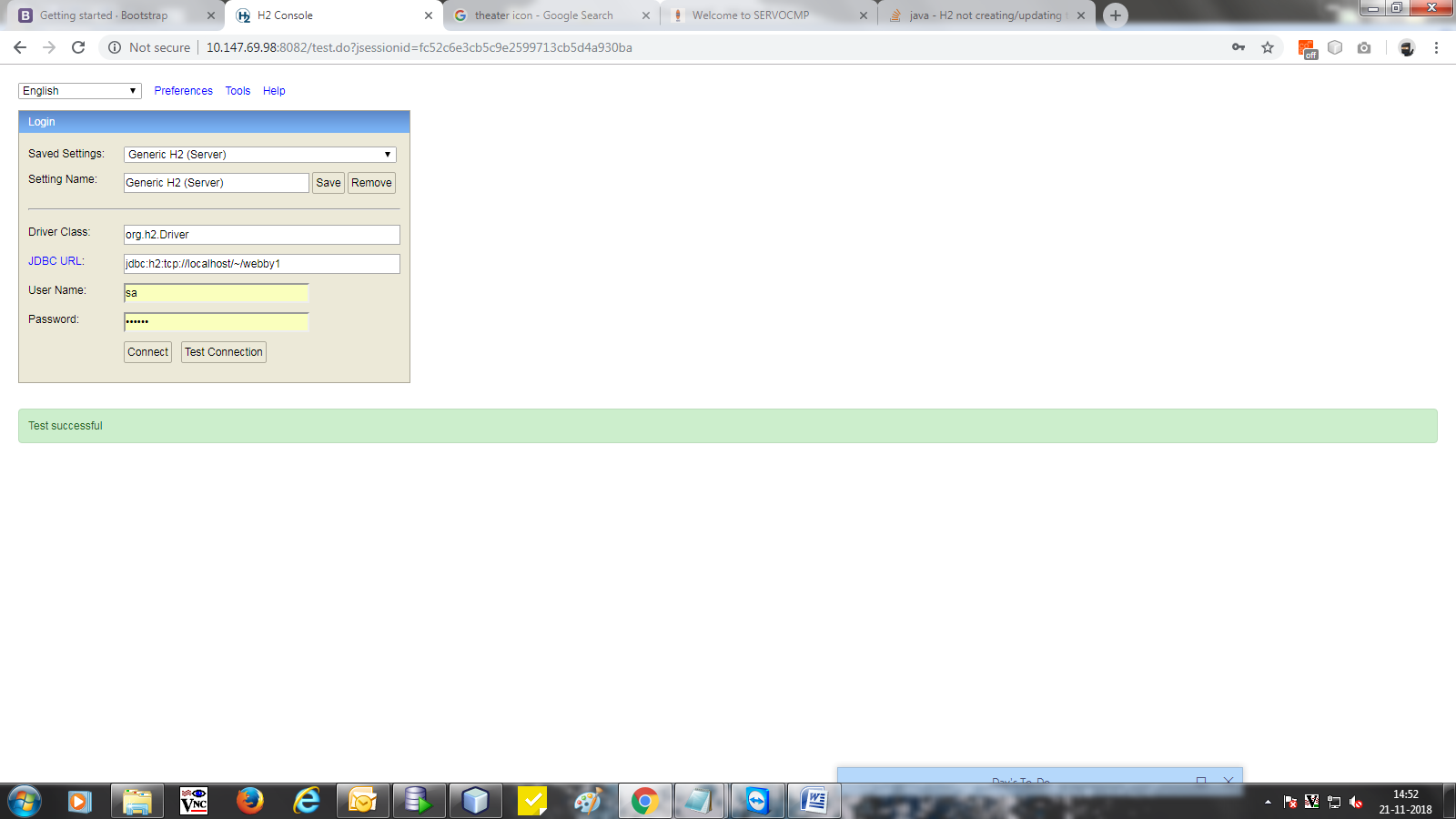
Dir: -- MyProjee\src\main\webapp\resources\dumps.



**Spring Boot Shopping Cart Web App**

**About**

This is a Final project for an institute exam build with Spring. The idea was to build some basic shopping cart web app.

It was made using **Spring MVC**, **Spring Security**, **Spring Data JPA**, **Spring Data REST and Hibernate**. Database is in memory **H2**.

There is a login and registration functionality included.

Users can shop for products. Each user has his own shopping cart (session functionality). Checkout is transactional.

**Configuration**

**Configuration Files**

Folder **src/WEB-INF/** contains config files for **shopping-website** Spring MVC application.

* **src/resources/** -Here all the images, js, fonts and css files are stored which are implemented in jsp pages.

**How to run**

There are several ways to run the application. You can run it from the command line with Maven or run using IDE with apache maven installed.

Once the app starts, go to the web browser and visit <http://localhost/MyProjee>

Admin username: **As enter in h2 database USERS table**

Admin password: **As enter in h2 database USERS table**

User username: **As enter in h2 database USERS table**

User password: **As enter in h2 database USERS table**

**Maven Wrapper**

**Using the Maven Plugin**

Go to the root folder of the application and type:

$ chmod +x scripts/mvnw

$ scripts/mvnw spring-boot:run

**Using Executable Jar**

Or you can build the JAR file with

$ scripts/mvnw clean package

Then you can run the JAR file:

$ java -jar target/shopping-cart-0.0.1-SNAPSHOT.jar

**Maven**

Open a terminal and run the following commands to ensure that you have valid versions of Java and Maven installed:

$ java -version

java version "1.8.0\_102"

Java(TM) SE Runtime Environment (build 1.8.0\_102-b14)

Java HotSpot(TM) 64-Bit Server VM (build 25.102-b14, mixed mode)

$ mvn -v

Apache Maven 3.3.9 (bb52d8502b132ec0a5a3f4c09453c07478323dc5; 2015-11-10T16:41:47+00:00)

Maven home: /usr/local/Cellar/maven/3.3.9/libexec

Java version: 1.8.0\_102, vendor: Oracle Corporation

**Using the Maven Plugin**

The Spring Boot Maven plugin includes a run goal that can be used to quickly compile and run your application. Applications run in an exploded form, as they do in your IDE. The following example shows a typical Maven command to run a Spring Boot application:

$ mvn spring-boot:run

**Using Executable Jar**

To create an executable jar run:

$ mvn clean package

To run that application, use the java -jar command, as follows:

$ java -jar target/shopping-cart-0.0.1-SNAPSHOT.jar

To exit the application, press **ctrl-c**.

**Docker**

It is possible to run **shopping-cart** using Docker:

Build Docker image:

$ mvn clean package

$ docker build -t shopping-cart:dev -f docker/Dockerfile .

Run Docker container:

$ docker run --rm -i -p 8070:8070 \

--name shopping-cart \

shopping-cart:dev

**Helper script**

It is possible to run all of the above with helper script:

$ chmod +x scripts/run\_docker.sh

$ scripts/run\_docker.sh

**Docker**

Folder **docker** contains:

* **docker/shopping-cart/Dockerfile** - Docker build file for executing shopping-cart Docker image. Instructions to build artifacts, copy build artifacts to docker image and then run app on proper port with proper configuration file.

**Util Scripts**

* **scripts/run\_docker.sh.sh** - util script for running shopping-cart Docker container using **docker/Dockerfile**

**Tests**

Tests can be run by executing following command from the root of the project:

$ mvn test

**Helper Tools**

**Browser**

Go to the web browser and visit http://localhost:8081/

You will need to be authenticated to be able to see this page.

**H2 Database web interface**

Go to the web browser and visit http://localhost:8070/h2-console

In field **JDBC URL** put

jdbc:h2:mem:shopping\_cart\_db

In /src/main/resources/application.properties file it is possible to change both web interface url path, as well as the datasource url.