**Deployment Steps of Account-Service Application**

**Functionality:**

It contains a nominal login page which can be used as Admin login. It creates a new “current account” of already existing Customers using their customerID property. It displays a particular Customer’s details based on its customerID.

**Software and Technologies used:**

* Spring Boot for publishing Rest Apis
* Gradle build tool
* JPA Hibernate for performing CRUD operations on database
* H2 in-memory database
* Angular8, HTML5, CSS, Bootstrap for UI components
* Apache Tomcat for server deployment
* Eclipse IDE

**Deploying Application using Jenkins**

1. Create a pipeline job in your Jenkins
2. Copy the groovy script which is provided in the source code named as “Jenkinsfile”.
3. Replace the respective paths into the script as per your system.
4. Make sure you have tomcat installed on the same server as your job created.
5. Build the job.
6. It will copy the source code from my Github Repository which is [**https://github.com/DeepaMotwani/AccountServiceAssignment.git**](https://github.com/DeepaMotwani/AccountServiceAssignment.git)and build both Rest API application and UI code and copy it into your tomcat directory and start it.

**Deploying application manually**

1. Clone the repository from the github:

[**https://github.com/DeepaMotwani/AccountServiceAssignment.git**](https://github.com/DeepaMotwani/AccountServiceAssignment.git)

1. Build the code using the command:

**gradle clean build**

1. Build the UI code by opening the terminal at the following location:

src/UI/account-service

Run the command if you have npm and node installed on your machine

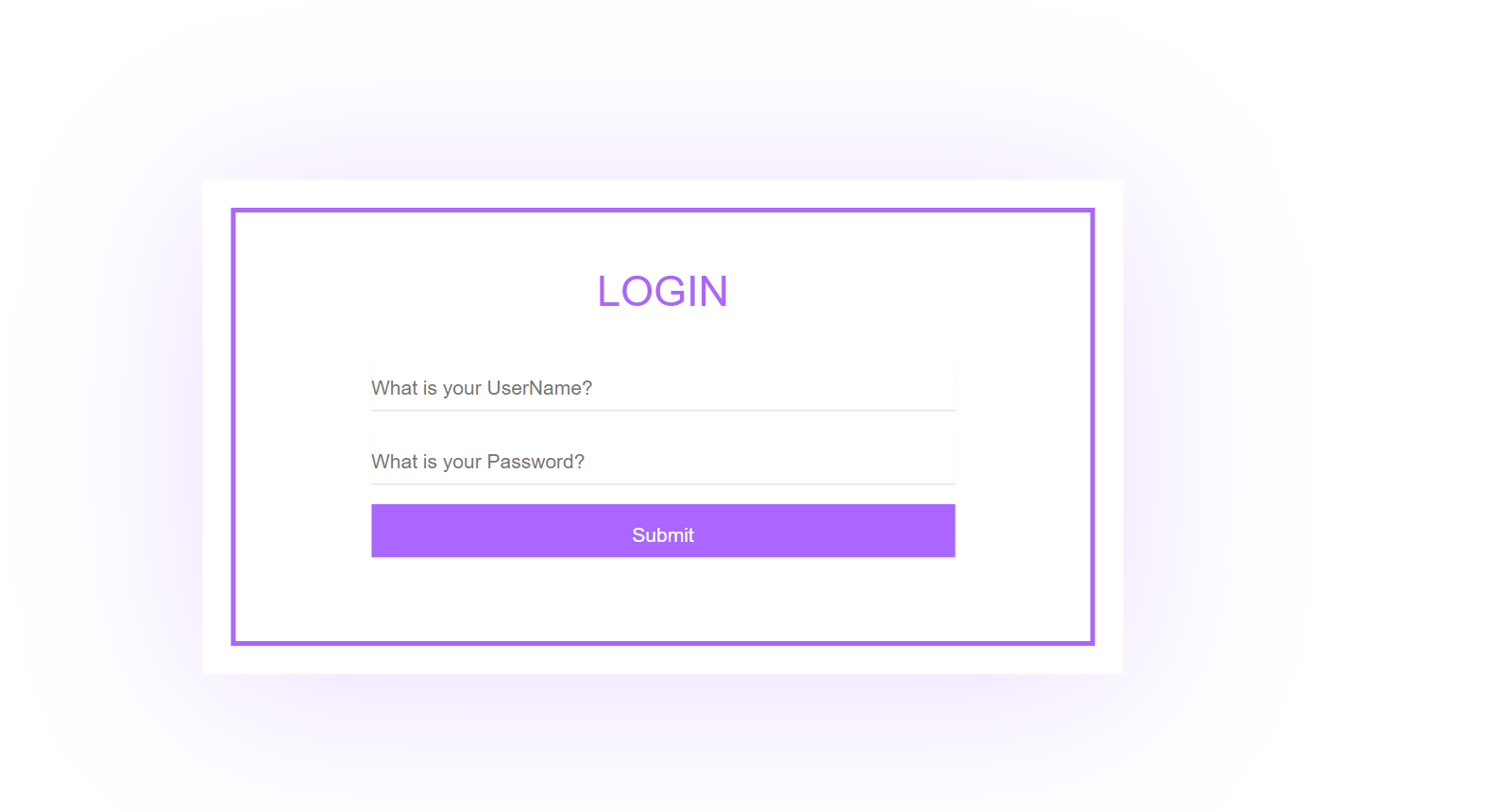
**npm run build**

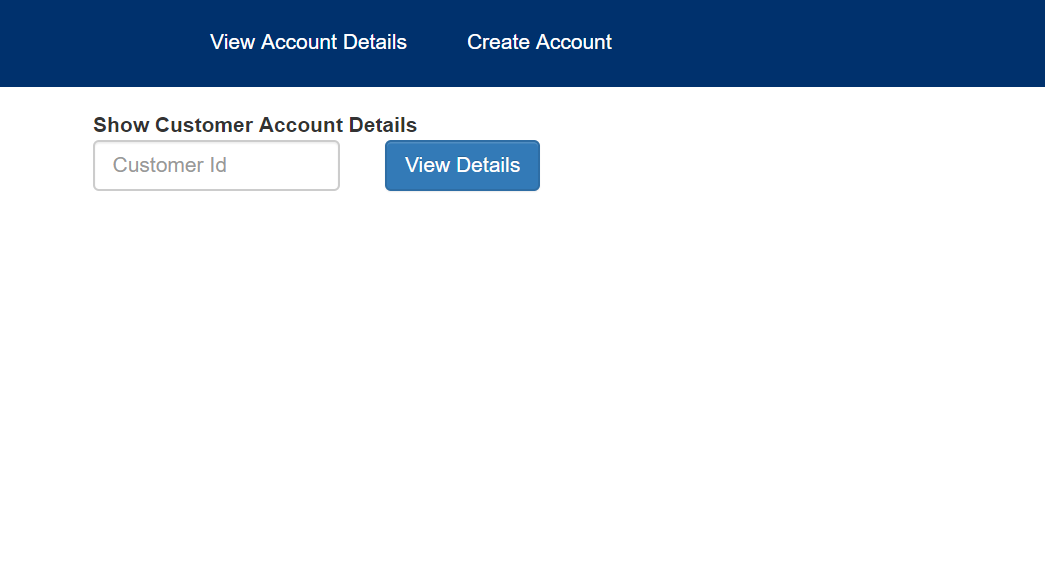
Or else there is already build folder in the repository which is at the below location:

Src/UI/dist/account-service

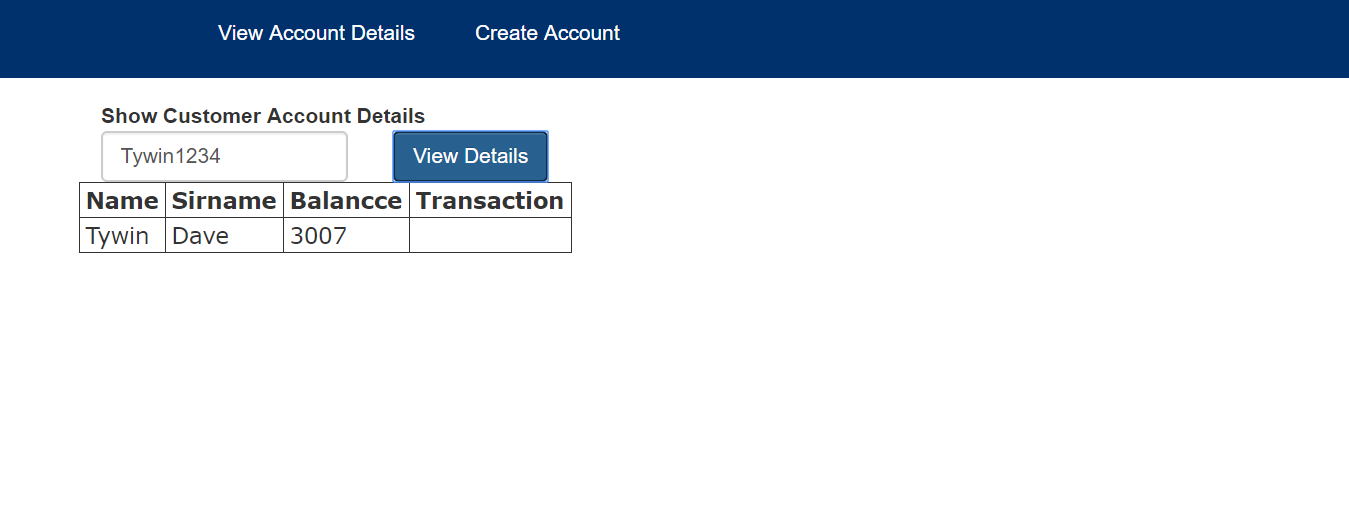
1. Copy this account-service folder and the war file in build/libs/ folder and paste it in the tomcat webapps directory.
2. Start your tomcat.

**Testing the application**

1. Once your application is ready, if you are using Linux system then you can run the script “enroll\_customers.sh” or if you are on Windows then you can use “[AddCustomers.postman\_collection.json](https://github.com/DeepaMotwani/AccountServiceAssignment/blob/master/AddCustomers.postman_collection.json" \o "AddCustomers.postman_collection.json)” collection in the Postman app to add all the existing customers in the database which can be used for further reference.
2. Now that your application is ready, go to the below link in the browser:
   1. **localhost:8080/AccountServiceApp**
3. You will see a login page as shown below:
   1. 
4. Login using credentials as username: admin
   * + - 1. Password: admin
5. You will be redirected to the Dashboard where you can see two tabs:
6. 
7. If you want to view account details, then simply click on “View Account Details” tab



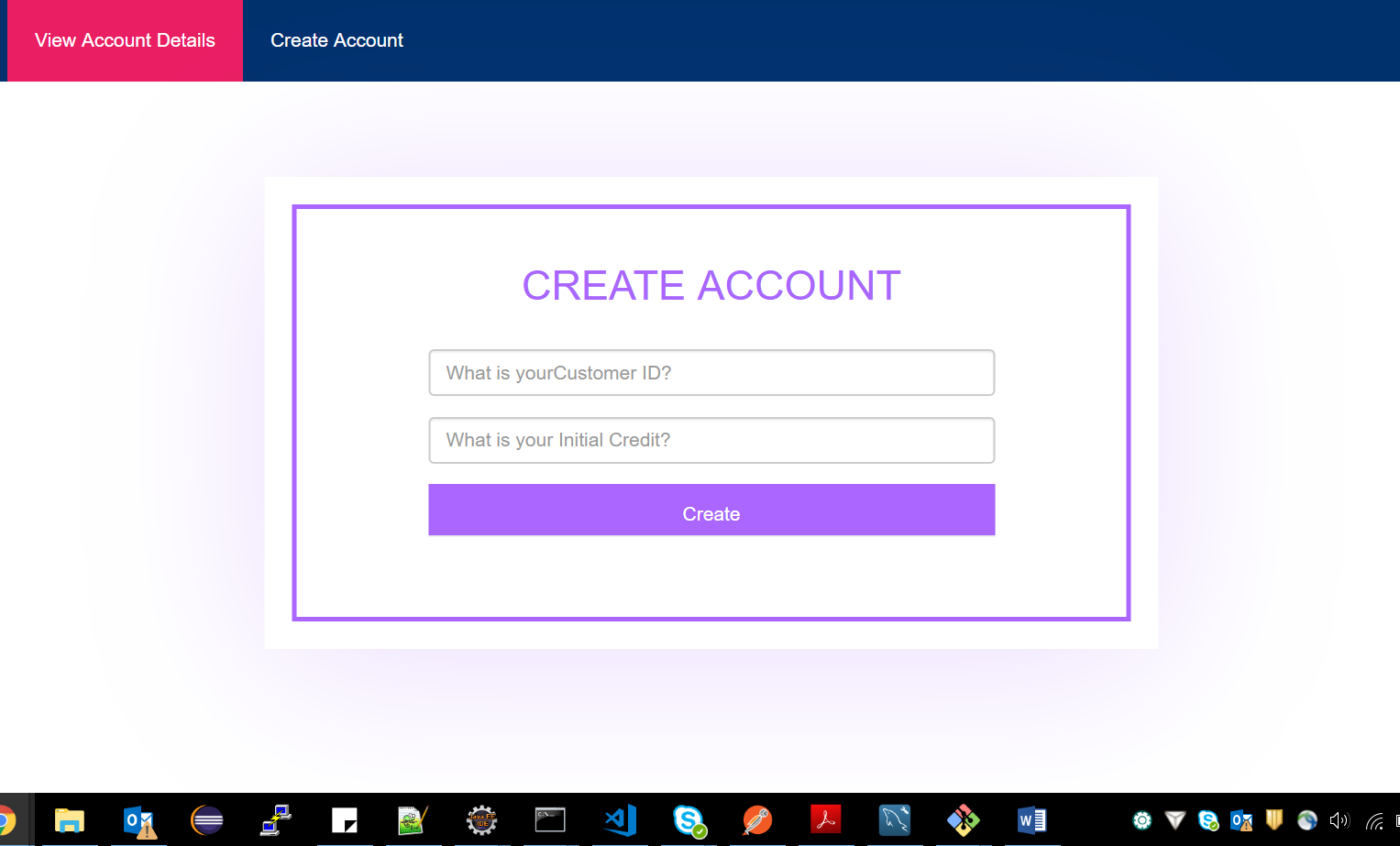
1. Input one of the customerID which you have recently added into the database and click on View Details button.
2. If you input a valid customerId who has not performed any transactions yet, it will give you following output:



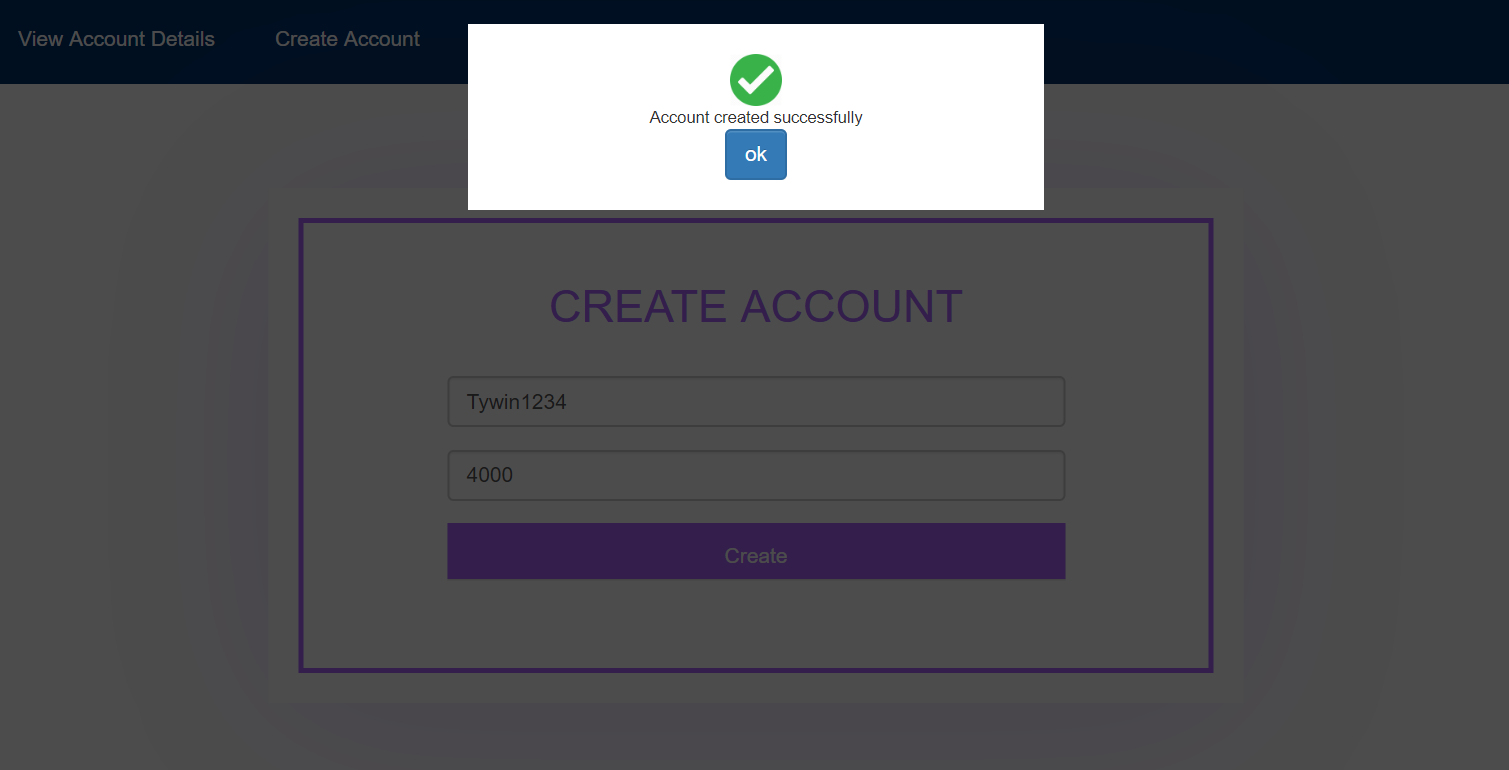
1. If you enter an invalid customerId, it will show you as below:



1. If you want to create a new account of exiting customer, then click on the “Create Account” tab. It will ask you customerId and initial credit if you want to provide.



1. Enter the values and click on Create button, you will get the following output:



1. Now that you have added a new account, auto-generated accountId is created and if you provided the initial credit while creating account, it will show you in the transactions.

