|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Chức năng | Class | Các phương thức sử dụng | Giải thích |
| 1 | Đăng nhập | UserAccount.java | Getter,setter | Quy định cấu trúc của đối tượng User  Là đối tượng để gán giá trị từ cơ sở dữ liệu, sau đó so sánh với dữ liệu nhập từ View |
| Login.jsp |  | Hiển thị giao diện đăng nhập |
| LoginController.java  LoginFBController.java  DBUtils.java  ValidateRegister.java | Doget()  Dopost()  Check\_login()  isValid()  Check\_login\_perrmission() | Lấy dữ liệu từ lớp view, sử dụng các phương thức hỗ trợ để kiểm tra đầu vào của dữ liệu, sau đó truyền vào UserAccount, so sánh với dữ liệu trong database. Nếu chính xác, trả lại thông tin đầy đủ cho UserAccount và gán vào session để duy trì đăng nhập và thực hiện các chức năng khác.Kiểm tra quyền và cho đến trang phù hợp |
| 2 | Đăng ký | UserAccount.java | Getter, Setter | Lưu trữ dữ liệu từ Controller và thực hiện các câu lệnh để kiểm tra dữ liệu |
| Register.jsp  Verify.jsp |  | Hiển thị giao diện để đăng ký cho người dùng nhập thông tin. |
| RegisterController.java  ValidateRegister.java  SendingMail.java  DBUtils.java | Doget()  Dopost()  Sendverifymail()  isValid() | Lấy dữ liệu từ View qua request và tiến hành kiểm tra. Nếu thành công, truyền dữ liệu vào model và lưu vào database. Đồng thời sử dụng phương thức để gửi email xác nhận. Khi người dùng bấm vào xác nhận mail, nếu thỏa điều kiện sẽ chuyển đến trang Verify.jsp |
| 3 | Tạo đơn hàng mới để thực hiện chức năng mua | Product.java  Order.java  OrderDetiail.java  UserAccount.java | Getter, Setter | Quy định cấu trúc của đơn hàng để lưu trữ thông tin cần thiết, bên cạnh đó trong đơn hang có nhiều đối tượng khác cũng cần được ghi lại |
| Cart.jsp  Order.jsp |  | Sau khi đơn hàng được tạo từ giỏ hàng cart(session), nếu tạo thành công sẽ chuyển đến trang order.jsp để kiểm tra |
| CartController.java  OrderController.java  DBUtils.java | Doget()  Dopost() | Lấy dữ liệu từ các session, kiểm tra lại dữ liệu và thực hiện các chức năng tương ứng, sau đó lần lượt gán vào các model ở trên và ghi xuống database. Sau khi thực hiện xong, sẽ nhận được tín hiệu và có hàm gọi Order.jsp để hiển thị |
| 4 | Dropdown chọn tỉnh, huyện, xã | Province.java  District.java  Town.java | Getter, Setter | Quy định cấu trúc của tỉnh, huyện, xã và lưu trữ dữ liệu từ database |
| AccountSetting.jsp  ThemSanpham.jsp | ajax | Hiển thị giao diện để chọn. Khi chọn 1 tỉnh thì thì ajax gọi controller để đưa lên dữ liệu tương ứng và dropdown còn lại sẽ hiển thị các huyện xã tương ứng |
| ProvinceController.java | Doget()  Dopost() | Xử lý tín hiệu từ view gọi về và đưa dữ liệu từ các đối tượng model trên lên view |
| 5 | Thêm xóa sửa sản phẩm | Product.java  UserAccount.java | Getter, Setter | Quy định cấu hình sản phẩm để lưu trữ thông tin sản phẩm trước khi ghi xuống database |
| ThemSanPham.jsp  XoaSanPham.jsp |  | Hiển thị sản phẩm hiện có, cũng như kết quả sau khi thao tác. Sau đó nhận tín hiệu để xử lý hành động tiếp theo |
| ThemSanPhamController.java  XoaSanPhamController.java  DBUtils.java | Doget()  Dopost() | Nhận dữ liệu từ view xong gán vào product và chạy các hàm kiểm tra xong. Tiến hành thêm. Nếu thêm thành công sẽ gửi tín hiệu refresh lại trang để xem sự thay đổi |