

Capstone Project - DESCRIPTION / Git link

MEDICARE PROJECT

- **Browser-based end user testing using Selenium WebDriver**
 - This is functional testing
 - Should include: search medication products, products, login functionality, adding products to the cart, checkout, address.
- **Unit testing for back-end elements of the website using TestNG**
 - Similar for admin. Add create user, add create product, add create category.
- **API testing with Postman**
 - **API request: product list user / product list admin**
- **Automating the whole testing process by a Jenkins job**
 - MAVEN project + Newman

DESCRIPTION

This project requires end-to-end development of a comprehensive QA and test environment for a healthcare website. This QA and test environment should be inclusive of the following testing layers:

1. Browser-based end user testing using Selenium WebDriver
2. Unit testing for back-end elements of the website using TestNG
3. API testing with Postman on AWS cloud
4. Automating the whole testing process by a Jenkins job

The end-deliverables will be executable scripts and modules, which can be run on demand to test the web app.

Final Capstone Project

Website setup

- 1- We Imported the provided war file into Eclipse
- 2- Created and imported the Database from the medicare backend folder sql file: MySQL-DatabaseQueries.sql
- 3- Run the medicare website through Tomcat server y the database through MySQL

Maven project

1- Create a Maven project

2- Repositories needed POM.xml

- Selenium-java
- TestNG
- Maven Compiler
- Surefire
- Rest assured (optional)

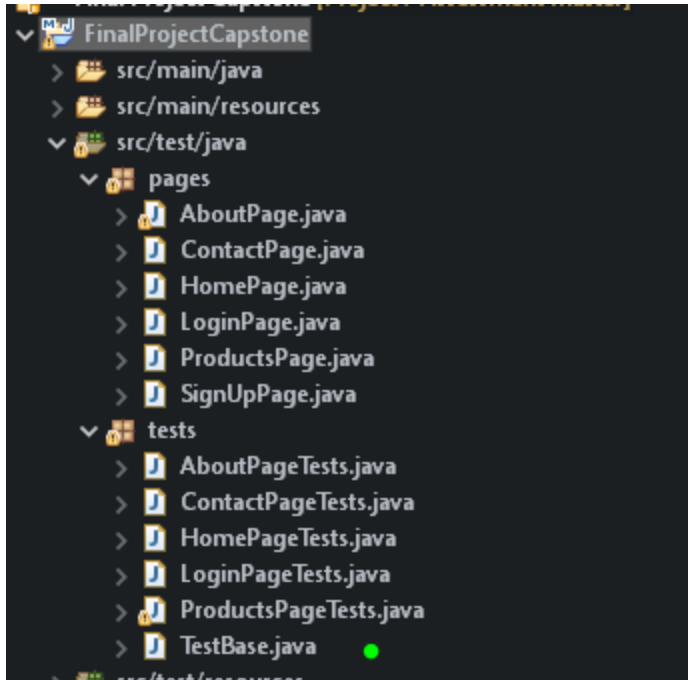
```
<build>
<plugins>
  <plugin>
    <groupId>org.apache.maven.plugins</groupId>
    <artifactId>maven-compiler-plugin</artifactId>
    <version>3.8.1</version>
    <configuration>
      <source>1.8</source>
      <target>1.8</target>
    </configuration>
  </plugin>
  <plugin>
    <groupId>org.apache.maven.plugins</groupId>
    <artifactId>maven-surefire-plugin</artifactId>
    <version>2.22.0</version>
    <configuration>
      <forkCount>0</forkCount>
      <suiteXmlFiles>
        <suiteXmlFile>testng.xml</suiteXmlFile>
      </suiteXmlFiles>
    </configuration>
  </plugin>
</plugins>
</build>

<dependencies>
  <dependency>
    <groupId>org.seleniumhq.selenium</groupId>
    <artifactId>selenium-java</artifactId>
    <version>3.141.59</version>
  </dependency>

  <dependency>
    <groupId>org.testng</groupId>
    <artifactId>testng</artifactId>
    <version>7.4.0</version>
    <scope>test</scope>
  </dependency>
</dependencies>
```

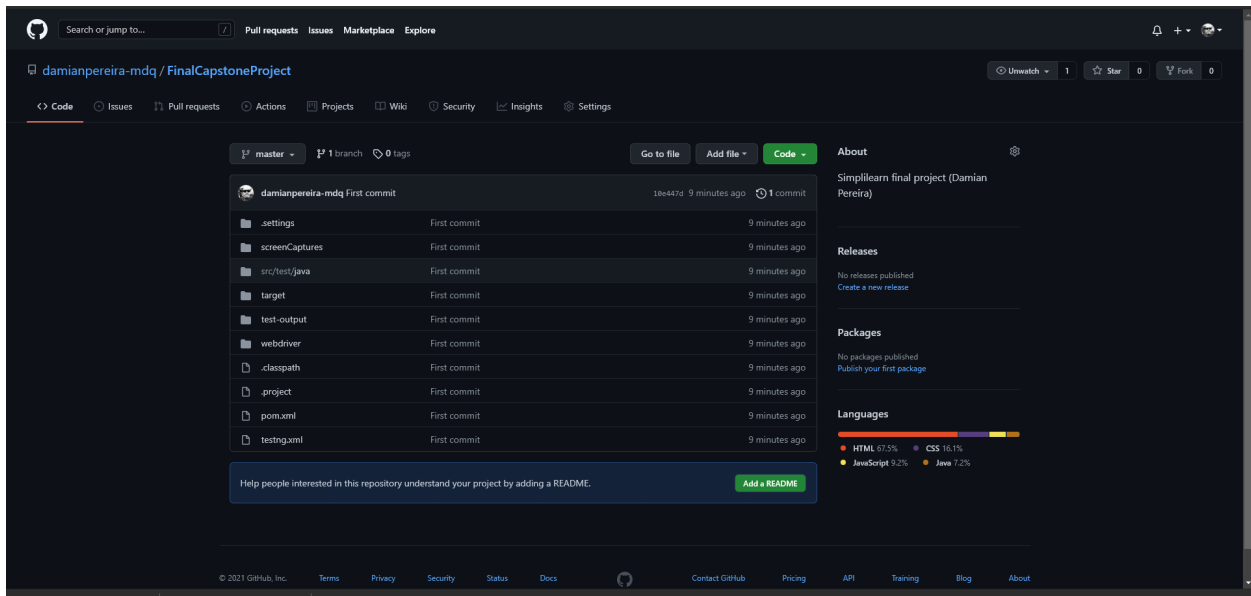
Page Object Model

We followed the POM, creating all Classes for pages functionalities with locators, and all the correspondent tests on different Classes. And also we used a Base Class with the Setup settings



GITHUB Repository

<https://github.com/damianpereira-mdq/FinalCapstoneProject.git>



JENKINS Integration

We created a Job in Jenkins that will run the MAVEN tests and also will execute a POSTman API test as requested, using the command prompt via Newman

The screenshot shows the Jenkins configuration page for a job named "Final Capstone Project". The "Pre Steps" tab is selected, showing the following configuration:

- Pre Steps:** A button "Add pre-build step" is visible.
- Build:**
 - Root POM: pom.xml
- Goals and options:**
 - clean test
- Post Steps:**
 - Run only if build succeeds: ☐
 - Run only if build succeeds or is unstable: ☐
 - Run regardless of build result: ☒
 - Should the post-build steps run only for successful builds, etc.
- Execute Windows batch command:**
 - Command: `newman run E:\SimpleLearn\FinalProjectCapstone\medicare.postman_collection`

At the bottom, there is a link "See the list of available environment variables" and an "Advanced..." button.