

(Phase-4) Swiggy App Project - Writeup

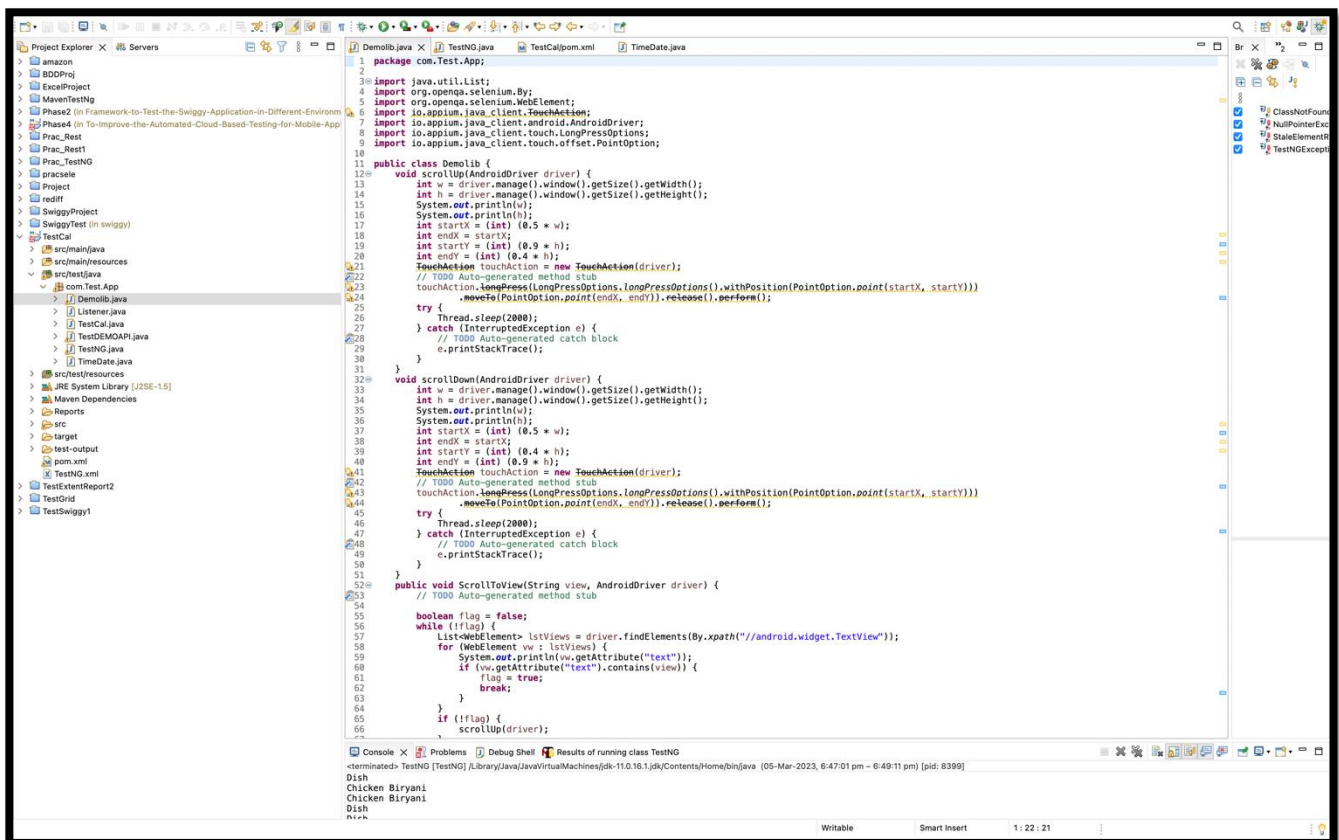
Introduction:-

Firstly, we have to create a maven project and we have to create a package name called com.Test.App

This com.Test.App Package contains classes:-

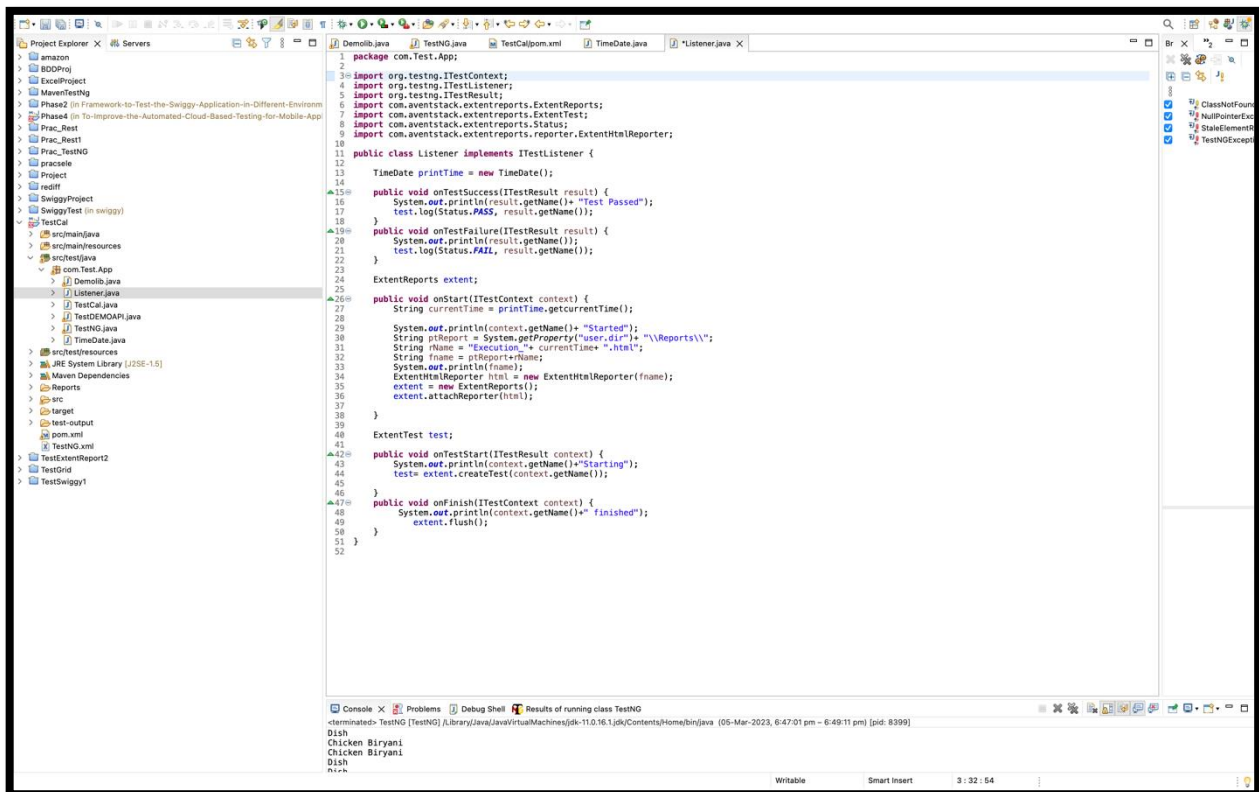
- Demolib.java
- Listener.java
- TestNG.java
- TimeDate.java

Then, we have created the pom.xml and TestNg.xml file inside the maven project. And we have to setup the android emulator/ real-time device and Appium server, Appium Inspector.



In this above file the imports are implemented as per requirement. The demolib file contains the startX, endX and StartY, endY. It will give the width and height of the window to sroll up and down in the app interface.

Then in the Terminal we have to give the adb devices cmd to verify the real-time is connected. And we have to give adb shell and dumsys window | grep mFocusedApp command. After this we can get the apppackage and appActivity.



In this above Listener file, we have implemented the imports as per requirement. In this file the TimeDate file is linked and using the extent report it will print the reports.

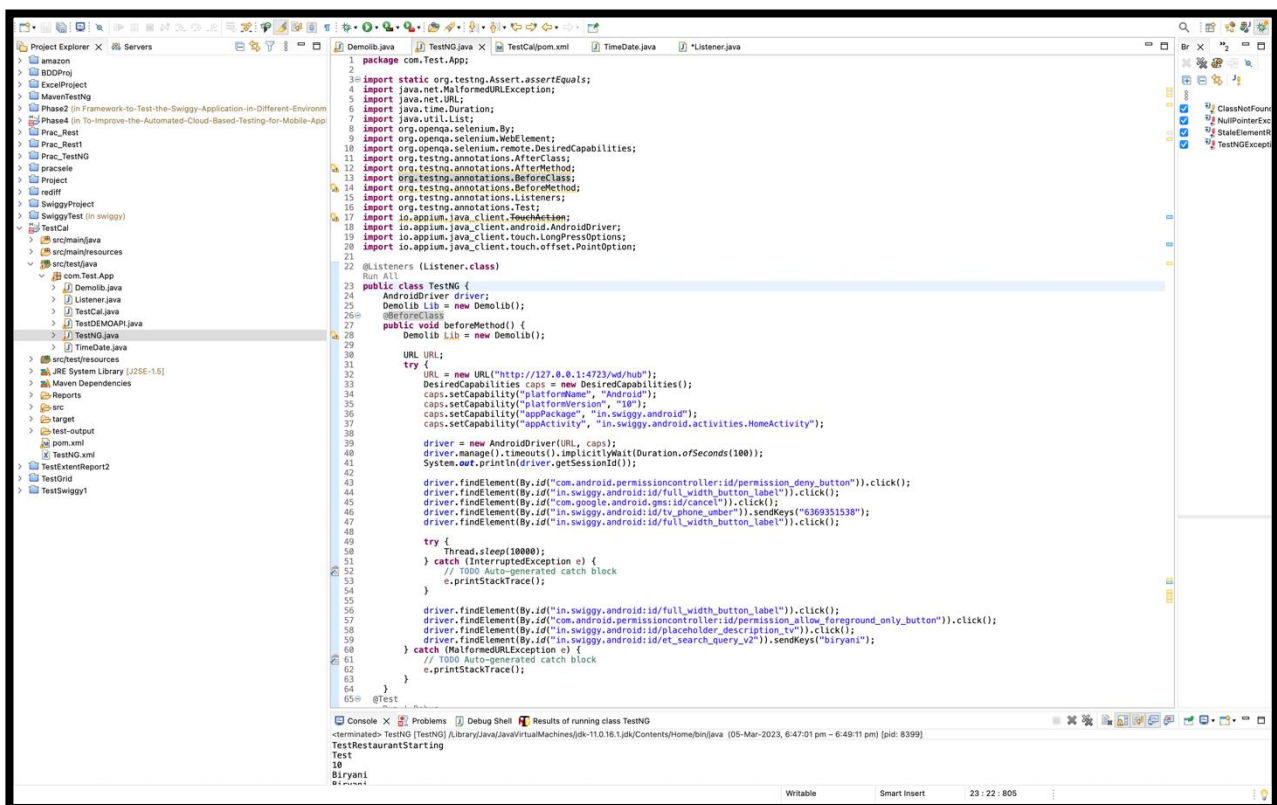
we need to create desired capabilities inside Appium inspector , like this

```
{
  "platformName": "Android",
  "appium:platformVersion": "10",
  "appium:appPackage": "in.swiggy.android",
  "appium:appActivity": "in.swiggy.android.activities.HomeActivity"
}
```

We have to do the changes and save the capabilities and start the session. And in the Appium inspector we have give the remote host, remote path and

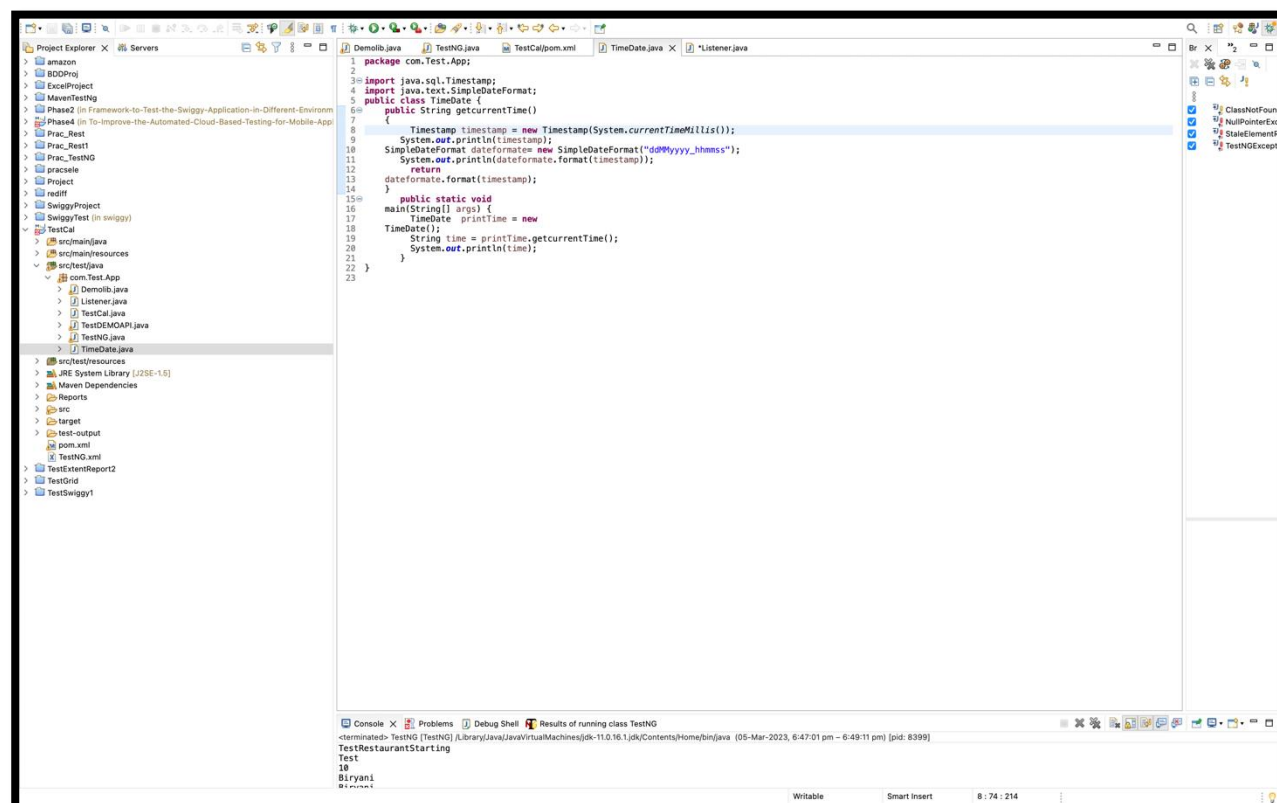
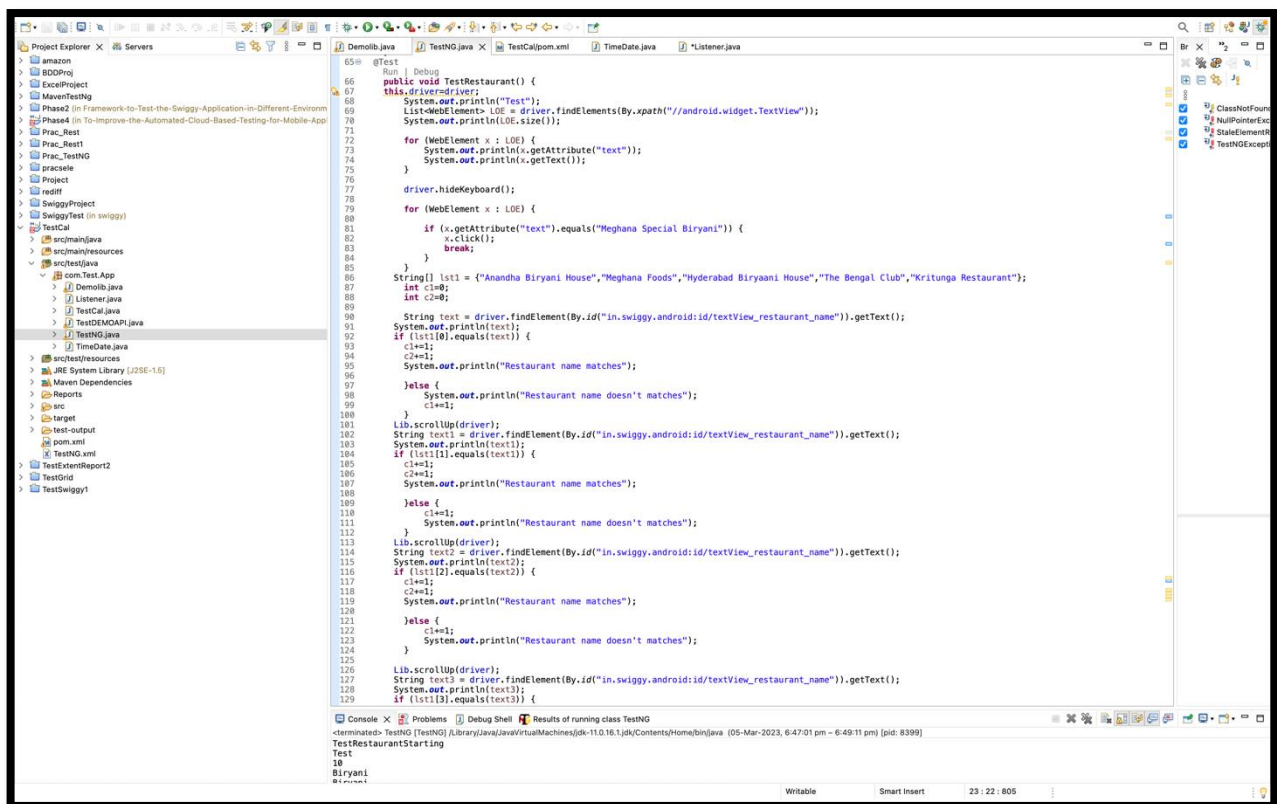
remote path. While using the real-time we have to give the attach to session option and we have give the session id and start the session.

In the Appium Gui, we have to first preset the option that we are going to use. In that we have to click the checkbox to use the allowCors option. After this we have to given start server option.

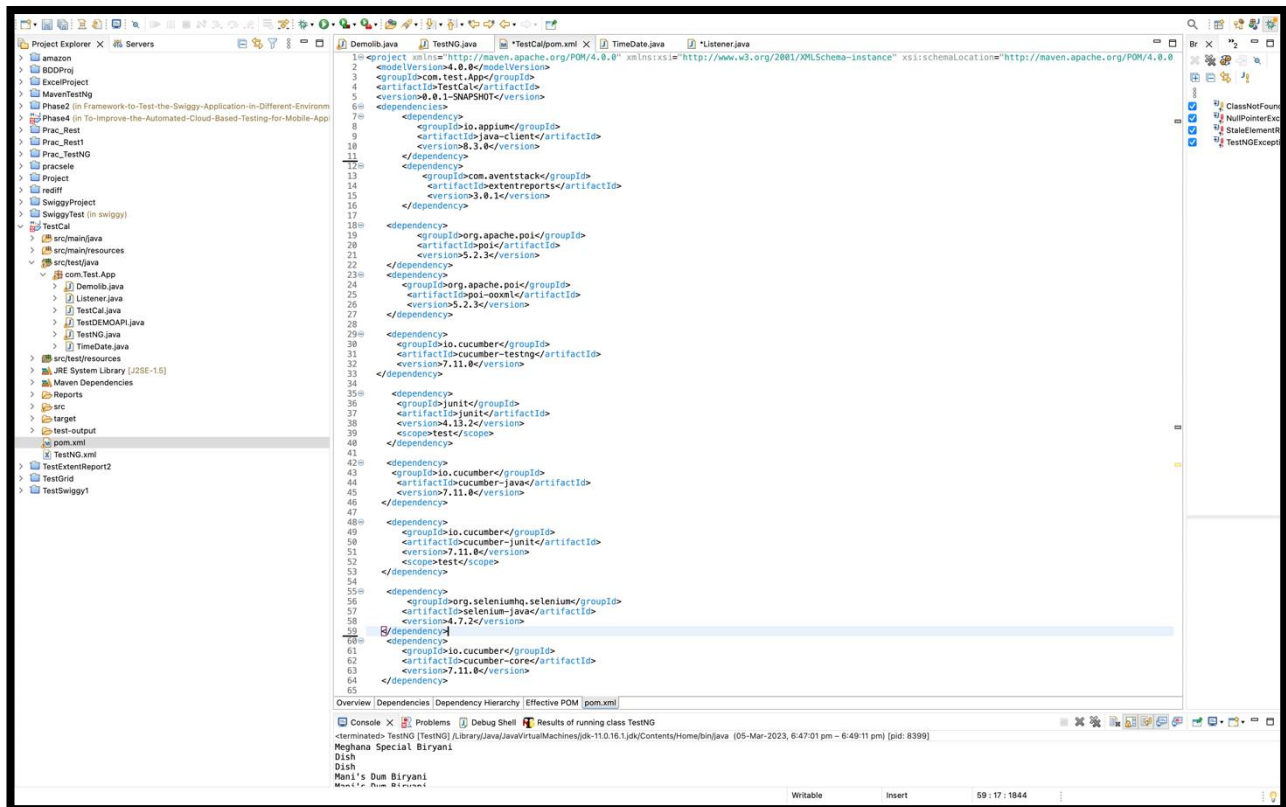


In this above TestNG file, we have implemented the imports as per requirement. It is a testng class, in that we have used Listener class, beforeClass, Test, AfterClass. In the beforeClass, we have given the url of the Appium inspector and the desired capabilities and we have used Android driver in this. Then we have implemented the id's of the element. At first it will open the swiggy app in the real-time device which is already connected. And it will click the get started button and we click the none of the above button which pop-up while asking for the mobile number. After that through sendkeys, we have given the mobile number and it redirect to the otp page and the otp is entering automatically and it will redirect to location page in that it will click the allow current location. And then it will redirect to home page and in the search bar it will search for biryani. And it will redirect to the next page, where it will show the list of dish in the various restaurant. After that it will scroll up and down. And get the unique id of the given restaurant and select the dish(biryani). After this it will redirect to

order page in that, it will give the number of items and click the payment option. Using this TestNg class we have called the listener class, it will help to create the extent report.



In the above TimeDate file, we have given the imports as per requirement. In this file we have used Timestamp and it will give the simpleDate format. The simple date format is “ddMMyyyy_hhmmss” it will give the current date, month, year and current time in hours, minutes and in seconds.



In this above pom.xml file, we have given the all the dependencies like appium, Extentreports, poi, poi-ooxml, cucumber testing, selenium, junit, cucumber, cucumber junit, cucumber core and maven surefire, compiler plugins.

