



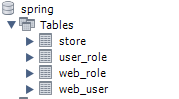
|  |  |
| --- | --- |
| **Student Name/ID Number:** | I Kadek Adi Astawa |
| **Academic Year:** | 2022 |
| **Unit Assessor:** |  |
| **Project Title:** | Assignment5 – Spring Security, Error Handling, and Logging |
| **Issue Date:** | February, 6 2023 |
| **Submission Date:** | February, 9 2023 |
| **Internal Verifier Name:** |  |
| **Date:** |  |

|  |
| --- |
| **Learner declaration** |
| I certify that the work submitted for this assignment is my own and research sources are fully acknowledged.  February, 9 2023  Student signature: Date: |

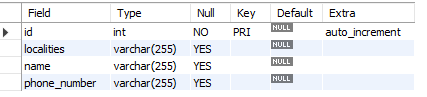
This assignment is about implementing Authentication & Authorization solution for Know-Your-Neighbourhood application.

**Spring Security**

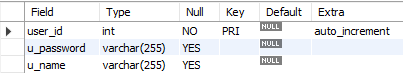
1. Create user, role and user\_role tables in the database.



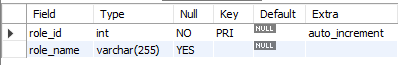
Store Table



User Table



Role Table



User Role Table



1. Add entries into user, role(roleid, role name) and user\_role tables.
   1. Add two entries into role table. One role is to view the store info (VIEW\_STORE) and the other role is to add/modify store info (ADD\_STORE).

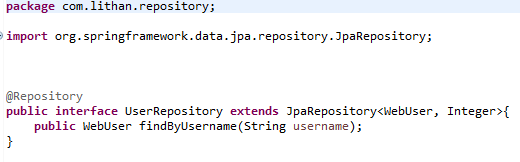


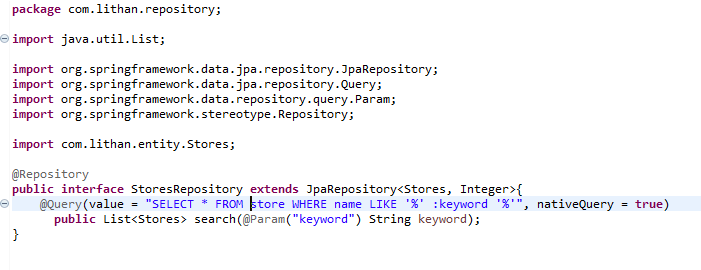
* 1. Add two entries into user table. Let one user have play VIEW\_STORE role, another user play only both VIEW\_STORE & ADD\_STORE roles.





1. Enhance Spring Data JPA enabled version of ‘Know-Your-Neighbourhood’ application to support authentication and authorization



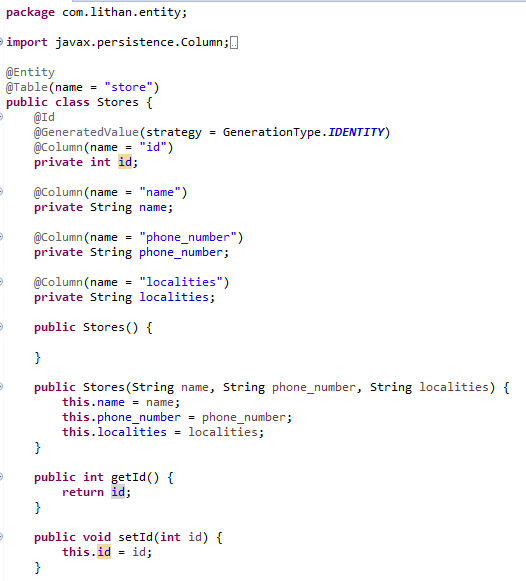


1. **Authenticatio**n:
   1. Create entities for Role and Store.

Role

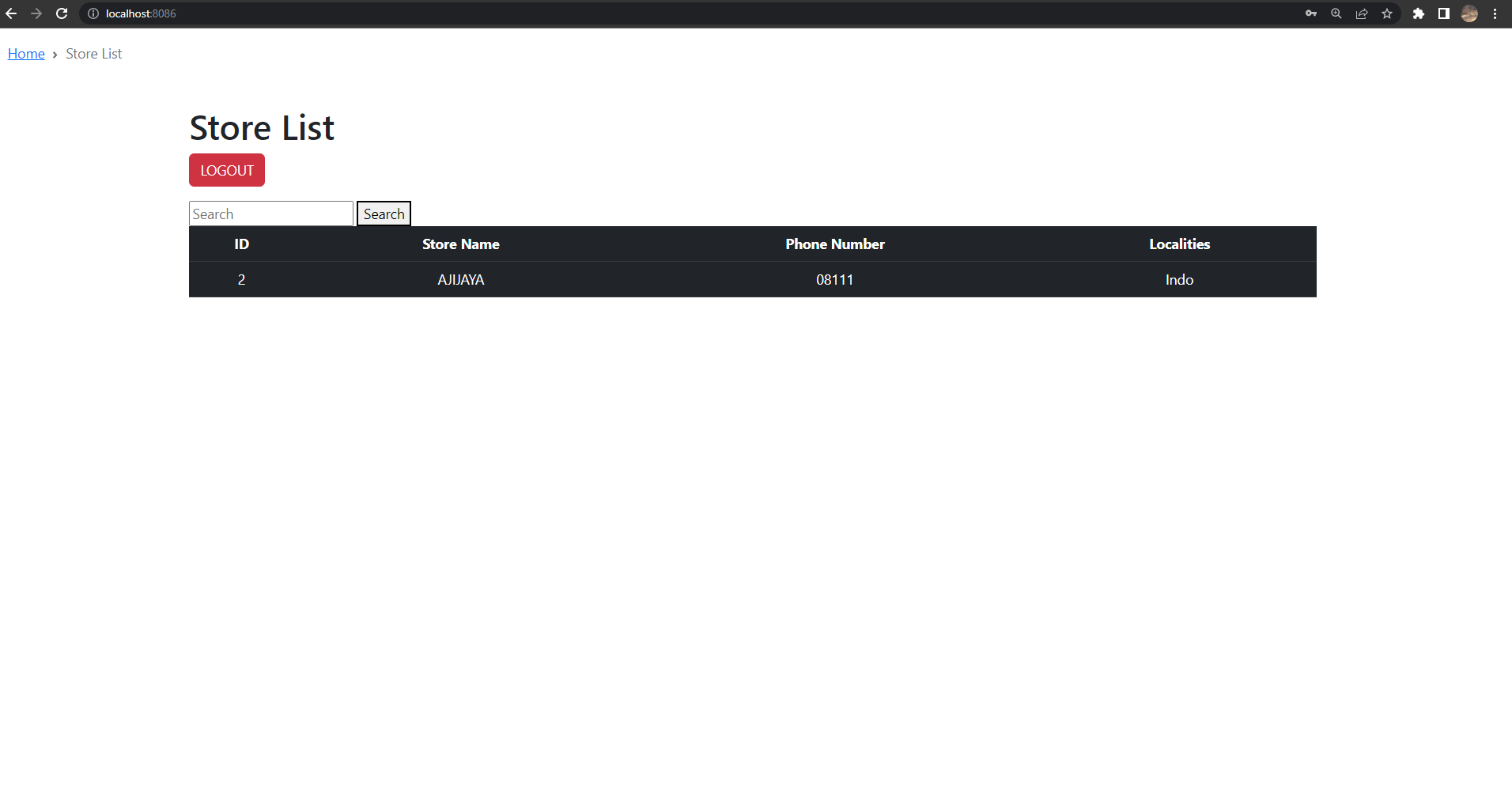


Store

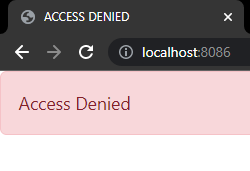


* 1. Create a login page for the application. User should be forced to login before visiting any of the application page.

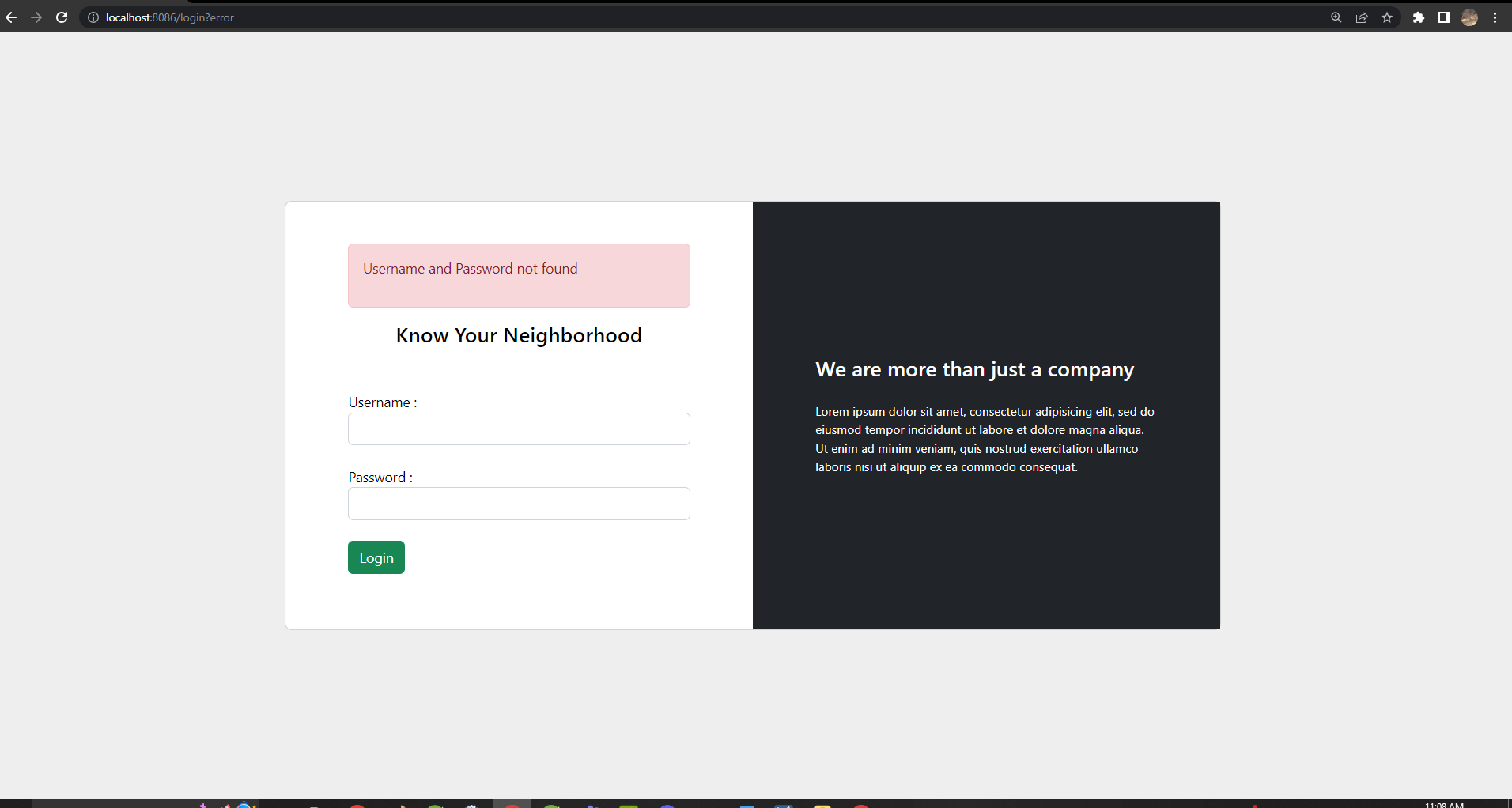
After login\*



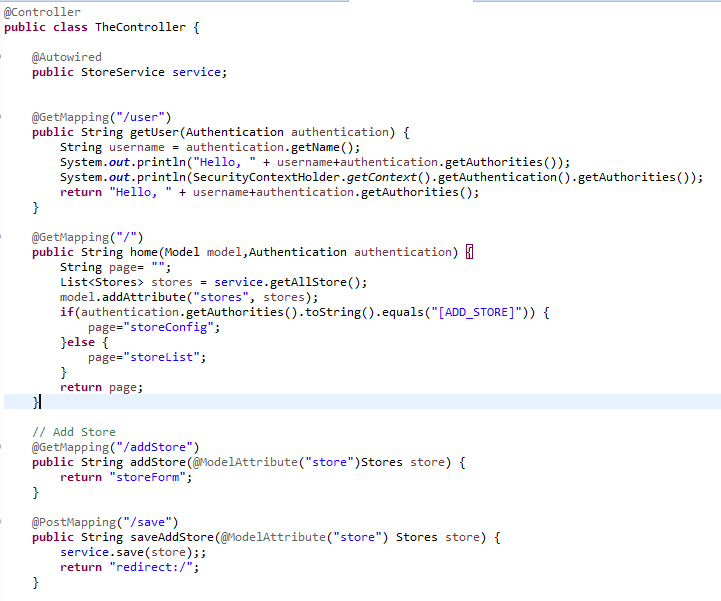
* 1. Authenticate user with the credentials entered against the database entries added in the previous step. 



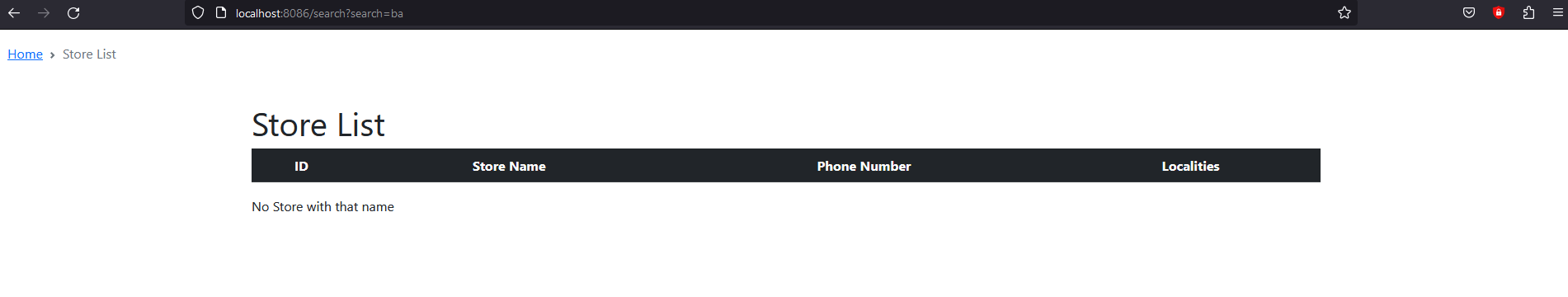
* 1. Show a message if user enters invalid user and password. And then allow him to re-enter the credentials.



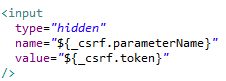
1. **Authorization:** 
   1. Once the user logged in, make sure that only the user with ADD\_STORE role can add/modify store data. The user with VIEW\_STORE data can only view the stores entered.



* 1. If there are no stores in the system, show appropriate message to the user.



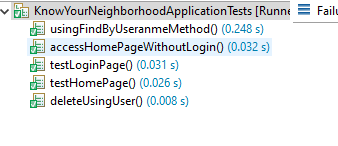
1. Add CSRF security to the application



**Logging**

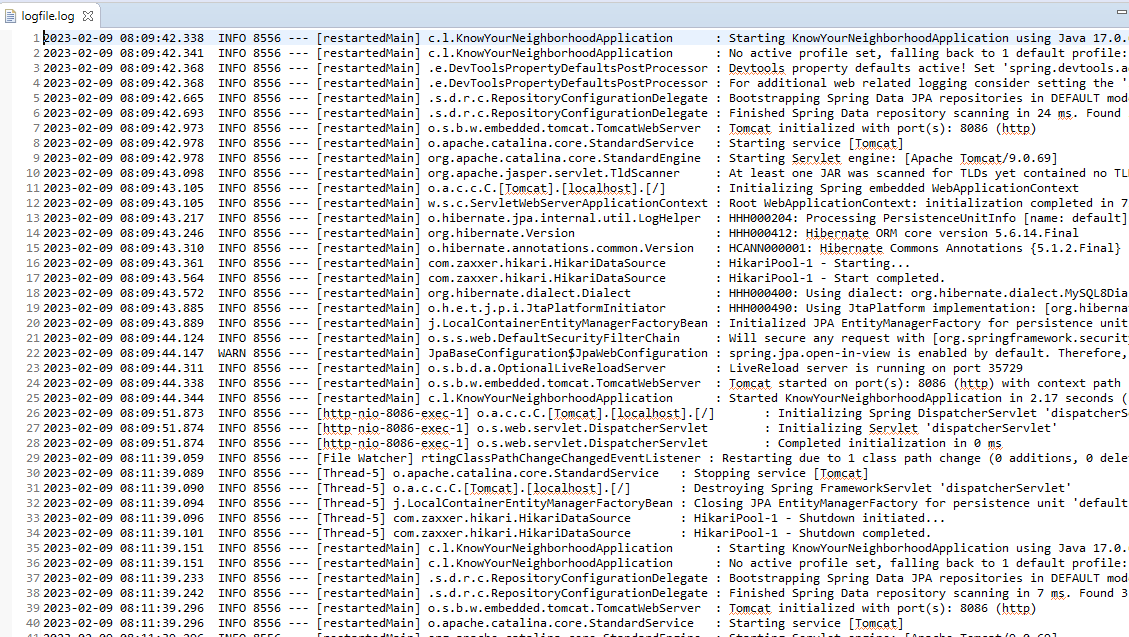
1. Develop 5 unit test cases to validate the role based authorization





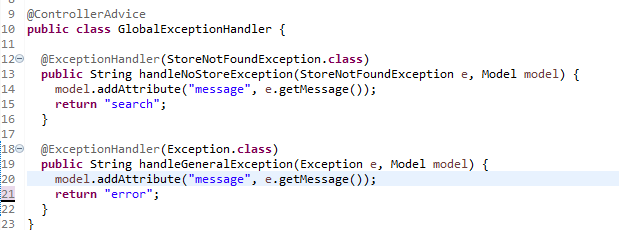
**Logging**

1. Implement the rolling file appended and log at the class level

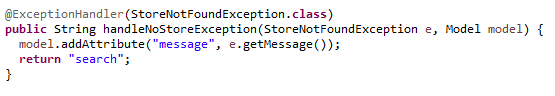


**Error Handling**

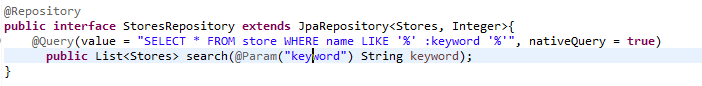
1. Create Global Exception Handler that can handle an exception. Add method to handle run away exception. [Run away exception is a RuntimeException that is not handled by any exception handler].



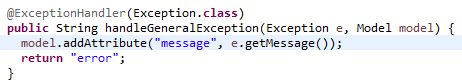
1. Create an exception class StoreNotFoundException.



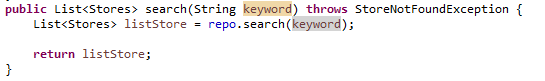
1. Create an end point to search store by email.



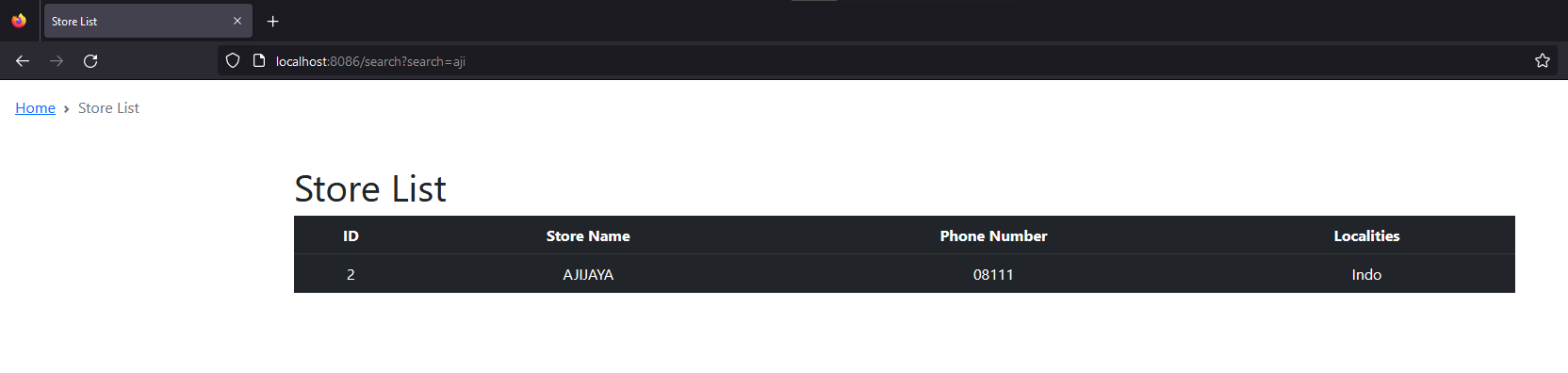
1. Create exception handler specific to the controller.

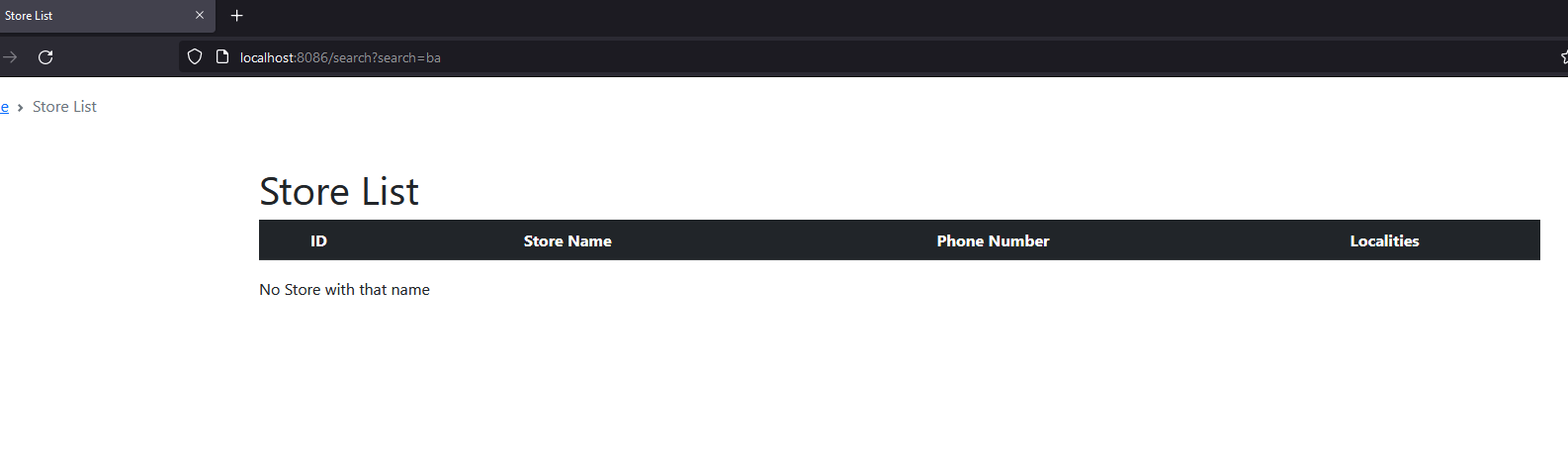


5. Let the service throw StoreNotFoundException if no store is found for given **email.**



1. Let the StoreNotFoundException handler route to view stores page and show the error message in view stores page.





1. Enter the url http://:/stores?email@test.com. This should route to view stores page and show the error message.



