Stock Management System

Team 9

AFRIN Rukaya (A0113802W)
CHAN Jian Liu (A0226741H)
LIU Lei (A0214899L)
RAMAKRISHNAN Niveditha (A0214867W)
XIAO Changwei (A0226757U)
ZHANG Hongduo (A0226744B)
ZHOU Yanjun (A0226701N)
ZHU Haokun (A0226723H)

Work Distribution

Jian Liu: UserList, and supplier detail page; contributed in the CRUD logic for users, supplier, products, inventory, stock usage (customer), stock usage inventory.

Rukaya: Login logout with spring security for role based authentication; beatified all html pages

Nivi: CRUD for Admin user detail page; pagination and sorting; reorder report and contribution to Stock usage report.

Liu Lei: CRUD for supplier detail page; separation of admin and mechanic view page

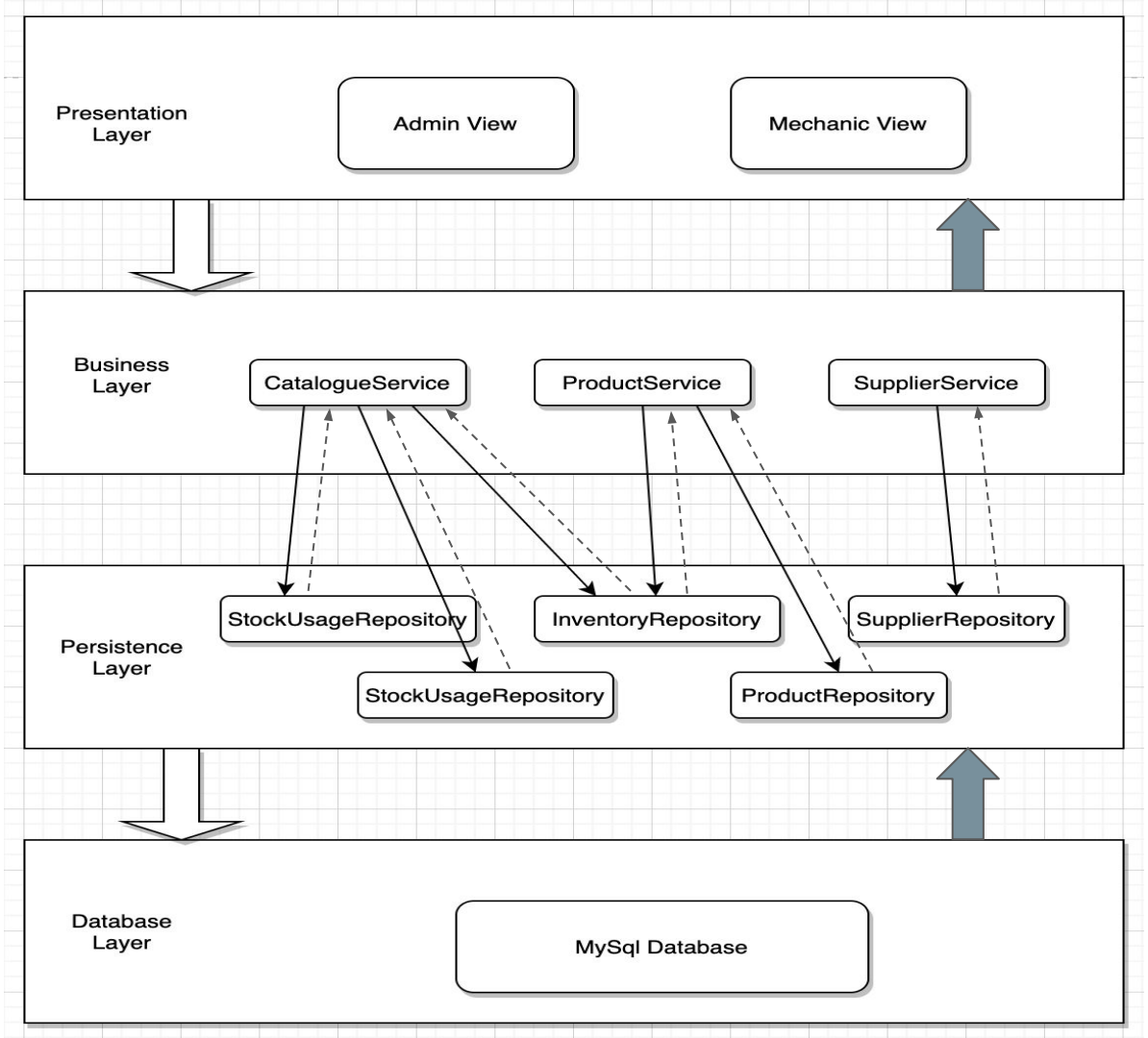
Hongduo: Inventory and product view

Haokun: Java mail and update product state; product list

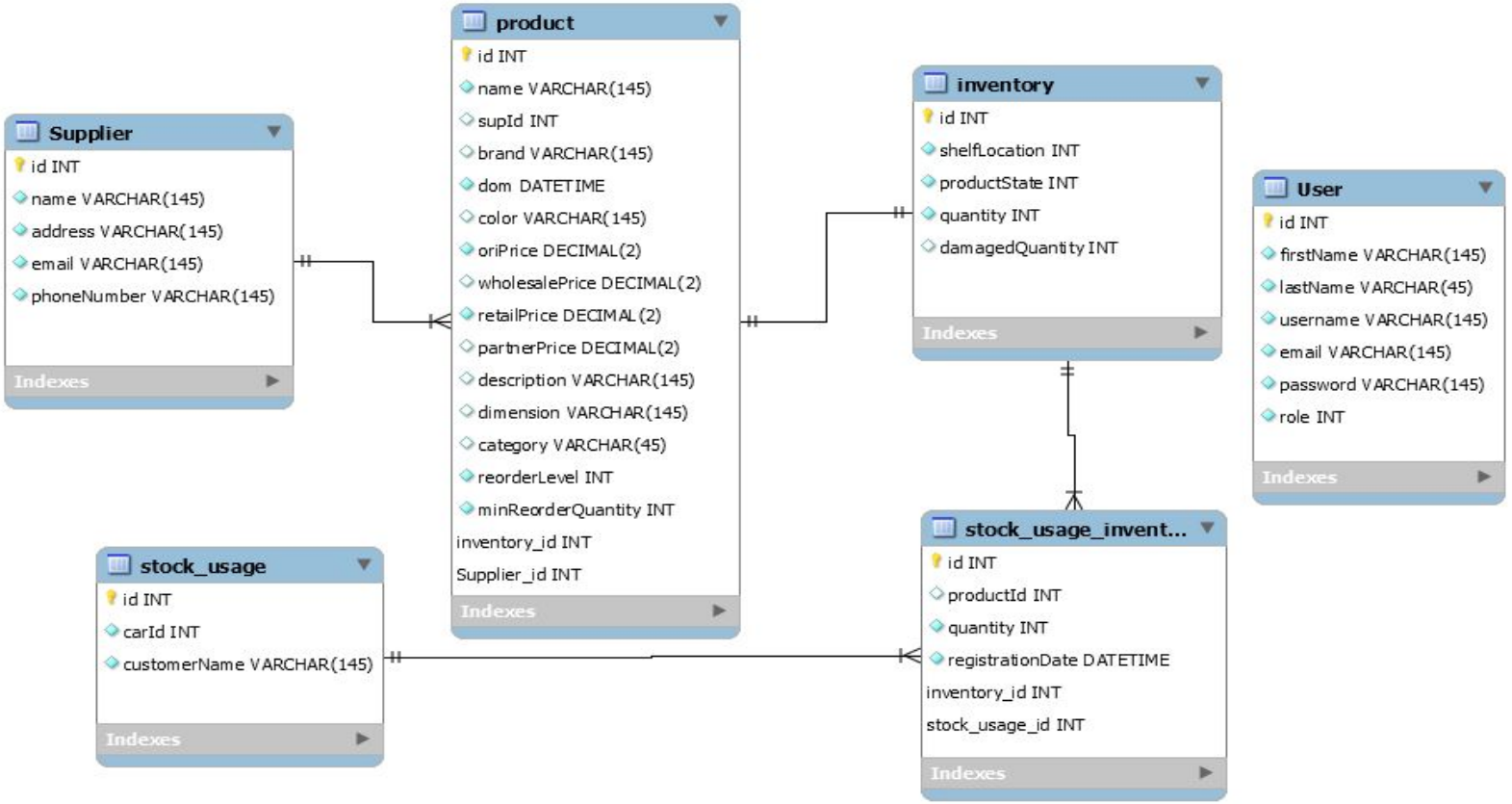
Changwei: Product Search; Supplier list

Yanjun: Stock Usage CRUD for mechanic; search stock usage record based on selected date range; update inventory stock status for admin; report damaged product quantity for admin

Architecture Diagram



ER Diagram


product
id INT
name VARCHAR(145)
supId INT
brand VARCHAR(145)
dom DATETIME
color VARCHAR(145)
oriPrice DECIMAL(2)
wholesalePrice DECIMAL(2)
retailPrice DECIMAL(2)
partnerPrice DECIMAL(2)
description VARCHAR(145)
dimension VARCHAR(145)
category VARCHAR(45)
reorderLevel INT
minReorderQuantity INT
inventory_id INT
Supplier_id INT
Indexes
Supplier
id INT
name VARCHAR(145)
address VARCHAR(145)
email VARCHAR(145)
phoneNumber VARCHAR(145)
Indexes
inventory
id INT
shelfLocation INT
productState INT
quantity INT
damagedQuantity INT
Indexes
User
id INT
firstName VARCHAR(145)
lastName VARCHAR(45)
username VARCHAR(145)
email VARCHAR(145)
password VARCHAR(145)
role INT
Indexes
stock_usage
id INT
carId INT
customerName VARCHAR(145)
Indexes
stock_usage_invent...
id INT
productId INT
quantity INT
registrationDate DATETIME
inventory_id INT
stock_usage_id INT
Indexes

Technologies Used

Spring Web framework

Spring Security

Spring Data JPA

Thymeleaf

MySQL Database

JavaMail

Validation

GitHub

Lessons we've learned

1. We learnt that having a setter for an entity's id is very important (despite having DB auto-generating the id), because without a setter for id, the Model class object in the html view would not be able to set the id, and the object sent back to the controller would not have an id. Therefore, when the service class ".save()" the object, it would not update the record but save it as a new record instead.
2. Database design is very important because an inefficient design can make code logic very complicated and even became a bottleneck when attempting to adopt new technologies. (eg: When we were trying to build APIs for react page, there was a infinite loop/nested structure in our JSON data..)
3. We learnt that apart from the technical work we did for the project, the communication among our group members is also equally important as it allows the workload to be handed over to one another smoothly. For example the structure and coding style are different, when we merged our codes, there were problems surfaced. After an effective communication with our members, we were able to solve the problem.
4. We learned that it is very efficient to subdivide functions and assign them to team members as tasks. Each person just needs to focus on completing their own tasks, greatly reducing the frequency of errors.
5. Peer coding and debugging are very helpful and effective, and can help team members build strong and trustful connections!