Cinema Documentation

By: Imran Mohd Raji

Description:

Backend is developed in Java using JPA to access MySQL queries. Front End is developed using Angular.

Frontend Link: <https://github.com/imranmr/cinema-FE.git>

Backend Link: <https://github.com/imranmr/cinema-BE.git>

Features:

* Client is able to login
* Client can reset password
* Client can see movies
* Client can sort by genre
* Client can search movies
* Client can search movie timing
* Client can search movie location
* Client can add movie to cart
* Client can adjust number of seats in cart
* Client can delete cart items
* Client can purchase movies
* Client can see all purchased movies
* Client can see client details such as username, email, name

Admin Features

* Admin can add movies
* Admin can update movies
* Admin can delete movies
* Admin can add movie timing
* Admin can update movie timings
* Admin can delete movietimings
* Admin can add genre
* Admin can update genre
* Admin can delete genre

Note: Current github versions are designed to run on AWS EC2 Instances. Please follow the steps below to proceed.

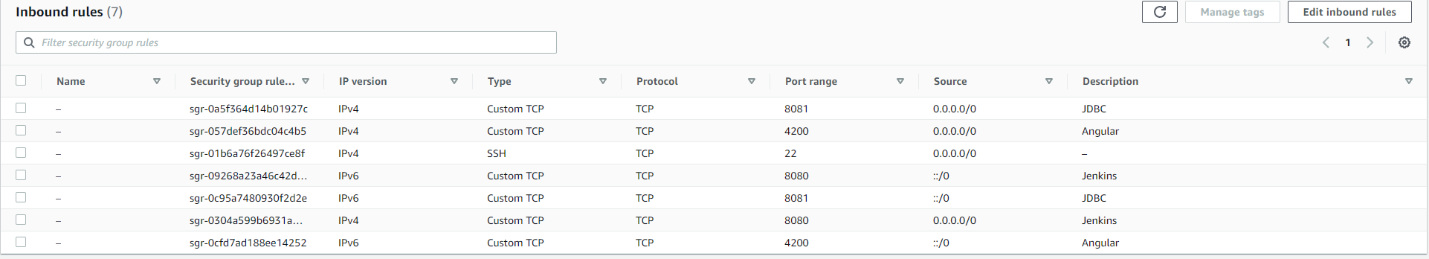
Note: Github versions can be updated to allow local instances.

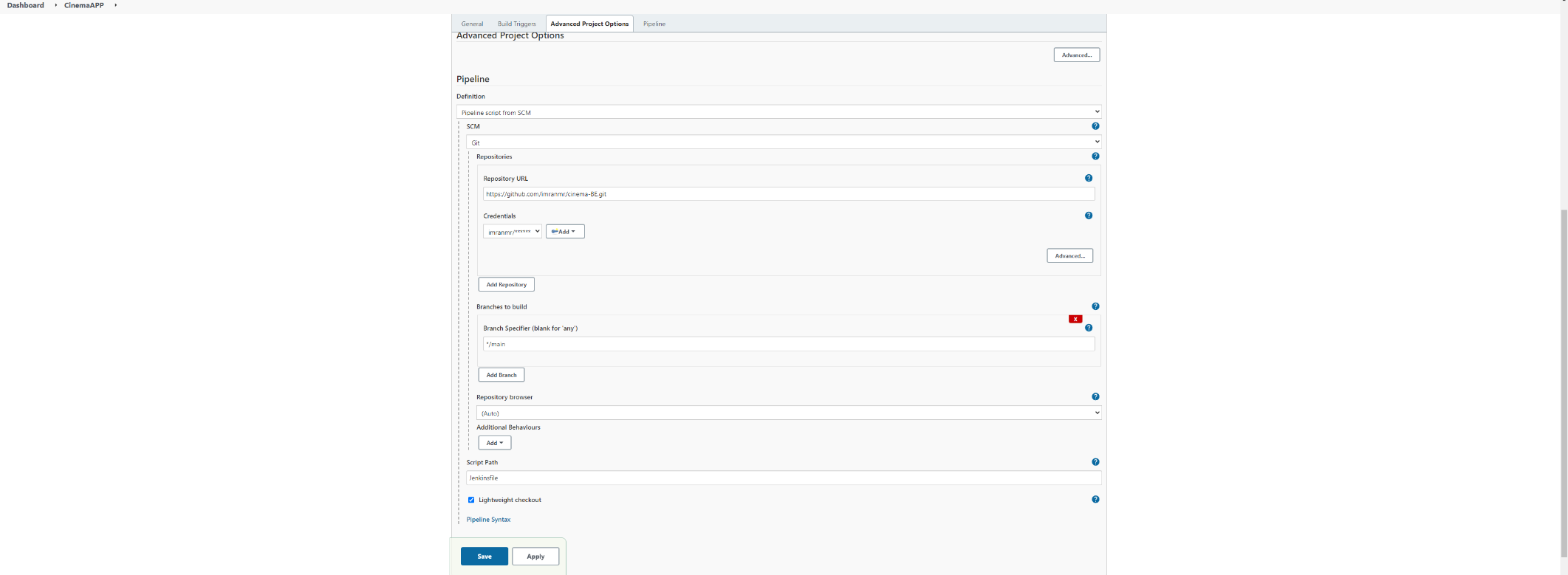
CI/CD Deployment for Springboot Application

This project requires access to AWS EC2 instances. The technologies used are as follows:

* Jenkins
* Spring boot
* Docker
* Eclipse
* AWS EC2

To setup the project, follow the instructions below:

1. **Setup EC2 instance on AWS**
   1. Make sure EC2 has following settings:
   2. 
   3. Once created, ssh into the EC2 instance
2. **Install Jenkins**
   1. sudo apt-get update
   2. sudo apt install default-jdk -y
   3. sudo apt install git -y
   4. sudo apt install maven -y
   5. wget -q -O - https://pkg.jenkins.io/debian-stable/jenkins.io.key | sudo apt-key add -
   6. sudo sh -c 'echo deb https://pkg.jenkins.io/debian-stable binary/ > /etc/apt/sources.list.d/jenkins.list'
   7. sudo apt-get update
   8. sudo apt-get install jenkins -y
3. **Install MySQL**
   1. sudo apt install docker.io -y
   2. sudo usermod -aG docker $USER
   3. sudo docker network create cinema-mysql
   4. sudo docker pull mysql/mysql-server:latest
   5. sudo docker run --rm --name=mysql-container --network cinema-mysql -p3306:3306 -e MYSQL\_ROOT\_HOST=% -e MYSQL\_ROOT\_PASSWORD=mysql-pw -e MYSQL\_DATABASE=cinema -d mysql/mysql-server
4. **Visit the EC2 host website at port 8080 and create a user**
   1. Get the admin password by typing:
   2. sudo su
   3. cat /var/lib/jenkins/secrets/initialAdminPassword
   4. Once user is created, create a new project of type pipeline
   5. Set the following definition:



* 1. Click apply and save
  2. sudo usermod -a -G docker jenkins
  3. sudo service jenkins restart

1. **Once mysql is running add data to it**
   1. sudo docker exec -it mysql-container bash -l
   2. mysql -uroot -pmysql-pw
   3. use cinema;
   4. insert into user (userid,adminrights, email, firstname, lastname, mobile, password) values (1,1, "imran.mohdraji@gmail.com", "Imran", "Raji", "1112223333", "admin");
   6. insert into movie (movieid, name, description, price,userid,hours,minutes,rating,releasedate,posterurl) values (1,"Interstellar","The earth is in trouble and we need to find a new place to live", 6, 1,3,35,"PG","2021-10-05","https://www.joblo.com/wp-content/uploads/2014/05/interstellar-poster-2-1-400x600.jpg");
   7. insert into movie (movieid, name, description, price,userid,hours,minutes,rating,releasedate,posterurl) values (2,"Shrek","Ogre lives in swamp with donkey", 6, 1,2,35,"PG","2001-10-05","https://originalvintagemovieposters.com/wp-content/uploads/2010/06/Shrek1.jpg");
   8. insert into movie (movieid, name, description, price,userid,hours,minutes,rating,releasedate,posterurl) values (3,"Lord of the Rings","The ring has been found in Middle Earth. It needs to be destroyed", 6, 1,3,20,"PG","2001-10-05","https://www.limitedruns.com/media/cache/7e/6b/7e6b6743a7e45d096838abd67b2464e2.jpg");
   10. insert into movietiming (movietimingid, language, seats, time, movieid,date,location) values (1, "English", 250, "21:00:00",1,"2022-03-20","New York");
   11. insert into movietiming (movietimingid, language, seats, time, movieid,date,location) values (2, "Spanish", 250, "22:00:00",1,"2022-03-22","New York");
   12. insert into movietiming (movietimingid, language, seats, time, movieid,date,location) values (3, "French", 250, "23:00:00",1,"2022-03-23","California");
   13. insert into movietiming (movietimingid, language, seats, time, movieid,date,location) values (4, "English", 250, "21:00:00",2,"2022-01-20","New York");
   14. insert into movietiming (movietimingid, language, seats, time, movieid,date,location) values (5, "Spanish", 250, "22:00:00",2,"2022-03-23","California");
   15. insert into movietiming (movietimingid, language, seats, time, movieid,date,location) values (6, "French", 250, "23:00:00",2,"2022-03-22","Pennsylvania");
   16. insert into movietiming (movietimingid, language, seats, time, movieid,date,location) values (7, "English", 250, "21:00:00",3,"2022-01-20","New York");
   17. insert into movietiming (movietimingid, language, seats, time, movieid,date,location) values (8, "Spanish", 250, "22:00:00",3,"2022-03-23","California");
   18. insert into movietiming (movietimingid, language, seats, time, movieid,date,location) values (9, "French", 250, "23:00:00",3,"2022-03-22","Pennsylvania");
   20. insert into moviegenre(moviegenreid,genre,movieid) values (1, "Action", 1);
   21. insert into moviegenre(moviegenreid,genre,movieid) values (2, "Suspense", 1);
   22. insert into moviegenre(moviegenreid,genre,movieid) values (3, "Mystery", 1);
   23. insert into moviegenre(moviegenreid,genre,movieid) values (4, "Comedy", 2);
   24. insert into moviegenre(moviegenreid,genre,movieid) values (5, "Fantasy", 2);
   25. insert into moviegenre(moviegenreid,genre,movieid) values (6, "Adventure", 2);
   26. insert into moviegenre(moviegenreid,genre,movieid) values (7, "Action", 3);
   27. insert into moviegenre(moviegenreid,genre,movieid) values (8, "Fantasy", 3);
   28. insert into moviegenre(moviegenreid,genre,movieid) values (9, "Adventure", 3);
2. Setup Front End:
   1. git clone <https://github.com/imranmr/cinema-FE.git>
   2. cd cinema-FE
3. Install Nodejs
   1. curl -sL https://deb.nodesource.com/setup\_14.x | sudo -E bash –
   2. sudo apt-get install -y nodejs
4. Install NPM
   1. sudo apt install npm -y
   2. sudo npm install -g @angular/cli
   3. sudo npm install -g npm@latest
5. Run Front end
   1. Npm install
   2. ng serve --host=0.0.0.0 --disable-host-check
6. Go to <http://34.230.58.0:4200/> to visit front end