Set up, Configure, and Build Projects

You Already Know

Before we begin, let’s see what you have covered till now:

- Agile
- Git
- SQL
- HTML or CSS
- JavaScript
- Angular
- Protractor

Recap

Agile

An iterative approach to manage the development of a software project

Git and GitHub

A distributed version control system that handles software projects

SQL or MySQL

Relational Database Management System to store data in a structured way using tables

HTML or CSS

Interactive Web Pages

Recap

JavaScript

A programming language for the Web Pages

Angular

A platform and framework by Google to create single page web applications using HTML and TypeScript

Protractor

A testing framework to perform end-to-end testing, run tests against the application running in a real browser, and interact with it

A Day in the Life of an Automation Test Engineer

As an Automation Test Engineer, our key role is to test both client and server software with the latest test automation tools.

We shall be testing food delivery application built in Angular, Node as the front end with Spring Boot, Java, and MySQL/MongoDB as the backend.

We have already cloned and synced the project with GitHub and also have our database and tables set up.

Moving forward, we will configure the Angular Projects and Java Backend Project for dependencies or database configurations. We will finally build the projects and run them so as to prepare for testing.

Here, we will also configure protractor for the Angular Projects.

Learning Objectives

By the end of this lesson, you will be able to:

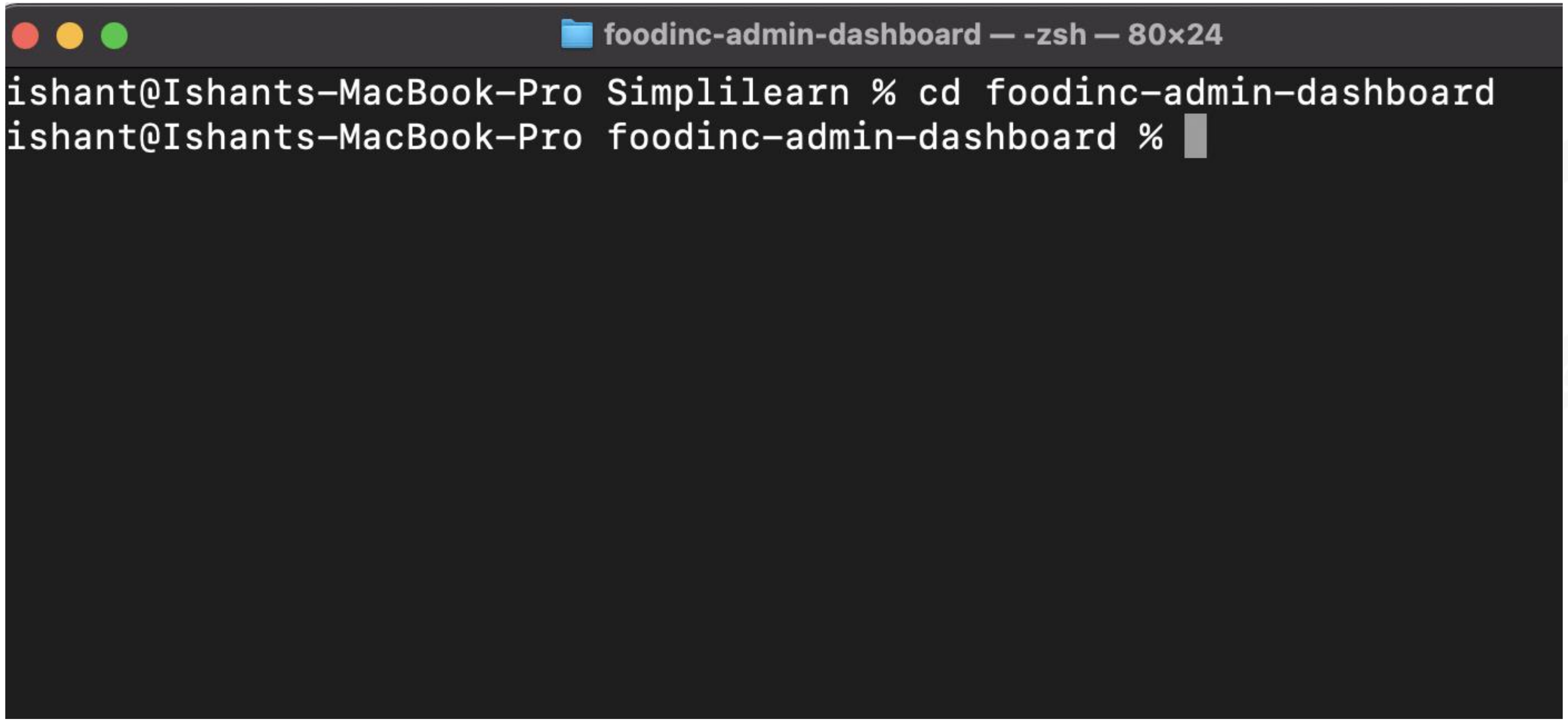
- Install dependencies for Angular project using npm install
- Configure Protractor for Angular projects
- Set up database configuration in Java backend project
- Build and execute the projects to start testing

Task 1: Set up Angular Admin Project

```
#include <iostream>
#include <cstdlib>
using namespace std;
int main()
{
    cout << "Hello, world!" << endl;
    return 0;
}
```

Configure Dependencies for Angular Admin Project

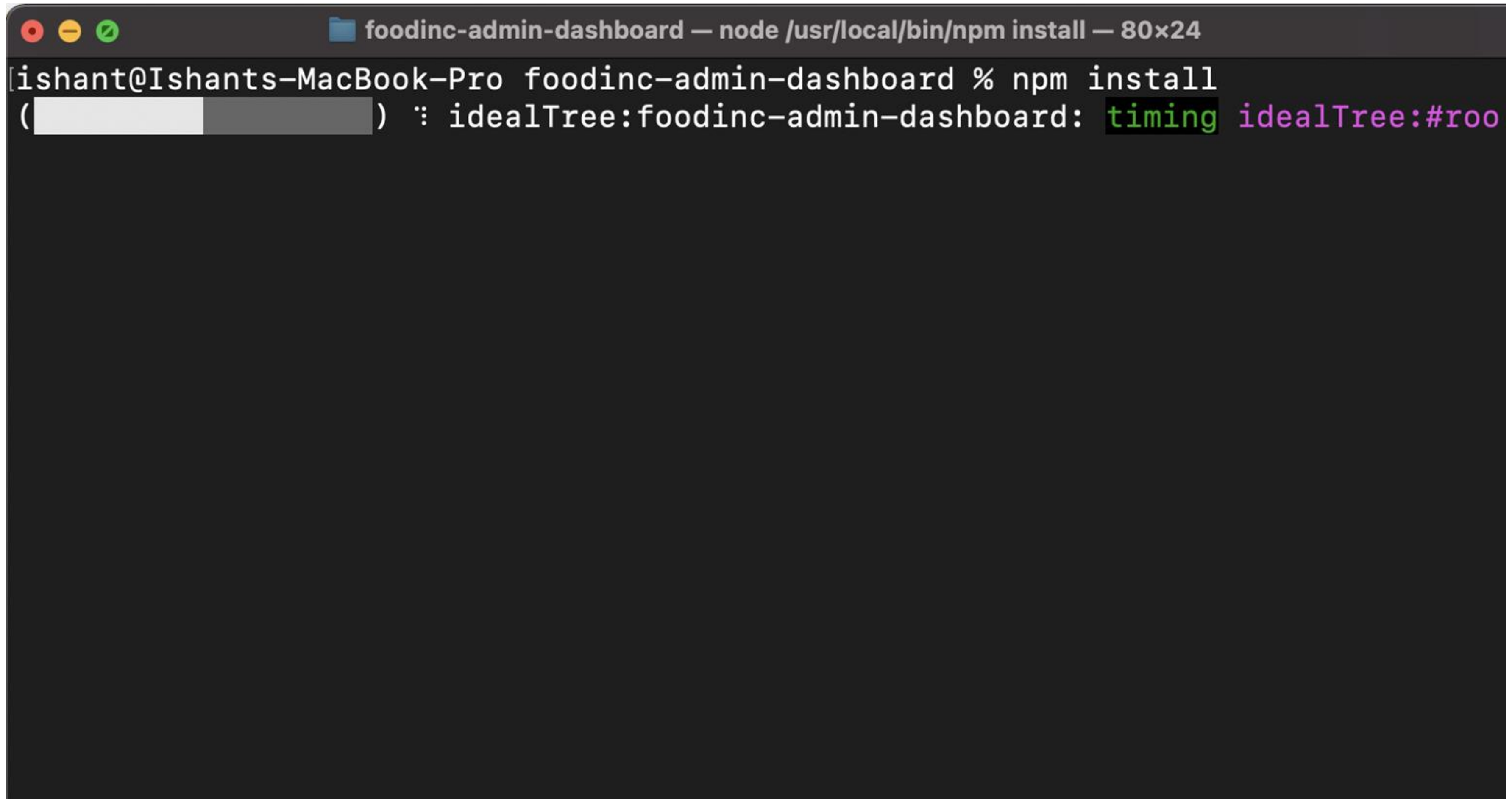
Open the Terminal Shell and change the director of your project



```
ishant@Ishants-MacBook-Pro Simplilearn % cd foodinc-admin-dashboard
ishant@Ishants-MacBook-Pro foodinc-admin-dashboard %
```

Configure Dependencies for Angular Admin Project

Now, you must execute the **npm install** command.
In case you get some issues, you can also try **npm install --legacy-peer-deps**.

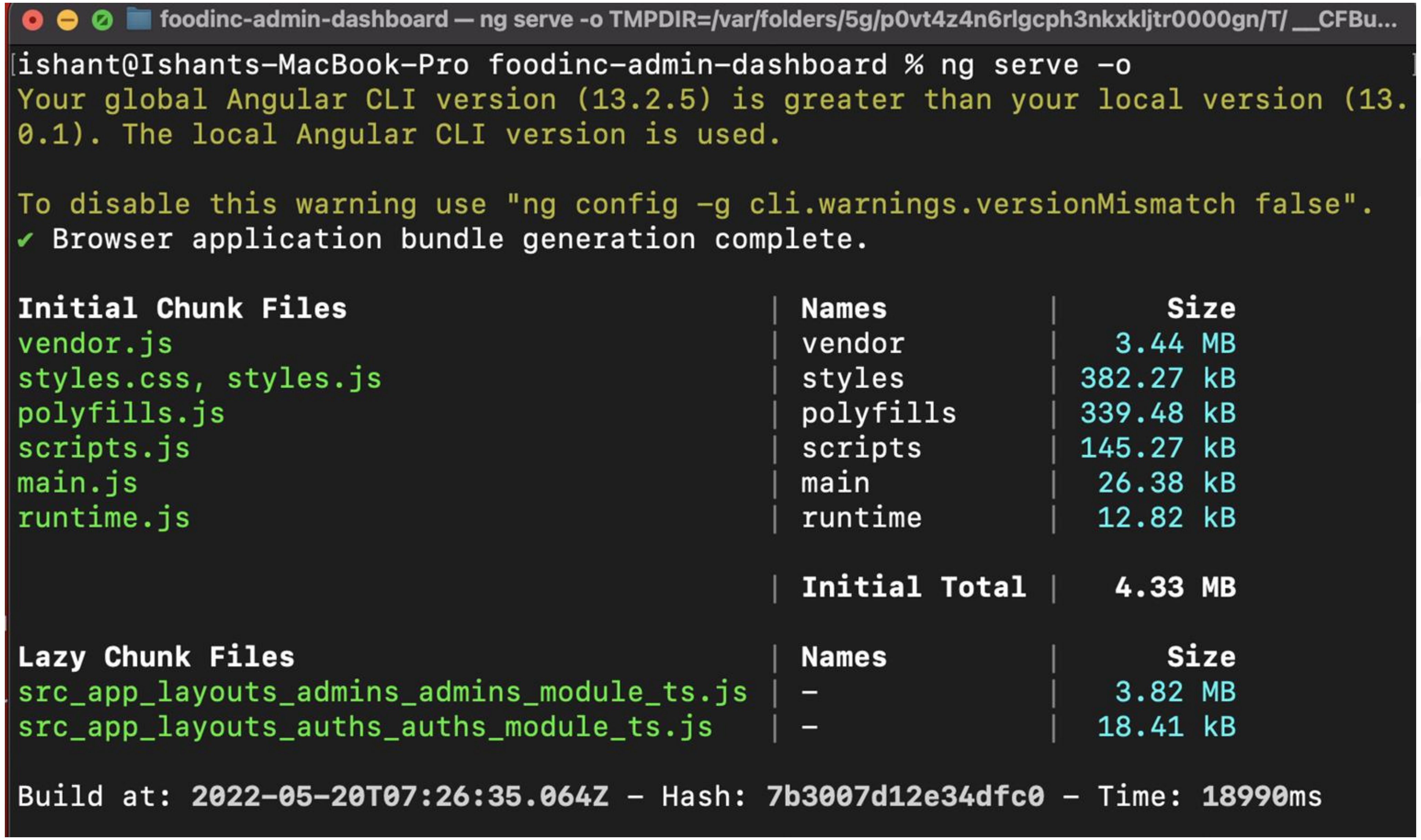


foodinc-admin-dashboard — node /usr/local/bin/npm install — 80×24

```
ishant@Ishants-MacBook-Pro foodinc-admin-dashboard % npm install
(                    ) ⠹ idealTree:foodinc-admin-dashboard: timing idealTree:#roo
```

Build and Execute Angular Admin Project

Since the dependencies such as node modules are installed , you can now execute **ng serve -o** command to build and execute the project.

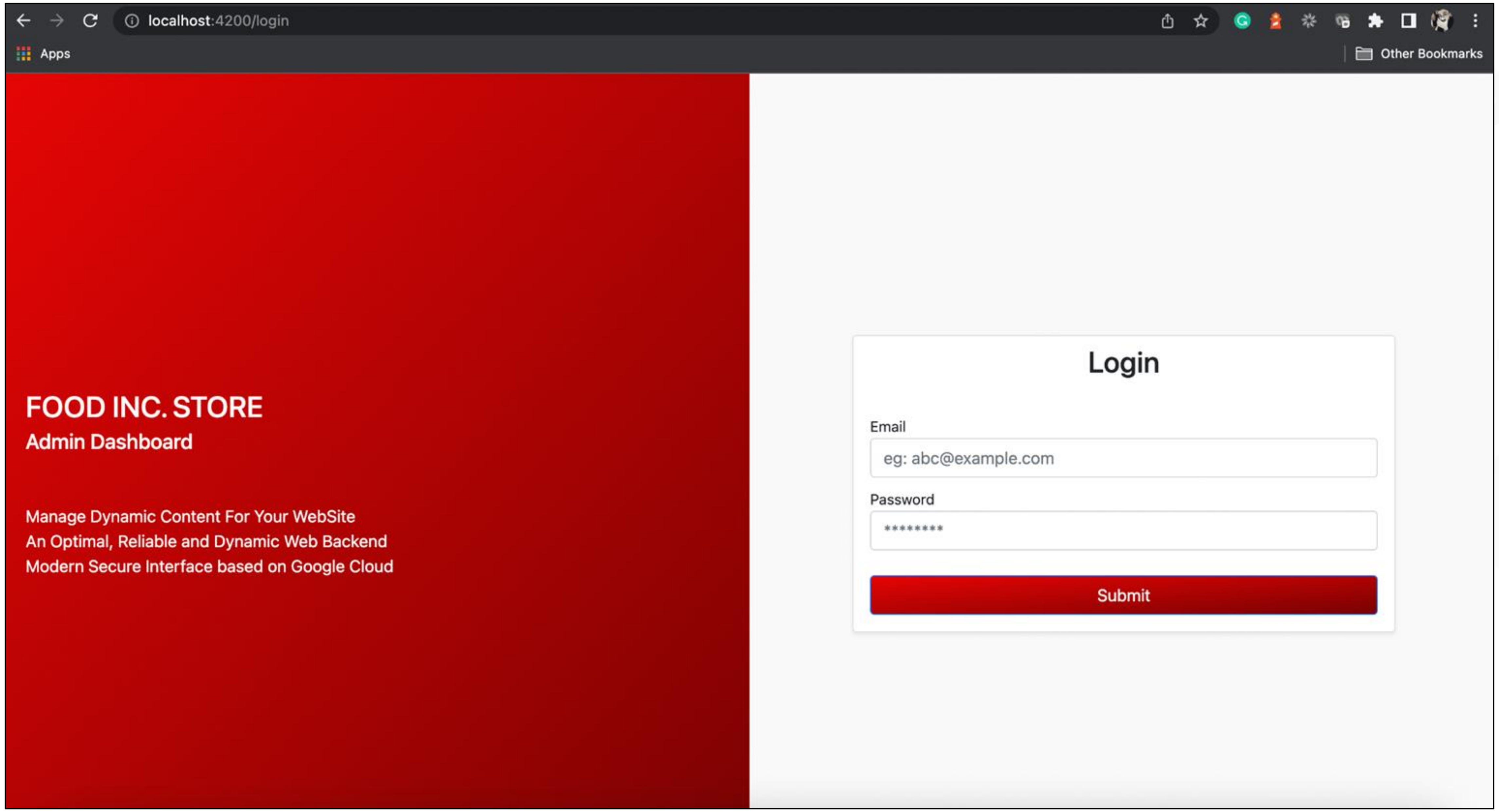


foodinc-admin-dashboard — ng serve -o TMPDIR=/var/folders/5g/p0vt4z4n6rlgcph3nkxkljtr0000gn/T/ __CFBu...

```
ishant@Ishants-MacBook-Pro foodinc-admin-dashboard % ng serve -o
Your global Angular CLI version (13.2.5) is greater than your local version (13.
0.1). The local Angular CLI version is used.

To disable this warning use "ng config -g cli.warnings.versionMismatch false".
✔ Browser application bundle generation complete.

Initial Chunk Files                          | Names          |      Size
vendor.js                                    | vendor         |   3.44 MB
styles.css, styles.js                        | styles         | 382.27 kB
polyfills.js                                 | polyfills      | 339.48 kB
scripts.js                                   | scripts        | 145.27 kB
main.js                                      | main           |  26.38 kB
runtime.js                                   | runtime        |  12.82 kB

                                             | Initial Total  |   4.33 MB

Lazy Chunk Files                             | Names          |      Size
src_app_layouts_admins_admins_module_ts.js   | -              |   3.82 MB
src_app_layouts_auths_auths_module_ts.js     | -              |  18.41 kB

Build at: 2022-05-20T07:26:35.064Z - Hash: 7b3007d12e34dfc0 - Time: 18990ms
```

Build and Execute Angular Admin Project

Your project is up and running fine on localhost port 4200, with the initial UI screen to login:



FOOD INC. STORE

Admin Dashboard

Manage Dynamic Content For Your WebSite
An Optimal, Reliable and Dynamic Web Backend
Modern Secure Interface based on Google Cloud

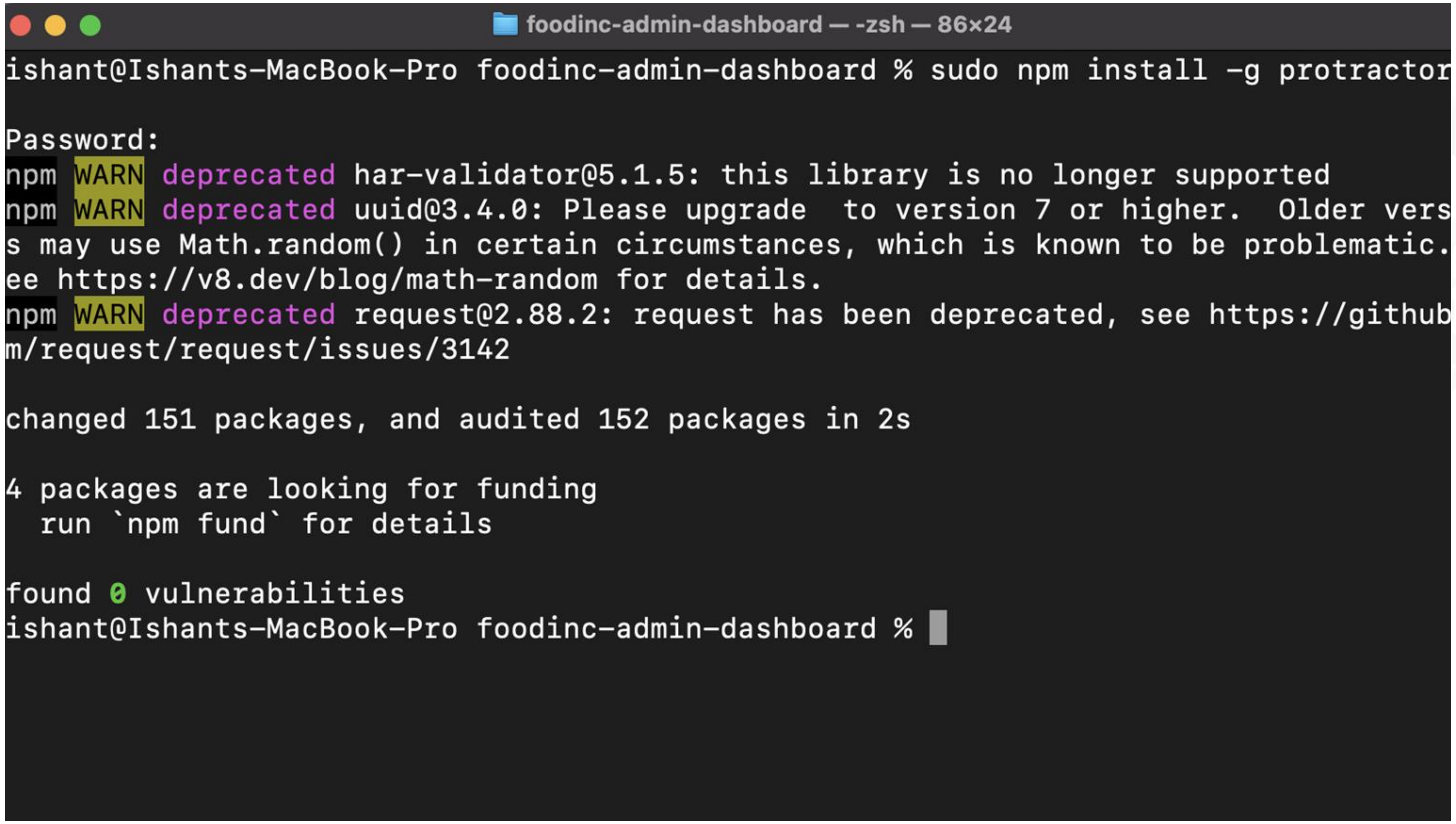
Login

Email: eg: abc@example.com

Password: ********

Configure Protractor

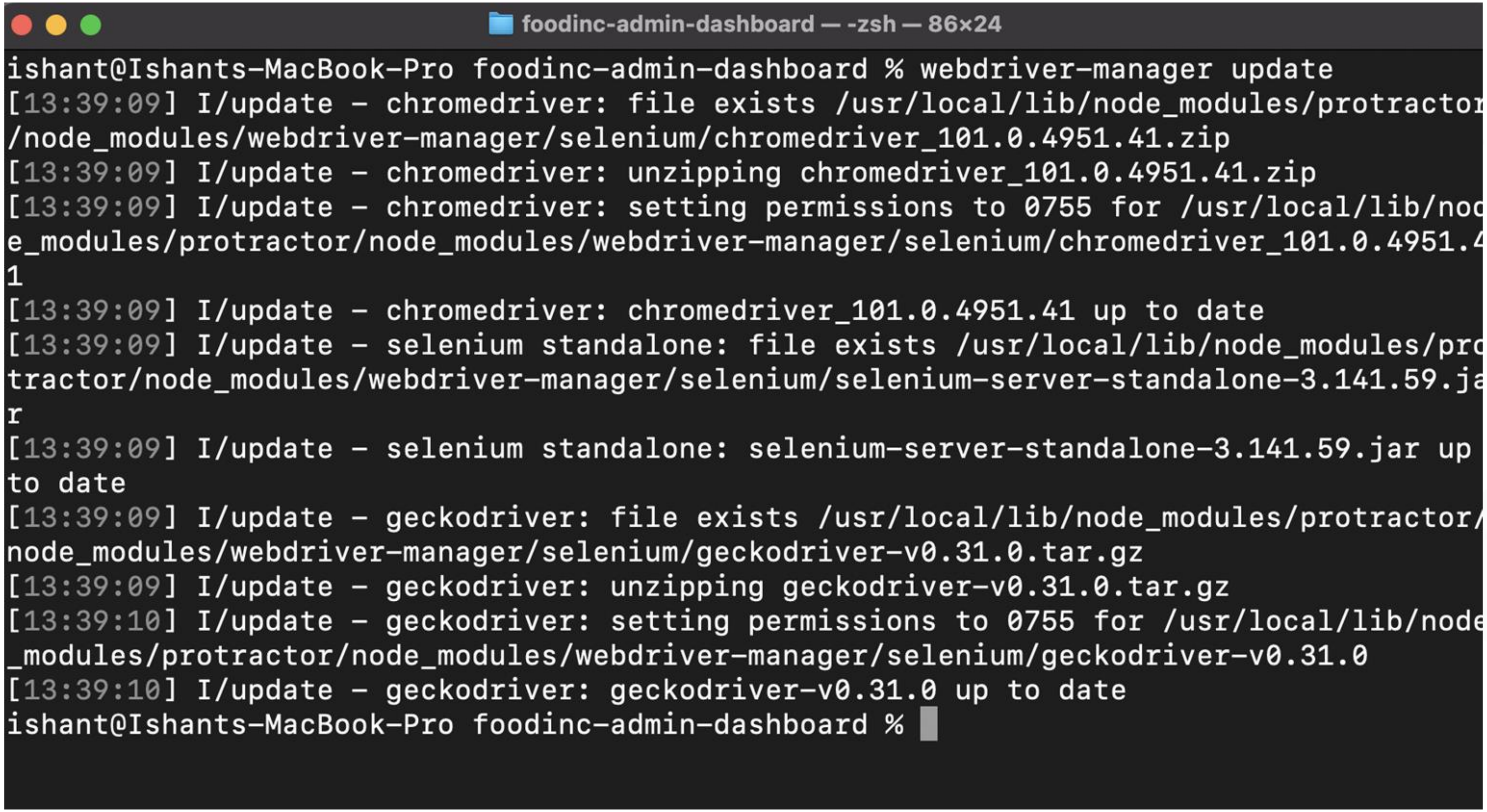
Now, execute the command **npm install -g protractor** to add dependency for Protractor. You can also use sudo, in case you face EACCESS errors**.**



```
ishant@Ishants-MacBook-Pro foodinc-admin-dashboard % sudo npm install -g protractor

Password:
npm WARN deprecated har-validator@5.1.5: this library is no longer supported
npm WARN deprecated uuid@3.4.0: Please upgrade  to version 7 or higher.  Older vers
s may use Math.random() in certain circumstances, which is known to be problematic.
ee https://v8.dev/blog/math-random for details.
npm WARN deprecated request@2.88.2: request has been deprecated, see https://github
m/request/request/issues/3142

changed 151 packages, and audited 152 packages in 2s

4 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
ishant@Ishants-MacBook-Pro foodinc-admin-dashboard %
```

Configure WebDriver

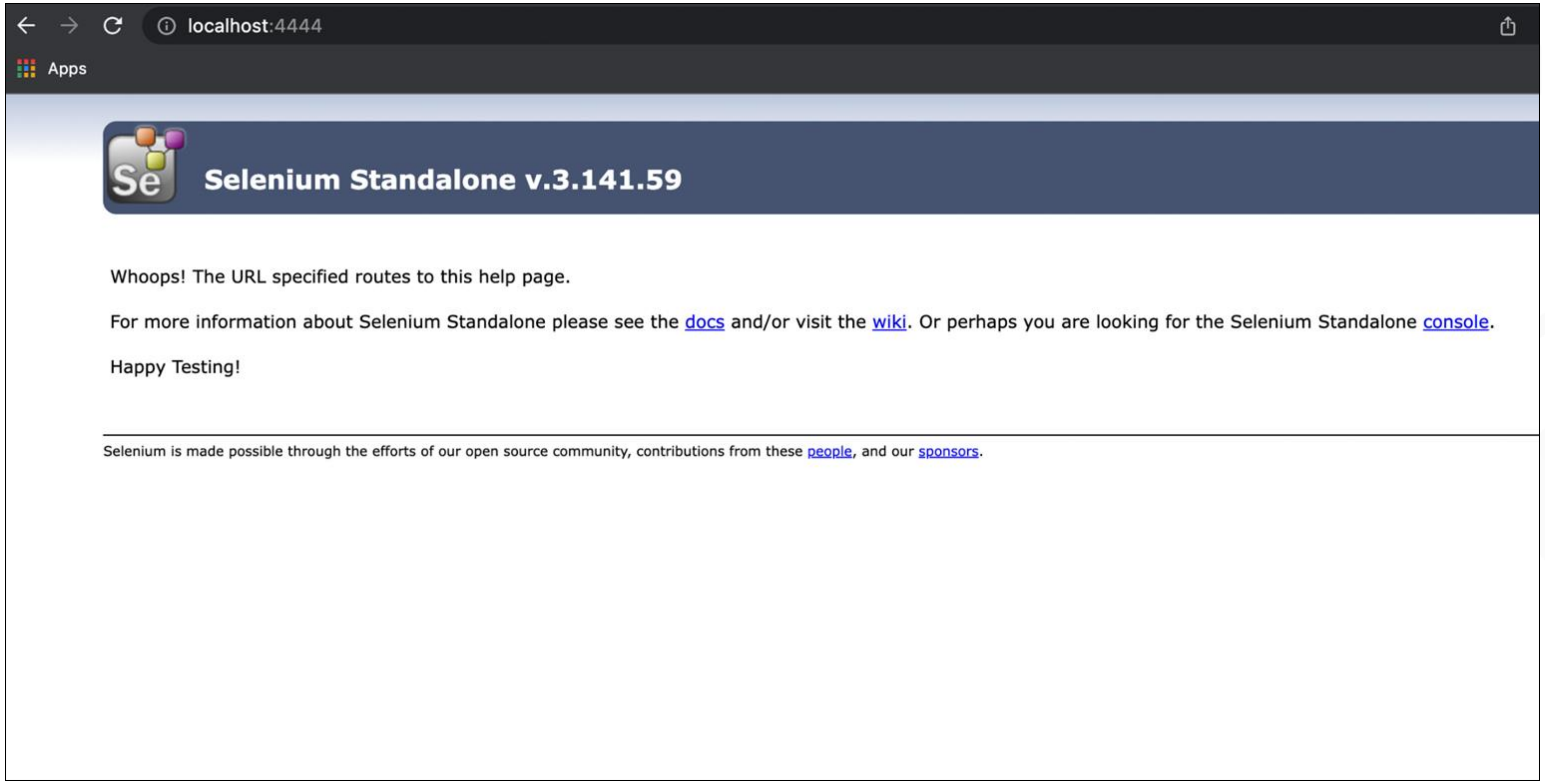
Once installed, you can run **webdriver-manager update** in order to update the webdriver-manager.



```
ishant@Ishants-MacBook-Pro foodinc-admin-dashboard % webdriver-manager update
[13:39:09] I/update - chromedriver: file exists /usr/local/lib/node_modules/protractor
/node_modules/webdriver-manager/selenium/chromedriver_101.0.4951.41.zip
[13:39:09] I/update - chromedriver: unzipping chromedriver_101.0.4951.41.zip
[13:39:09] I/update - chromedriver: setting permissions to 0755 for /usr/local/lib/noc
e_modules/protractor/node_modules/webdriver-manager/selenium/chromedriver_101.0.4951.4
1
[13:39:09] I/update - chromedriver: chromedriver_101.0.4951.41 up to date
[13:39:09] I/update - selenium standalone: file exists /usr/local/lib/node_modules/prc
tractor/node_modules/webdriver-manager/selenium/selenium-server-standalone-3.141.59.ja
r
[13:39:09] I/update - selenium standalone: selenium-server-standalone-3.141.59.jar up
to date
[13:39:09] I/update - geckodriver: file exists /usr/local/lib/node_modules/protractor/
node_modules/webdriver-manager/selenium/geckodriver-v0.31.0.tar.gz
[13:39:09] I/update - geckodriver: unzipping geckodriver-v0.31.0.tar.gz
[13:39:10] I/update - geckodriver: setting permissions to 0755 for /usr/local/lib/node
_modules/protractor/node_modules/webdriver-manager/selenium/geckodriver-v0.31.0
[13:39:10] I/update - geckodriver: geckodriver-v0.31.0 up to date
ishant@Ishants-MacBook-Pro foodinc-admin-dashboard %
```

Configure WebDriver

Now, you can start up a server by executing the command **webdriver-manager start** and open the localhost port 4444 to check the webdriver.



Selenium Standalone v.3.141.59

Whoops! The URL specified routes to this help page.

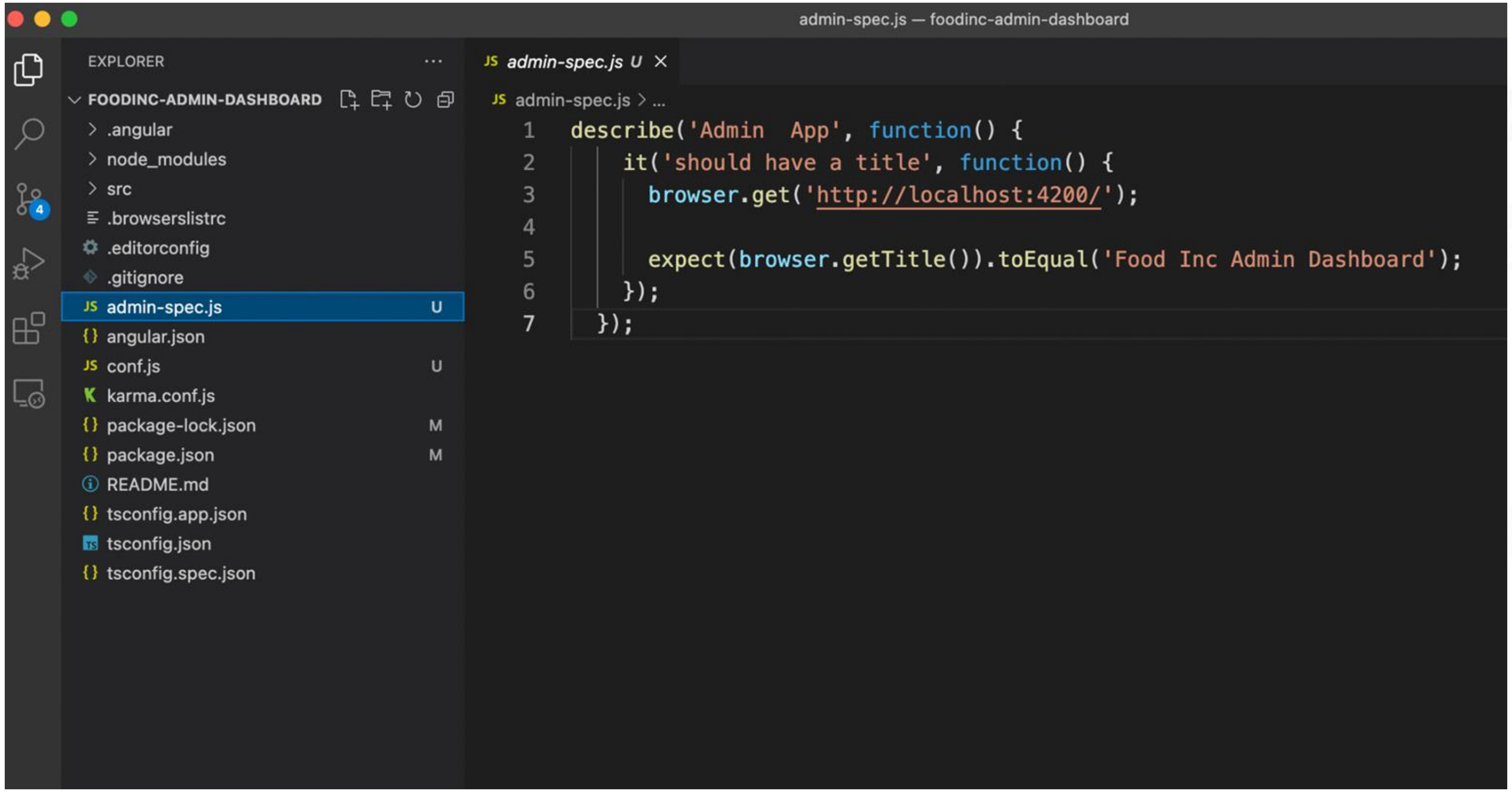
For more information about Selenium Standalone please see the docs and/or visit the wiki. Or perhaps you are looking for the Selenium Standalone console.

Happy Testing!

Selenium is made possible through the efforts of our open source community, contributions from these people, and our sponsors.

Set up Spec and Config Files

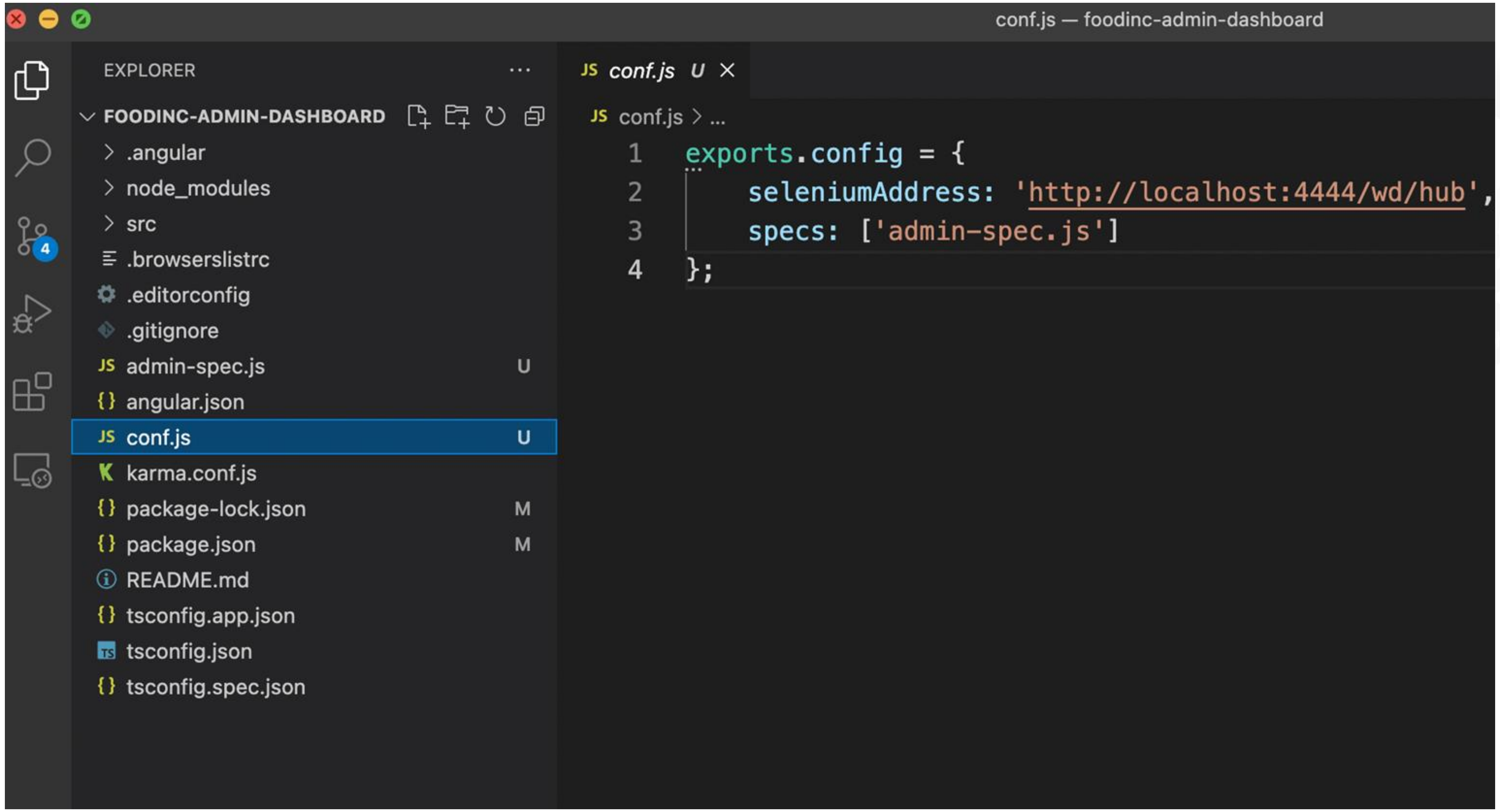
You can create a spec file named admin-spec.js and write a small test case to validate the home page title.



```
describe('Admin  App', function() {
    it('should have a title', function() {
      browser.get('http://localhost:4200/');

      expect(browser.getTitle()).toEqual('Food Inc Admin Dashboard');
    });
  });
```

Set up Spec and Config Files

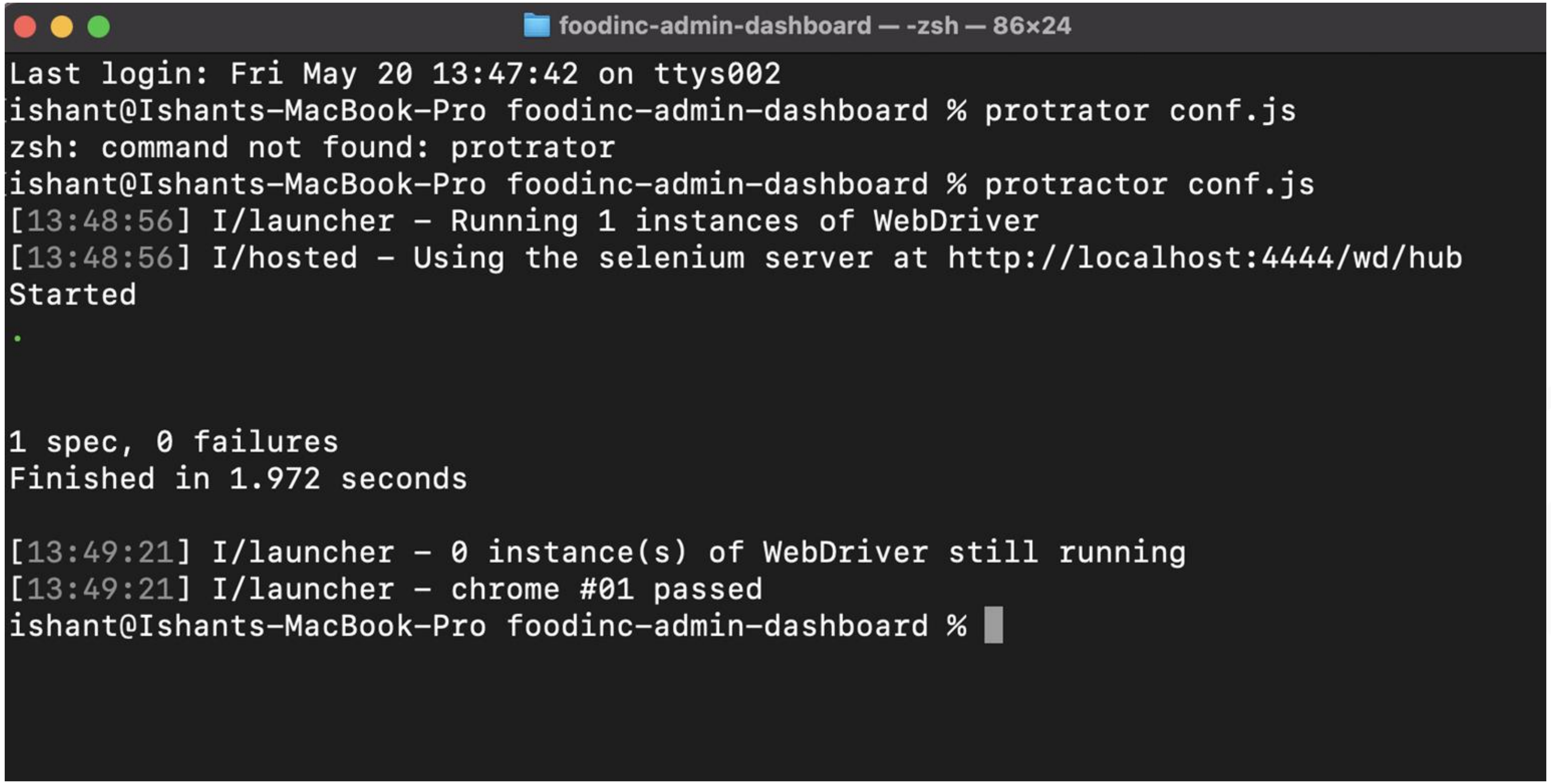
In order to configure protractor, we need a configuration file. You must create a new **conf.js** where the configuration is specified.



```
exports.config = {
    seleniumAddress: 'http://localhost:4444/wd/hub',
    specs: ['admin-spec.js']
};
```

Execute Protractor

Now, let us run the protractor conf.js on the terminal to validate if the protractor works fine.

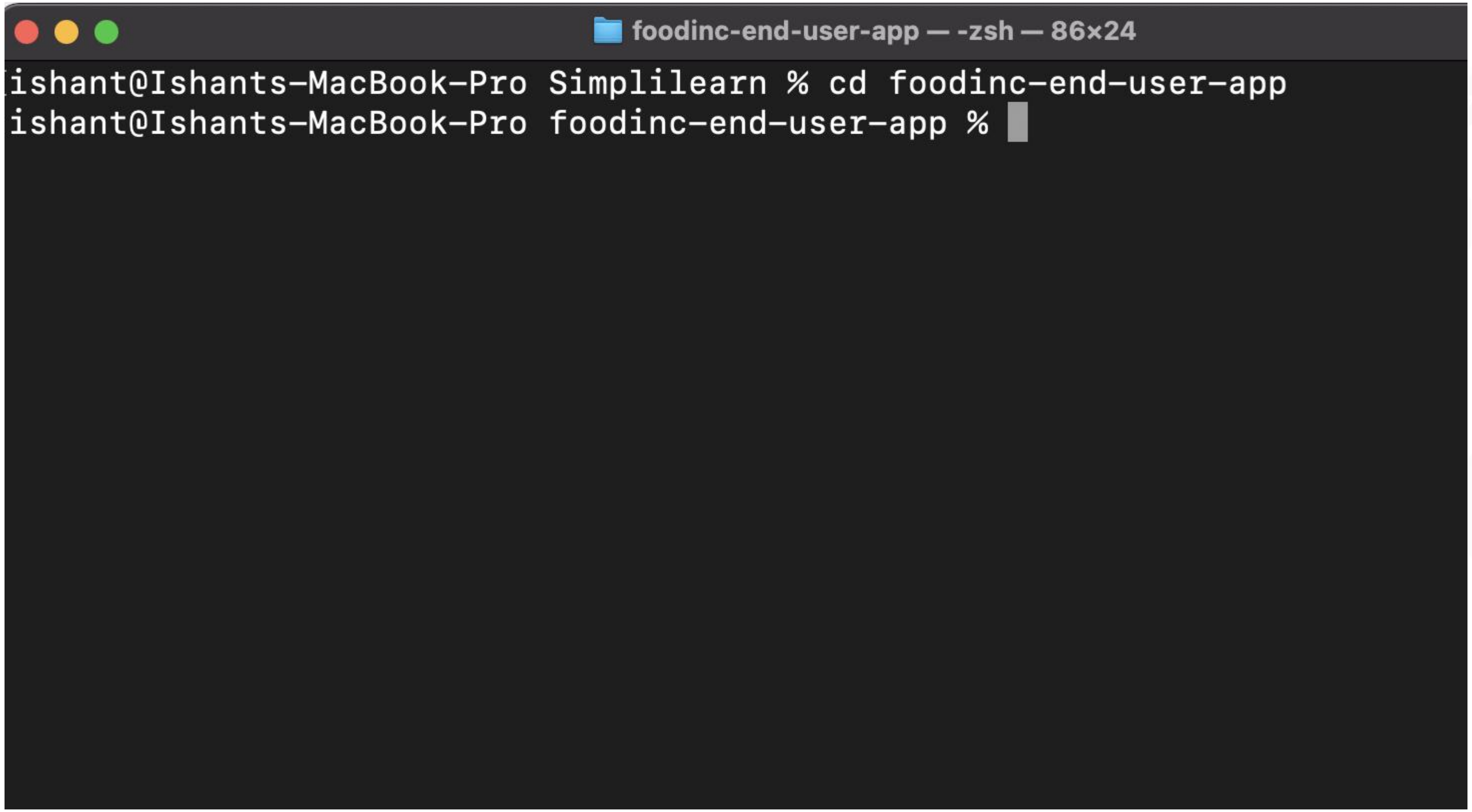


```
Last login: Fri May 20 13:47:42 on ttys002
ishant@Ishants-MacBook-Pro foodinc-admin-dashboard % protrator conf.js
zsh: command not found: protrator
ishant@Ishants-MacBook-Pro foodinc-admin-dashboard % protractor conf.js
[13:48:56] I/launcher - Running 1 instances of WebDriver
[13:48:56] I/hosted - Using the selenium server at http://localhost:4444/wd/hub
Started
.


1 spec, 0 failures
Finished in 1.972 seconds

[13:49:21] I/launcher - 0 instance(s) of WebDriver still running
[13:49:21] I/launcher - chrome #01 passed
ishant@Ishants-MacBook-Pro foodinc-admin-dashboard %
```

Task 2: Set up Angular End User Project

Configure Dependencies for Angular End User Project

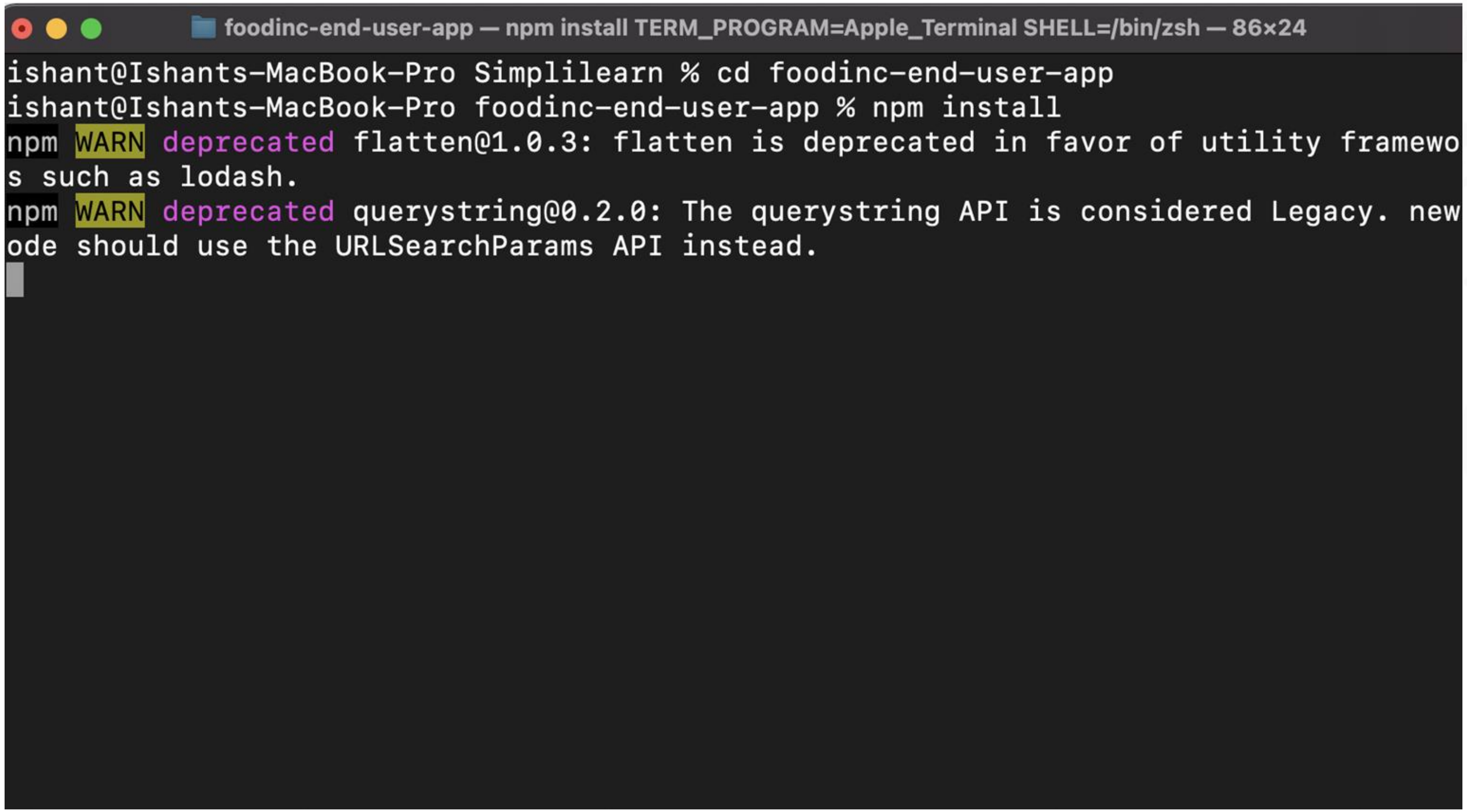
Open the Terminal Shell and change the director of your project:



```
ishant@Ishants-MacBook-Pro Simplilearn % cd foodinc-end-user-app
ishant@Ishants-MacBook-Pro foodinc-end-user-app % 
```

Configure Dependencies for Angular End User Project

Now, you must execute the **npm install** command.
In case you get some issues, you can also try **npm install --legacy-peer-deps**.

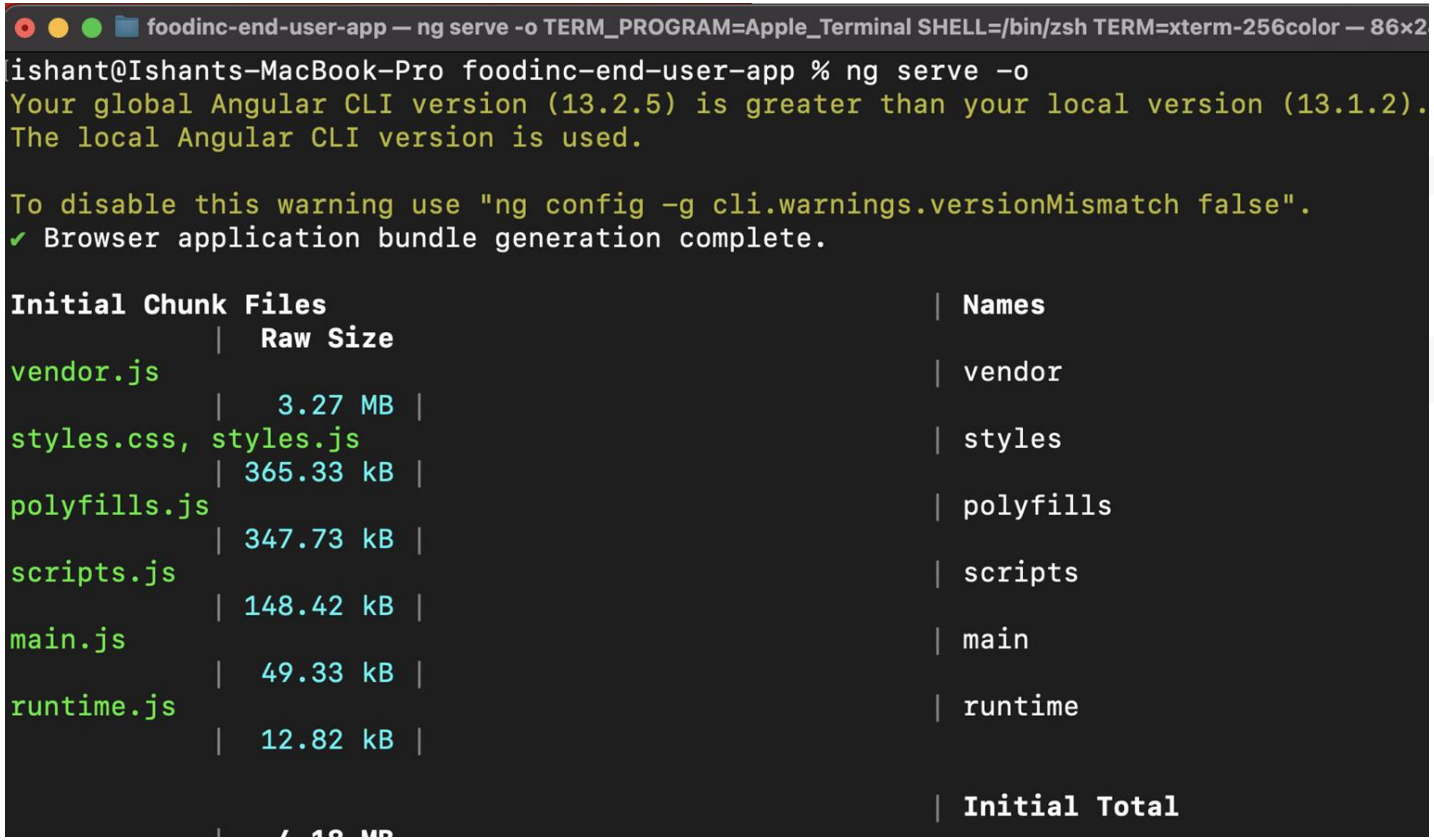


foodinc-end-user-app — npm install TERM_PROGRAM=Apple_Terminal SHELL=/bin/zsh — 86×24

```
ishant@Ishants-MacBook-Pro Simplilearn % cd foodinc-end-user-app
ishant@Ishants-MacBook-Pro foodinc-end-user-app % npm install
npm WARN deprecated flatten@1.0.3: flatten is deprecated in favor of utility framewo
s such as lodash.
npm WARN deprecated querystring@0.2.0: The querystring API is considered Legacy. new
ode should use the URLSearchParams API instead.
```

Build and execute Angular End User Project

Since the dependencies are installed i.e. node modules, you can now execute **ng serve -o** command to build and execute the project

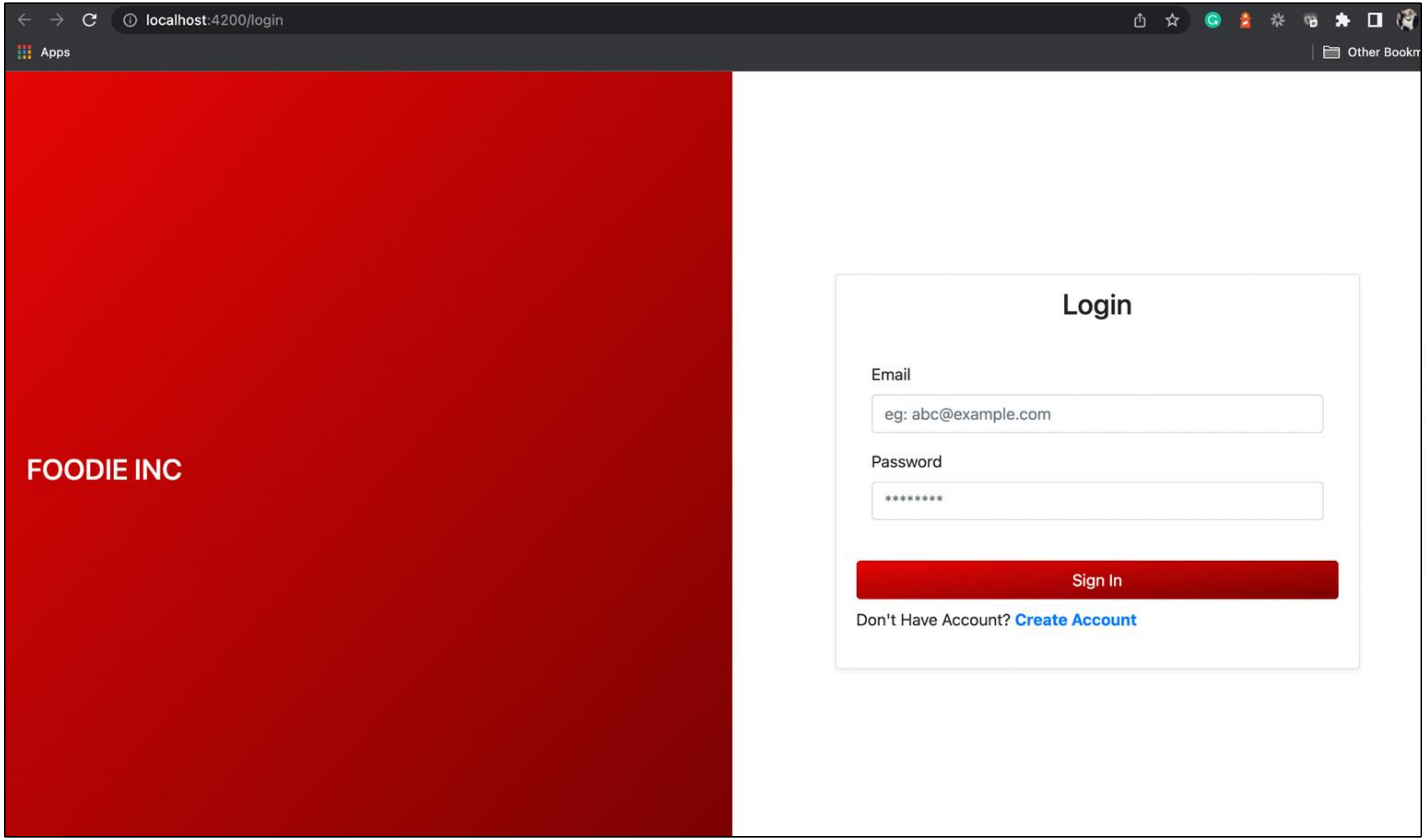


foodinc-end-user-app — ng serve -o TERM_PROGRAM=Apple_Terminal SHELL=/bin/zsh TERM=xterm-256color — 86×2

```
ishant@Ishants-MacBook-Pro foodinc-end-user-app % ng serve -o
Your global Angular CLI version (13.2.5) is greater than your local version (13.1.2).
The local Angular CLI version is used.

To disable this warning use "ng config -g cli.warnings.versionMismatch false".
✔ Browser application bundle generation complete.

Initial Chunk Files                                    | Names
            |  Raw Size
vendor.js                                              | vendor
            |   3.27 MB |
styles.css, styles.js                                  | styles
            | 365.33 kB |
polyfills.js                                           | polyfills
            | 347.73 kB |
scripts.js                                             | scripts
            | 148.42 kB |
main.js                                                | main
            |  49.33 kB |
runtime.js                                             | runtime
            |  12.82 kB |

                                                       | Initial Total
            |   4.18 MB
```

Build and execute Angular End User Project

Your project is up and running fine on localhost port 4200, with initial UI screen to login:



localhost:4200/login

Apps

Other Bookm

FOODIE INC

Login

Email

eg: abc@example.com

Password

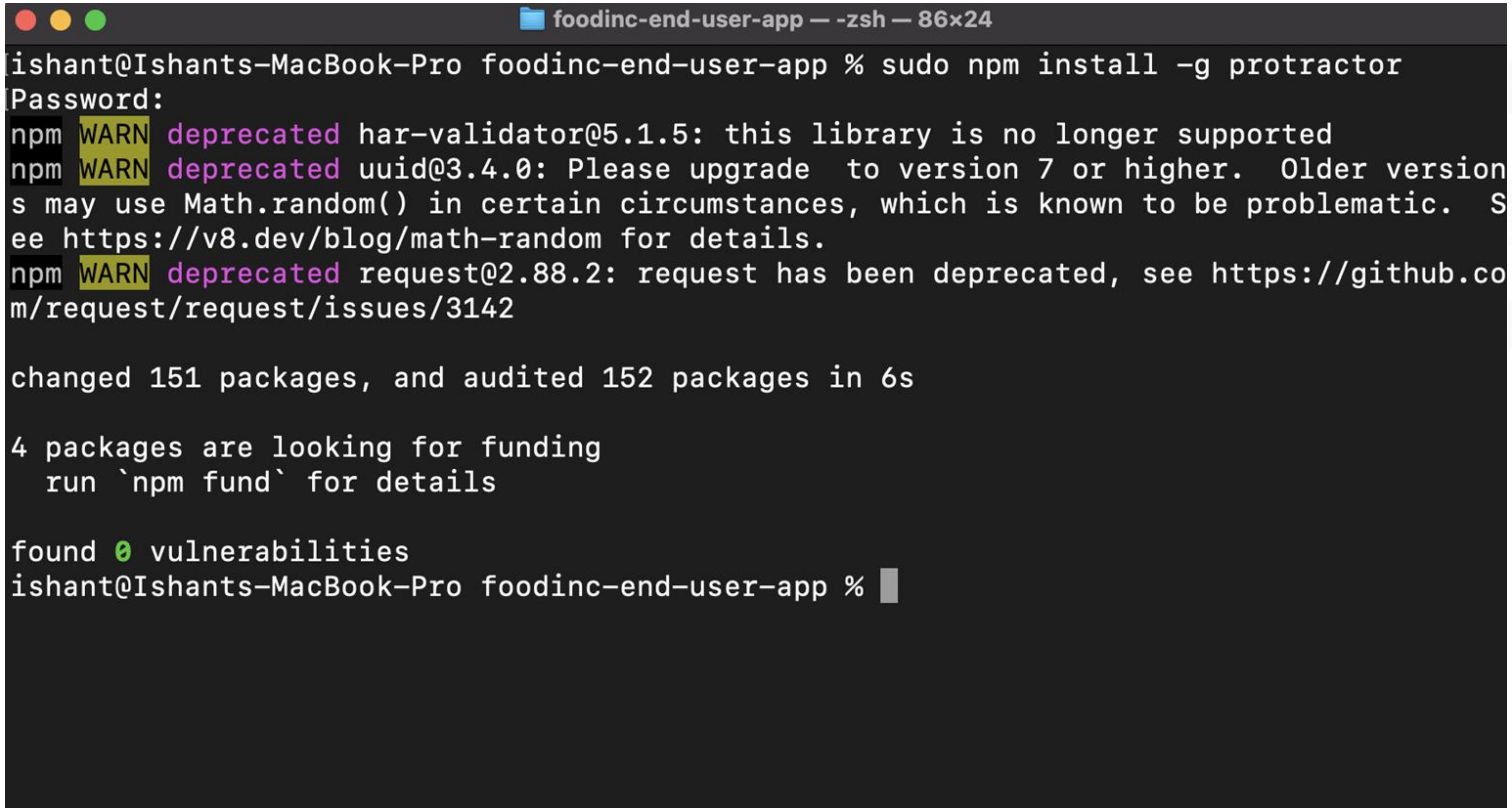
****

Sign In

Don't Have Account? **Create Account**

Configure Protractor

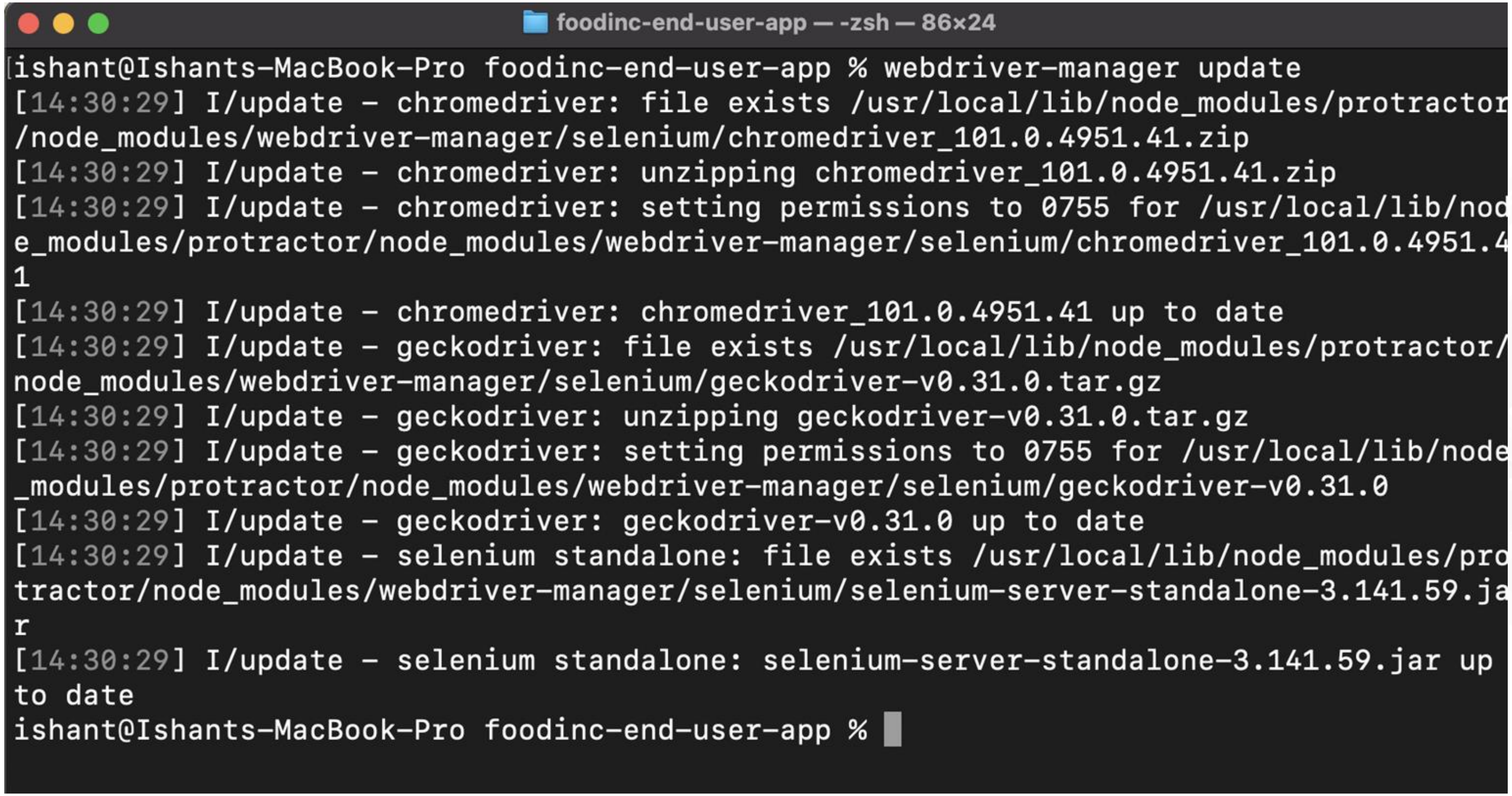
Now, we will add dependency for Protractor so that we can run tests for the application running in a real browser. Execute the command **npm install -g protractor.**You can also use sudo in case you face EACCESS errors.



```
ishant@Ishants-MacBook-Pro foodinc-end-user-app % sudo npm install -g protractor
Password:
npm WARN deprecated har-validator@5.1.5: this library is no longer supported
npm WARN deprecated uuid@3.4.0: Please upgrade  to version 7 or higher.  Older version
s may use Math.random() in certain circumstances, which is known to be problematic.  S
ee https://v8.dev/blog/math-random for details.
npm WARN deprecated request@2.88.2: request has been deprecated, see https://github.co
m/request/request/issues/3142

changed 151 packages, and audited 152 packages in 6s

4 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
ishant@Ishants-MacBook-Pro foodinc-end-user-app % 
```

Configure WebDriver

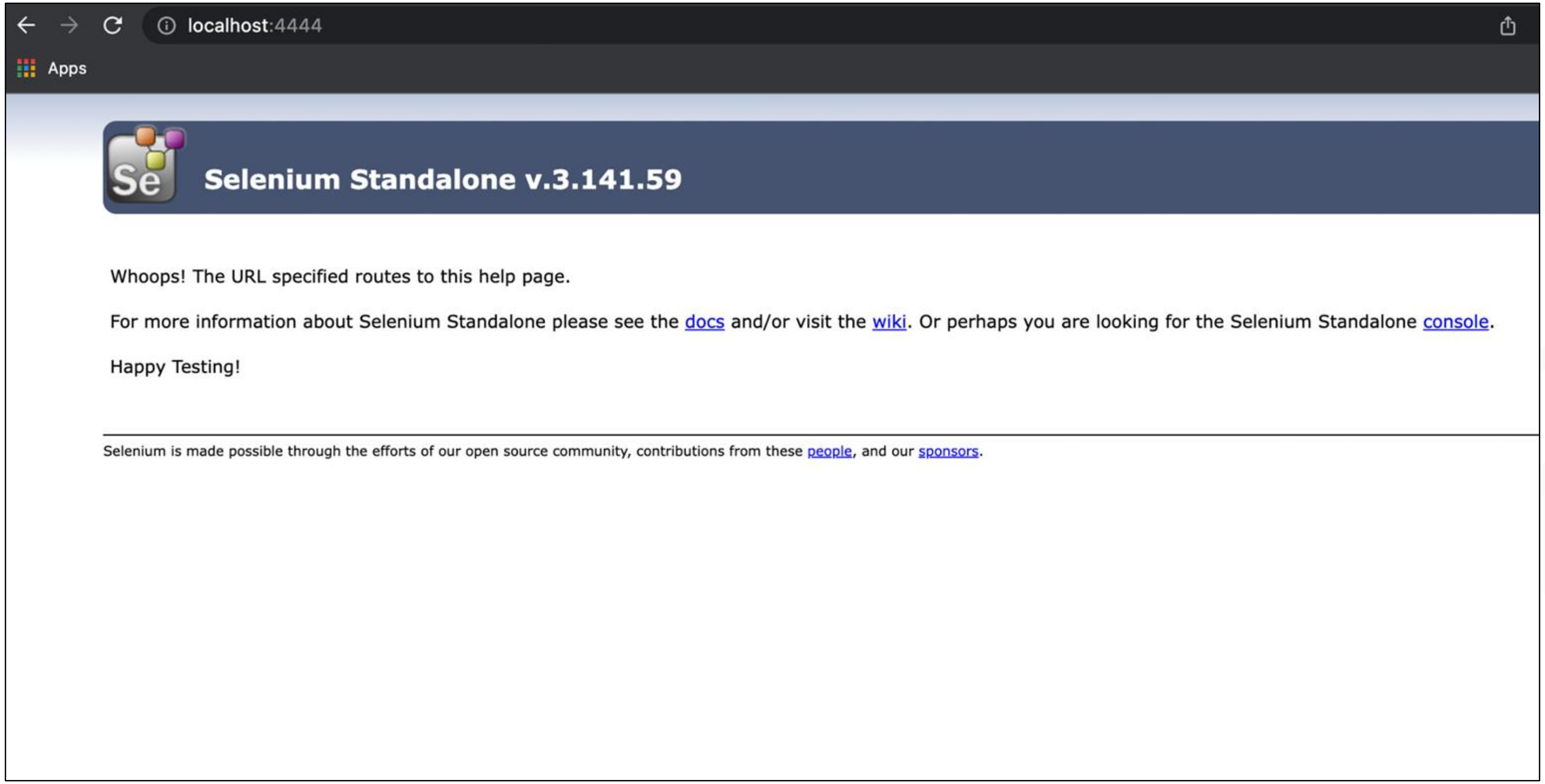
Once installed, you can run **webdriver-manager update** in order to update the webdriver-manager.



```
[ishant@Ishants-MacBook-Pro foodinc-end-user-app % webdriver-manager update
[14:30:29] I/update - chromedriver: file exists /usr/local/lib/node_modules/protractor
/node_modules/webdriver-manager/selenium/chromedriver_101.0.4951.41.zip
[14:30:29] I/update - chromedriver: unzipping chromedriver_101.0.4951.41.zip
[14:30:29] I/update - chromedriver: setting permissions to 0755 for /usr/local/lib/nod
e_modules/protractor/node_modules/webdriver-manager/selenium/chromedriver_101.0.4951.4
1
[14:30:29] I/update - chromedriver: chromedriver_101.0.4951.41 up to date
[14:30:29] I/update - geckodriver: file exists /usr/local/lib/node_modules/protractor/
node_modules/webdriver-manager/selenium/geckodriver-v0.31.0.tar.gz
[14:30:29] I/update - geckodriver: unzipping geckodriver-v0.31.0.tar.gz
[14:30:29] I/update - geckodriver: setting permissions to 0755 for /usr/local/lib/node
_modules/protractor/node_modules/webdriver-manager/selenium/geckodriver-v0.31.0
[14:30:29] I/update - geckodriver: geckodriver-v0.31.0 up to date
[14:30:29] I/update - selenium standalone: file exists /usr/local/lib/node_modules/pro
tractor/node_modules/webdriver-manager/selenium/selenium-server-standalone-3.141.59.ja
r
[14:30:29] I/update - selenium standalone: selenium-server-standalone-3.141.59.jar up
to date
ishant@Ishants-MacBook-Pro foodinc-end-user-app %
```

Configure WebDriver

Now, you can start up a server by executing the command **webdriver-manager start** and open the localhost port 4444 to check the webdriver.



Selenium Standalone v.3.141.59

Whoops! The URL specified routes to this help page.

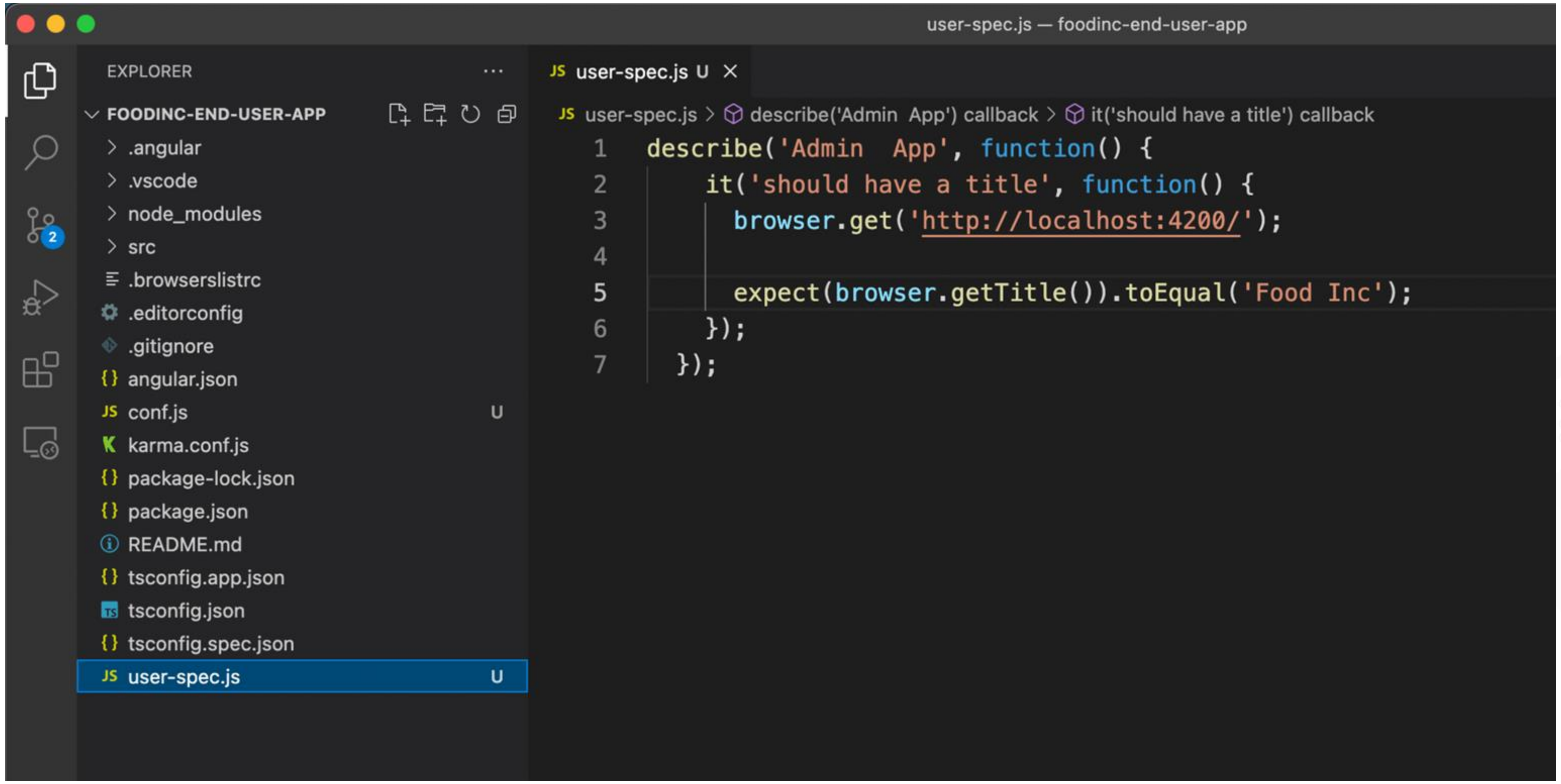
For more information about Selenium Standalone please see the docs and/or visit the wiki. Or perhaps you are looking for the Selenium Standalone console.

Happy Testing!

Selenium is made possible through the efforts of our open source community, contributions from these people, and our sponsors.

Setup Spec and Config files for Protractor Working

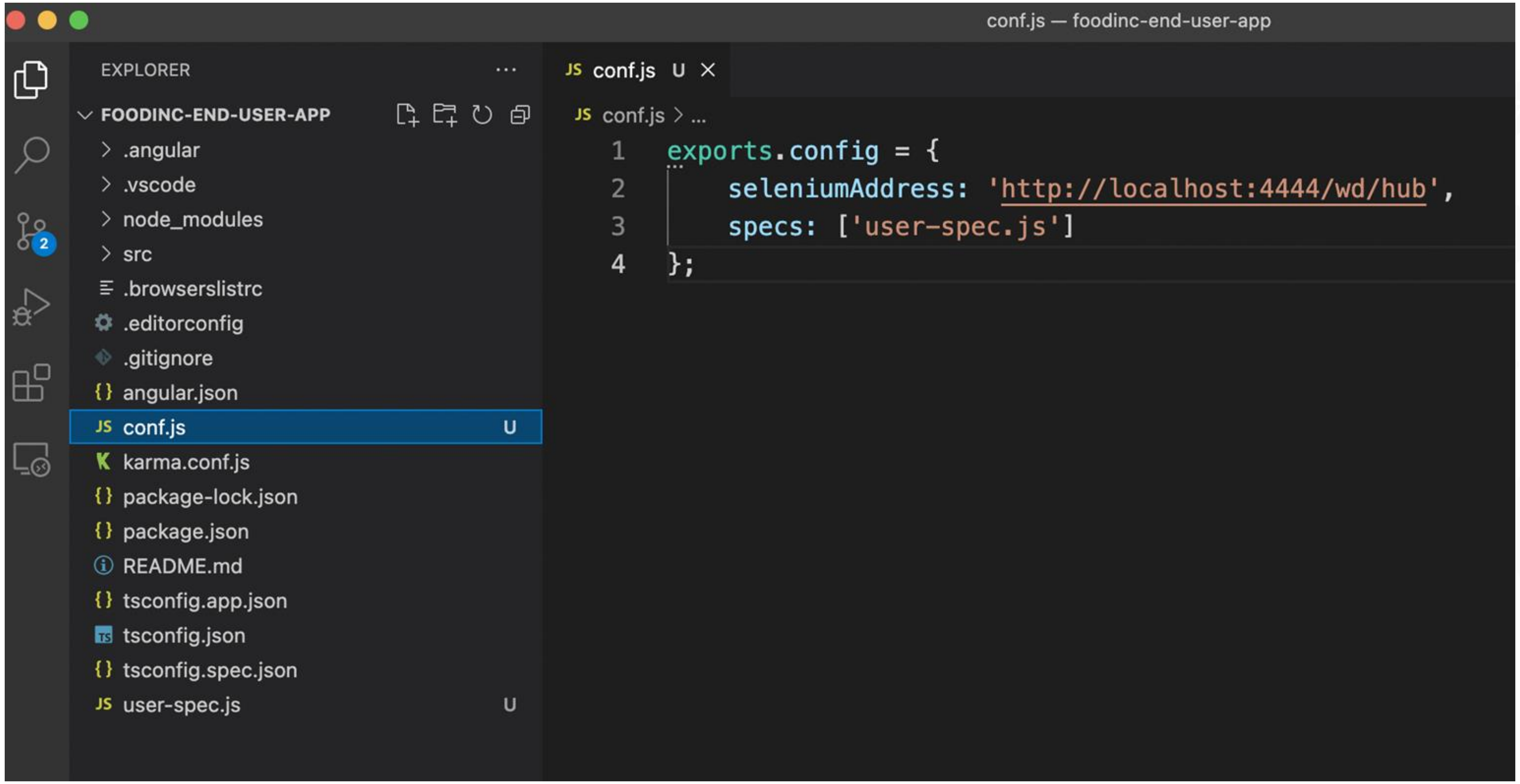
You can create a spec file named user-spec.js and write a small test case to validate the home page title.



```
describe('Admin  App', function() {
    it('should have a title', function() {
      browser.get('http://localhost:4200/');

      expect(browser.getTitle()).toEqual('Food Inc');
    });
  });
```

Setup Spec and Config files for Protractor Working

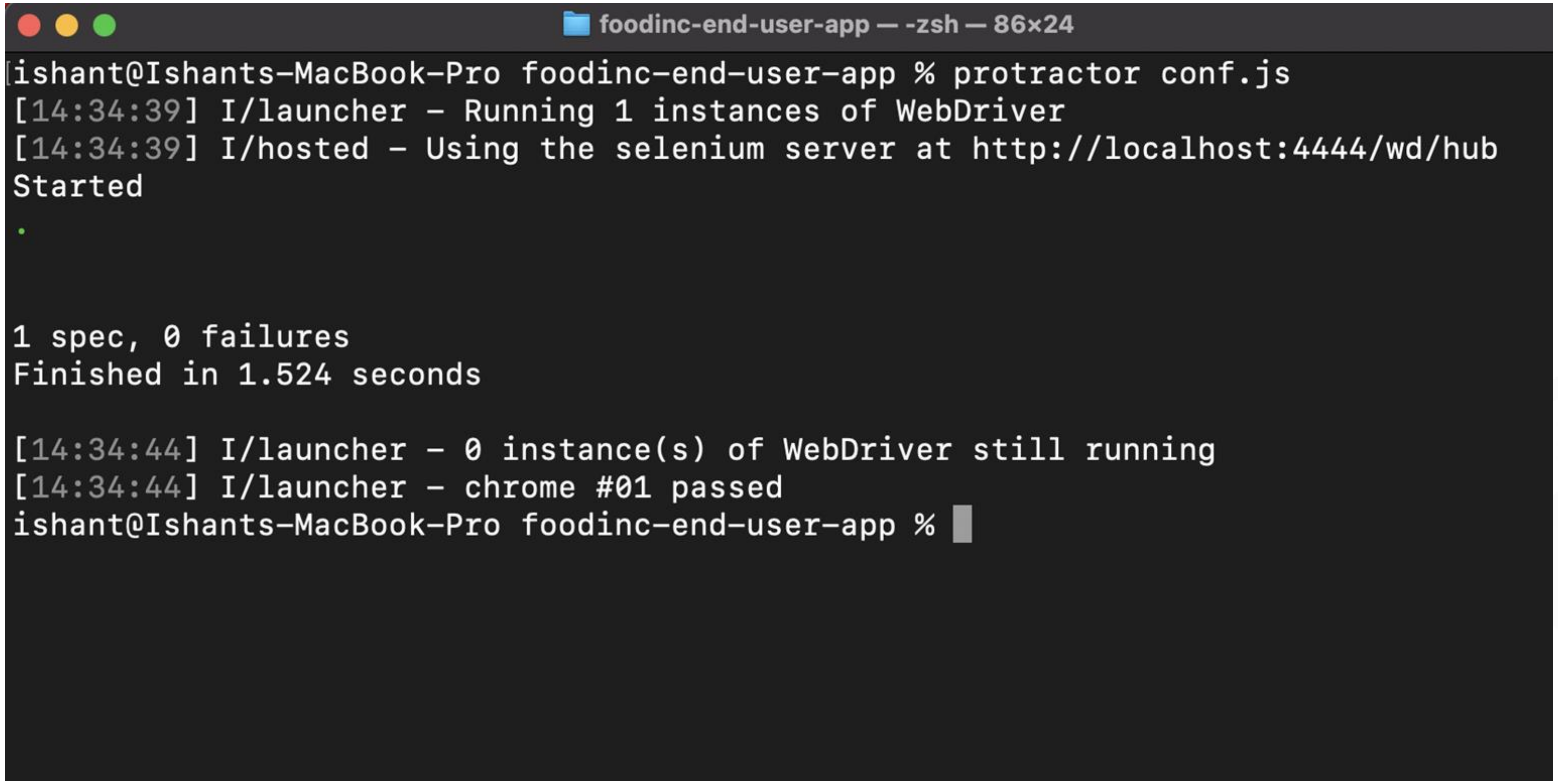
In order to configure Protractor, we need a configuration file. Create a new **conf.js** where the configuration is specified.



```
exports.config = {
    seleniumAddress: 'http://localhost:4444/wd/hub',
    specs: ['user-spec.js']
};
```

Execute Protractor

Now, let us run the Protractor conf.js on the terminal to validate if it works fine.



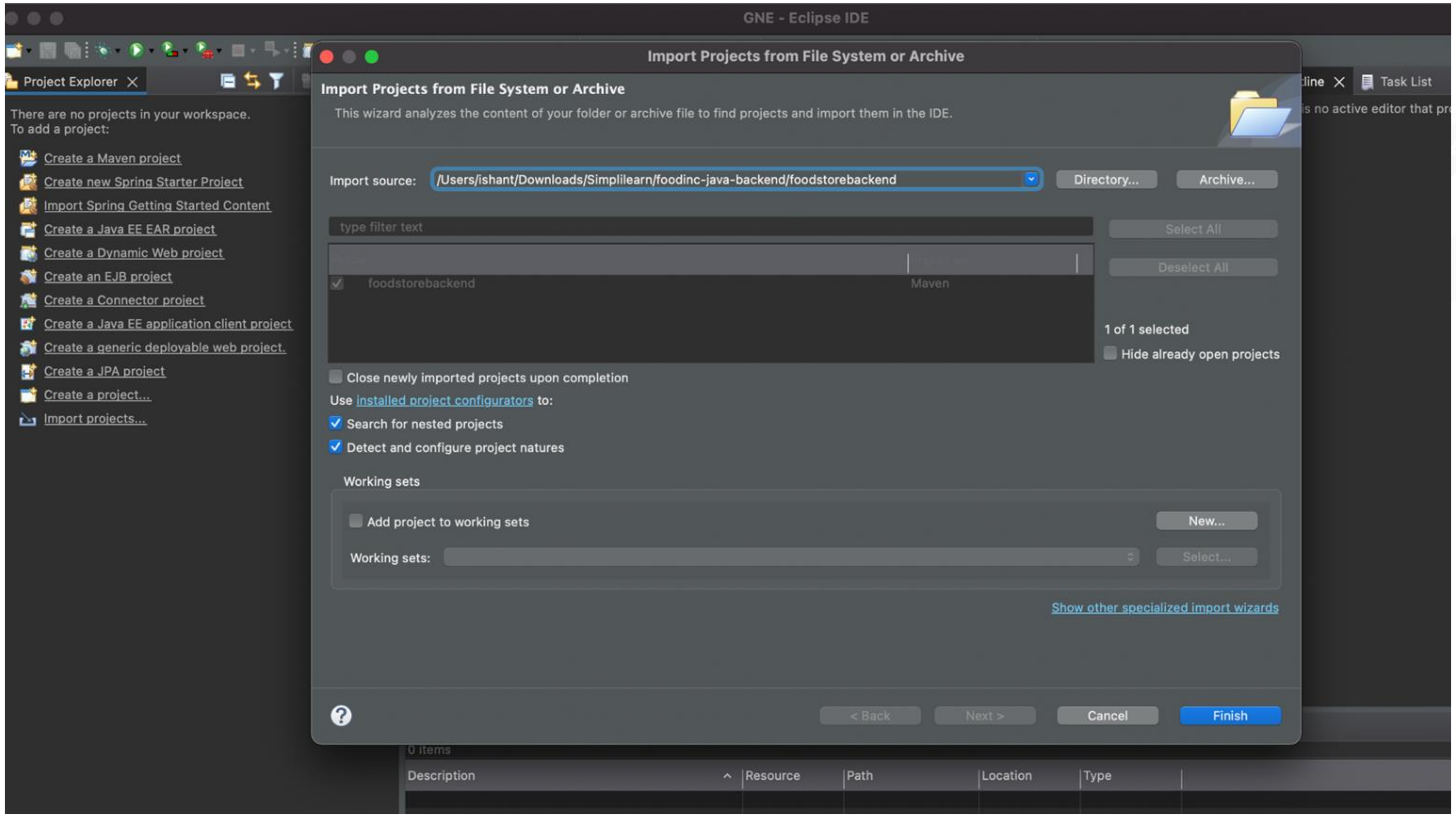
```
ishant@Ishants-MacBook-Pro foodinc-end-user-app % protractor conf.js
[14:34:39] I/launcher - Running 1 instances of WebDriver
[14:34:39] I/hosted - Using the selenium server at http://localhost:4444/wd/hub
Started
.



1 spec, 0 failures
Finished in 1.524 seconds

[14:34:44] I/launcher - 0 instance(s) of WebDriver still running
[14:34:44] I/launcher - chrome #01 passed
ishant@Ishants-MacBook-Pro foodinc-end-user-app %
```

Task 3: Set up Java Backend Project

```
#include <iostream>
#include <cstdlib>
using namespace std;
int main()
{
    cout << "Hello, world!" << endl;
    return 0;
}
```

Import Java Backend Project in Eclipse EE

Open the Eclipse EE, choose File, open Projects from File System, and import the Java Backend Project



GNE - Eclipse IDE

Project Explorer

There are no projects in your workspace.
To add a project:

- Create a Maven project
- Create new Spring Starter Project
- Import Spring Getting Started Content
- Create a Java EE EAR project
- Create a Dynamic Web project
- Create an EJB project
- Create a Connector project
- Create a Java EE application client project
- Create a generic deployable web project.
- Create a JPA project
- Create a project...
- Import projects...

Import Projects from File System or Archive

Import Projects from File System or Archive

This wizard analyzes the content of your folder or archive file to find projects and import them in the IDE.

Import source: /Users/ishant/Downloads/Simplilearn/foodinc-java-backend/foodstorebackend

☑ foodstorebackend	Maven

1 of 1 selected

□ Hide already open projects

□ Close newly imported projects upon completion

Use installed project configurators to:

☑ Search for nested projects

☑ Detect and configure project natures

Working sets

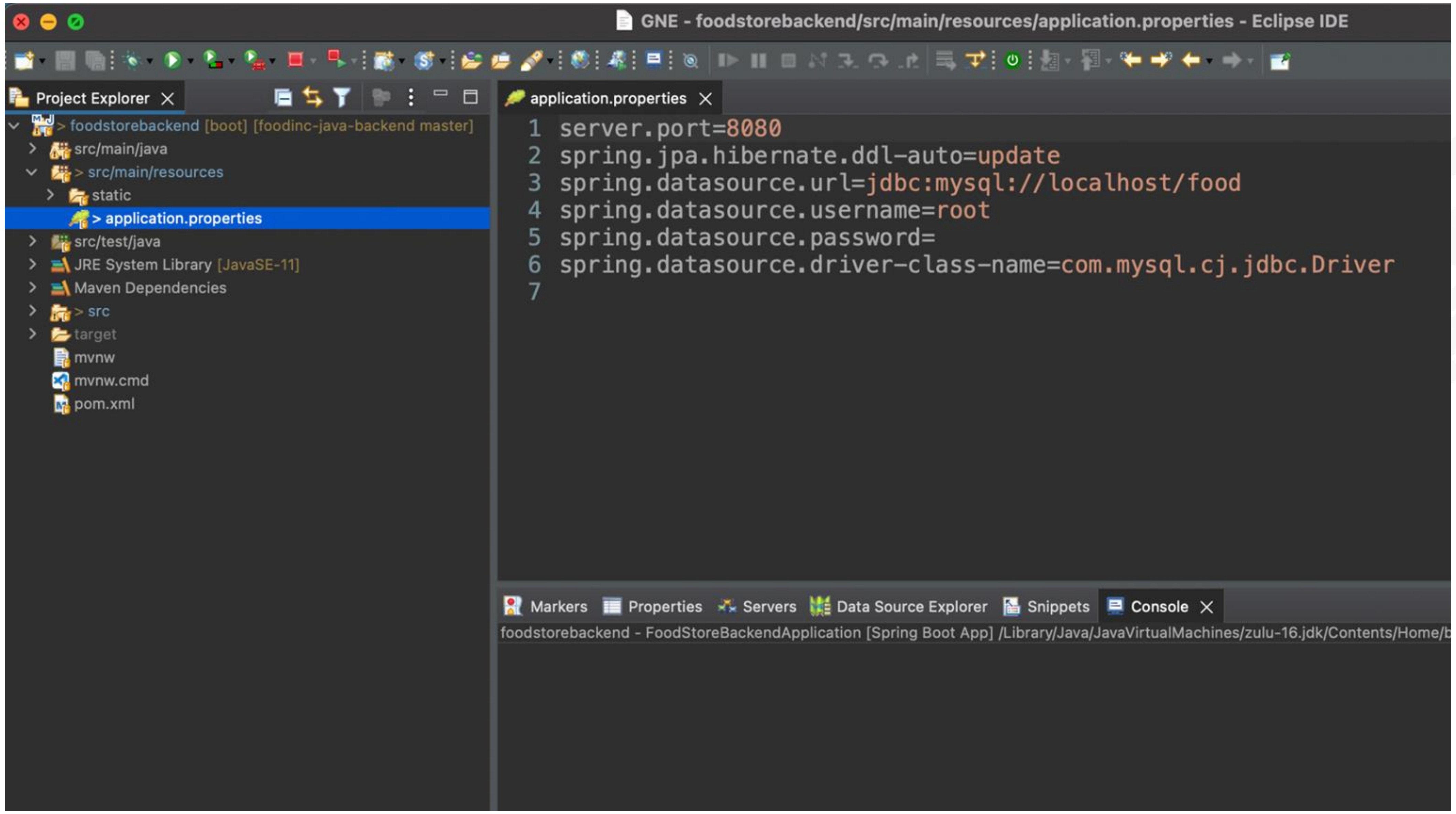
□ Add project to working sets

Working sets:

Show other specialized import wizards

Configure DataBase Connectivity

Once the project is imported, open application.properties file and configure database connection parameters for the DB Communication

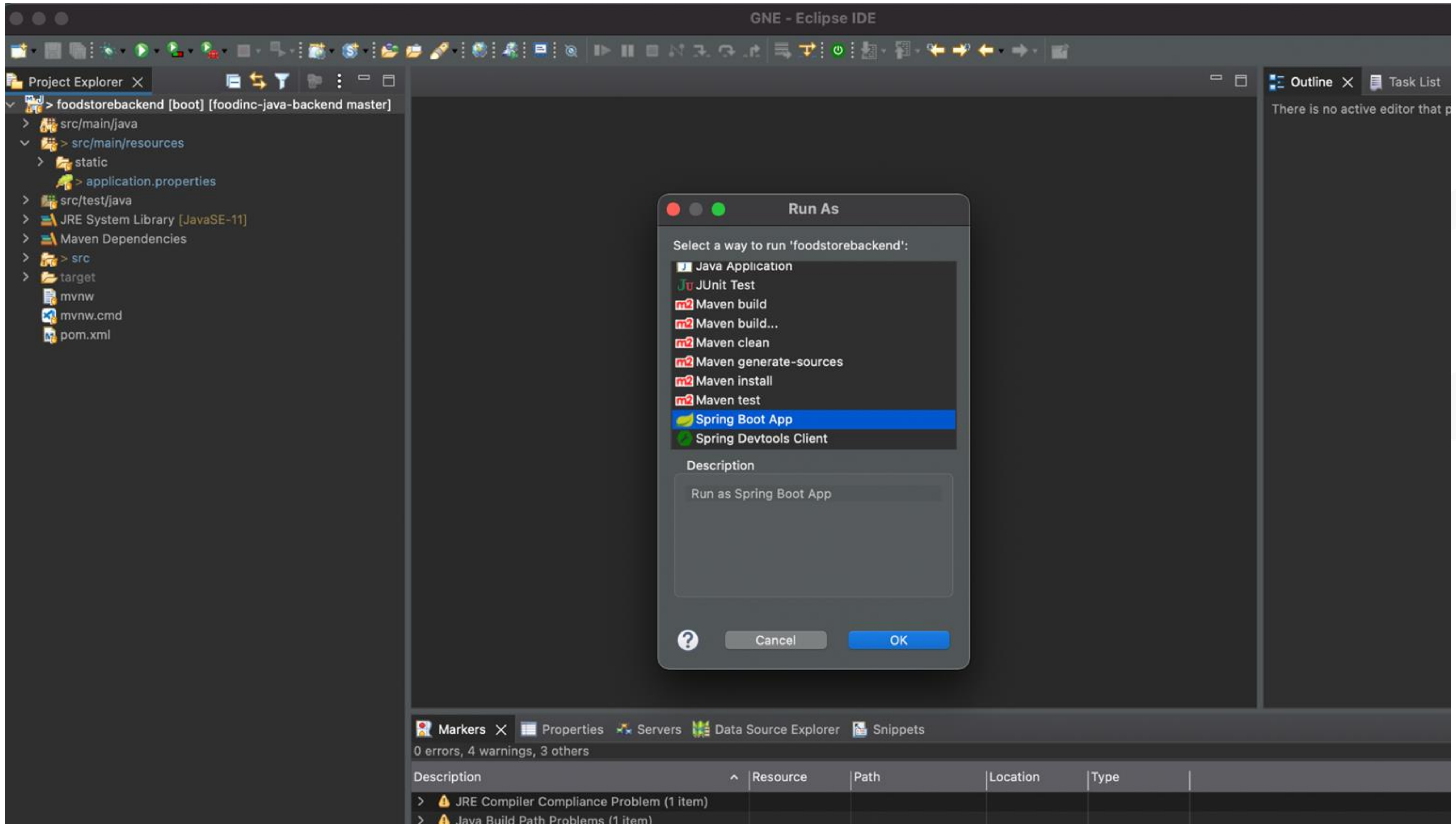


```
server.port=8080
spring.jpa.hibernate.ddl-auto=update
spring.datasource.url=jdbc:mysql://localhost/food
spring.datasource.username=root
spring.datasource.password=
spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
```

```
foodstorebackend - FoodStoreBackendApplication [Spring Boot App] /Library/Java/JavaVirtualMachines/zulu-16.jdk/Contents/Home/b
```

Build and Execute the Maven Project

Select the Green Build Icon and choose the Run Option as Spring Boot App



Select a way to run 'foodstorebackend':

Spring Boot App

Description

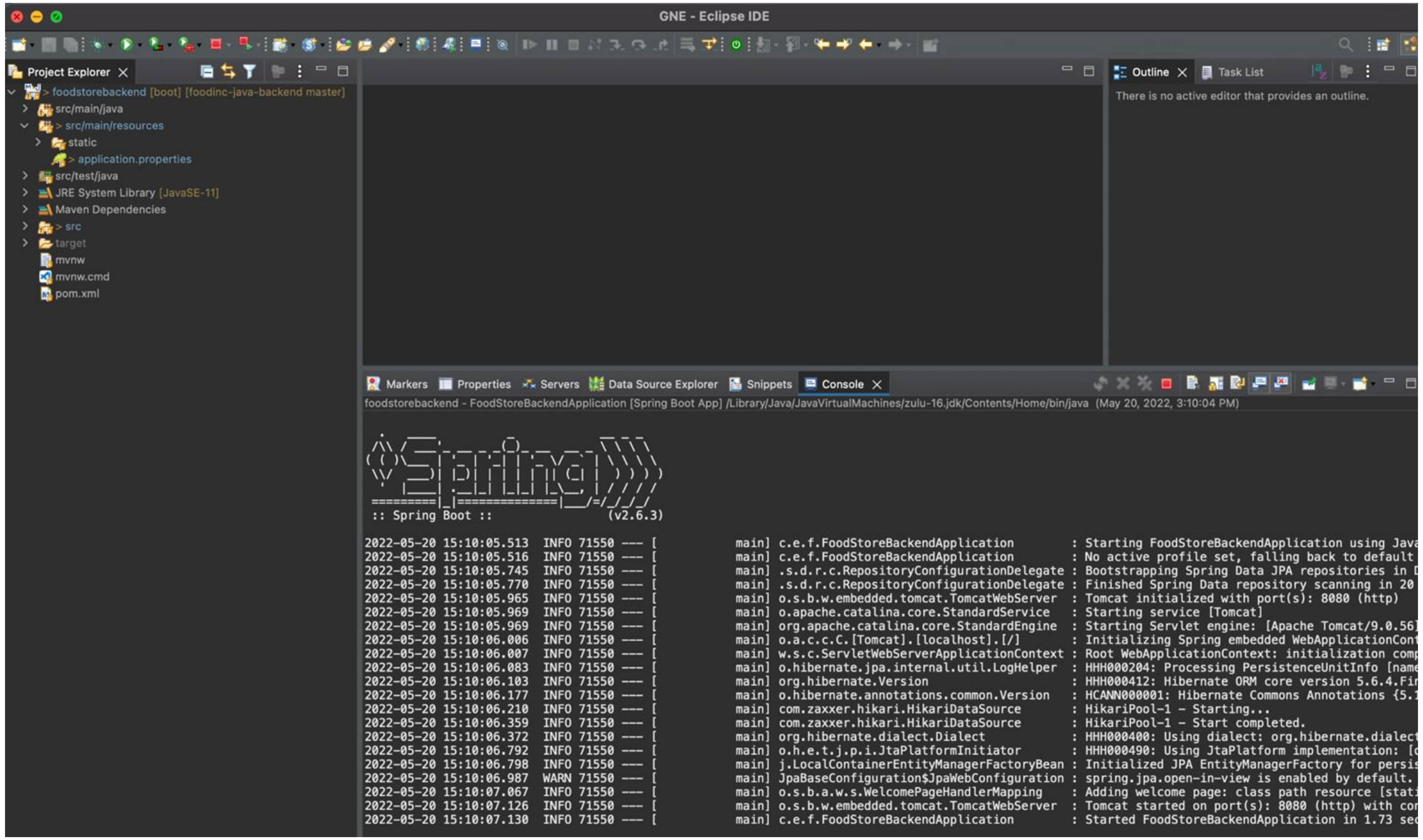
Run as Spring Boot App

0 errors, 4 warnings, 3 others

Description	Resource	Path	Location	Type
JRE Compiler Compliance Problem (1 item)				
Java Build Path Problems (1 item)				

Build and Execute the Maven Project

Once, the project is built, you can see Tomcat Server in the console for Apache running on port 8080 successfully:

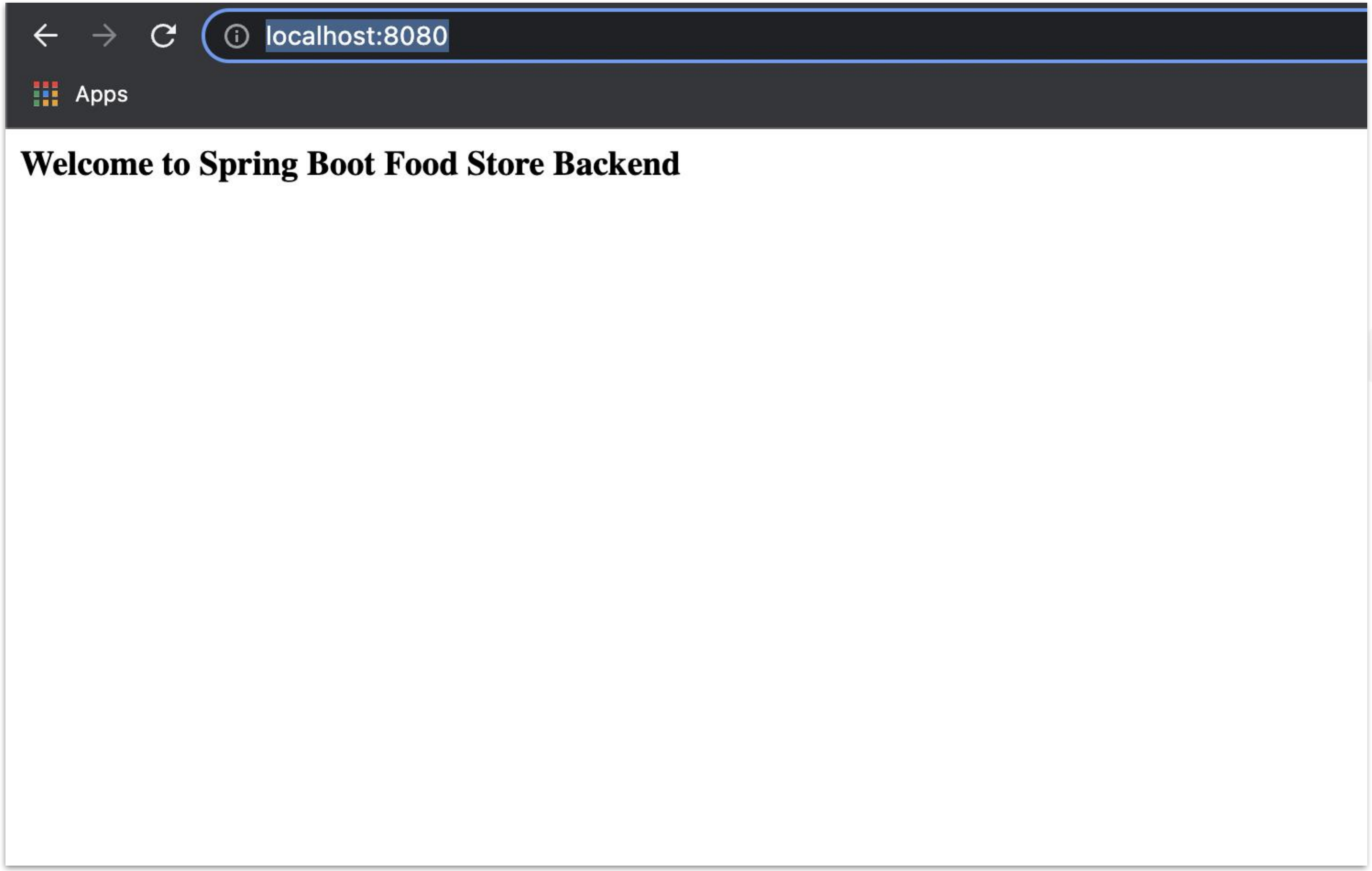


foodstorebackend - FoodStoreBackendApplication [Spring Boot App] /Library/Java/JavaVirtualMachines/zulu-16.jdk/Contents/Home/bin/java (May 20, 2022, 3:10:04 PM)

```
  .   ____          _            __ _ _
 /\\ / ___'_ __ _ _(_)_ __  __ _ \ \ \ \
( ( )\___ | '_ | '_| | '_ \/ _` | \ \ \ \
 \\/  ___)| |_)| | | | | || (_| |  ) ) ) )
  '  |____| .__|_| |_|_| |_\__, | / / / /
 =========|_|==============|___/=/_/_/_/
 :: Spring Boot ::                (v2.6.3)

2022-05-20 15:10:05.513  INFO 71550 --- [           main] c.e.f.FoodStoreBackendApplication        : Starting FoodStoreBackendApplication using Java
2022-05-20 15:10:05.516  INFO 71550 --- [           main] c.e.f.FoodStoreBackendApplication        : No active profile set, falling back to default
2022-05-20 15:10:05.745  INFO 71550 --- [           main] .s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping Spring Data JPA repositories in D
2022-05-20 15:10:05.770  INFO 71550 --- [           main] .s.d.r.c.RepositoryConfigurationDelegate : Finished Spring Data repository scanning in 20
2022-05-20 15:10:05.965  INFO 71550 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat initialized with port(s): 8080 (http)
2022-05-20 15:10:05.969  INFO 71550 --- [           main] o.apache.catalina.core.StandardService   : Starting service [Tomcat]
2022-05-20 15:10:05.969  INFO 71550 --- [           main] org.apache.catalina.core.StandardEngine  : Starting Servlet engine: [Apache Tomcat/9.0.56]
2022-05-20 15:10:06.006  INFO 71550 --- [           main] o.a.c.c.C.[Tomcat].[localhost].[/]       : Initializing Spring embedded WebApplicationCont
2022-05-20 15:10:06.007  INFO 71550 --- [           main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization comp
2022-05-20 15:10:06.083  INFO 71550 --- [           main] o.hibernate.jpa.internal.util.LogHelper  : HHH000204: Processing PersistenceUnitInfo [name
2022-05-20 15:10:06.103  INFO 71550 --- [           main] org.hibernate.Version                    : HHH000412: Hibernate ORM core version 5.6.4.Fin
2022-05-20 15:10:06.177  INFO 71550 --- [           main] o.hibernate.annotations.common.Version   : HCANN000001: Hibernate Commons Annotations {5.1
2022-05-20 15:10:06.210  INFO 71550 --- [           main] com.zaxxer.hikari.HikariDataSource       : HikariPool-1 - Starting...
2022-05-20 15:10:06.359  INFO 71550 --- [           main] com.zaxxer.hikari.HikariDataSource       : HikariPool-1 - Start completed.
2022-05-20 15:10:06.372  INFO 71550 --- [           main] org.hibernate.dialect.Dialect            : HHH000400: Using dialect: org.hibernate.dialect
2022-05-20 15:10:06.792  INFO 71550 --- [           main] o.h.e.t.j.p.i.JtaPlatformInitiator       : HHH000490: Using JtaPlatform implementation: [o
2022-05-20 15:10:06.798  INFO 71550 --- [           main] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerFactory for persis
2022-05-20 15:10:06.987  WARN 71550 --- [           main] JpaBaseConfiguration$JpaWebConfiguration : spring.jpa.open-in-view is enabled by default.
2022-05-20 15:10:07.067  INFO 71550 --- [           main] o.s.b.a.w.s.WelcomePageHandlerMapping    : Adding welcome page: class path resource [stati
2022-05-20 15:10:07.126  INFO 71550 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat started on port(s): 8080 (http) with con
2022-05-20 15:10:07.130  INFO 71550 --- [           main] c.e.f.FoodStoreBackendApplication        : Started FoodStoreBackendApplication in 1.73 sec
```

Check Project Status as Running

Open the web browser and type **http://localhost:8080/** to check if your project is up and running



Welcome to Spring Boot Food Store Backend

Key Takeaways

- Users can configure Angular admin project.
- They can configure Angular end user project.
- They can configure and execute Protractor.
- They can configure and build Java projects.

Before the Next Class

Since you have successfully completed this session, in the next discussion, you should know how to:

- Review JavaScript
- Brush up angular CLI
- Go through the Protractor configuration
- Review page object model in Protractor

What Next?

In the next class, you will be able to:

- Work with Protractor to perform testing in real browser
- Write simple test scripts for admin Angular project
- Use page object to test admin Angular project
- Write simple test scripts for end user Angular project
- Use page object to test end user project