# AndroidShop

In this assignment, we designed a module for placing an order. User chooses products and sets counts, and places order. The server will process the request, changing balance, product storage, order history accordingly. And it is error-tolerant by detecting invalid input, inadequate storage or money, etc. on both client and server side.

**Structure of package:**

This package is composed as below:

./Server/AndroidApp

./Client/OrderApp

./Database/AndroidShop

The “AndroidApp” is the Server side which can run by Eclipse. Use Android Studio to run “OrderApp”. The “AndroidShop/” is for restore our database based on MongoDB. The way to restore it will be introduced.

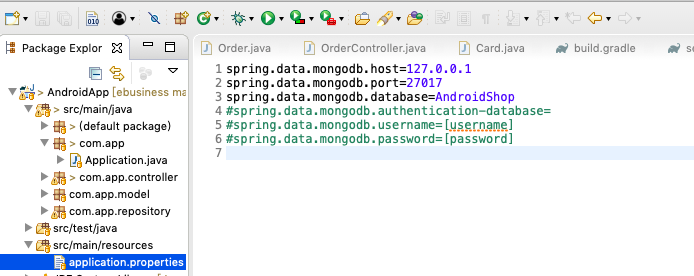
**Stack:**

Client: Android

Server: Spring, MongoDb

**How to run:**

1. Restore database:
2. Below is how we set the attribute of database in server side. You should set it according to your database settings.



1. Restore

We already dumped our database “AndroidShop”

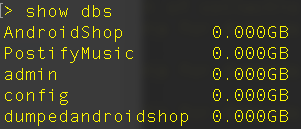
Use the command:

mongorestore -h $(your\_mongodb\_ip\_address) --port $(your\_mongodb\_port) -u $(username) -p $(password) -d $(database\_name) --drop $(path\_to\_dumped\_file\_AndroidShop)

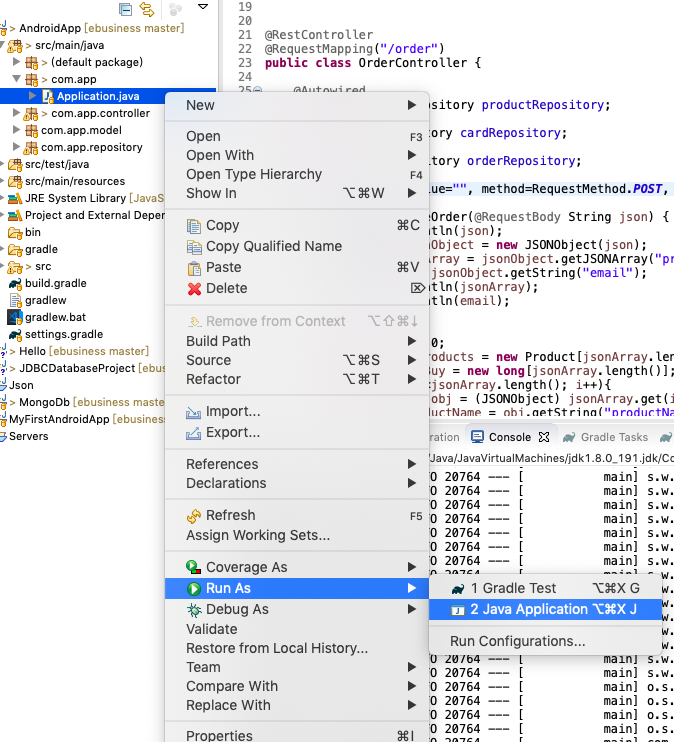
For example:



Then we get the new database “dumpandroidshop”:

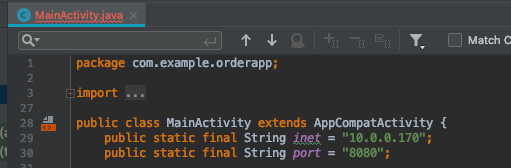


1. Run Server: right click on “Application.java” in “com.app” package, and run as Java Application. The implementation of placing an order is in “OrderController.java” under “com.app.controller” package.



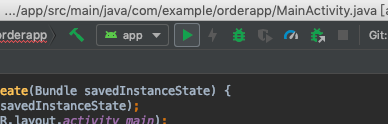
In addition, in “com.app.model” package, we defined several collections, which describe our data.

1. Open Android Studio:
2. Change IP address, and port (if necessary):

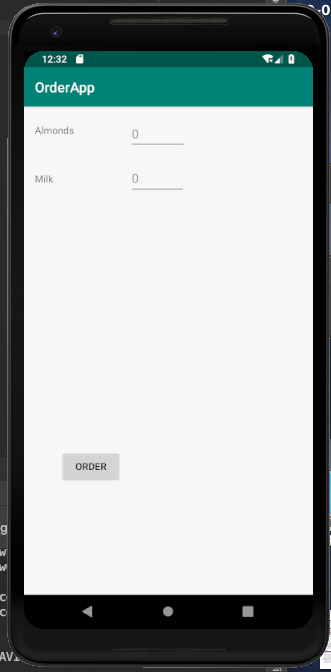


The IP is your IP in local network, which can be acquired by “ifconfig” or “netstat” in Unix system. “8080” is the default port of server.

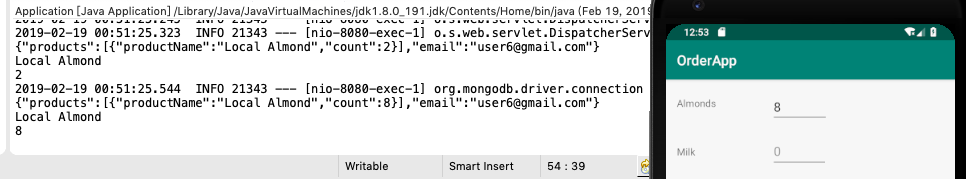
1. Run the project by clicking the green button



1. Check the item you want and input the count. Any invalid operation will not trigger the server to process



1. At the server side, we can see the order is processed.



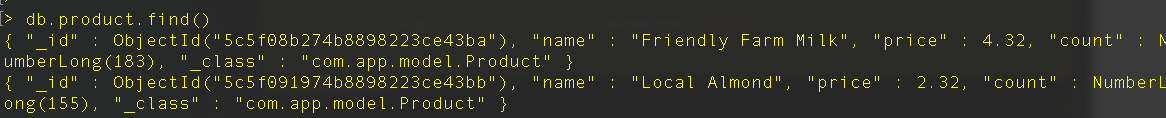
The successful connection to database is shown:



Check the order collection:



Check the product collection:



Check the card collection (users’ wallet):

