Advanced Internet Programming –Assignment 1

## Installation

Server – Oracle WebLogic 10.03.06

* The project is running in server of Oracle WebLogic server and the version needs to be 10.03.06, project deployment could be done via putting the AIP\_ASS1.war file in the WebLogic domain under the auto deploy directory.

Database - MySQL

* The Database uses MySQL, and the URL used for the database connection is 127.0.0.1
* The Database file is named as schema.sql under the folder of WEB\_INFO/lib
* Running the database in the command line with($ mysql –uroot –proot < schema.sql ), root is your MySQL username of –u**root** and root is password of –p**root**
* In order to running the database schema successfully at the first time you has to remove/comment out the first line (DROP DATABASE fcSportsware ;), this is because this line of code would drop the old database and creating a new database.

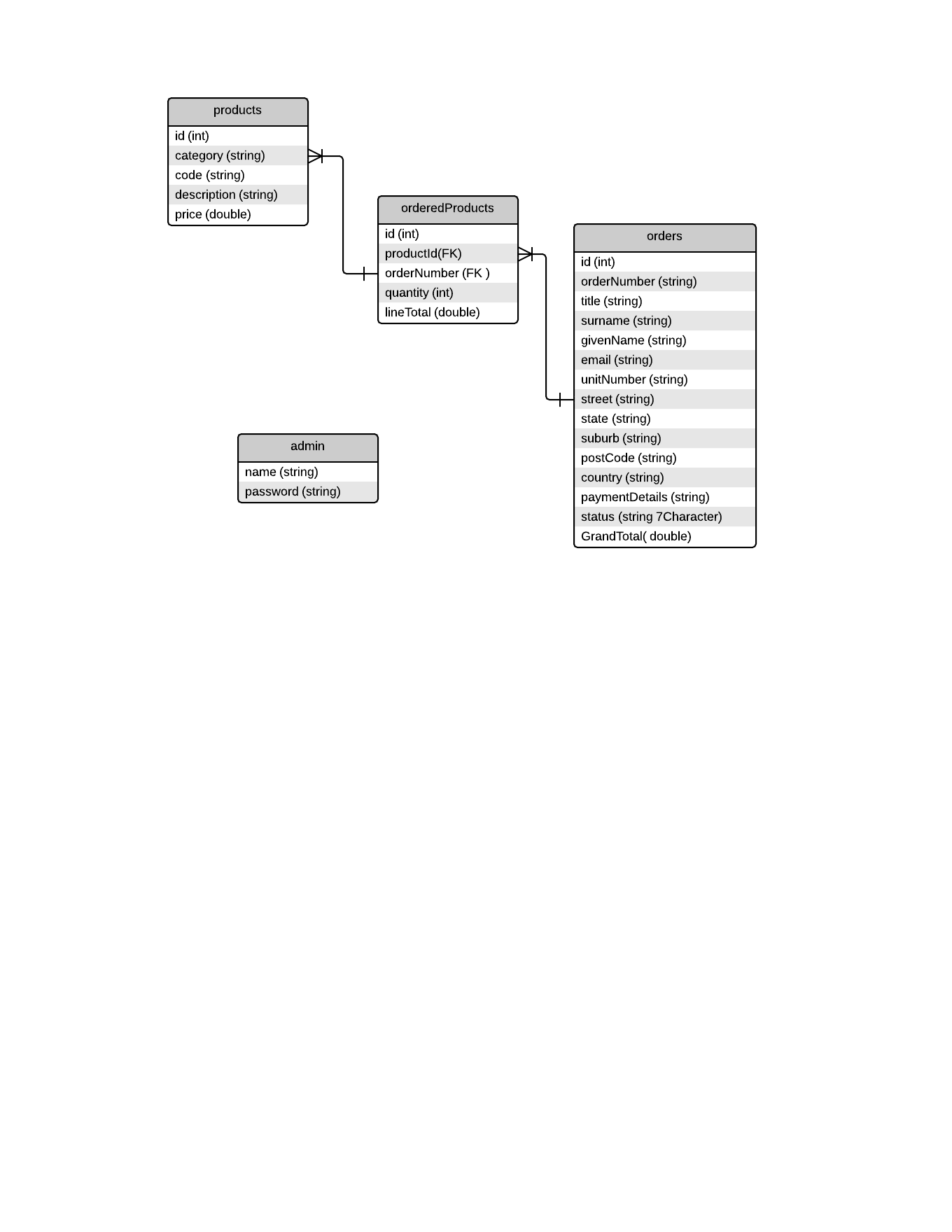
## C:\Users\11376860\Downloads\AIP_SystemArchitecture - New Page.pngSystem Architecture

* This project uses MVC Architecture associated with DAO and DTO

## Database Design

The Database is named as fcSportsware and totally it includes 3 tables, products, orderedProducts and orders.

* Products tables stores the product details includes category, code, description and price, and it which is generated by the database as auto incremental ID.
* OrderedProducts tables stores the quantity and line Total, productId is referenced to the id in products table as foreign key to join the two tables, and orderNumber is referenced as foreign key for join the order table.
* Orders table includes all the data for the customer details, orderNumber is generated by the server as a unique id for each order.
* Admin table stores the username and password for the admin, the default username is orders and password is orderspw



## Class Description

### JSP pages

**index.jsp**

* This is index page for the project and it shows the link for each page with the overview description of the company business, also admin could login in this page.

**products.jsp**

* This page displays ordering products that is offered by the fcSportsware company, it has a category filter allows customer to filter the products by category, customer can also add or check products in their shopping cart.

**orders.jsp**

* Customer can check their shopping cart details include the selected products, quantity and total price that applied to this order; they could cancel the entire order, purchase the order or continue shopping.

**purchase.jsp**

* This page shows the purchase form allows customer to input their ordering details.

**comfirmPurchase.jsp**

* This page shows a confirmation dialog for customer to confirm process the order or cancel the order

**purchaseNotice.jsp**

* Once customer has purchased order successfully, then this page shows their order number.

**viewOrder.jsp**

* This page allows customer to view an existing order, they could searching their order via their order number and surname.

**admin.jsp**

* This is default admin page allows admin to manages customer orders, like view outstanding orders and update orders

**admin\_update.jsp**

* This page allows admin to update customer order status.

### Java Classes

**Model**

* Admin.java, Reusable java bean for admin object and allows user to login as admin
* Order.java, Reusable java bean for order object
* OrderedProduct.java, Reusable java bean for orderedProduct object
* Product.java, Reusable java bean for product object

**Database**

* MysqlDatabase.java, Reusable java bean for mysql database connection

**DAO**

* OrderDAO.java, java interface for processing and searching orders
* OrderDAOImple.java, processing and searching orders via user input
* ProductDAO.java, java interface for searching products
* ProductDAOImple.java, searching product via category or get all the products if no category is selected

**Controller**

* IndexServlet.java, handles request from index.jsp page
* ProductServlet.java, handles request from products.jsp page to display the products
* OrderServlet.java, handles request from orders.jsp page for showing ordering details
* ProcessOrderServlet.java, handles request from comfirmPurchase.jsp page to confirm the purchase process or cancel it
* PurchaseServlet.java, handles request from purchase.jsp page for saving order and customer information
* ViewOrderServlet.java, handles http request from viewOrder.jsp page for searching order
* adminServlet.java, handles request from admin.jsp and admin\_update.jsp page for searching outstanding orders or update order status