**Các package và chức năng**

**1. bean: Để định nghĩa các đối tượng đơn giản để lưu dữ liệu thường có các thuộc tính getter – setter,**

**- DanhMucBean.java:**

**2. controller: Điểu khiển hành động cho các giao diện.**

**- ChuyenManHinhController.java:**

+ Dùng để chuyển giữa các chức năng trong menu.

**- QuanLySinhVienController.java:**

+ Dùng để quản lí, điều khiển chức năng: Quản lí sinh viên.

**- SinhVienController.java:**

+ Dùng để quản lí, điều khiển nơi: Nhập thông tin sinh viên.

**- TaiKhoanController.java:**

+ Dùng để quản lí, điều khiển nơi: Đăng nhập.

**3. dao: Data Access Objec, chỉ để quản lí các kết nối tới DataBase.**

**Sau đó đưa tới Service để thực hiện các logic riêng.**

**- DBConnect.java** + Dùng để kết nối project với Database

**- SinhVienDAO.java**

+ Là một interface

+ Chứa các phương thức chung chung, cơ bản :

**Trả về danh sách sinh viên vào DataBase**

**Cập nhật thông tin sinh viên vào DataBase**

**- SinhVienDAOImpl.java** + Triển khai và xử lí các phương thức cụ thể, tỉ mỉ hơn so   
 với SinhVienDao.

+ Truy cập vào phần sinh\_vien ở trong DataBase

**- TaiKhoanDAO.java**

+ Tương tự như trên, nó có phương thức xác nhận việc đăng nhập của người dùng, truy vấn tới DataBase.

**- TaiKhoanDAOImpl.java**

+ Truy cập vào phần tai\_khoan trong DataBase để tìm kiếm tài khoản – mật khẩu được lưu từ trước.

**4. images: Dùng để lưu chữ các ảnh cho giao diện**

**5. main**

**- Main.java** + Chạy chương trình chính

**6. model: Dùng để định nghĩa, khai báo cho các lớp đại diện**

**- KhoaHoc.java**

+ Thiết lập các giá trị liên quan đến khóa học  
 (mã sinh viên, tên khóa học, ngày bắt đầu, kết thúc,...)

+ In các giá trị đó ra.

**- SinhVien.java**

+ Thiết lập các giá trị liên quan đến sinh viên  
 (mã sv, tên, sđt, ngày sinh, ...)

**- LopHoc.java**

+ Chứa hết các thông tin của sinh viên trong lớp

< lấy từ class KhoaHoc và class SinhVien>

+ Có thể coi class LopHoc là cha của SinhVien và KhoaHoc.

**7. service: thực hiện các chức năng, logic riêng để xử lí   
 Sau đó sẽ đưa tới Controller để xử lí in ra ngoài giao diện.**

**- SinhVienService.java:**

+ Là một interface

+ Dùng để hiển thị sinh viên ra giao diện.

+ Cập nhật thông tin sinh viên hiển thị ra giao diện

**- SinhVienServiceImpl.java**

**- TaiKhoanService.java**

+ Là một interface

+ Chứa các phương thức chung

**- TaiKhoanServiceImpl.java**

+ Sẽ xử lí kĩ hơn so với TaiKhoanService

VD: Sau khi người dùng nhập tk-mk, thì phần DAO sẽ truy cập để truy vấn tới DataBase để lấy thông tin rồi gửi tới Service để xác nhận, xử lí luồng “ Có thằng vừa nhập tk-mk”  
 Sau khi đã nhập xong tk-mk rồi thì nó sẽ đưa nốt tới **Controller** để xử lí, in cái gì ra ngoài giao diện phụ thuộc vào yêu cầu trong phần **Controller** xử lí.

**8. utility**

**- ClassTableModel.java**

+ Dùng để hiển thị danh sách sinh viên, được cấu hình theo một dạng bảng.

**9. view: Thiết kế các giao diện**

**- MainJFrame.java**

+ Hiện thị giao diện chính của ứng dụng

**- TrangChuJPanel.java**

+ Hiện thông tin của chức năng: Màn hình chính

**- SinhVienJFrame.java**

+ Nhập thông tin sinh viên.

**- SinhVienJPanel.java**

+ Hiện ra thông tin sinh viên của lớp.

**- KhoaHocJPanel.java**