|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Chức năng | Class | Các phương thức sử dụng | Giải thích |
| 1 | Đăng nhập bằng facebook | Contants.java |  | Cấu hình đối tượng ứng dụng facebook |
| Login.jsp  Index.jsp |  | Hiển thị giao diện đăng nhập và trả về trang chủ khi đăng nhập thành công |
| LoginFBController.java  DBUtils.java  Restfb.java  getToken()  getInfo() | Doget()  Dopost() | Sau bấm khi login ở trang đăng nhập, tiến hành dẫn đến 1 đường dẫn của facebook. Sau đó trả lại 1 đường dẫn xác nhận và lấy được token và đã đăng nhập thành công. Dùng 2 hàm trên và token để cấu hình ứng dụng facebook và lưu dữ liệu xuống database. |
| 2 | Hash mật khẩu  Bằng md5 | UserAccount.java | Getter, Setter | Lưu dữ liệu vào model để chuẩn bị lưu xuống database |
| Register.jsp |  | Hiển thị giao diện để đăng ký cho người dùng nhập thông tin. |
| RegisterController.java | Doget()  Dopost() | Ngoài việc lưu trữ thông tin người dung mới, hash mật khẩu bằng thư viện Digest của java. Sau khi lấy mật khẩu từ view, mã hóa mật khẩu rồi mới đưa xuống database |
| 3 | Filter | JDBCFilter.java  CookieFilter.java  EncodingFilter.java  Security.java | Sử dung lớp Filter | Cả 4 filter đều được dung để lọc dữ liệu theo điều kiện mong muốn và đều được đặt annotoion=\*/, nghĩa là bất kỳ thực hiện chuyển qua trang mới hay hành động mới thì sẽ kích hoạt filter này trước. Sau khi hoành thành mới được thực hiện các công việc phía sau.  JDBC có chức năng kiểm tra xem hành động muốn thực hiện có phải Servlet hay JSP hay không để mở cổng kết nối database để tránh bị lạm dung và cải thiện tốc độ.  CookieFilter sẽ kiểm tra nếu xem nếu như tắt trình duyệt và không có session hiện có thì sẽ load thông tin từ cookie lên và tự động đăng nhập  SecurityFilter sẽ kiểm tra hành động nào bắt buộc phải đăng nhập, nếu thỏa mãn sẽ kiểm tra tiếp có phân quyền sử dụng hay không và chuyển đến trang phù hợp hoặc dừng trùy cập  EncodingFilter có chức năng mã hóa session để tăng tính bảo mật |
|  | | |