eCommerce Book Portal using Angular & Spring Boot

Date: 28/06/2020

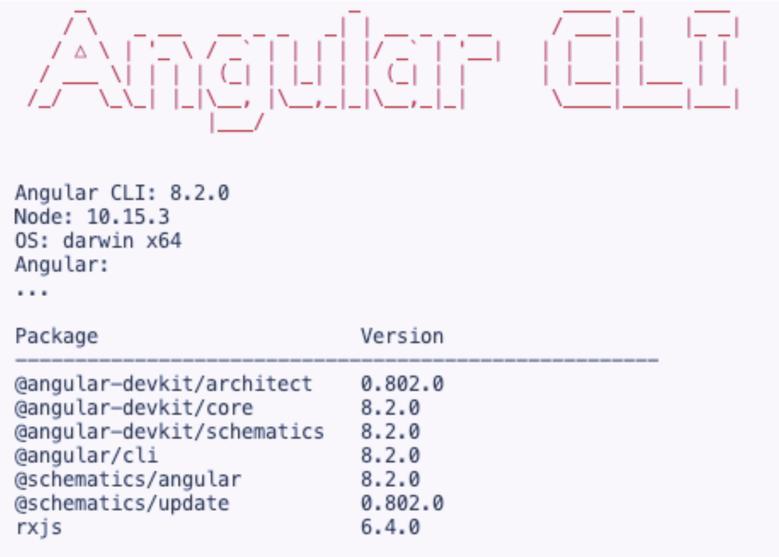
Introduction :

This application is built on top of Udemy course given by client, Also it has some few custom modification and few custom components are added based on requirement. This portal deal with normal shopping site functionality like add to cart, modify cart, check out and place order.

System Requirement:

- Node.js
- Angular 2+
- Java 8
- Maven
- Spring Boot
- Redis Server (Download from *https://medium.com/@petehouston/install-and-config-redis-on-mac-os-x-via-homebrew-eb8df9a4f298*)
- My Sql Database (If you have already then Ignore)
- Java Home properly set in Env Variable(If you have already then Ignore).

Product Front End Development Platform:



```
     _                      _                 ____ _     ___
    / \   _ __   __ _ _   _| | __ _ _ __     / ___| |   |_ _|
   / △ \ | '_ \ / _` | | | | |/ _` | '__|   | |   | |    | |
  / ___ \| | | | (_| | |_| | | (_| | |      | |___| |___ | |
 /_/   \_\_| |_|\__, |\__,_|_|\__,_|_|       \____|_____|___|
                |___/


Angular CLI: 8.2.0
Node: 10.15.3
OS: darwin x64
Angular:
...

Package                      Version
------------------------------------------------------
@angular-devkit/architect    0.802.0
@angular-devkit/core         8.2.0
@angular-devkit/schematics   8.2.0
@angular/cli                 8.2.0
@schematics/angular          8.2.0
@schematics/update           0.802.0
rxjs                         6.4.0
```

Product Backend Version:

java version "1.8.0_231"
Java(TM) SE Runtime Environment (build 1.8.0_231-b11)
Java HotSpot(TM) 64-Bit Server VM (build 25.231-b11, mixed mode)

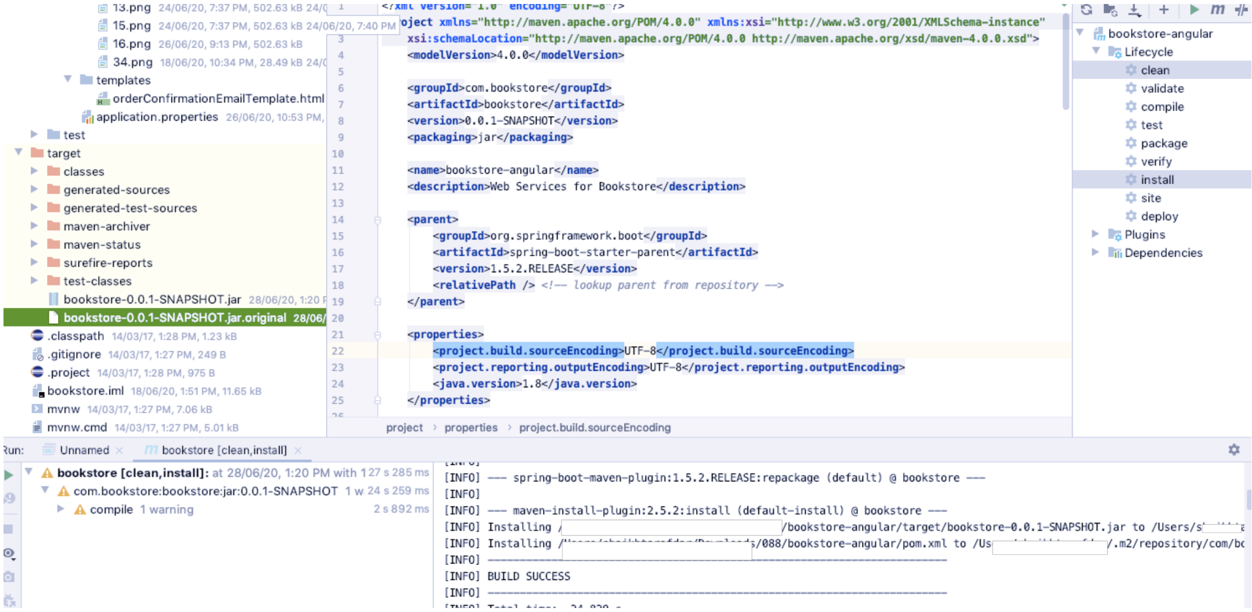
Set up Backend /Spring Boot :

DataBase Setup:

Go to your MySql Workbench, create a database named bookstore should be match exactly what you have mentioned inside application.properties in spring boot.

Start the redis-server by executing .redis-server in your terminal post installation. It’s an in-memory database.

Import bookstore-angular project in to your IDE , go to pom.xml , right click and select clean , install as in below picture.



13.png 24/06/20, 7:37 PM, 502.63 kB 24/0
15.png 24/06/20, 7:37 PM, 502.63 kB 24/06/20, 7:40 PM
16.png 26/06/20, 9:13 PM, 502.63 kB
34.png 18/06/20, 10:34 PM, 28.49 kB 24/0
templates
orderConfirmationEmailTemplate.html
application.properties 26/06/20, 10:53 PM,
test
target
classes
generated-sources
generated-test-sources
maven-archiver
maven-status
surefire-reports
test-classes
bookstore-0.0.1-SNAPSHOT.jar 28/06/20, 1:20 P
bookstore-0.0.1-SNAPSHOT.jar.original 28/06/
.classpath 14/03/17, 1:28 PM, 1.23 kB
.gitignore 14/03/17, 1:27 PM, 249 B
.project 14/03/17, 1:28 PM, 975 B
bookstore.iml 18/06/20, 1:51 PM, 11.65 kB
mvnw 14/03/17, 1:27 PM, 7.06 kB
mvnw.cmd 14/03/17, 1:27 PM, 5.01 kB
bookstore-angular
Lifecycle
clean
validate
compile
test
package
verify
install
site
deploy
Plugins
Dependencies
project › properties › project.build.sourceEncoding
Run: Unnamed × bookstore [clean,install] ×

```
<?xml version="1.0" encoding="UTF-8"?>
oject xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
    <modelVersion>4.0.0</modelVersion>

    <groupId>com.bookstore</groupId>
    <artifactId>bookstore</artifactId>
    <version>0.0.1-SNAPSHOT</version>
    <packaging>jar</packaging>

    <name>bookstore-angular</name>
    <description>Web Services for Bookstore</description>

    <parent>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-parent</artifactId>
        <version>1.5.2.RELEASE</version>
        <relativePath /> <!-- lookup parent from repository -->
    </parent>

    <properties>
        <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
        <project.reporting.outputEncoding>UTF-8</project.reporting.outputEncoding>
        <java.version>1.8</java.version>
    </properties>
```

```
bookstore [clean,install]: at 28/06/20, 1:20 PM with 127 s 285 ms
  com.bookstore:bookstore:jar:0.0.1-SNAPSHOT  1 w 24 s 259 ms
    compile  1 warning                           2 s 892 ms
```

```
[INFO] --- spring-boot-maven-plugin:1.5.2.RELEASE:repackage (default) @ bookstore ---
[INFO]
[INFO] --- maven-install-plugin:2.5.2:install (default-install) @ bookstore ---
[INFO] Installing /[redacted]/bookstore-angular/target/bookstore-0.0.1-SNAPSHOT.jar to /Users/s[redacted]
[INFO] Installing /[redacted]s/088/bookstore-angular/pom.xml to /Us[redacted]/.m2/repository/com/bo
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
```

Before that verify your mail server and database setting inside application.properties file.

Database settings:

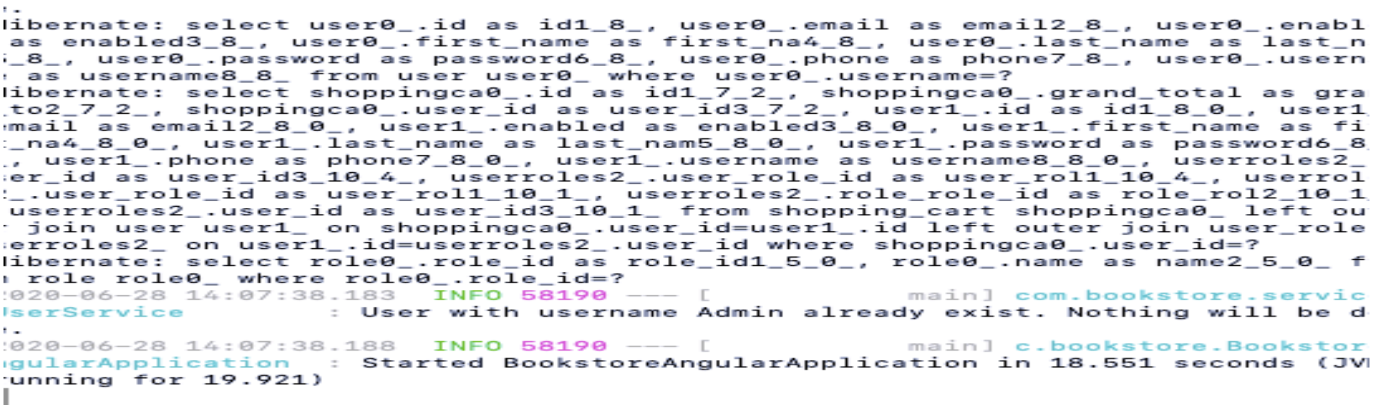
```
spring.datasource.username = your database user name
spring.datasource.password = your database password
```

Mail Server Setting

```
spring.mail.host = letgarden.com
spring.mail.username = support@letgarden.com
spring.mail.password = SmH6YLp{G
spring.mail.properties.mail.smtp.auth = true
spring.mail.properties.mail.smtp.socketFactory.port = 465
spring.mail.properties.mail.smtp.socketFactory.class = javax.net.ssl.SSLSocketFactory
spring.mail.properties.mail.smtp.socketFactory.fallback = false
support.email=support@letgarden.com
```

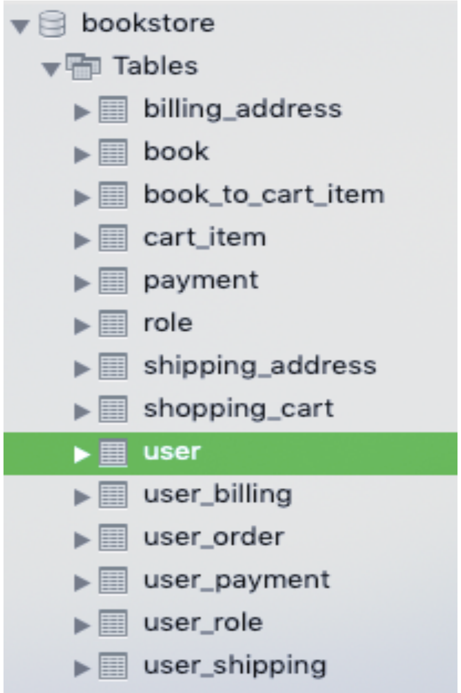
When the build is successful a jar is generated under **/bookstore-angular/target/bookstore-0.0.1-SNAPSHOT.jar.Go to this path hit command**

Java -jar bookstore-0.0.1-SNAPSHOT.jar , it will start spring boot backend server



```
:.
Iibernate: select user0_.id as id1_8_, user0_.email as email2_8_, user0_.enabl
 as enabled3_8_, user0_.first_name as first_na4_8_, user0_.last_name as last_n
;_8_, user0_.password as password6_8_, user0_.phone as phone7_8_, user0_.usern
: as username8_8_ from user user0_ where user0_.username=?
Iibernate: select shoppingca0_.id as id1_7_2_, shoppingca0_.grand_total as gra
.to2_7_2_, shoppingca0_.user_id as user_id3_7_2_, user1_.id as id1_8_0_, user1
:mail as email2_8_0_, user1_.enabled as enabled3_8_0_, user1_.first_name as fi
:_na4_8_0_, user1_.last_name as last_nam5_8_0_, user1_.password as password6_8
., user1_.phone as phone7_8_0_, user1_.username as username8_8_0_, userroles2_
:er_id as user_id3_10_4_, userroles2_.user_role_id as user_rol1_10_4_, userrol
:_.user_role_id as user_rol1_10_1_, userroles2_.role_role_id as role_rol2_10_1
 userroles2_.user_id as user_id3_10_1_ from shopping_cart shoppingca0_ left ou
· join user user1_ on shoppingca0_.user_id=user1_.id left outer join user_role
:erroles2_ on user1_.id=userroles2_.user_id where shoppingca0_.user_id=?
Iibernate: select role0_.role_id as role_id1_5_0_, role0_.name as name2_5_0_ f
ı role role0_ where role0_.role_id=?
:020-06-28 14:07:38.183  INFO 58190 --- [           main] com.bookstore.servic
IserService           : User with username Admin already exist. Nothing will be d
:.
:020-06-28 14:07:38.188  INFO 58190 --- [           main] c.bookstore.Bookstor
ıgularApplication     : Started BookstoreAngularApplication in 18.551 seconds (JVI
·unning for 19.921)
```

Go to your database and check , it will create below table structures.



```
▼ bookstore
  ▼ Tables
    ▶ billing_address
    ▶ book
    ▶ book_to_cart_item
    ▶ cart_item
    ▶ payment
    ▶ role
    ▶ shipping_address
    ▶ shopping_cart
    ▶ user
    ▶ user_billing
    ▶ user_order
    ▶ user_payment
    ▶ user_role
    ▶ user_shipping
```

Set up Front end /Spring Boot :

Open VS Code. The folder has two modules store-front and Store-Admin, Go to respective directory let say Store-Admin, and execute command **npm install** It will installed necessary node modules inside directory node_modules, After it successfully completed execute **npm start** , you will be get something like as below.

```
webpack: Compiled successfully.
webpack: Compiling...
Hash: 5ca73ddadaa1c9af53df
Time: 977ms
chunk    {0} main.bundle.js, main.bundle.map (main) 262 kB {3} [initial] [rendered]
chunk    {1} styles.bundle.js, styles.bundle.map (styles) 218 kB {4} [initial]
chunk    {2} scripts.bundle.js, scripts.bundle.map (scripts) 645 kB {4} [initial]
chunk    {3} vendor.bundle.js, vendor.bundle.map (vendor) 3.23 MB [initial]
chunk    {4} inline.bundle.js, inline.bundle.map (inline) 0 bytes [entry]
webpack: Compiled successfully.
```

Hit the browser http://localhost:4300/login for Admin portal (username :**Admin , password :password**) .If you want to change port, just open package.json and change the port as below.(replace 4300 with your new value)

```
{
  "name": "bookstore-admin",
  "version": "0.0.0",
  "license": "MIT",
  "angular-cli": {},
  "scripts": {
    "ng": "ng",
    "start": "ng serve --port 4300",
    "lint": "tslint \"src/**/*.ts\" --project src/tsconfig.json --type-check && tslint
\"e2e/**/*.ts\" --project e2e/tsconfig.json --type-check",
    "test": "ng test",
    "pree2e": "webdriver-manager update --standalone false --gecko false",
    "e2e": "protractor"
  },
```

Repeat similar for Store-front and hit the url in browser http://localhost:4200/home

(username : **Azmat123, password :password**)