MINDTECK ACADEMY

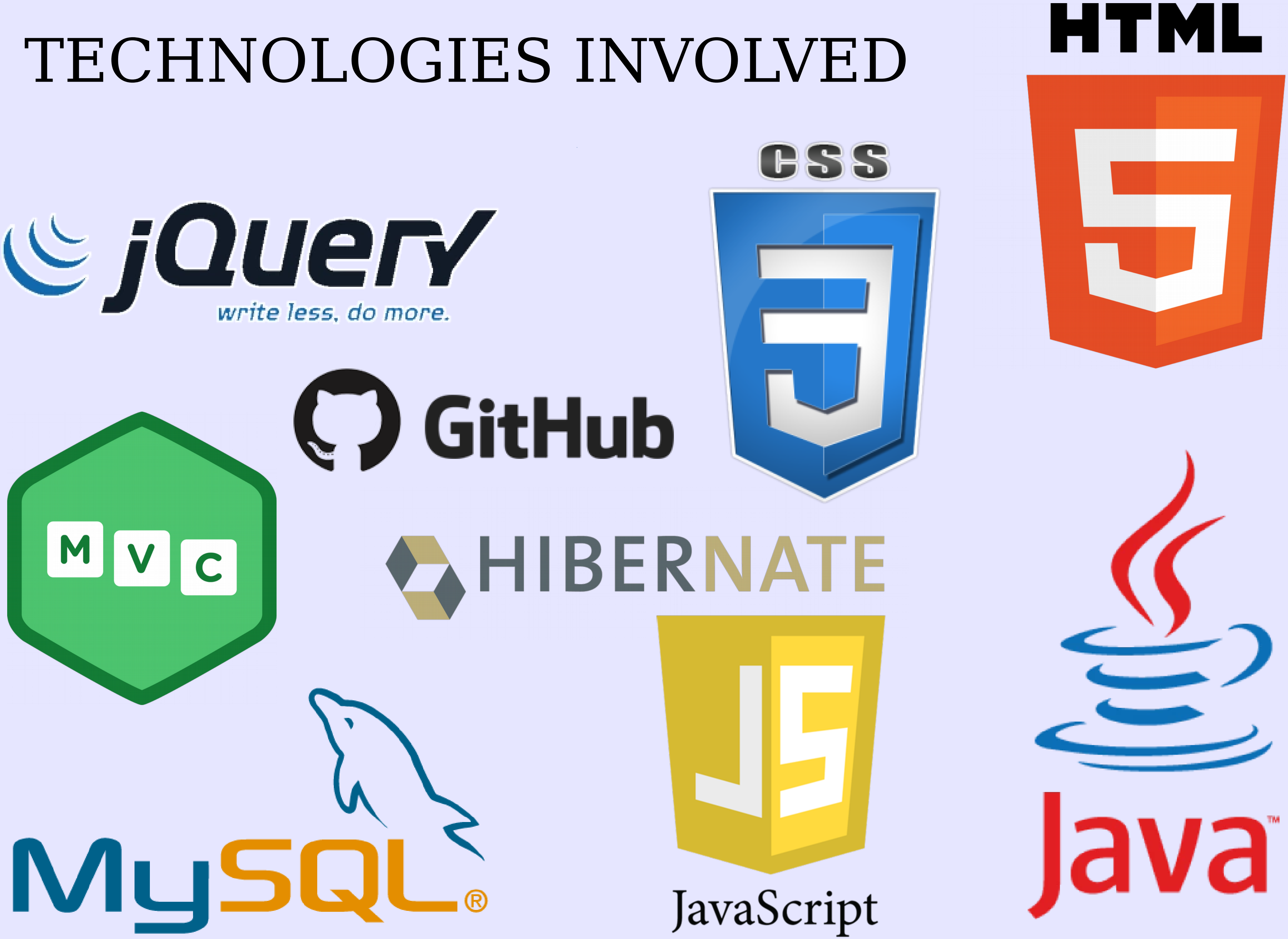
FOSTERING TECHNOLOGY CAREERS

Final Project
E commerce Design

Dover Hellfeldt

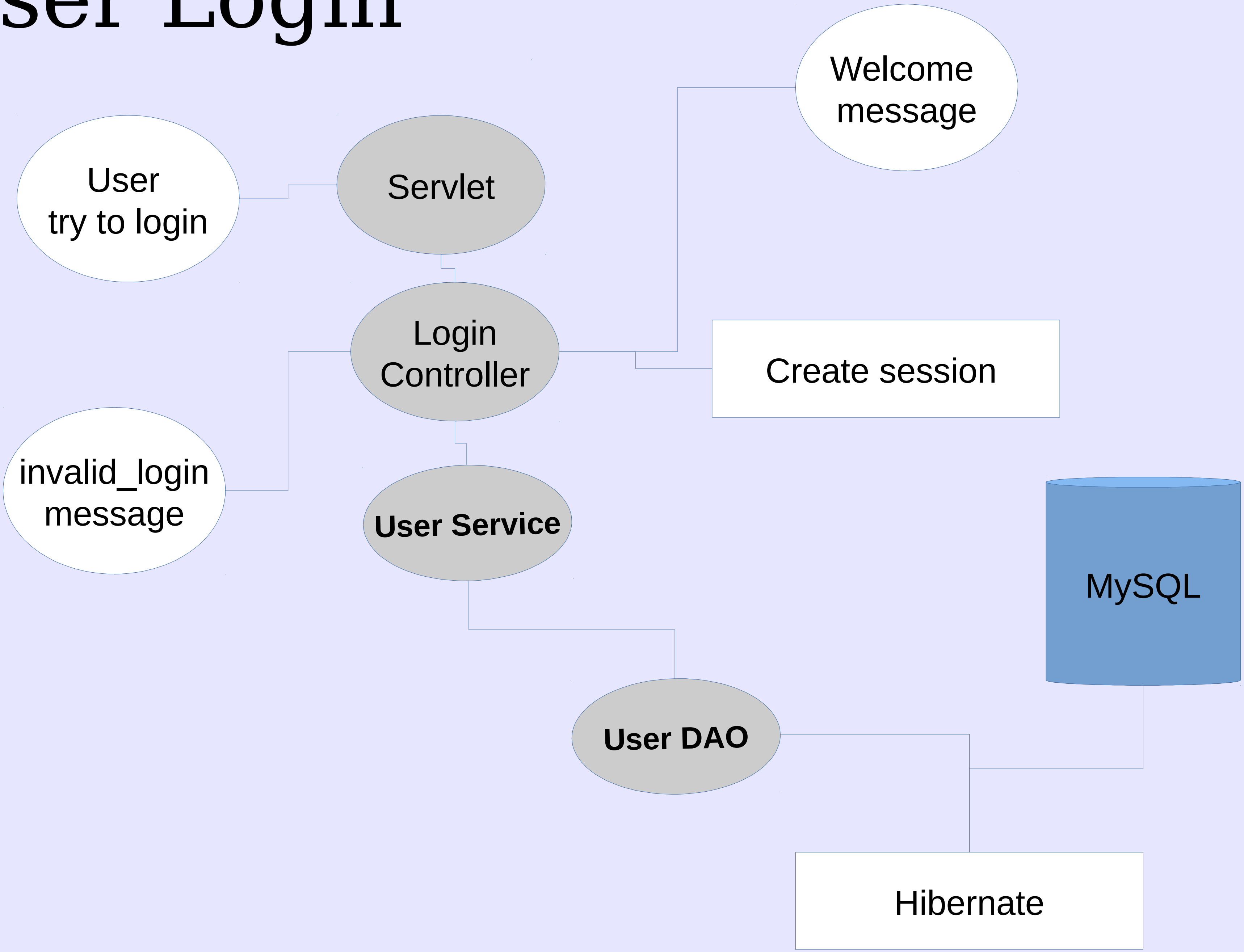
Pittsburgh June 16th, 2017

TECHNOLOGIES INVOLVED

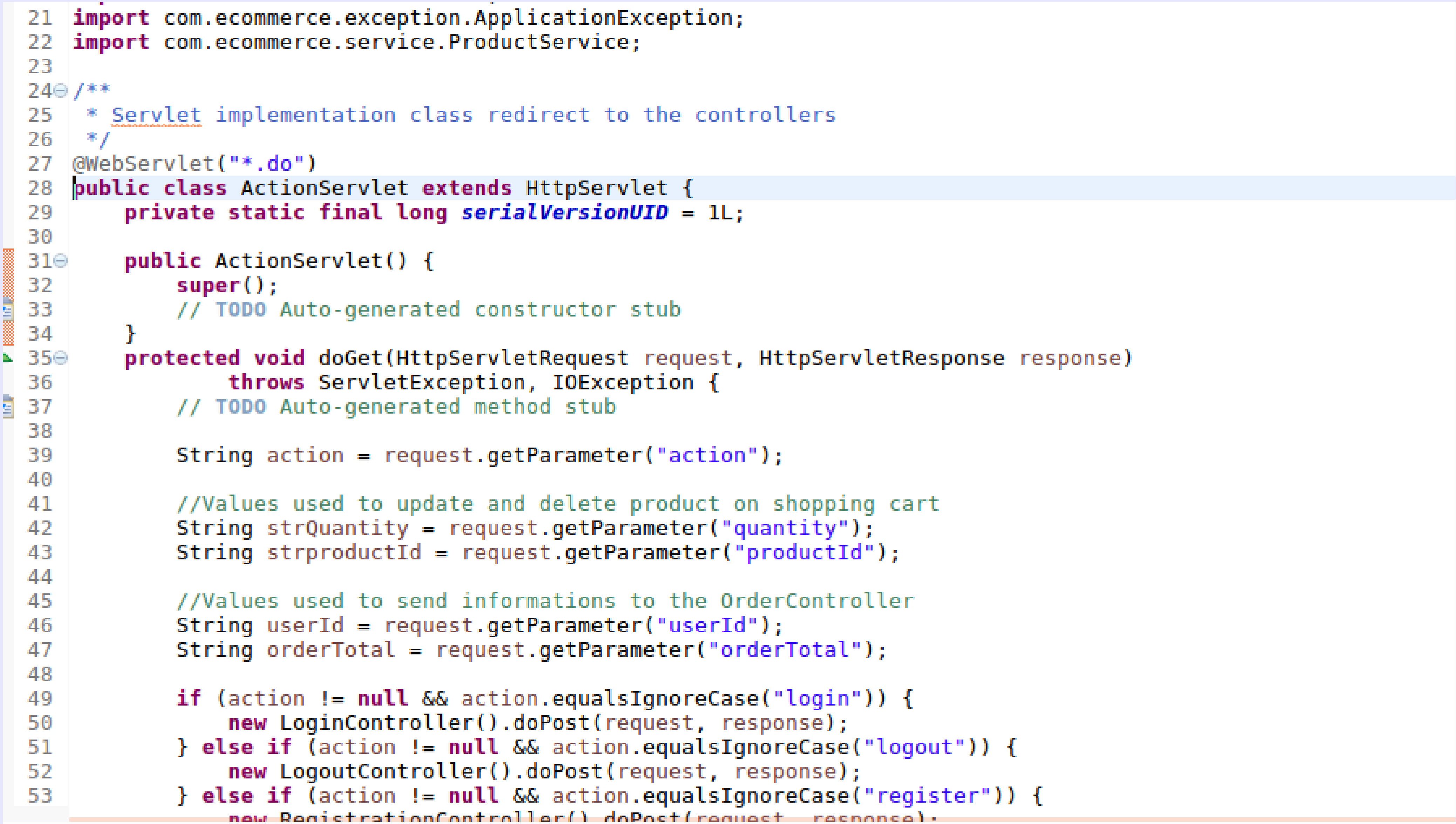


USE CASES

User Login

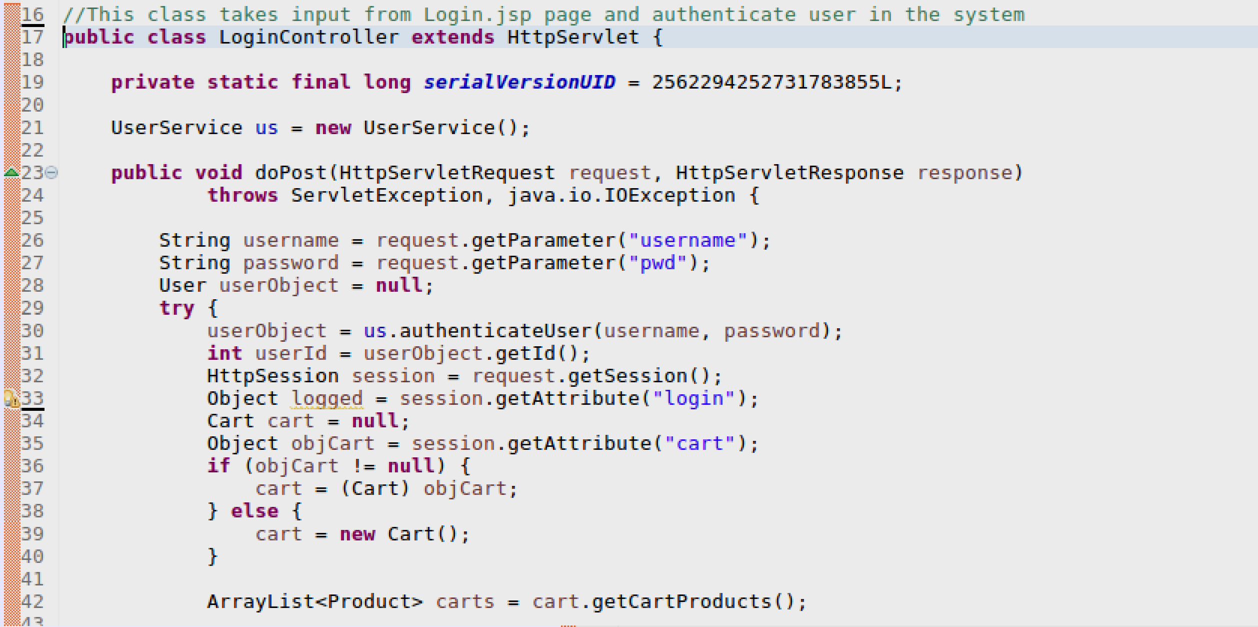

Welcome message
User try to login
Servlet
Login Controller
Create session
invalid_login message
User Service
MySQL
User DAO
Hibernate

Servlet

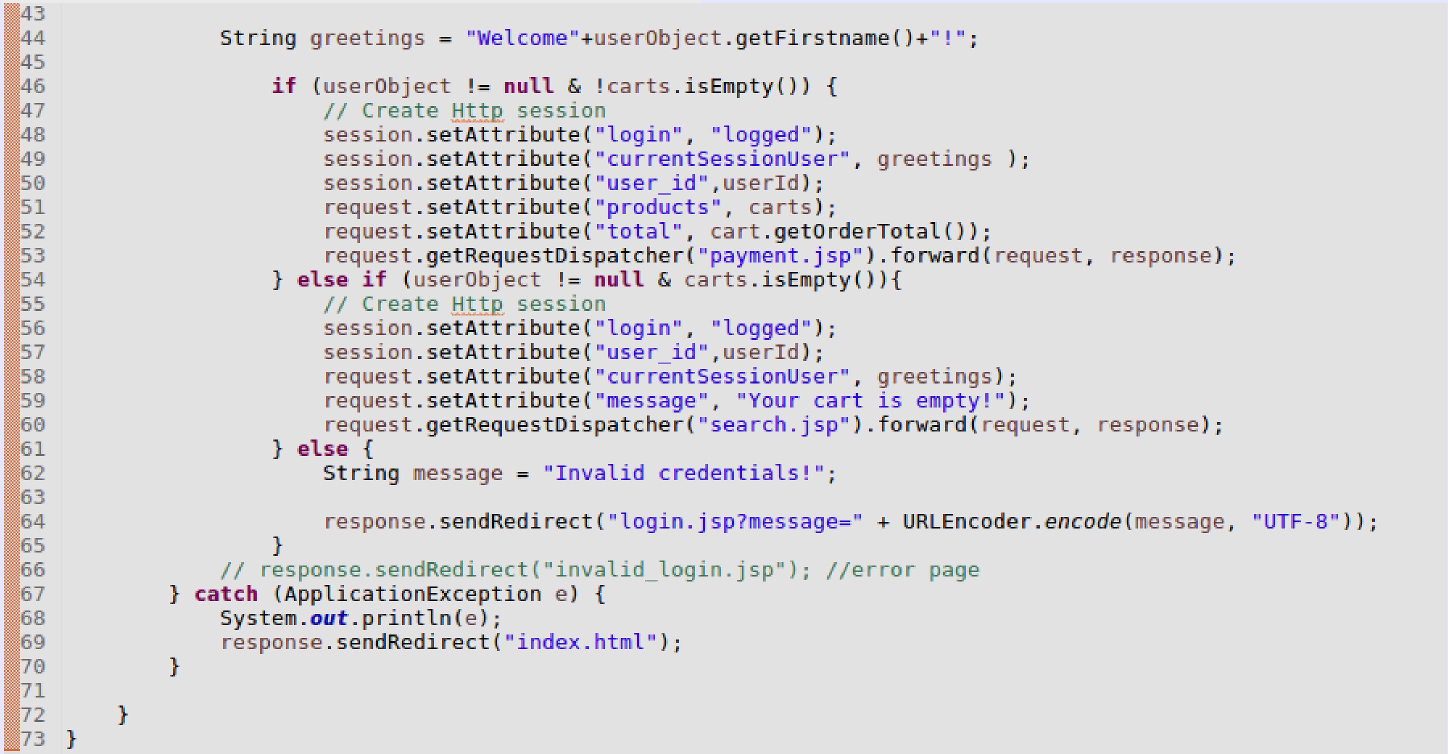


```java
import com.ecommerce.exception.ApplicationException;
import com.ecommerce.service.ProductService;

/**
 * Servlet implementation class redirect to the controllers
 */
@WebServlet("*.do")
public class ActionServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    public ActionServlet() {
        super();
        // TODO Auto-generated constructor stub
    }
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
            throws ServletException, IOException {
        // TODO Auto-generated method stub

        String action = request.getParameter("action");

        //Values used to update and delete product on shopping cart
        String strQuantity = request.getParameter("quantity");
        String strproductId = request.getParameter("productId");

        //Values used to send informations to the OrderController
        String userId = request.getParameter("userId");
        String orderTotal = request.getParameter("orderTotal");

        if (action != null && action.equalsIgnoreCase("login")) {
            new LoginController().doPost(request, response);
        } else if (action != null && action.equalsIgnoreCase("logout")) {
            new LogoutController().doPost(request, response);
        } else if (action != null && action.equalsIgnoreCase("register")) {
            new RegistrationController().doPost(request, response);
```

Login Controller

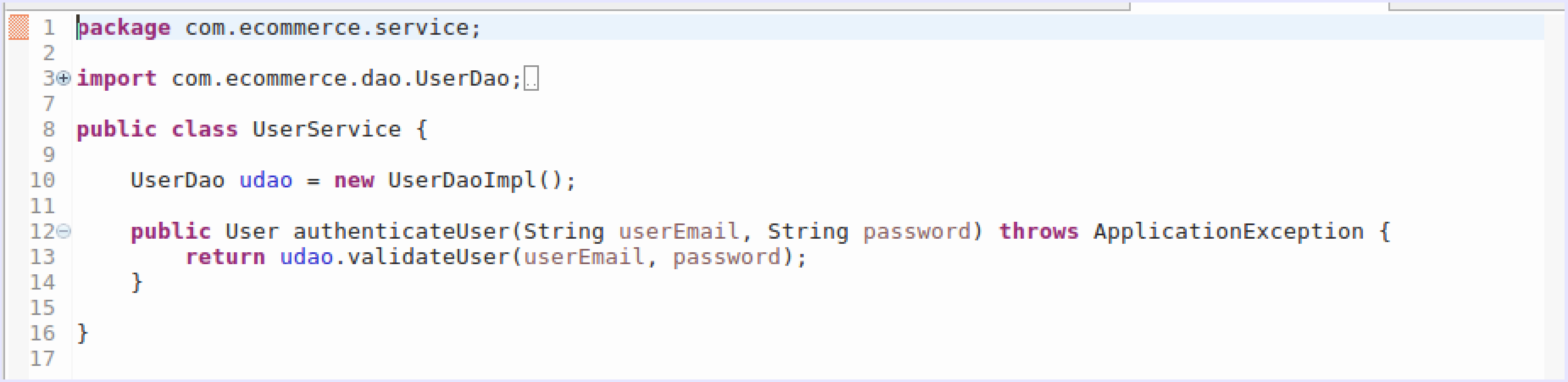


```java
//This class takes input from Login.jsp page and authenticate user in the system
public class LoginController extends HttpServlet {

    private static final long serialVersionUID = 2562294252731783855L;

    UserService us = new UserService();

    public void doPost(HttpServletRequest request, HttpServletResponse response)
            throws ServletException, java.io.IOException {

        String username = request.getParameter("username");
        String password = request.getParameter("pwd");
        User userObject = null;
        try {
            userObject = us.authenticateUser(username, password);
            int userId = userObject.getId();
            HttpSession session = request.getSession();
            Object logged = session.getAttribute("login");
            Cart cart = null;
            Object objCart = session.getAttribute("cart");
            if (objCart != null) {
                cart = (Cart) objCart;
            } else {
                cart = new Cart();
            }

            ArrayList<Product> carts = cart.getCartProducts();
```



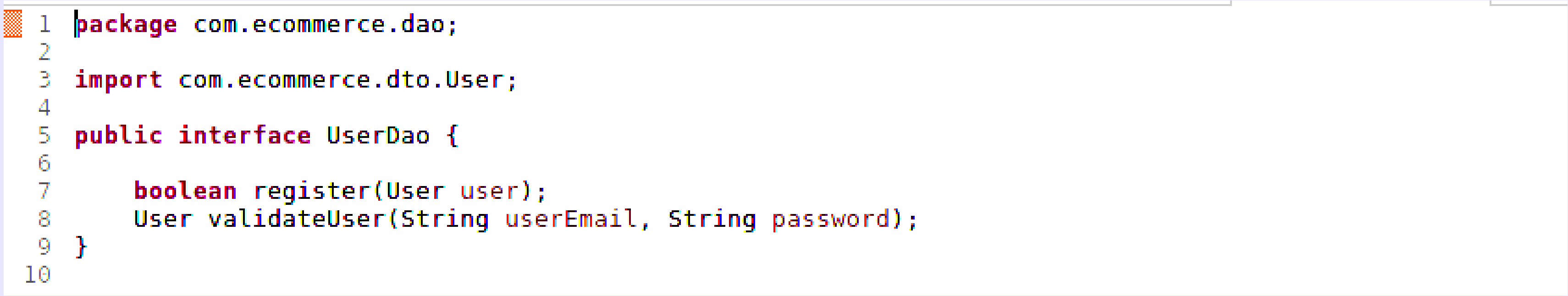
```java

            String greetings = "Welcome"+userObject.getFirstname()+"!";

                if (userObject != null & !carts.isEmpty()) {
                    // Create Http session
                    session.setAttribute("login", "logged");
                    session.setAttribute("currentSessionUser", greetings );
                    session.setAttribute("user_id",userId);
                    request.setAttribute("products", carts);
                    request.setAttribute("total", cart.getOrderTotal());
                    request.getRequestDispatcher("payment.jsp").forward(request, response);
                } else if (userObject != null & carts.isEmpty()){
                    // Create Http session
                    session.setAttribute("login", "logged");
                    session.setAttribute("user_id",userId);
                    request.setAttribute("currentSessionUser", greetings);
                    request.setAttribute("message", "Your cart is empty!");
                    request.getRequestDispatcher("search.jsp").forward(request, response);
                } else {
                    String message = "Invalid credentials!";

                    response.sendRedirect("login.jsp?message=" + URLEncoder.encode(message, "UTF-8"));
                }
            // response.sendRedirect("invalid_login.jsp"); //error page
        } catch (ApplicationException e) {
            System.out.println(e);
            response.sendRedirect("index.html");
        }

    }
}
```

User Service

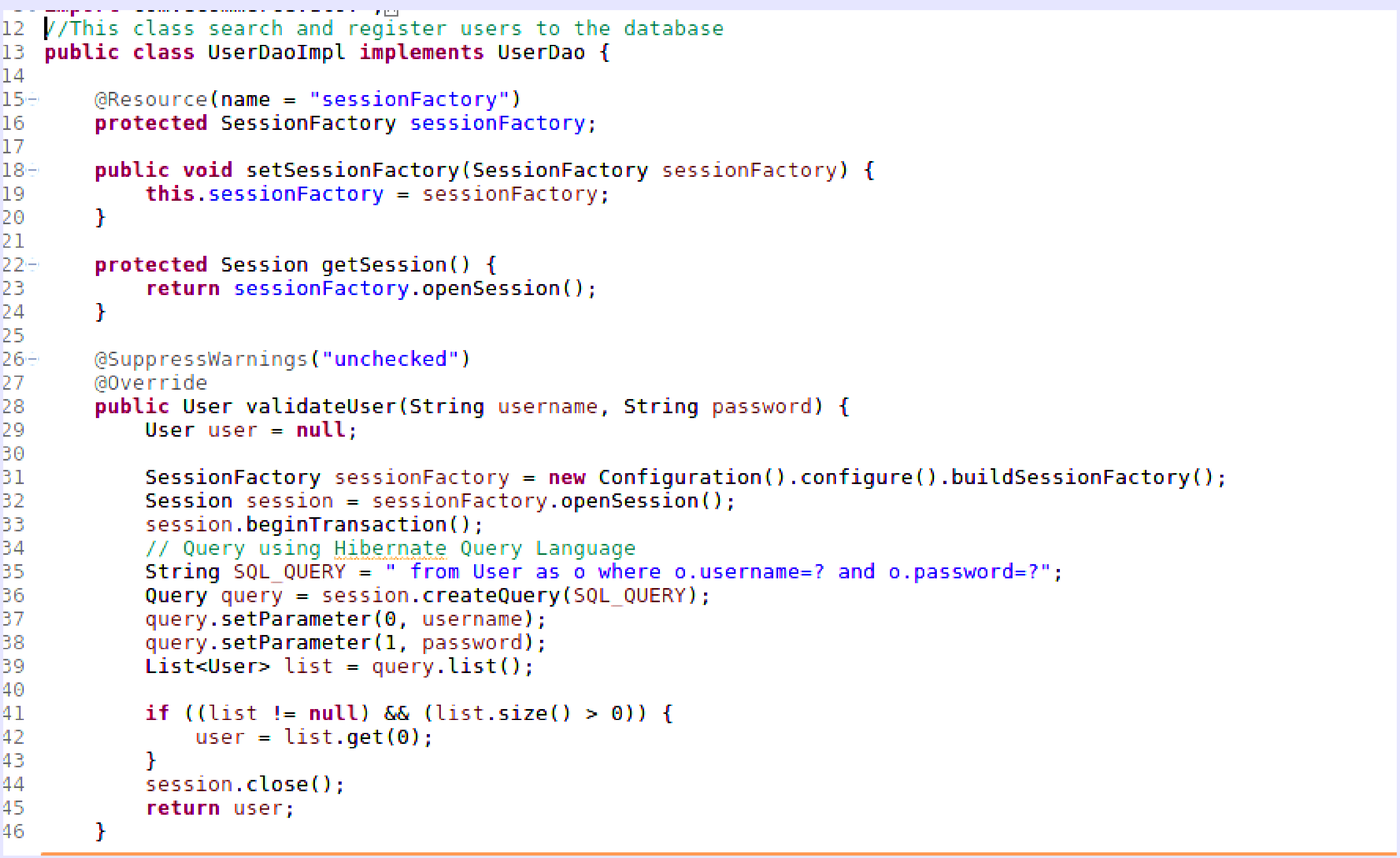


```java
package com.ecommerce.service;

import com.ecommerce.dao.UserDao;

public class UserService {

    UserDao udao = new UserDaoImpl();

    public User authenticateUser(String userEmail, String password) throws ApplicationException {
        return udao.validateUser(userEmail, password);
    }

}

```

User DAO and HIBERNATE

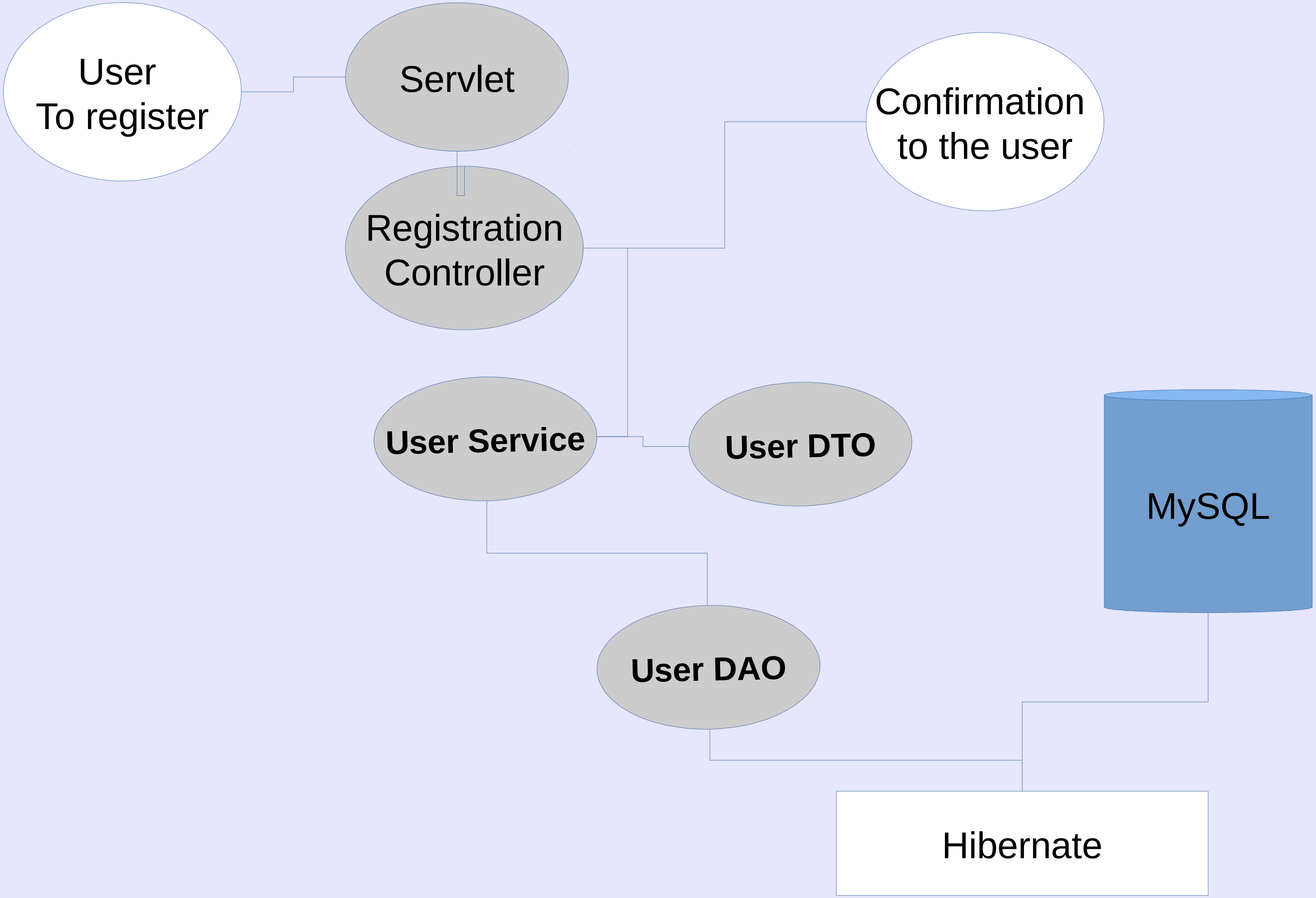


```java
package com.ecommerce.dao;

import com.ecommerce.dto.User;

public interface UserDao {

    boolean register(User user);
    User validateUser(String userEmail, String password);
}

```

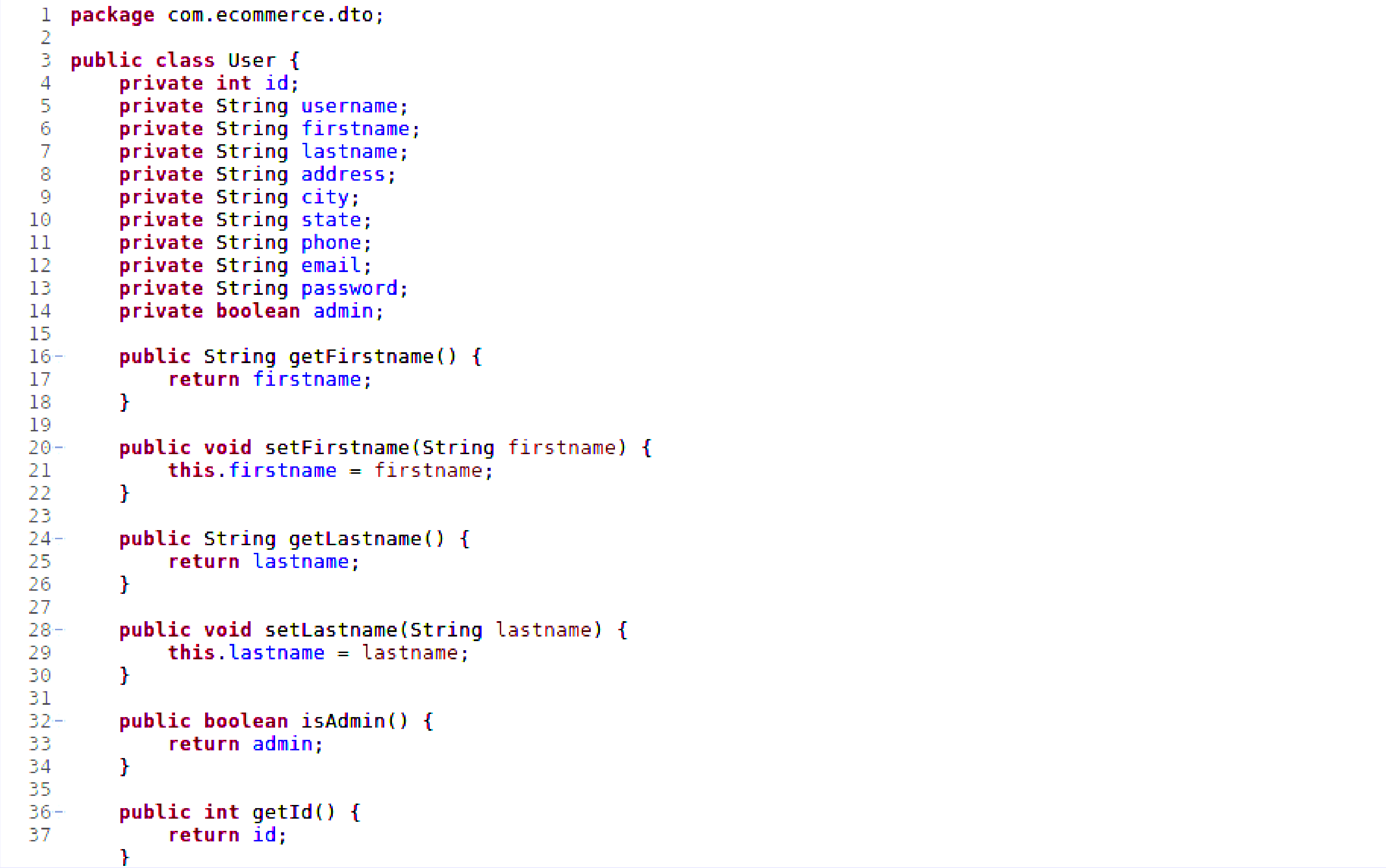


```java
//This class search and register users to the database
public class UserDaoImpl implements UserDao {

    @Resource(name = "sessionFactory")
    protected SessionFactory sessionFactory;

    public void setSessionFactory(SessionFactory sessionFactory) {
        this.sessionFactory = sessionFactory;
    }

    protected Session getSession() {
        return sessionFactory.openSession();
    }

    @SuppressWarnings("unchecked")
    @Override
    public User validateUser(String username, String password) {
        User user = null;

        SessionFactory sessionFactory = new Configuration().configure().buildSessionFactory();
        Session session = sessionFactory.openSession();
        session.beginTransaction();
        // Query using Hibernate Query Language
        String SQL_QUERY = " from User as o where o.username=? and o.password=?";
        Query query = session.createQuery(SQL_QUERY);
        query.setParameter(0, username);
        query.setParameter(1, password);
        List<User> list = query.list();

        if ((list != null) && (list.size() > 0)) {
            user = list.get(0);
        }
        session.close();
        return user;
    }
```

User Registration

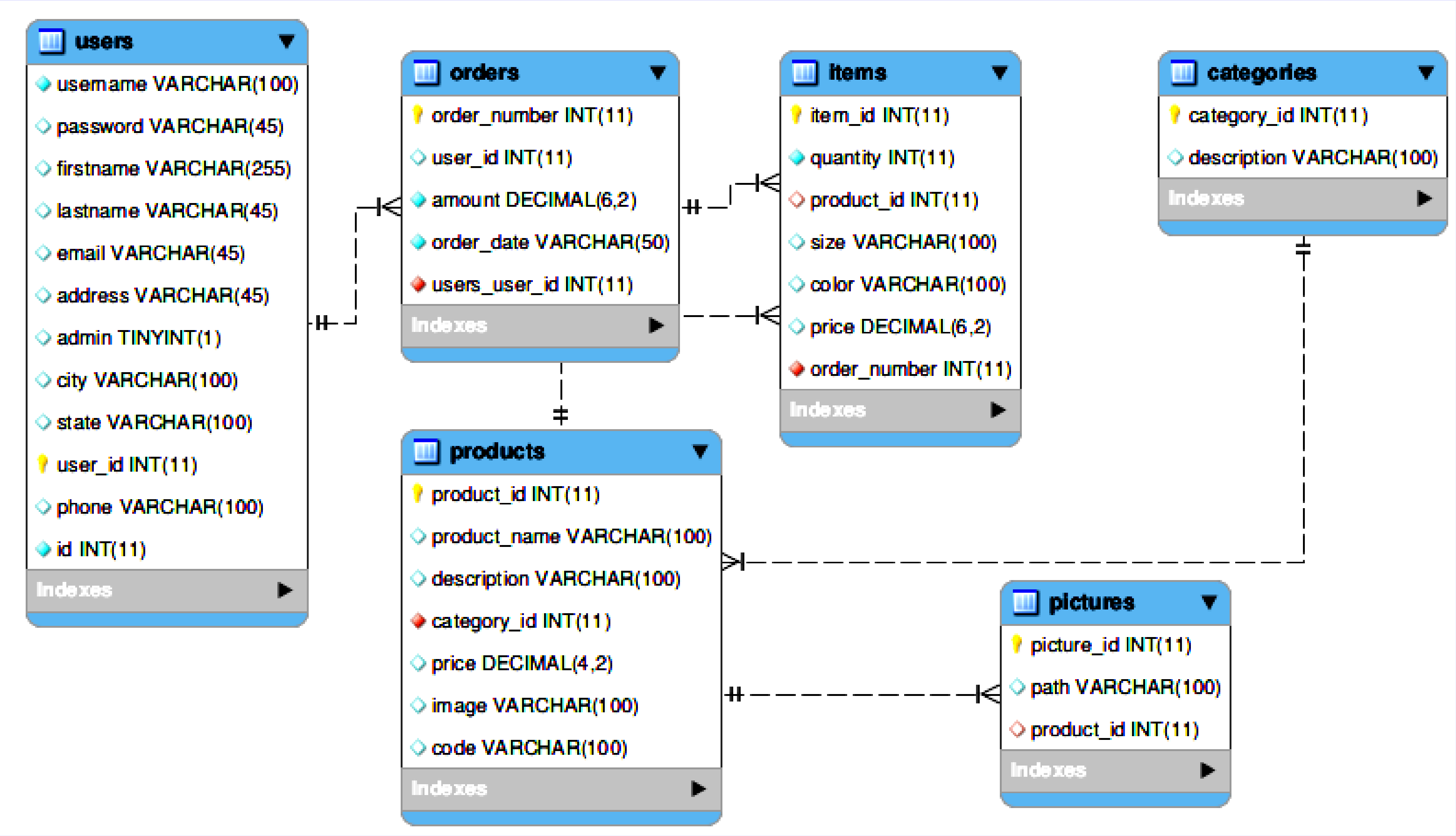


User DTO



```java
package com.ecommerce.dto;

public class User {
    private int id;
    private String username;
    private String firstname;
    private String lastname;
    private String address;
    private String city;
    private String state;
    private String phone;
    private String email;
    private String password;
    private boolean admin;

    public String getFirstname() {
        return firstname;
    }

    public void setFirstname(String firstname) {
        this.firstname = firstname;
    }

    public String getLastname() {
        return lastname;
    }

    public void setLastname(String lastname) {
        this.lastname = lastname;
    }

    public boolean isAdmin() {
        return admin;
    }

    public int getId() {
        return id;
    }
```

MSQL



users

username VARCHAR(100)
password VARCHAR(45)
firstname VARCHAR(255)
lastname VARCHAR(45)
email VARCHAR(45)
address VARCHAR(45)
admin TINYINT(1)
city VARCHAR(100)
state VARCHAR(100)
user_id INT(11)
phone VARCHAR(100)
id INT(11)
Indexes

orders

order_number INT(11)
user_id INT(11)
amount DECIMAL(6,2)
order_date VARCHAR(50)
users_user_id INT(11)
Indexes

items

item_id INT(11)
quantity INT(11)
product_id INT(11)
size VARCHAR(100)
color VARCHAR(100)
price DECIMAL(6,2)
order_number INT(11)
Indexes

categories

category_id INT(11)
description VARCHAR(100)
Indexes

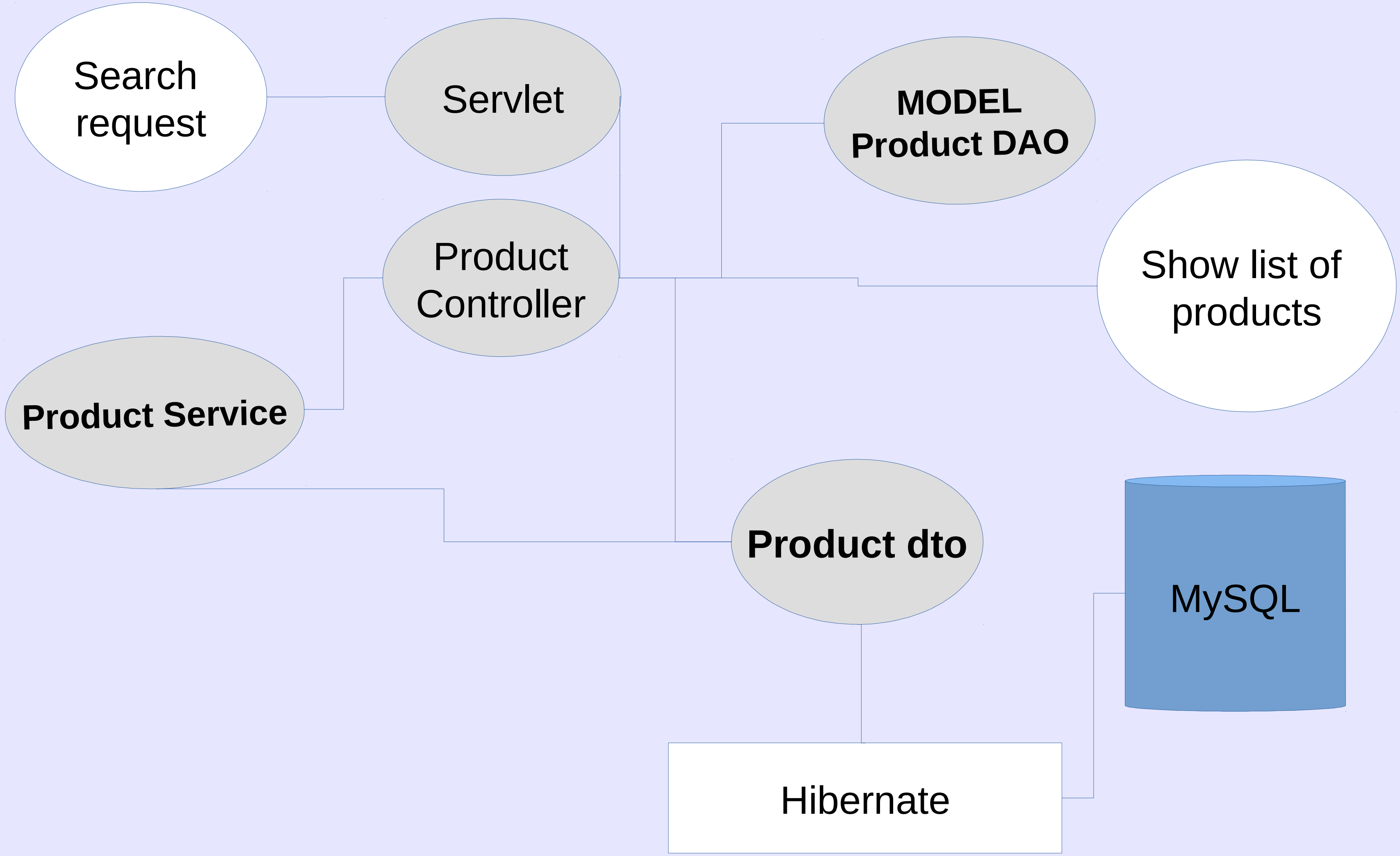
products

product_id INT(11)
product_name VARCHAR(100)
description VARCHAR(100)
category_id INT(11)
price DECIMAL(4,2)
image VARCHAR(100)
code VARCHAR(100)
Indexes

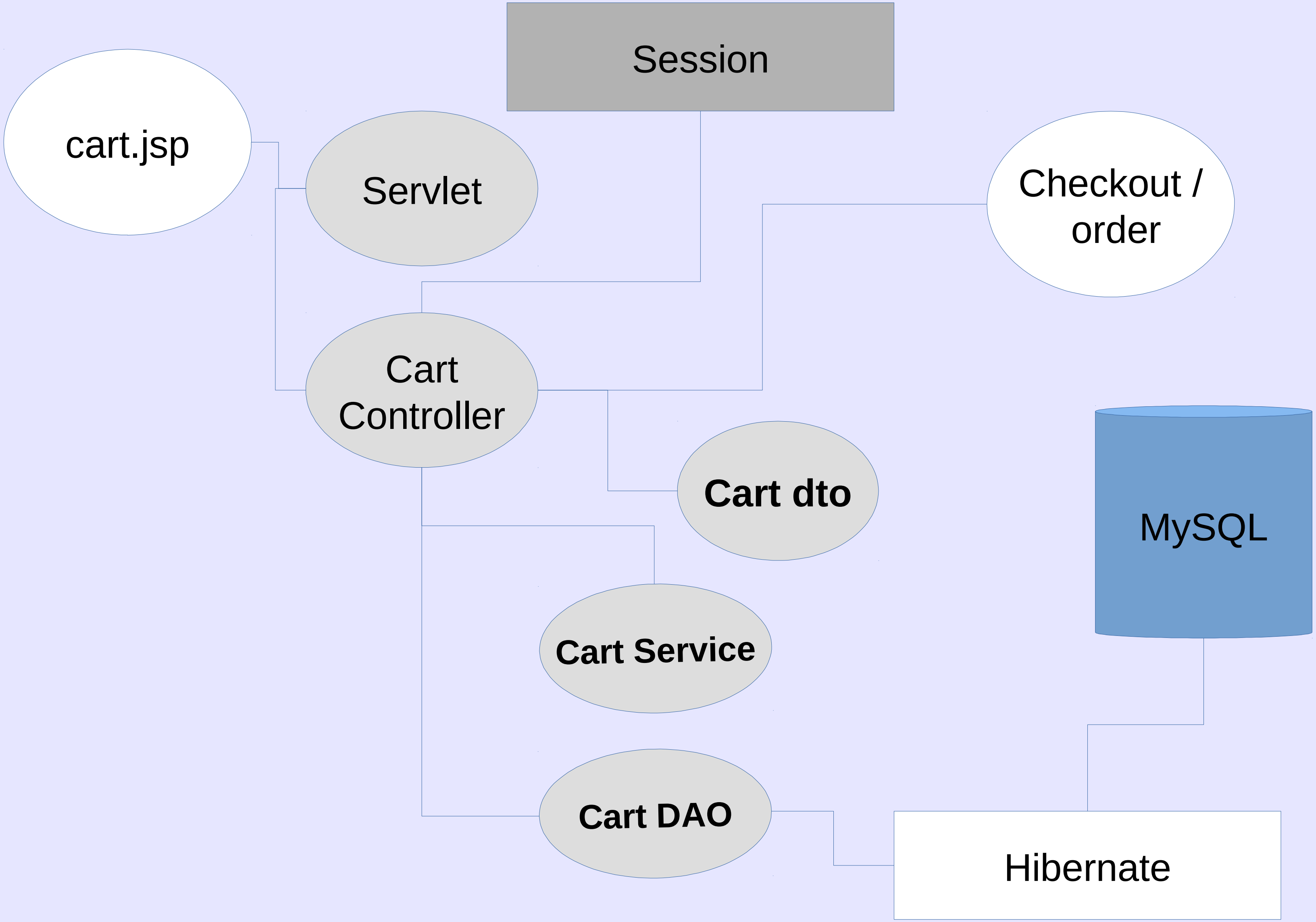
pictures

picture_id INT(11)
path VARCHAR(100)
product_id INT(11)
Indexes

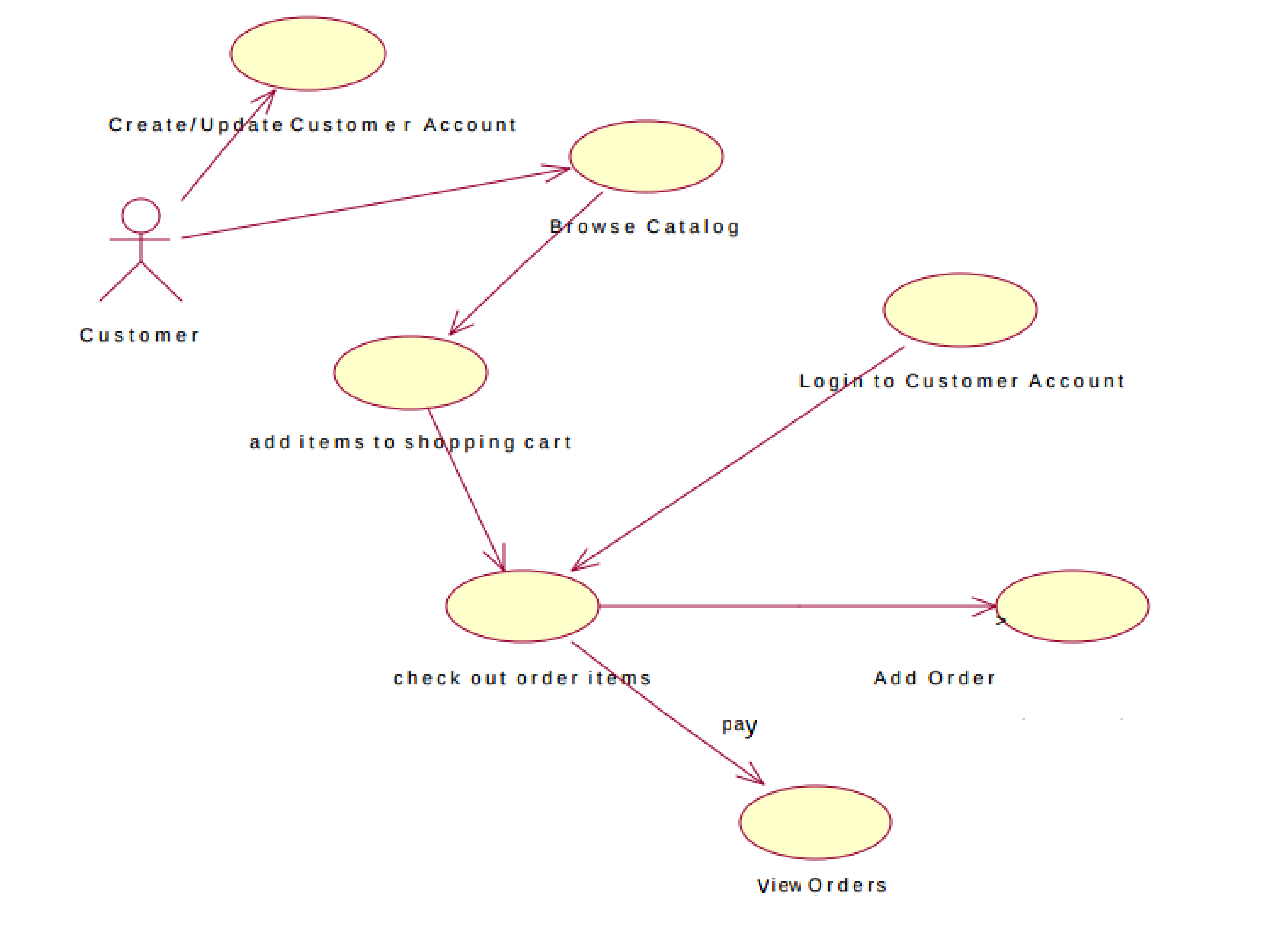
Browse Catalog



Shopping Cart



How the system works?


Create/Update Customer Account
Browse Catalog
Customer
Login to Customer Account
add items to shopping cart
check out order items
Add Order
pay
View Orders

Next steps…

- Improve coverage of unit tests;
- Design admin module
- Improve design of UI

THANK YOU!