh"));  list.add(entity);  }  rs.getStatement().getConnection().close();  return list;  } catch (SQLException e) {  throw new RuntimeException(e);  }  }  public CayTrong selectByIdInt(Integer mact) {  List<CayTrong> list = this.selectBySql(SELECT\_BY\_ID\_SQL, mact);  if (list.isEmpty()) {  return null;  }  return list.get(0);  }  } |
| --- |

* + - 1. **DanTrongDAO**

| package com.farmsys.dao;  import com.farmsys.Entity.GianTrong;  import com.farmsys.Helper.JdbcHelper;  import java.sql.ResultSet;  import java.sql.SQLException;  import java.util.ArrayList;  import java.util.List;  import java.util.logging.Level;  import java.util.logging.Logger;  public class GianTrongDAO extends FarmSysDAO<GianTrong, String> {  String INSERT\_SQL = "INSERT INTO GianTrong (TenGian, TrangThai) VALUES(?,?)";  String UPDATE\_SQL = "UPDATE GianTrong SET TenGian =?, TrangThai =? WHERE MaGian =?";  String DELETE\_SQL = "DELETE FROM GianTrong WHERE MaGian =?";  String SELECT\_ALL\_SQL = "SELECT \* FROM GianTrong";  String SELECT\_BY\_ID\_SQL = "SELECT \* FROM GianTrong WHERE MaGian =?";  String SELECT\_BY\_ID\_TT = "SELECT \* FROM GianTrong WHERE TrangThai = ?";  String SELECT\_BY\_ID\_TenGian = "SELECT \* FROM GianTrong WHERE TenGian =?";  String UPDATE\_TrangThai\_SQL = "UPDATE GianTrong SET TrangThai = 1 WHERE TenGian = ?";  String UPDATE\_TraVeCbo\_SQL = "UPDATE GianTrong SET TrangThai = 0 WHERE TenGian = ?";    @Override  public void insert(GianTrong entity) {  try {  JdbcHelper.update(INSERT\_SQL, entity.getTenDan(), entity.isTrangThai());  } catch (SQLException ex) {  Logger.getLogger(GianTrongDAO.class.getName()).log(Level.SEVERE, null, ex);  }  }  @Override  public void update(GianTrong entity) {  try {  JdbcHelper.update(UPDATE\_SQL, entity.getTenDan(), entity.isTrangThai(), entity.getMaDan());  } catch (SQLException ex) {  Logger.getLogger(GianTrongDAO.class.getName()).log(Level.SEVERE, null, ex);  }  }  public void updateTrangThai(String TenGian) {  try {  JdbcHelper.update(UPDATE\_TrangThai\_SQL, TenGian);  } catch (SQLException ex) {  Logger.getLogger(GianTrongDAO.class.getName()).log(Level.SEVERE, null, ex);  }  }  public void updateAgain(String TenGian) {  try {  JdbcHelper.update(UPDATE\_TraVeCbo\_SQL, TenGian);  } catch (SQLException ex) {  Logger.getLogger(GianTrongDAO.class.getName()).log(Level.SEVERE, null, ex);  }  }  @Override  public void delete(String key) {  try {  JdbcHelper.update(DELETE\_SQL, key);  } catch (SQLException ex) {  Logger.getLogger(GianTrongDAO.class.getName()).log(Level.SEVERE, null, ex);  }  }  @Override  public List<GianTrong> selectAll() {  return this.selectBySql(SELECT\_ALL\_SQL);  }  public List<GianTrong> selectByTT(Integer key) {  return this.selectBySql(SELECT\_BY\_ID\_TT, key);  }  public GianTrong selectById(Integer key) {  List<GianTrong> list = this.selectBySql(SELECT\_BY\_ID\_SQL, key);  if (list.isEmpty()) {  return null;  }  return list.get(0);  }  public GianTrong selectByTenGian(String key) {  List<GianTrong> list = this.selectBySql(SELECT\_BY\_ID\_TenGian, key);  if (list.isEmpty()) {  return null;  }  return list.get(0);  }  @Override  protected List<GianTrong> selectBySql(String sql, Object... args) {  List<GianTrong> list = new ArrayList<GianTrong>();  try {  ResultSet rs = JdbcHelper.query(sql, args);  while (rs.next()) {  GianTrong entity = new GianTrong();  entity.setMaDan(rs.getInt("MaGian"));  entity.setTenDan(rs.getString("TenGian"));  entity.setTrangThai(rs.getBoolean("TrangThai"));  list.add(entity);  }  rs.getStatement().getConnection().close();  return list;  } catch (SQLException e) {  throw new RuntimeException(e);  }  }  } |
| --- |

* + - 1. **CongViecDAO**

| package com.farmsys.dao;  import com.farmsys.Entity.CongViec;  import com.farmsys.Helper.JdbcHelper;  import java.sql.ResultSet;  import java.sql.SQLException;  import java.util.ArrayList;  import java.util.List;  public class CongViecDAO extends FarmSysDAO<CongViec, String> {  String SELECT\_ALL\_SQL = "SELECT \* FROM CongViec";  String SELECT\_BY\_ID\_SQL = "SELECT \* FROM CongViec WHERE TenCV=?";    @Override  public List<CongViec> selectAll() {  return this.selectBySql(SELECT\_ALL\_SQL);  }    @Override  public CongViec selectById(String key) {  List<CongViec> list = this.selectBySql(SELECT\_BY\_ID\_SQL, key);  if (list.isEmpty()) {  return null;  }  return list.get(0);  }  @Override  protected List<CongViec> selectBySql(String sql, Object... args) {  List<CongViec> list = new ArrayList<CongViec>();  try {  ResultSet rs = JdbcHelper.query(sql, args);  while (rs.next()) {  CongViec entity = new CongViec();  //entity.setMaCV(rs.getInt("MaCV"));  entity.setTenCV(rs.getString("TenCV"));  list.add(entity);  }  rs.getStatement().getConnection().close();  return list;  } catch (SQLException e) {  throw new RuntimeException(e);  }  }  @Override  public void insert(CongViec entity) {  throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.  }  @Override  public void update(CongViec entity) {  throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.  }  @Override  public void delete(String key) {  throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.  }    } |
| --- |

* + - 1. **NhatKyDAO**

| package com.farmsys.dao;  import com.farmsys.Entity.CayTrong;  import com.farmsys.Entity.NhatKy;  import com.farmsys.Helper.JdbcHelper;  import java.util.List;  import java.sql.ResultSet;  import java.sql.SQLException;  import java.util.ArrayList;  import java.util.logging.Level;  import java.util.logging.Logger;  public class NhatKyDAO extends FarmSysDAO<NhatKy, String> {  String INSERT\_SQL = "INSERT INTO NhatKy (TenCV, TenCay, TenGian, ChiTiet, NguoiTao, Nhanvien, NgayBatDau, NgayKetThuc, TrangThai) VALUES(?,?,?,?,?,?,?,?,?)";  String select\_all\_sql = "select\*from NhatKy";  String select\_by\_ten\_sql = "select \* from NhatKy where TenCV = ?";  String select\_by\_trangthai\_sql = "select \* from NhatKy where TrangThai = ?";  String select\_by\_trangthaivaten\_sql = "select \* from NhatKy where TrangThai = ? and NhanVien like ?";  String select\_formtodoanddoing\_sql = "select \* from NhatKy where STT = ?";  String UPDATE\_Done\_SQL = "UPDATE NhatKy SET TrangThai = 3 where STT = ?";  String UPDATE\_SQL = "UPDATE NhatKy SET TrangThai = 1 where STT = ?";  String UPDATE\_TuChoi\_SQL = "UPDATE NhatKy SET TrangThai = 2 where STT = ?";  String select\_by\_Done\_Month\_SQL ="select \*from NhatKy where TrangThai = 3 and NhanVien like ? and NgayKetThuc between (select CONVERT(varchar,dateadd(d,-(day(getdate()-1)),getdate()),106)) and (select CONVERT(varchar,dateadd(d,-(day(dateadd(m,1,getdate()))),dateadd(m,1,getdate())),106))" ;  String UPDATE\_TrangThaiTH\_SQL = "UPDATE NhatKy SET TrangThai = 5 where TenGian = ? ";    @Override  public void insert(NhatKy entity) {  try {  JdbcHelper.update(INSERT\_SQL, entity.getTenCV(), entity.getTenCay(), entity.getTenGian(), entity.getChiTiet(), entity.getNguoiTao(), entity.getNhanVien(), entity.getNgayBatDau(), entity.getNgayKetThuc(), entity.getTrangThai());  } catch (SQLException ex) {  Logger.getLogger(CayTrongDAO.class.getName()).log(Level.SEVERE, null, ex);  }  }  public void update(int entity) {  try {  JdbcHelper.update(UPDATE\_SQL, entity);  } catch (SQLException ex) {  Logger.getLogger(NhanVienDAO.class.getName()).log(Level.SEVERE, null, ex);  }  }  public void updateTuChoi(int entity) {  try {  JdbcHelper.update(UPDATE\_TuChoi\_SQL, entity);  } catch (SQLException ex) {  Logger.getLogger(NhanVienDAO.class.getName()).log(Level.SEVERE, null, ex);  }  }  @Override  public void delete(String key) {  }  @Override  public List<NhatKy> selectAll() {  return selectBySql(select\_all\_sql);  }  public void updateTrangThai(int stt) {  try {  JdbcHelper.update(UPDATE\_Done\_SQL, stt);  } catch (SQLException ex) {  Logger.getLogger(NhatKyDAO.class.getName()).log(Level.SEVERE, null, ex);  }  }  @Override  public NhatKy selectById(String macv) {  List<NhatKy> list = this.selectBySql(select\_by\_ten\_sql, macv);  if (list.isEmpty()) {  return null;  }  return list.get(0);  }  public NhatKy selectformtodoanddoing(Integer macv) {  List<NhatKy> list = this.selectBySql(select\_formtodoanddoing\_sql, macv);  if (list.isEmpty()) {  return null;  }  return list.get(0);  }  public NhatKy selectById(Integer macv) {  List<NhatKy> list = this.selectBySql(select\_by\_ten\_sql, macv);  if (list.isEmpty()) {  return null;  }  return list.get(0);  }  public List<NhatKy> selectByTen(String tencv) {  return selectBySql(select\_by\_ten\_sql, tencv);  }  public List<NhatKy> selectByTrangThai(int trangthai) {  return selectBySql(select\_by\_trangthai\_sql, trangthai);  }  public List<NhatKy> selectByTrangThaivaTennv(int trangthai, String tennv) {  return selectBySql(select\_by\_trangthaivaten\_sql, trangthai, tennv);  }  @Override  protected List<NhatKy> selectBySql(String sql, Object... args) {  List<NhatKy> list = new ArrayList<NhatKy>();  try {  ResultSet rs = JdbcHelper.query(sql, args);  while (rs.next()) {  NhatKy entity = new NhatKy();//Tạo đối tượng NhatKy  entity.setStt(rs.getInt("STT"));  entity.setTenCV(rs.getString("TenCV"));  entity.setTenCay(rs.getString("TenCay"));  entity.setTenGian(rs.getString("TenGian"));  entity.setChiTiet(rs.getString("ChiTiet"));  entity.setNguoiTao(rs.getString("NguoiTao"));  entity.setNhanVien(rs.getString("NhanVien"));  entity.setNgayBatDau(rs.getDate("NgayBatDau"));  entity.setNgayKetThuc(rs.getDate("NgayKetThuc"));  entity.setTrangThai(rs.getInt("TrangThai"));  list.add(entity);  }  rs.getStatement().getConnection().close();  return list;  } catch (SQLException ex) {  throw new RuntimeException(ex);  }  }  @Override  public void update(NhatKy entity) {  throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.  }  public List<NhatKy> selectDoneByMonth(String MaNV) {  return selectBySql(select\_by\_Done\_Month\_SQL,MaNV);  }  public void updateTrangThaiTH(String TenGian) {  try {  JdbcHelper.update(UPDATE\_TrangThaiTH\_SQL, TenGian);  } catch (SQLException ex) {  Logger.getLogger(NhanVienDAO.class.getName()).log(Level.SEVERE, null, ex);  }  }  } |
| --- |

* + - 1. **KhoHangDAO**

| package com.farmsys.dao;  import com.farmsys.Entity.KhoHang;  import com.farmsys.Helper.JdbcHelper;  import java.sql.ResultSet;  import java.sql.SQLException;  import java.util.ArrayList;  import java.util.List;  import java.util.logging.Level;  import java.util.logging.Logger;  public class KhoHangDAO {  String INSERT\_SQL = "INSERT INTO KhoHang (TenGian, TenCay, TrongLuong, ThoiGianThuHoach) VALUES(?,?,?,?)";  String SELECT\_ALL\_SQL = "select\*from KhoHang where TrongLuong > 0";  String SELECT\_BY\_TEN\_SQL = "select \* from KhoHang where TenGian = ?";  String DELETE\_SQL = "DELETE FROM KhoHang where TenDan = ?";  String UPDATE\_SQL = "UPDATE KhoHang SET TrongLuong =? WHERE MaTH =?";  String SELECT\_SLBan\_SQL = "UPDATE KhoHang set TrongLuong = ? , Coin = ? where MaTH = ?";  public List<KhoHang> selectAll() {  return this.selectBySql(SELECT\_ALL\_SQL);  }  public void insert(KhoHang entity) {  try {  JdbcHelper.update(INSERT\_SQL, entity.getTenGian(), entity.getTenCay(), entity.getTrongLuong(), entity.getNgayTH());  } catch (SQLException ex) {  Logger.getLogger(CayTrongDAO.class.getName()).log(Level.SEVERE, null, ex);  }  }  public void update(KhoHang entity) {  try {  JdbcHelper.update(UPDATE\_SQL, entity.getTrongLuong(), entity.getMaTH());  } catch (SQLException ex) {  Logger.getLogger(KhoHangDAO.class.getName()).log(Level.SEVERE, null, ex);  }  }  public void update(Float TrongLuong, Double Coin, int MaTH) {  try {  JdbcHelper.update(SELECT\_SLBan\_SQL, TrongLuong, Coin, MaTH);  } catch (SQLException ex) {  Logger.getLogger(KhoHangDAO.class.getName()).log(Level.SEVERE, null, ex);  }  }  public void delete(String key) {  try {  JdbcHelper.update(DELETE\_SQL, key);  } catch (SQLException ex) {  Logger.getLogger(KhoHangDAO.class.getName()).log(Level.SEVERE, null, ex);  }  }  public KhoHang selectById(String key) {  List<KhoHang> list = this.selectBySql(SELECT\_BY\_TEN\_SQL, key);  if (list.isEmpty()) {  return null;  }  return list.get(0);  }  protected List<KhoHang> selectBySql(String sql, Object... args) {  List<KhoHang> list = new ArrayList<KhoHang>();  try {  ResultSet rs = JdbcHelper.query(sql, args);  while (rs.next()) {  KhoHang entity = new KhoHang();  entity.setMaTH(rs.getInt("MaTH"));  entity.setTenGian(rs.getString("TenGian"));  entity.setTenCay(rs.getString("TenCay"));  entity.setTrongLuong(rs.getFloat("TrongLuong"));  entity.setNgayTH(rs.getDate("ThoiGianThuHoach"));  entity.setGiaThanh(rs.getDouble("Coin"));  list.add(entity);  }  rs.getStatement().getConnection().close();  return list;  } catch (SQLException e) {  throw new RuntimeException(e);  }  }  public List<KhoHang> selectByKeyword(String keyword) {  String SELECT\_BY\_NAME\_SQL = "SELECT \* FROM KhoHang WHERE TenCay LIKE ? and TrongLuong > 0";  return this.selectBySql(SELECT\_BY\_NAME\_SQL, "%" + keyword + "%");  }  } |
| --- |

* + - 1. **ThuHoachDAO**

| package com.farmsys.dao;  import com.farmsys.Entity.CongViec;  import com.farmsys.Entity.ThuHoach;  import com.farmsys.Helper.JdbcHelper;  import java.sql.ResultSet;  import java.sql.SQLException;  import java.util.ArrayList;  import java.util.List;  public class ThuHoachDAO extends FarmSysDAO<ThuHoach, String> {    String SELECT\_ThuHoach\_SQL = "select NhatKy.TenGian,LoaiCay.TenCay,(SELECT DATEADD(day, +(LoaiCay.ThoiGianThuHoach) , NhatKy.NgayKetThuc)) as 'NgayTH' from LoaiCay inner join NhatKy on LoaiCay.TenCay = NhatKy.TenCay where TenCV like N'Trồng cây' and GETDATE() >= (SELECT DATEADD(day, +(LoaiCay.ThoiGianThuHoach) , NhatKy.NgayKetThuc))and TrangThai = 3";        @Override  public void insert(ThuHoach entity) {  throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.  }  @Override  public void update(ThuHoach entity) {  throw new UnsupportedOperationException("Not supported yet.");  }  @Override  public void delete(String key) {  throw new UnsupportedOperationException("Not supported yet.");  }  @Override  public List<ThuHoach> selectAll() {  return this.selectBySql(SELECT\_ThuHoach\_SQL);  }    @Override  public ThuHoach selectById(String key) {  throw new UnsupportedOperationException("Not supported yet.");  }  @Override  protected List<ThuHoach> selectBySql(String sql, Object... args) {  List<ThuHoach> list = new ArrayList<ThuHoach>();  try {  ResultSet rs = JdbcHelper.query(sql, args);  while (rs.next()) {  ThuHoach entity = new ThuHoach();  entity.setTenGian(rs.getString("TenGian"));  entity.setTenCay(rs.getString("TenCay"));  entity.setNgayTH(rs.getDate("NgayTH"));  list.add(entity);  }  rs.getStatement().getConnection().close();  return list;  } catch (SQLException e) {  throw new RuntimeException(e);  }  }  } |
| --- |

* + - 1. **ChartDAO**

| package com.farmsys.dao;  import com.farmsys.Entity.Chart;  import com.farmsys.Entity.ThuHoach;  import com.farmsys.Helper.JdbcHelper;  import java.sql.ResultSet;  import java.sql.SQLException;  import java.time.Year;  import java.util.ArrayList;  import java.util.List;  public class ChartDAO extends FarmSysDAO<Chart, String> {  String SELECT\_ALL\_SQL = "select distinct (select SUM(coin) from KhoHang where ThoiGianThuHoach between" + "'" + Year.now() + "/01/01' and" + "'" + Year.now() + "/01/31') as 'TongTien01',\n"  + " (select SUM(coin) from KhoHang where ThoiGianThuHoach between" + "'" + Year.now() + "/02/01' and" + "'" + Year.now() + "/02/28') as 'TongTien02',\n"  + " (select SUM(coin) from KhoHang where ThoiGianThuHoach between" + "'" + Year.now() + "/03/01' and" + "'" + Year.now() + "/03/31') as 'TongTien03',\n"  + " (select SUM(coin) from KhoHang where ThoiGianThuHoach between" + "'" + Year.now() + "/04/01' and" + "'" + Year.now() + "/04/30') as 'TongTien04',\n"  + " (select SUM(coin) from KhoHang where ThoiGianThuHoach between" + "'" + Year.now() + "/05/01' and" + "'" + Year.now() + "/05/31') as 'TongTien05',\n"  + " (select SUM(coin) from KhoHang where ThoiGianThuHoach between" + "'" + Year.now() + "/06/01' and" + "'" + Year.now() + "/06/30') as 'TongTien06',\n"  + " (select SUM(coin) from KhoHang where ThoiGianThuHoach between" + "'" + Year.now() + "/07/01' and" + "'" + Year.now() + "/07/31') as 'TongTien07',\n"  + " (select SUM(coin) from KhoHang where ThoiGianThuHoach between" + "'" + Year.now() + "/08/01' and" + "'" + Year.now() + "/08/31') as 'TongTien08',\n"  + " (select SUM(coin) from KhoHang where ThoiGianThuHoach between" + "'" + Year.now() + "/09/01' and" + "'" + Year.now() + "/09/30') as 'TongTien09',\n"  + " (select SUM(coin) from KhoHang where ThoiGianThuHoach between" + "'" + Year.now() + "/10/01' and" + "'" + Year.now() + "/10/31') as 'TongTien10',\n"  + " (select SUM(coin) from KhoHang where ThoiGianThuHoach between" + "'" + Year.now() + "/11/01' and" + "'" + Year.now() + "/11/30') as 'TongTien11',\n"  + " (select SUM(coin) from KhoHang where ThoiGianThuHoach between" + "'" + Year.now() + "/12/01' and" + "'" + Year.now() + "/12/31') as 'TongTien12'\n"  + " from KhoHang";  @Override  public void insert(Chart entity) {  throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.  }  @Override  public void update(Chart entity) {  throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.  }  @Override  public void delete(String key) {  throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.  }  @Override  public List<Chart> selectAll() {  return this.selectBySql(SELECT\_ALL\_SQL);  }  @Override  public Chart selectById(String key) {  throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.  }  @Override  protected List<Chart> selectBySql(String sql, Object... args) {  List<Chart> list = new ArrayList<Chart>();  try {  ResultSet rs = JdbcHelper.query(sql, args);  while (rs.next()) {  Chart entity = new Chart();  entity.setThang1(rs.getDouble("TongTien01"));  entity.setThang2(rs.getDouble("TongTien02"));  entity.setThang3(rs.getDouble("TongTien03"));  entity.setThang4(rs.getDouble("TongTien04"));  entity.setThang5(rs.getDouble("TongTien05"));  entity.setThang6(rs.getDouble("TongTien06"));  entity.setThang7(rs.getDouble("TongTien07"));  entity.setThang8(rs.getDouble("TongTien08"));  entity.setThang9(rs.getDouble("TongTien09"));  entity.setThang10(rs.getDouble("TongTien10"));  entity.setThang11(rs.getDouble("TongTien11"));  entity.setThang12(rs.getDouble("TongTien12"));  list.add(entity);  }  rs.getStatement().getConnection().close();    return list;  } catch (SQLException e) {  throw new RuntimeException(e);  }  }  } |
| --- |

* 1. **Viết mã cho ứng dụng**
     1. **Màn hình chào**

| private void init() {  this.setLocationRelativeTo(null);  pgbLogging.setForeground(Color.black);  new Timer(40, (ActionEvent e) -> {  int value = pgbLogging.getValue();  if (value < pgbLogging.getMaximum()) {  if (value <= 0) {  pgbLogging.setString("Đang vào chương trình....");  }  if (value >= 40) {  pgbLogging.setString("Đang vào chương trình....");  }  if (value >= 60) {  pgbLogging.setString("Đang kết nối với database....");  }  if (value == 70) {  pgbLogging.setString("Đang kết nối với database....");  }  if (value == 80) {  JdbcHelper.ConnectDAO();  }  pgbLogging.setValue(value + 5);  } else {  ChaoJdialog.this.dispose();  }  }).start();  } |
| --- |

* + 1. **Đăng nhập**

| void SendOTP() {  try {  manvotp = txttaikhoanlaymk.getText();  NhanVien nhanVien = dao.selectById(manvotp);  if (nhanVien != null) {//check tk có tồn tại không  emailNV = dao.selectById(manvotp).getEmail();//check mail nv  if (emailNV == null) {  MsgBox.alert(this, "Tài khoản này chưa có email");  } else {//tài khoản có mail --> gửi mail -->check otp  MsgBox.alert(this, "Đang gửi mail...");  this.randomString();  MailHelper.sendText(emailNV, "OTP - Quên mật khẩu", "OTP:" + " " + OTP);  MsgBox.alert(this, "Có 1 mã OTP đã được gửi vào mail của bạn !");  }  } else {  MsgBox.alert(this, "Tài khoản không tồn tại");  }  } catch (Exception e) {  }  }  void dangNhap() {  try {  String taikhoan = txtMaNV.getText().trim();  String matKhau = new String(txtMatKhau.getPassword());  String QRcode1 = txtQRcode.getText().trim();  NhanVien nhanVien = dao.selectById(taikhoan);  NhanVien nhanVienEM = dao.selectByEmail(taikhoan);  NhanVien nhanvienQRcode = dao.selectByQRcodeFormNV(QRcode1);  if (nhanVien == null) {  if (nhanVienEM == null) {  if (nhanvienQRcode == null) {  // MsgBox.alert(this, "Sai tên đăng nhập hoặc Email!");  // clearForm();  lblsaitk.setText("Sai tài khoản !");  lblsaimk.setText("");  } else if (!QRcode1.equals(nhanvienQRcode.getQRcodeString())) {  MsgBox.alert(this, "Mã QR code không đúng !");  } else {  Auth.user = nhanvienQRcode;  webcam.close();  this.dispose();  }  } else if (!matKhau.equals(nhanVienEM.getMatKhau())) {  // clearForm();  } else {  Auth.user = nhanVienEM;  webcam.close();  this.dispose();  }  } else if (!matKhau.equals(nhanVien.getMatKhau())) {  // MsgBox.alert(this, "Sai mật khẩu!");  lblsaimk.setText("Sai mật khẩu !");  lblsaitk.setText("");  // clearForm();  } else {  Auth.user = nhanVien;  this.dispose();  webcam.close();  this.dispose();  }  } catch (Exception e) {  }  }  void ketThuc() {  if (MsgBox.confirm(this, "Bạn muốn kết thúc ứng dụng?")) {  System.exit(0);  }  }  public static String HienThihoten() {  return Auth.user.getHoTen();  }  public static String HienThivaitro() {  return (Auth.user.isVaiTro() ? "Trưởng phòng" : "Nhân viên");  }  public static String HienThiTen() {  return Auth.user.getMaNV();  }  private void randomString() {  OTP = UUID.randomUUID().toString();  }  private void doiMatKhau(String manv) {  String matKhauMoi1 = new String(txtmknew.getPassword());  String matKhauMoi2 = new String(txtxnmknew.getPassword());  if (!matKhauMoi1.equals(matKhauMoi2)) {  MsgBox.alert(this, "Xác nhận mật khẩu không đúng!");  this.clearForm();  } else {  tempOTP = txtOTP.getText();  if (tempOTP.equals(OTP)) {//đúng OTP thì hiện form đổi mk  nv.setMatKhau(matKhauMoi1);  nv.setMaNV(manv);  dao.resetPass(nv);  MsgBox.alert(this, "Đổi mật khẩu thành công!");  } else {  MsgBox.alert(this, "Bạn nhập sai OTP");  txtMaNV.setText("");  txtOTP.setText("");  txtmknew.setText("");  txtxnmknew.setText("");  }  }  } |
| --- |

* + 1. **Quên mật khẩu**

|  |
| --- |

* + 1. **Tổng quan**

| void setThoiTiet() {  String weath = " Weather: " + result.getList()[idx].getWeather()[0].getDescription();  lblweather.setText(weath);  String text = lblweather.getText();  if (text.contains("clouds")) {  lblsunrain.setIcon(new ImageIcon("src\\com\\farmsys\\icons\\cloud\_100px.png"));  } else if (text.contains("sun")) {  lblsunrain.setIcon(new ImageIcon("src\\com\\farmsys\\icons\\sun\_100px.png"));  } else if (text.contains("rain")) {  lblsunrain.setIcon(new ImageIcon("src\\com\\farmsys\\icons\\rain\_100px.png"));  }  }  private void loadLbl() {  lblhoten.setText(DangNhapJDialog.HienThihoten());  lblvaitro.setText(DangNhapJDialog.HienThivaitro());  edit();  }  private void openDangXuat() {  int ask = JOptionPane.showConfirmDialog(this, "Bạn có chắc là mình muốn đăng xuất ?", "Xác nhận", JOptionPane.YES\_NO\_OPTION);  if (ask == 0) {  this.dispose();  new TongQuan().setVisible(true);  }  }  private void OpenNhanVien() {  if (Auth.isManager()) {  tabs.setSelectedIndex(6);  } else {  MsgBox.alert(this, "Bạn không có quyền truy cập!");  }  }  private void OpenGianTrong() {  if (Auth.isManager()) {  tabs.setSelectedIndex(8);  } else {  MsgBox.alert(this, "Bạn không có quyền truy cập!");  }  }  private void OpenCayTrong() {  if (Auth.isManager()) {  tabs.setSelectedIndex(4);  } else {  MsgBox.alert(this, "Bạn không có quyền truy cập!");  }  }  private void OpenNhatky() {  if (Auth.isLogin()) {  tabs.setSelectedIndex(2);  } else {  MsgBox.alert(this, "Vui lòng đăng nhập!");  }  }  private void OpenGiaoViec() {  if (Auth.isManager()) {  tabs.setSelectedIndex(5);  } else {  MsgBox.alert(this, "Bạn không có quyền truy cập!");  }  }  private void OpenTodo() {  if (Auth.isLogin()) {  tabs.setSelectedIndex(3);  } else {  MsgBox.alert(this, "Vui lòng đăng nhập!");  }  }  private void Opendoing() {  if (Auth.isLogin()) {  tabs.setSelectedIndex(7);  } else {  MsgBox.alert(this, "Vui lòng đăng nhập!");  }  }  private void Openchart() {  if (Auth.isLogin()) {  tabs.setSelectedIndex(9);  } else {  MsgBox.alert(this, "Vui lòng đăng nhập!");  }  }  private void OpenLive() {  if (Auth.isLogin()) {  this.start();  tabs.setSelectedIndex(1);  } else {  MsgBox.alert(this, "Vui lòng đăng nhập!");  }  }  private void OpenKhohang() {  if (Auth.isManager()) {  tabs.setSelectedIndex(10);  } else {  MsgBox.alert(this, "Bạn không có quyền truy cập!");  }  } |
| --- |

* + 1. **Quản lý nhân viên**

| void init() {  if (!Auth.isLogin()) {  MsgBox.alert(this, "Chưa đăng nhập, vô cái loz");  System.exit(0);  }  this.updateStatus();  new Timer(60000, (ActionEvent e) -> {  this.fillTable();  }).start();  }  void insert() {  if (!Auth.isManager()) {  MsgBox.alert(this, "Không có quyền xóa nhân viên!");  } else {  if (timKiemNhanVien() == -1) {  NhanVien nv = getForm();  String mk1 = txtMatKhau.getText();  String mk2 = txtMatKhau2.getText();  if (!mk2.equals(mk1)) {  MsgBox.alert(this, "Xác nhận mật khẩu không đúng");  } else {  try {  this.createQRcode();  dao.insert(nv);  this.fillTable();  MsgBox.alert(this, "Thêm mới thành công!");  this.Sendmail();  } catch (Exception e) {  MsgBox.alert(this, "Thêm thất bại!");  System.out.println(e);  }  }  }  }  }  void update() {  NhanVien nv = getForm();  String mk1 = new String(txtMatKhau.getPassword());  String mk2 = new String(txtMatKhau2.getPassword());  if (!mk2.equals(mk1)) {  MsgBox.alert(this, "Xác nhận mật khẩu không đúng!");  } else {  try {  this.createQRcode();  dao.update(nv);  this.fillTable();  MsgBox.alert(this, "Cập nhật thành công!");  this.Sendmail();  } catch (Exception e) {  MsgBox.alert(this, "Cập nhật thất bại!");  System.out.println(e);  }  }  }  void delete() {  if (!Auth.isManager()) {  MsgBox.alert(this, "Không có quyền xóa nhân viên!");  } else {  String manv = txtMaNV.getText();  System.out.println(Auth.user.getMaNV());  System.out.println(manv);  if (manv.equals(Auth.user.getMaNV())) {  MsgBox.alert(this, "Bạn không được xóa chính bạn!");  } else if (MsgBox.confirm(this, "bạn thực sự muốn xóa nhân viên này?")) {  try {  dao.delete(manv);  this.fillTable();  this.clearForm();  MsgBox.alert(this, "Xóa thành công!");  } catch (Exception e) {  MsgBox.alert(this, "Xóa thất bại!");  System.out.println(e);  }  }  }  }  void clearForm() {  NhanVien nv = new NhanVien();  this.setForm(nv);  this.row = -1;  lblHinh1.setIcon(new javax.swing.ImageIcon(getClass().getResource("/com/farmsys/icons/avatar.png")));  this.updateStatus();  }  void edit() {  String manv = (String) tblNhanVien.getValueAt(this.row, 0);  NhanVien nv = dao.selectById(manv);  this.setForm(nv);  this.updateStatus();  }  void first() {  this.row = 0;  this.edit();  }  void prev() {  if (this.row > 0) {  this.row--;  this.edit();  }  }  void next() {  if (this.row < tblNhanVien.getRowCount() - 1) {  this.row++;  this.edit();  }  }  void last() {  this.row = tblNhanVien.getColumnCount() - 1;  this.edit();  }  void fillTable() {  DefaultTableModel model = (DefaultTableModel) tblNhanVien.getModel();  model.setRowCount(0);  try {  List<NhanVien> list = dao.selectAll();  for (NhanVien nv : list) {  Object[] row = {  nv.getMaNV(), "\*\*\*\*\*\*\*\*\*\*\*", nv.getHoTen(),  nv.isVaiTro() ? "Trưởng phòng" : "Nhân viên",  nv.getEmail(), nv.isGioiTinh() ? "Nam" : "Nữ", nv.getLuong()  };  model.addRow(row);  }  } catch (Exception e) {  MsgBox.alert(this, "Lỗi truy vấn dữ liệu!");  }  }  void setForm(NhanVien nv) {  txtMaNV.setText(nv.getMaNV());  txtHoTen.setText(nv.getHoTen());  txtMatKhau.setText(nv.getMatKhau());  txtMatKhau2.setText(nv.getMatKhau());  rdoTruongPhong.setSelected(nv.isVaiTro());  rdoNhanVien.setSelected(!nv.isVaiTro());  rdoNam.setSelected(nv.isGioiTinh());  rdoNu.setSelected(!nv.isGioiTinh());  txtEmail.setText(nv.getEmail());  txtLuong.setText(nv.getLuong() + "");  txtqrcode.setText(nv.getQRcodeString() + "");  if (nv.getHinh() != null) {  lblHinh1.setToolTipText(nv.getHinh());  ImageIcon icon = XImage.read(nv.getHinh()); // Lấy địa chỉ của file Icon  //Chuyển Icon sang image và điều chỉnh kích thước  Image scaleIcon = icon.getImage().getScaledInstance(lblHinh1.getWidth(), lblHinh1.getHeight(), Image.SCALE\_DEFAULT);  lblHinh1.setIcon(new javax.swing.ImageIcon(scaleIcon));  } else {  lblHinh1.setIcon(new javax.swing.ImageIcon(getClass().getResource("/com/farmsys/icons/avatar.png")));  }  }  NhanVien getForm() {  if (validation()) {  NhanVien nv = new NhanVien();  nv.setMaNV(txtMaNV.getText());  nv.setHoTen(txtHoTen.getText());  nv.setMatKhau(new String(txtMatKhau.getPassword()));  nv.setVaiTro(rdoTruongPhong.isSelected());  //xét giới tính  nv.setGioiTinh(rdoNam.isSelected());  nv.setEmail(txtEmail.getText());  nv.setLuong(Integer.parseInt(txtLuong.getText()));  nv.setQRcodeString(txtqrcode.getText());  nv.setHinh(lblHinh1.getToolTipText());  return nv;  }  return null;  }  void updateStatus() {  boolean edit = (this.row >= 0);  boolean first = (this.row == 0);  boolean last = (this.row == tblNhanVien.getColumnCount() - 1);  //Trạng thái form  txtMaNV.setEditable(!edit);  btnThem.setEnabled(!edit);  btnSua.setEnabled(edit);  btnXoa.setEnabled(edit);  }  private void randomString() {  QRcoderandomString = UUID.randomUUID().toString();  txtqrcode.setText(QRcoderandomString);  }  private void createQRcode() {  try {  String qrCodeData = QRcoderandomString;  String filePath = "src\\QRcode\\a.png";  String charset = "UTF-8"; // or "ISO-8859-1"  Map< EncodeHintType, ErrorCorrectionLevel> hintMap = new EnumMap<>(EncodeHintType.class);  hintMap.put(EncodeHintType.ERROR\_CORRECTION, ErrorCorrectionLevel.L);  BitMatrix matrix = new MultiFormatWriter().encode(  new String(qrCodeData.getBytes(charset), charset),  BarcodeFormat.QR\_CODE, 200, 200, hintMap);  MatrixToImageWriter.writeToFile(matrix, filePath.substring(filePath  .lastIndexOf('.') + 1), new File(filePath));  System.out.println("QR Code image created successfully!");  } catch (WriterException | IOException e) {  System.err.println(e);  }  }  void Sendmail() {  MsgBox.alert(this, "Đang gửi mail...");  String body = "Đây là mã QR code cá nhân. Vui lòng không để cho người khác có được mã này !";  try {  manv = txtMaNV.getText();  NhanVien nhanVien = dao.selectById(manv);  if (nhanVien != null) {//check tk có tồn tại không  emailNV = dao.selectById(manv).getEmail();//check mail nv  if (emailNV == null) {  MsgBox.alert(this, "Tài khoản này chưa có email");  } else {//tài khoản có mail --> gửi mail -->check otp  MailHelper.sendFile(emailNV, "QRCode to FarmSys", body, "src\\QRCODE\\a.png");  MsgBox.alert(this, "Gửi mail thành công !");  }  } else {  MsgBox.alert(this, "Tài khoản không tồn tại");  }  } catch (Exception e) {  }  }  void chonAnh() {  if (fileChooser.showOpenDialog(this) == JFileChooser.APPROVE\_OPTION) {  File file = fileChooser.getSelectedFile();  XImage.save(file); // Lưu hình vào thư mục logos  ImageIcon icon = XImage.read(file.getName()); // Đọc hình từ logos  Image scaleIcon = icon.getImage().getScaledInstance(lblHinh1.getWidth(), lblHinh1.getHeight(), Image.SCALE\_DEFAULT);  lblHinh1.setIcon(new javax.swing.ImageIcon(scaleIcon));  lblHinh1.setToolTipText(file.getName()); //Giữ tên hình trong toolTip  }  }  private boolean validation() {  if (txtMaNV.getText().isEmpty()) {  MsgBox.alert(this, "Bạn chưa nhập mã nhân viên!");  txtMaNV.requestFocus();  return false;  }  if (txtMatKhau.getPassword().toString().isEmpty()) {  MsgBox.alert(this, "Bạn chưa nhập mật khẩu!");  txtMatKhau.requestFocus();  return false;  }  if (txtMatKhau2.getPassword().toString().isEmpty()) {  MsgBox.alert(this, "Bạn chưa nhập xác nhận mật khẩu!");  txtMatKhau2.requestFocus();  return false;  }  if (txtHoTen.getText().isEmpty()) {  MsgBox.alert(this, "Bạn chưa nhập họ tên nhân viên!");  txtHoTen.requestFocus();  return false;  }  if (txtEmail.getText().isEmpty()) {  MsgBox.alert(this, "Bạn chưa nhập email!");  txtEmail.requestFocus();  return false;  }  if (txtLuong.getText().isEmpty()) {  MsgBox.alert(this, "Bạn chưa nhập lương!");  txtLuong.requestFocus();  return false;  }  if (!txtMatKhau2.getText().equals(txtMatKhau.getText())) {  txtMatKhau2.requestFocus();  return false;  }  try {  Integer.parseInt(txtLuong.getText());  } catch (Exception e) {  MsgBox.alert(this, "Lương phải là số!");  txtLuong.requestFocus();  return false;  }  if (Integer.parseInt(txtLuong.getText()) < 0) {  MsgBox.alert(this, "Lương không được âm!");  txtLuong.requestFocus();  return false;  }  //kiểm tra tính hợp lệ đữ liệu bằng biểu thức chính quy  if (!(txtEmail.getText().matches("\\w+@\\w+\\.\\w+"))) {  MsgBox.alert(this, "Bạn nhập sai định dạng email!");  txtEmail.requestFocus();  return false;  }  return true;  } |
| --- |

* + 1. **Quản lý cây trồng**

| public void init() {  load();  tblCayTrong.setDefaultEditor(Object.class, null);  new Timer(60000, (ActionEvent e) -> {  this.load();  }).start();  clear();  }  void load() {  DefaultTableModel model = (DefaultTableModel) tblCayTrong.getModel();  model.setRowCount(0);  try {  List<CayTrong> list = ctdao.selectAll();  for (CayTrong ct : list) {  Object[] row = {  ct.getTenCay(),  ct.getThoiGianThuHoach(),  ct.getDoTDS(),  ct.getDoPH(),  ct.getNhietDo(),  ct.getDoAm(),  ct.getHinh()  };  model.addRow(row);  }  } catch (Exception e) {  e.printStackTrace();  }  }  void insert() {  CayTrong model = getModelInsert();  try {  ctdao.insert(model);  this.load();  this.clear();  MsgBox.alert(this, "Thêm mới thành công!");  } catch (Exception e) {  e.printStackTrace();  }  }  void update() {  CayTrong model = getModel();  try {  ctdao.update(model);  this.load();  this.clear();  MsgBox.alert(this, "Cập nhật thành công!");  } catch (Exception e) {  MsgBox.alert(this, "Cập nhật thất bại!");  }  }  void delete() {  if (txtTenCay.getText().isEmpty()) {  MsgBox.alert(this, "Bạn chưa chọn cây cần xóa");  } else {  if (MsgBox.confirm(this, "Bạn có muốn xóa hay không?")) {  String Tencay = txtTenCay.getText();  try {  ctdao.delete(Tencay);  this.load();  this.clear();  MsgBox.alert(this, "Xóa thành công!");  } catch (Exception e) {  MsgBox.alert(this, "Xóa thất bại!");  }  }  }  }  void clear() {  CayTrong model = new CayTrong();  txtTenCay.setText("");  txtThoiGianThuHoach.setText("");  txtDoAm.setText("60");  txtDoPH.setText("");  txtDoTDS.setText("");  txtNhietDo.setText("");  lblHinh.setIcon(new ImageIcon("src\\com\\farmsys\\icons\\FarmSys.png"));  this.getstatus(true);  }  boolean Validation() {  if (txtTenCay.getText().isEmpty()) {  MsgBox.alert(this, "Tên cây không được để trống ");  txtTenCay.requestFocus();  return false;  }  if (txtThoiGianThuHoach.getText().isEmpty()) {  MsgBox.alert(this, "Thời gian thu hoạch cây không được để trống ");  txtThoiGianThuHoach.requestFocus();  return false;  }  if (txtDoTDS.getText().isEmpty()) {  MsgBox.alert(this, "Độ TDS của cây không được để trống ");  txtDoTDS.requestFocus();  return false;  }  if (txtDoPH.getText().isEmpty()) {  MsgBox.alert(this, "Độ PH của cây không được để trống ");  txtDoPH.requestFocus();  return false;  }  if (txtNhietDo.getText().isEmpty()) {  MsgBox.alert(this, "Nhiệt độ cây không được để trống ");  txtNhietDo.requestFocus();  return false;  }  if (txtDoAm.getText().isEmpty()) {  MsgBox.alert(this, "Độ ẩm của cây không được để trống ");  txtDoAm.requestFocus();  return false;  }  String tencay = txtTenCay.getText();  if (tencay.length() > 30) {  MsgBox.alert(this, "Tên cây không được quá 30 kí tự ");  txtTenCay.requestFocus();  return false;  }  //bắt không nhập chữ  try {  int ngayTH = Integer.parseInt(txtThoiGianThuHoach.getText());  } catch (NumberFormatException e) {  MsgBox.alert(this, "Thời gian thu hoạch không được nhập chữ!");  txtThoiGianThuHoach.requestFocus();  return false;  }  //  try {  Float.parseFloat(txtDoPH.getText());  } catch (Exception e) {  MsgBox.alert(this, "Độ PH không được nhập chữ!");  txtDoPH.requestFocus();  return false;  }  //  try {  Integer.parseInt(txtNhietDo.getText());  } catch (Exception e) {  MsgBox.alert(this, "Nhiệt độ không được nhập chữ!");  txtNhietDo.requestFocus();  return false;  }  //  try {  Integer.parseInt(txtDoAm.getText());  } catch (Exception e) {  MsgBox.alert(this, "Độ PH không được nhập chữ!");  txtDoAm.requestFocus();  return false;  }  //  try {  Float.parseFloat(txtDoTDS.getText());  } catch (Exception e) {  MsgBox.alert(this, "Độ TDS không được nhập chữ!");  txtDoTDS.requestFocus();  return false;  }  //  float tds = Float.parseFloat(txtDoTDS.getText());  if (tds < 0) {  MsgBox.alert(this, "Độ TDS phải là số dương");  txtDoTDS.requestFocus();  return false;  }  int nhietdo = Integer.parseInt(txtNhietDo.getText());  if (nhietdo < 24 || nhietdo > 27) {  MsgBox.alert(this, "Nhiệt độ tốt cho cây phải từ 24°C -> 27°C");  txtNhietDo.requestFocus();  return false;  }  int doAM = Integer.parseInt(txtDoAm.getText());  if (doAM < 60 || doAM > 65) {  MsgBox.alert(this, "Độ ẩm tốt cho cây phải từ 60% -> 65%");  txtDoAm.requestFocus();  return false;  }  //doph  if ((Float.parseFloat(txtDoPH.getText())) < 0 || (Float.parseFloat(txtDoPH.getText())) > 14) {  MsgBox.alert(this, "Độ PH của cây từ 1 -> 14");  txtDoPH.requestFocus();  return false;  }  //ngayth  if ((Integer.parseInt(txtThoiGianThuHoach.getText())) < 3 || (Integer.parseInt(txtThoiGianThuHoach.getText())) > 60) {  MsgBox.alert(this, "Thời gian thu hoạch phải từ 3 -> 60 ngày");  txtThoiGianThuHoach.requestFocus();  return false;  }  return true;  } |
| --- |

* + 1. **Quản lý giàn trồng**

| private void init() {  this.load();  this.setStatus(true);  }  void load() {  DefaultTableModel model = (DefaultTableModel) tblGianTrong.getModel();  model.setRowCount(0);  try {  List<GianTrong> list = gtdao.selectAll();  for (GianTrong gt : list) {  Object[] row = {  gt.getMaDan(),  gt.getTenDan(),  gt.isTrangThai() ? "Đã trồng" : "Chưa trồng"  };  model.addRow(row);  }  } catch (Exception e) {  MsgBox.alert(this, "Lỗi truy vấn dữ liệu!");  }  }  void fillTableToComboBoxLoc() {  DefaultTableModel model = (DefaultTableModel) tblGianTrong.getModel();  model.setRowCount(0);  try {  List<GianTrong> list = gtdao.selectByTT(cboLoc.getSelectedIndex() - 1);  for (GianTrong gt : list) {  Object[] row = {  gt.getMaDan(),  gt.getTenDan(),  gt.isTrangThai() ? "Đã trồng" : "Chưa trồng"  };  model.addRow(row);  }  } catch (Exception e) {  MsgBox.alert(this, "Lỗi truy vấn dữ liệu!");  }  }  void insert() {  GianTrong model = getModelInsert();  try {  gtdao.insert(model);  this.load();  this.clear();  MsgBox.alert(this, "Thêm mới thành công !");  } catch (Exception e) {  MsgBox.alert(this, "Thêm mới không thành công !");  }  }  void update() {  GianTrong model = getModel();  try {  gtdao.update(model);  this.load();  MsgBox.alert(this, "Cập nhật thành công!");  } catch (Exception e) {  MsgBox.alert(this, "Cập nhật thất bại!");  System.out.println(e);  }  }  void delete() {  if (MsgBox.confirm(this, "Bạn thực sự muốn xóa dàn trồng này?")) {  String manv = txtMaDanTrong.getText();  try {  gtdao.delete(manv);  this.load();  this.clear();  MsgBox.alert(this, "Xóa thành công!");  } catch (Exception e) {  MsgBox.alert(this, "Xóa thất bại!");  }  }  }  void edit() {  try {  Integer magt = (Integer) tblGianTrong.getValueAt(this.index, 0);  GianTrong model = gtdao.selectById(magt);  this.setModel(model);  this.setStatus(false);  } catch (Exception e) {  MsgBox.alert(this, "Lỗi truy vấn dữ liệu!");  }  }  void clear() {  GianTrong gt = new GianTrong();  this.setModel(gt);  this.index = -1;  this.setStatus(true);  txtMaDanTrong.setText("");  txtTrangThai.setText("Chưa hoạt động");  }  private boolean validation() {  if (txtTenDanTrong.getText().isEmpty()) {  MsgBox.alert(this, "Bạn chưa nhập tên dàn trồng !");  txtTenDanTrong.requestFocus();  return false;  }  if (txtTenDanTrong.getText().equalsIgnoreCase("Nhập tên dàn trồng !")) {  MsgBox.alert(this, "Bạn chưa nhập tên dàn trồng !");  txtTenDanTrong.requestFocus();  return false;  }  return true;  } |
| --- |

* + 1. **Giao việc**

| void init() {  txtnguoitao.setText(Auth.user.getMaNV());  fillComboBoxCongViec();  fillComboBoxCayTrong();  // fillComboBoxGianTrong();  fillComboBoxNhanVien();  }  void fillComboBoxCayTrong() {  DefaultComboBoxModel model = (DefaultComboBoxModel) cboCayTrong.getModel();  model.removeAllElements();  List<CayTrong> list = ctdao.selectAll();  for (CayTrong ct : list) {  model.addElement(ct);  }  }  void fillComboBoxGianTrong() {  DefaultComboBoxModel model = (DefaultComboBoxModel) cboGianTrong.getModel();  model.removeAllElements();  List<GianTrong> list = dtdao.selectByTT(0);  for (GianTrong dt : list) {  model.addElement(dt);  }  }  void fillComboBoxGianTrongTrongCay() {  DefaultComboBoxModel model = (DefaultComboBoxModel) cboGianTrong.getModel();  model.removeAllElements();  List<GianTrong> list = dtdao.selectByTT(0);  for (GianTrong dt : list) {  model.addElement(dt);  }  }  void fillComboBoxGianTrongChamSoc() {  DefaultComboBoxModel model = (DefaultComboBoxModel) cboGianTrong.getModel();  model.removeAllElements();  List<GianTrong> list = dtdao.selectByTT(1);  for (GianTrong dt : list) {  model.addElement(dt);  }  }  void fillComboBoxGianTrongThuHoach() {  DefaultComboBoxModel model = (DefaultComboBoxModel) cboGianTrong.getModel();  model.removeAllElements();  List<ThuHoach> list = thdao.selectAll();  System.out.println(thdao.selectAll());  for (ThuHoach th : list) {  model.addElement(th);  }  }  void fillComboBoxCongViec() {  DefaultComboBoxModel model = (DefaultComboBoxModel) cboCongViec.getModel();  model.removeAllElements();  List<CongViec> list = cvdao.selectAll();  for (CongViec cv : list) {  model.addElement(cv);  }  }  void fillComboBoxNhanVien() {  DefaultComboBoxModel model = (DefaultComboBoxModel) cboNhanVien.getModel();  model.removeAllElements();  List<NhanVien> list = nvdao.selectNhanVien();  for (NhanVien nv : list) {  model.addElement(nv);  }  }  private boolean Validation() {  //kiểm lỗi ngày bắt đầu  try {  txtNgayBatDau.getCalendar().getTime();  } catch (Exception e) {  MsgBox.alert(this, "Bạn chưa chọn ngày bắt đầu");  txtNgayBatDau.requestFocus();  return false;  }  //kiểm lỗi ngày kết thúc  try {  txtNgayKetThuc.getCalendar().getTime();  } catch (Exception e) {  MsgBox.alert(this, "Bạn chưa chọn ngày kết thúc");  txtNgayKetThuc.requestFocus();  return false;  }  return true;  } |
| --- |

* + 1. **Nhật ký**

| private void init() {  this.fillTableNhatALL();  this.TheNVJDialog.setLocationRelativeTo(null);  new Timer(60000, (ActionEvent e) -> {  this.fillTableNhatALL();  }).start();  }  private void fillTableNhatALL() {  DefaultTableModel model = (DefaultTableModel) tblNhatKy.getModel();  model.setRowCount(0);  List<NhatKy> list = nkdao.selectAllMonth();  for (NhatKy nk : list) {  String status = trangThai(nk);  model.addRow(new Object[]{  nk.getTenCV(),  nk.getTenCay(),  nk.getTenGian(),  nk.getChiTiet(),  nk.getNguoiTao(),  nk.getNhanVien(),  nk.getNgayBatDau(),  nk.getNgayKetThuc(),  status  });  }  }  private void fillTableNhatKyByTenCay() {  DefaultTableModel model = (DefaultTableModel) tblNhatKy.getModel();  model.setRowCount(0);  String TimKiem = txttimkiem.getText();  List<NhatKy> list = nkdao.selectByTenCay(TimKiem);  for (NhatKy nk : list) {  String status = trangThai(nk);  model.addRow(new Object[]{  nk.getTenCV(),  nk.getTenCay(),  nk.getTenGian(),  nk.getChiTiet(),  nk.getNguoiTao(),  nk.getNhanVien(),  nk.getNgayBatDau(),  nk.getNgayKetThuc(),  status  });  }  }  private void fillTableNhatKyByTenCV() {  DefaultTableModel model = (DefaultTableModel) tblNhatKy.getModel();  model.setRowCount(0);  String TimKiem = txttimkiem.getText();  List<NhatKy> list = nkdao.selectByTenCay(TimKiem);  for (NhatKy nk : list) {  String status = trangThai(nk);  model.addRow(new Object[]{  nk.getTenCV(),  nk.getTenCay(),  nk.getTenGian(),  nk.getChiTiet(),  nk.getNguoiTao(),  nk.getNhanVien(),  nk.getNgayBatDau(),  nk.getNgayKetThuc(),  status  });  }  }  private void fillTableNhatKyByTenNV() {  DefaultTableModel model = (DefaultTableModel) tblNhatKy.getModel();  model.setRowCount(0);  String TimKiem = txttimkiem.getText();  List<NhatKy> list = nkdao.selectByTenNV(TimKiem);  for (NhatKy nk : list) {  String status = trangThai(nk);  model.addRow(new Object[]{  nk.getTenCV(),  nk.getTenCay(),  nk.getTenGian(),  nk.getChiTiet(),  nk.getNguoiTao(),  nk.getNhanVien(),  nk.getNgayBatDau(),  nk.getNgayKetThuc(),  status  });  }  } |
| --- |

* + 1. **…..**
    2. **……**
    3. **……..**
    4. **…….**
    5. **…..**

1. **Kiểm tra chương trình**
   1. **Kiểm thử form Đăng Nhập**

| **Form kiểm thử** | **Các bước kiểm thử** | **Kết quả xảy ra** | **Kết quả đạt được** | **Người thực hiện** | **Ghi chú** |
| --- | --- | --- | --- | --- | --- |
| Đăng nhập | Nhập tên tài khoản có chứa kí tự | Hiển thị thông báo  “Tên tài khoản không được chứa kí tự” | Login thất bại | Tuấn |  |
| Đăng nhập | Nhập password không đúng với password trưởng phòng cấp | Hiển thị thông báo  “Tên tài khoản hoặc password không chính xác” | Login thất bại | Tuấn |  |
| Đăng nhập | Nhập tên tài khoản không đúng với tên tài khoản công ty cấp | Hiển thị thông báo  “Tên tài khoản hoặc password không chính xác” | Login thất bại | Tuấn |  |
| Đăng nhập | Mã QR không đúng với hệ thống cấp | Hiển thị thông báo  “Mã QR không chính xác hoặc đã hết hạn” | Login thất bại | Tuấn |  |
| Đăng nhập | Đăng nhập thất bại nhiều lần | Hiển thị thông báo  “Tài khoản của bạn đã bị khoá tạm thời” | Login thất bại | Tuấn |  |
| Đăng nhập | Đăng nhập với đúng tài khoản, mật khẩu | Hiển thị thông báo  “Đăng nhập thành công !” | Login thành công | Tuấn |  |
| Đăng nhập | Không nhập dữ liệu vào tên tài khoản và mật khẩu | Hiển thị thông báo  “Bạn chưa nhập thông tin đăng nhập !” | Login thất bại | Tuấn |  |
| Đăng nhập | Không nhập tên tài khoản | Hiển thị thông báo “Bạn chưa nhập tên tài khoản” | Login thất bại | Triều |  |
| Đăng nhập | Không |  |  |  |  |

* 1. **Kiểm thử form Nhân Viên**

| **Form kiểm thử** | **Các bước kiểm thử** | **Kết quả xảy ra** | **Kết quả đạt được** | **Người thực hiện** | **Ghi chú** |
| --- | --- | --- | --- | --- | --- |
| Nhân Viên | 1) Click button "New"  2) Nhập vào form những thông tin:  • Mã nhân viên: trieu  • Họ và tên: Đông Triều  • Mật khẩu: 123456  • Email: [trieu@gmail.com](mailto:trieu@gmail.com)  • Mật khẩu: 123456  • Lương CB: 1000000  3) Click button "Thêm"  4) Show message:"Thêm nhân viên thành công !" | - Thêm một nhân viên vào Database và chương trình thông báo: "Thêm nhân viên thành công !" | Thêm nhân viên thành công | Triều |  |
| Nhân Viên | 1) Click button "New"  2) Nhập vào form những thông tin:  • Mã nhân viên: trieu  • Họ và tên: Đông Triều 1234  • Mật khẩu: 123456  • Email: [trieu@gmail.com](mailto:trieu@gmail.com)  • Mật khẩu: 123456  • Lương CB: 1000000  3) Click button "Thêm"  4) Show message:"Thêm nhân viên thành công !" | - Thêm một nhân viên vào Database và chương trình thông báo: "Thêm nhân viên thất bại !" | Thêm nhân viên thất bại | Triều |  |
| Nhân Viên | 1) Click button "New"  2) Nhập vào form những thông tin:  • Mã nhân viên: **null**  • Họ và tên: **null**  • Mật khẩu: **null**  • Email: **null**  • Mật khẩu: **null**  • Lương CB: **null**  3) Click button "Thêm"  4) Show message:"Thêm nhân viên thành công !" | - Thêm một nhân viên vào Database và chương trình thông báo: "Bạn chưa nhập thông tin nhân viên !" | Thêm nhân viên thất bại | Triều |  |
| Nhân Viên | Mã QR không hợp lệ | Hiển thị thông báo  “Mã QR không chính xác hoặc đã hết hạn” | Lấy mã QR thất bại | Triều |  |
| Nhân Viên | Lấy mã QR thành công | Hiển thị thông báo  “Lấy Mã QR thành công” | Lấy mã QR thành công | Triều |  |
| Nhân Viên | 1) Click 1 dòng trên Table Nhân Viên  2) Nhập vào form những thông tin:  • Mã nhân viên: trieu  • Họ và tên: Đông Triều  • Mật khẩu: 123456  • Email: [trieu@gmail.com](mailto:trieu@gmail.com)  • Mật khẩu: 123456  • Lương CB: 1000000  3) Click button "Thêm"  4) Show message:"Cập nhật nhân viên thành công !" | - Thêm một nhân viên vào Database và chương trình thông báo: "Cập nhật nhân viên thành công !" | Cập nhật nhân viên thành công | Triều |  |
| Nhân Viên | 1) Click 1 dòng trên Table Nhân Viên  2) Nhập vào form những thông tin:  • Mã nhân viên: trieu  • Họ và tên: - Đông Triều @@@@  • Mật khẩu: 123456  • Email: [trieu@gmail.com](mailto:trieu@gmail.com)  • Mật khẩu: 123456  • Lương CB: 1000000  3) Click button "Thêm"  4) Show message:"Cập nhật nhân viên thất bại !" | - Thêm một nhân viên vào Database và chương trình thông báo: "Cập nhật nhân viên thành công !" | Cập nhật nhân viên thất bại | Triều |  |
| Nhân Viên | 1) Click button "New"  2) Nhập vào form những thông tin:  • Mã nhân viên: trieu  • Họ và tên: Đông Triều 123  • Mật khẩu: 123456  • Email: [trieu@gmail.com](mailto:trieu@gmail.com)  • Mật khẩu: 123456  • Lương CB: - 1000000  3) Click button "Thêm"  4) Show message:"Thêm nhân viên thành công !" | - Thêm một nhân viên vào Database và chương trình thông báo: "Cập nhật nhân viên thất bại !" | Cập nhật nhân viên thất bại | Triều |  |
| Nhân Viên | 1) Click button "New"  2) Nhập vào form những thông tin:  • Mã nhân viên: **null**  • Họ và tên: **null**  • Mật khẩu: **null**  • Email: **null**  • Mật khẩu: **null**  • Lương CB: - **null**  3) Click button "Thêm"  4) Show message:"Thêm nhân viên thành công !" | - Thêm một nhân viên vào Database và chương trình thông báo: "Cập nhật nhân viên thất bại !" | Cập nhật nhân viên thất bại | Triều |  |
| Nhân Viên | 1) Click 1 dòng trên Table Nhân Viên  2) Click button "Xóa"  3) Show message:"Bạn có muốn xóa nhân viên này không !" => Yes  4) Trở về form Nhân Viên | Xoá một nhân viên trong Database và hiển thị thông báo "Xoá nhân viên thành công !" | Xóa thành công | Triều |  |
| Nhân Viên | 1) Click 1 dòng trên Table Nhân Viên  2) Click button "Xóa"  3) Show message:"Bạn có muốn xóa nhân viên này không !" => Cancel  4) Trở về form Nhân Viên | Xoá một nhân viên trong Database và hiển thị thông báo "Xoá nhân viên thất bại !" | Xóa thất bại | Triều |  |
| Nhân Viên | 1) Click 1 dòng trên Table Nhân Viên  2) Click button "Mới"  3) Các TextField được làm mới | Nhấn vào button New và các TextField được làm mới | Tạo mới thành công | Triều |  |

* 1. **Kiểm thử form Cây Trồng**

| **Form kiểm thử** | **Các bước kiểm thử** | **Kết quả xảy ra** | **Kết quả đạt được** | **Người thực hiện** | **Ghi chú** |
| --- | --- | --- | --- | --- | --- |
| Cây Trồng | 1) Click button "New"  2) Nhập vào form những thông tin:  • Tên cây trồng: A100  3) Click button "Thêm"  4) Show message:"Thêm cây trồng thành công !" | - Thêm một cây trồng vào Database và chương trình thông báo: "Thêm cây trồng thành công !" | Thêm cây trồng thành công | Hải |  |
| Cây Trồng | 1) Click button "New"  2) Nhập vào form những thông tin:  • Tên cây trồng: A@#%&^  3) Click button "Thêm"  4) Show message:"Tên giàn phải từ 1 - 15 kí tự. Chữ hoa, thường hoặc số !" | - Thêm một giàn trồng vào Database và chương trình thông báo: "Thêm cây trồng thất bại !" | Thêm cây trồng thất bại | Hải |  |
| Cây Trồng | 1) Click button "New"  2) Nhập vào form những thông tin:  • Tên cây trồng: A1  3) Click button "Thêm"  4) Show message:"Tên giàn đã tồn tại !" | - Thêm một giàn trồng vào Database và chương trình thông báo: "Tên cây trồng đã tồn tại !" | Thêm cây trồng thất bại | Hải |  |
| Cây Trồng | 1) Click 1 dòng trên Table Giàn Trồng  2) Nhập vào form những thông tin:  • Tên giàn trồng: null  3) Click button "Thêm"  4) Show message:"Tên cây không được để trống !" | - Thêm một giàn trồng vào Database và chương trình thông báo: "Tên cây trồng không được để trống !" | Thêm cây trồng thất bại | Hải |  |
| Cây Trồng | 1) Click 1 dòng trên Table Giàn Trồng  2) Nhập vào form những thông tin:  • Tên cây trồng: null  3) Click button "Sửa"  4) Show message:"Tên cây không được để trống !" | - Cập nhật một giàn trồng vào Database và chương trình thông báo: "Tên cây trồng không được để trống !" | Cập nhật thất bại | Hải |  |
| Cây Trồng | 1) Click button "New"  2) Nhập vào form những thông tin:  • Tên cây trồng: A1 => A09  3) Click button "Sửa"  4) Show message:"Cập nhật giàn trồng thành công !" | - Cập nhật một giàn trồng vào Database và chương trình thông báo: "Cập nhật cây trồng thành công!" | Cập nhật thành công | Hải |  |
| Cây Trồng | 1) Click 1 dòng trên Table Giàn Trồng  2) Nhập vào from những thông tin:  • Tên cây trồng: A@#%&^  3) Click button "Sửa"  4) Show message:"Tên cây phải từ 1 - 15 kí tự. Chữ hoa, thường hoặc số !" | - Cập nhật một giàn trồng vào Database và chương trình thông báo: "Cập nhật cây trồng thất bại !" | Cập nhật thất bại | Hải |  |
| Cây Trồng | 1) Click 1 dòng trên Table Cây Trồng  2) Nhập vào from những thông tin:  • Tên cây trồng: A1  3) Click button "Sửa"  4) Show message:"Tên cây đã tồn tại !" | - Cập nhật một giàn trồng vào Database và chương trình thông báo: "Tên cây trồng đã tồn tại !" | Cập nhật thất bại | Hải |  |
| Cây Trồng | 1) Click 1 dòng trên Table Cây Trồng  2) Click button "Xóa"  3) Show message:"Bạn có muốn xóa cây trồng này không !" => Yes  4) Trở về form Cây Trồng | Xoá một giàn trồng trong Database và hiển thị thông báo "Xoá cây trồng thành công !" | Xóa thành công | Hải |  |
| Cây Trồng | 1) Click 1 dòng trên Table Cây Trồng  2) Click button "Xóa"  3) Show message:"Bạn có muốn xóa cây trồng này không !" => Cancel  4) Trở về form Cây Trồng | Xoá một giàn trồng trong Database và hiển thị thông báo "Xoá cây trồng thất bại !" | Xóa thất bại | Hải |  |
| Cây Trồng | 1) Click 1 dòng trên Table Cây Trồng  2) Click button "Mới"  3) Các TextField được làm mới | Nhấn vào button New và các TextField được làm mới | Tạo mới thành công | Hải |  |

* 1. **Kiểm thử form Giàn Trồng**

| **Form kiểm thử** | **Các bước kiểm thử** | **Kết quả xảy ra** | **Kết quả đạt được** | **Người thực hiện** | **Ghi chú** |
| --- | --- | --- | --- | --- | --- |
| Giàn Trồng | 1) Click button "New"  2) Nhập vào form những thông tin:  • Tên cây trồng: A100  3) Click button "Thêm"  4) Show message:"Thêm giàn trồng thành công !" | - Thêm một giàn trồng vào Database và chương trình thông báo: "Thêm cây trồng thành công !" | Thêm giàn trồng thành công | Tri |  |
| Giàn Trồng | 1) Click button "New"  2) Nhập vào form những thông tin:  • Tên cây trồng: A@#%&^  3) Click button "Thêm"  4) Show message:"Tên cây phải từ 1 - 15 kí tự. Chữ hoa, thường hoặc số !" | - Thêm một giàn trồng vào Database và chương trình thông báo: "Thêm giàn trồng thất bại !" | Thêm giàn trồng thất bại | Tri |  |
| Giàn Trồng | 1) Click button "New"  2) Nhập vào form những thông tin:  • Tên giàn trồng: A1  3) Click button "Thêm"  4) Show message:"Tên giàn đã tồn tại !" | - Thêm một giàn trồng vào Database và chương trình thông báo: "Tên giàn trồng đã tồn tại !" | Thêm giàn trồng thất bại | Tri |  |
| Giàn Trồng | 1) Click 1 dòng trên Table Giàn Trồng  2) Nhập vào form những thông tin:  • Tên giàn trồng: null  3) Click button "Thêm"  4) Show message:"Tên giàn không được để trống !" | - Thêm một giàn trồng vào Database và chương trình thông báo: "Tên giàn trồng không được để trống !" | Thêm giàn trồng thất bại | Tri |  |
| Giàn Trồng | 1) Click 1 dòng trên Table Giàn Trồng  2) Nhập vào form những thông tin:  • Tên giàn trồng: null  3) Click button "Sửa"  4) Show message:"Tên giàn không được để trống !" | - Cập nhật một giàn trồng vào Database và chương trình thông báo: "Tên giàn trồng không được để trống !" | Cập nhật thất bại | Tri |  |
| Giàn Trồng | 1) Click button "New"  2) Nhập vào form những thông tin:  • Tên giàn trồng: A1 => A09  3) Click button "Sửa"  4) Show message:"Cập nhật giàn trồng thành công !" | - Cập nhật một giàn trồng vào Database và chương trình thông báo: "Cập nhật giàn trồng thành công!" | Cập nhật thành công | Tri |  |
| Giàn Trồng | 1) Click 1 dòng trên Table Giàn Trồng  2) Nhập vào form những thông tin:  • Tên giàn trồng: A@#%&^  3) Click button "Sửa"  4) Show message:"Tên giàn phải từ 1 - 15 kí tự. Chữ hoa, thường hoặc số !" | - Cập nhật một giàn trồng vào Database và chương trình thông báo: "Cập nhật giàn trồng thất bại !" | Cập nhật thất bại | Tri |  |
| Giàn Trồng | 1) Click 1 dòng trên Table Giàn Trồng  2) Nhập vào form những thông tin:  • Tên giàn trồng: A1  3) Click button "Sửa"  4) Show message:"Tên giàn đã tồn tại !" | - Cập nhật một giàn trồng vào Database và chương trình thông báo: "Tên giàn trồng đã tồn tại !" | Cập nhật thất bại | Tri |  |
| Giàn Trồng | 1) Click 1 dòng trên Table Giàn Trồng  2) Click button "Xóa"  3) Show message:"Bạn có muốn xóa giàn trồng này không !" => Yes  4) Trở về form Giàn Trồng | Xoá một giàn trồng trong Database và hiển thị thông báo "Xoá giàn trồng thành công !" | Xóa thành công | Tri |  |
| Giàn Trồng | 1) Click 1 dòng trên Table Giàn Trồng  2) Click button "Xóa"  3) Show message:"Bạn có muốn xóa giàn trồng này không !" => Cancel  4) Trở về form Giàn Trồng | Xoá một giàn trồng trong Database và hiển thị thông báo "Xoá giàn trồng thất bại !" | Xóa thất bại | Tri |  |
| Giàn Trồng | 1) Click 1 dòng trên Table Giàn Trồng  2) Click button "Mới"  3) Các TextField được làm mới | Nhấn vào button New và các TextField được làm mới | Tạo mới thành công | Tri |  |

* 1. **Kiểm thử form Giao Việc**

| **Form kiểm thử** | **Các bước kiểm thử** | **Kết quả xảy ra** | **Kết quả đạt được** | **Người thực hiện** | **Ghi chú** |
| --- | --- | --- | --- | --- | --- |
| Giao Việc | 1)Nhập đầy đủ thông tin | - Hiển thị thông báo “Giao việc thành công” | Giao việc thành công | Tuấn |  |
| Giao Việc | 1)Nhập đầy đủ thông tin chừa ngày bắt đầu | - Hiển thị thông báo “Ngày bắt đầu không được để trống” | Giao việc thất bại | Tuấn |  |
| Giao Việc | 1)Nhập đầy đủ thông tin chừa ngày kết thúc | - Hiển thị thông báo “Ngày kết thúc không được để trống” | Giao việc thất bại | Tuấn |  |
| Giao Việc | 1) Click button "Giao Việc"  2) Nhập vào form những thông tin chưa chọn  3) Click button "Giao việc"  4) Show message:"Tên giàn phải từ 1 - 15 kí tự. Chữ hoa, thường hoặc số !" | - Thêm việc vào Database và chương trình thông báo: "Giao việc thất bại !" | Giao việc thất bại | Hải | xuống dưới làm đi t làm phụ mấy cái cho |
| Giao Việc | 1) Click button "New"  2) Nhập vào form những thông tin:  • Tên cây trồng: A1  3) Click button "Thêm"  4) Show message:"Tên giàn đã tồn tại !" | - Thêm một giàn trồng vào Database và chương trình thông báo: "Tên cây trồng đã tồn tại !" | Thêm cây trồng thất bại | Hải |  |
| Giao Việc | 1) Click 1 dòng trên Table Giàn Trồng  2) Nhập vào form những thông tin:  • Tên giàn trồng: null  3) Click button "Thêm"  4) Show message:"Tên cây không được để trống !" | - Thêm một giàn trồng vào Database và chương trình thông báo: "Tên cây trồng không được để trống !" | Thêm cây trồng thất bại | Hải |  |
| Giao Việc |  |  |  |  |  |
| Giao Việc |  |  |  |  |  |
| Giao Việc |  |  |  |  |  |
| Giao Việc |  |  |  |  |  |
| Giao Việc |  |  |  |  |  |

* 1. **Kiểm thử form Nhận Việc**

| **Form kiểm thử** | **Các bước kiểm thử** | **Kết quả xảy ra** | **Kết quả đạt được** | **Người thực hiện** | **Ghi chú** |
| --- | --- | --- | --- | --- | --- |
| Nhận việc | 1)Click button “Nhận việc“ | Nhấn vào button “Nhận việc” và công việc được fill trên table nhận việc được chuyển sang trạng thái “đang làm” đồng thời hiện thông báo “nhận việc thành công” | Nhận việc thành công | Nguyên Trung |  |
| Nhận việc | 1)Click button “Từ chối“ | Nhấn vào button “Từ chối” và công việc được fill trên table nhận việc được chuyển sang trạng thái “Từ chối” đồng thời hiện thông báo “Đã từ chối” | Từ chối nhận việc thành công | Nguyên Trung |  |

* 1. **Kiểm thử form Hoàn Thành Công Việc**

| **Form kiểm thử** | **Các bước kiểm thử** | **Kết quả xãy ra** | **Kết quả đạt được** | **Người thực hiện** | **Ghi chú** |
| --- | --- | --- | --- | --- | --- |
| Hoàn thành | 1)Click button “Hoàn thành“ | 1)Chọn công việc”Trồng cây” trên table  2)Nhấn vào button “Hoàn thành” và trạng thái công việc đó được đổi thành trạng thái “hoàn thành” | Xác nhận hoàn thành thành công | Nguyên Trung |  |
| Hoàn thành | 1)Click button “Hoàn thành“ | 1)Chọn công việc”Chăm sóc” trên table  2)Nhấn vào button “Hoàn thành” và trạng thái công việc đó được đổi thành trạng thái “hoàn thành” | Xác nhận hoàn thành thành công | Nguyên Trung |  |
| Hoàn thành | 1)Click button “Hoàn thành“ | 1)Chọn công việc”Thu hoạch” trên table  2)Nhấn vào button “Hoàn thành”  3)Nhập vào textfield số lượng cây đã thu hoạch được  3) bấm button”OK” trạng thái của công việc chuyển thành đang bán và được fill qua table kho hàng | Xác nhận hoàn thành thành công | Nguyên Trung |  |
| Hoàn thành | 1)Click button “Hoàn thành“ | 1)Chọn công việc”Thu hoạch” trên table  2)Nhấn vào button “Hoàn thành”  3)Nhập vào textfield số lượng cây đã thu hoạch được  3) bấm button”Cancel” trạng thái của công việc chuyển thành đang bán và được fill qua table kho hàng | Xác nhận hoàn thành không thành công | Nguyên Trung |  |
| Hoàn thành | 1)Nhập chữ vào textfield số lượng | 1)hiện thông báo “Không được nhập số lượng bằng chữ” | Xác nhận hoàn thành không thành công |  |  |
| Hoàn thành | 1)Nhập số vào textfield số lượng | 1)Không hiện bắt lỗi  2)Nhấn button “OK”  Trạng thái công việc chuyển thành đang bán | Xác nhận hoàn thành thành công |  |  |

* 1. **Kiểm thử form Nhật Ký**

| **Form kiểm thử** | **Các bước kiểm thử** | **Kết quả xảy ra** | **Kết quả đạt được** | **Người thực hiện** | **Ghi chú** |
| --- | --- | --- | --- | --- | --- |
| Nhật Ký | 1) Click button “Lọc”  2) Chọn ngày bắt đầu 2021/12/04  3) Chọn ngày bắt đầu 2021/12/31  4) Show danh sách nhật ký trong khoảng thời gian đã chọn | - Show danh sách nhật ký trong khoảng thời gian đã chọn | Show danh sách nhật ký trong khoảng thời gian đã chọn | Dũng |  |
| Nhật Ký | 1) Click button “Lọc”  2) Chọn ngày bắt đầu 2021/12/04  3) Chọn ngày bắt đầu 2021/12/31  4) Form không thay đổi | - Danh sách nhật ký không thay đổi trong khoảng thời gian đã chọn | - Danh sách nhật ký không thay đổi trong khoảng thời gian đã chọn | Dũng |  |
| Nhật Ký | 1) Click button "Print"  2) Không xuất hiện form địa chỉ lưu file | 1) Click button "Print"  2) Không xuất hiện form địa chỉ lưu file | Print thất bại | Dũng |  |
| Nhật Ký | 1) Click button "Print"  2) Hiển thị form địa chỉ lưu file  3) Chọn nơi lưu file rồi bấm “Save” | Lưu dược file pdf của nút “Print” | Print thành công | Dũng |  |
| Nhật Ký | 1) Click vào label Tìm kiếm  2) Nhập từ khóa liên quan đến thứ cần tìm  3) Show tìm kiếm thất bại | Tìm kiếm không hoạt động | Tìm kiếm thất bại | Dũng |  |
| Nhật Ký | 1) Click vào label Tìm kiếm  2) Nhập từ khóa liên quan đến thứ cần tìm  3) Hiển thị danh sách cần tìm | Tìm kiếm được những thứ liên quan đén từ khóa | Tìm kiếm thành công | Dũng |  |
| Nhật Ký | Không hiển thị dữ liệu trong form | Đổ dữ kiệu từ database thất bại | Load dữ liệu không thành công | Dũng |  |

1. **Build chương trình**