

Groupon Application Test Automation



Project Report

CMPE 287

Software Quality Assurance and Testing

Submitted by:

Apoorv Mehta (011436074)
Darshil Saraiya (010825321)
Parvez Patel (011490531)
Shachi Shah (010466573)
Siddharth Daftari (011457004)

Under the guidance of:

Prof. Santhi Narayananankutty

Date: April 27, 2017

Table Of Contents

Table Of Contents	2
1. Introduction	3
1.1 Overview of Groupon Photos Test Project	3
2. Test Automation Plan and Scope	4
2.1 Test Automation Approach and Strategy	4
2.2 Test Scope	4
2.2.1 In Scope	4
2.2.2 Out of Scope	4
2.3 Entrance / Exit Criteria	4
2.3.1 Entry Criteria	4
2.3.2 Exit Criteria	4
2.4 Test Automation Requirements	5
2.5 Tools Used	5
2.5.1 Platform: Web	5
2.5.2 Platform: Android	5
3. Test Automation Design	6
3.1 Platform: Web	6
3.1.1 Steps to Run Selenium	6
3.1.2 End-To-End Test Automations	7
3.1.2.1 Add to Wishlist	7
3.1.2.2 Buy Deal	12
3.2 Platform: Android	18
3.2.1 Steps to Run Appium	18
3.2.2 End-To-End Test Automations	21
3.2.2.1 Signup using Email address	21
3.2.2.2 Login and see deals in specific pincode	30
4. Test Automation Deliverables	41
5. Test Automation Summary	42
6. Team Members	43
6.1 Roles and Responsibilities	43
7. Conclusion	44

1. Introduction

1.1 Overview of Groupon Photos Test Project

Groupon Test is an test automation project. This exercise is aimed at understanding and implementing the automation black box testing for a real-world live app. This project scope covers the automation of an application from two of its most used platforms i.e Web and Android App.

In this project, end-to-end flow automation for the Groupon application will be done. The automation will be done in both of the platforms i.e. the web and mobile platform. It will be done through the software-testing framework Selenium WebDriver API for the web application and an open source test automation tool Appium. In this project, two distinct end-to-end flow automation is done for each of the platforms.

2. Test Automation Plan and Scope

2.1 Test Automation Approach and Strategy

In this test automation project, the strategy is to have a shared responsibility model to achieve the overall target of quality control. As this is a learning exercise over a live deployed app, the approach is to focus on developing the implementation of different end-to-end flow automation of the application in both platforms. This exercise will also help in understanding the fundamentals of test automation and allow to exploration of Selenium as well as Appium test automation tools.

2.2 Test Scope

2.2.1 In Scope

- End-to-end flow Automation Testing

2.2.2 Out of Scope

- Automation testing of dynamic elements
- Features involving Payment

2.3 Entrance / Exit Criteria

2.3.1 Entry Criteria

- All the testers are familiar i.e. have the basic knowledge of Groupon App
- The team should have procured the necessary devices/hardware to perform test automation across desired platforms.
- Team should work in a collaborative environment.
- For the mobile application, the testing hardware should be connected and the application should be installed in order to automate.

2.3.2 Exit Criteria

- All the end-to-end test automation flows are executed successfully.

- No high level critical defects are pending or unresolved.
- The Application is bug free.

2.4 Test Automation Requirements

- Two different end-to-end test automation flows for each of the platforms should be implemented.
- No defects and bugs for core functionality

2.5 Tools Used

2.5.1 Platform: Web

- Browser: Firefox 52.0.2 (64-bit)
- Selenium IDE 2.9.1
- Eclipse Java EE IDE for Web Developers Version: Neon.2 Release (4.6.2)
- Geckodriver
<https://github.com/mozilla/geckodriver/releases/download/v0.16.0/geckodriver-v0.16.0-macos.tar.gz>
- Code at : https://github.com/siddharth-daftari/groupon_testing_selenium

2.5.2 Platform: Android

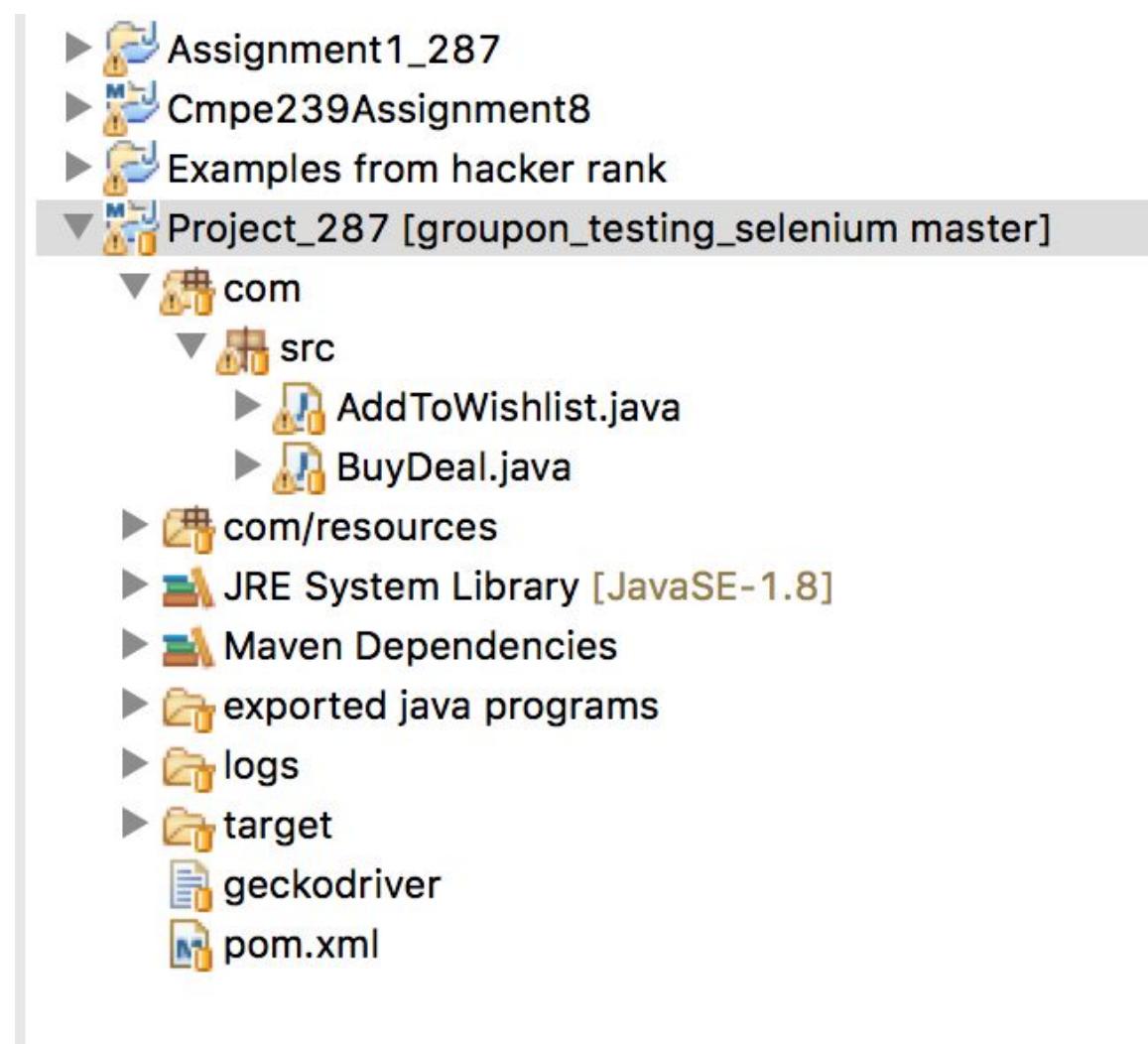
- Browser/Client: Android Native App
- Appium Version: v1.6.4
- Android Appium Webdriver: 1.17.1
- Android Studio: 2.2.3

3. Test Automation Design

3.1 Platform: Web

3.1.1 Steps to Run Selenium

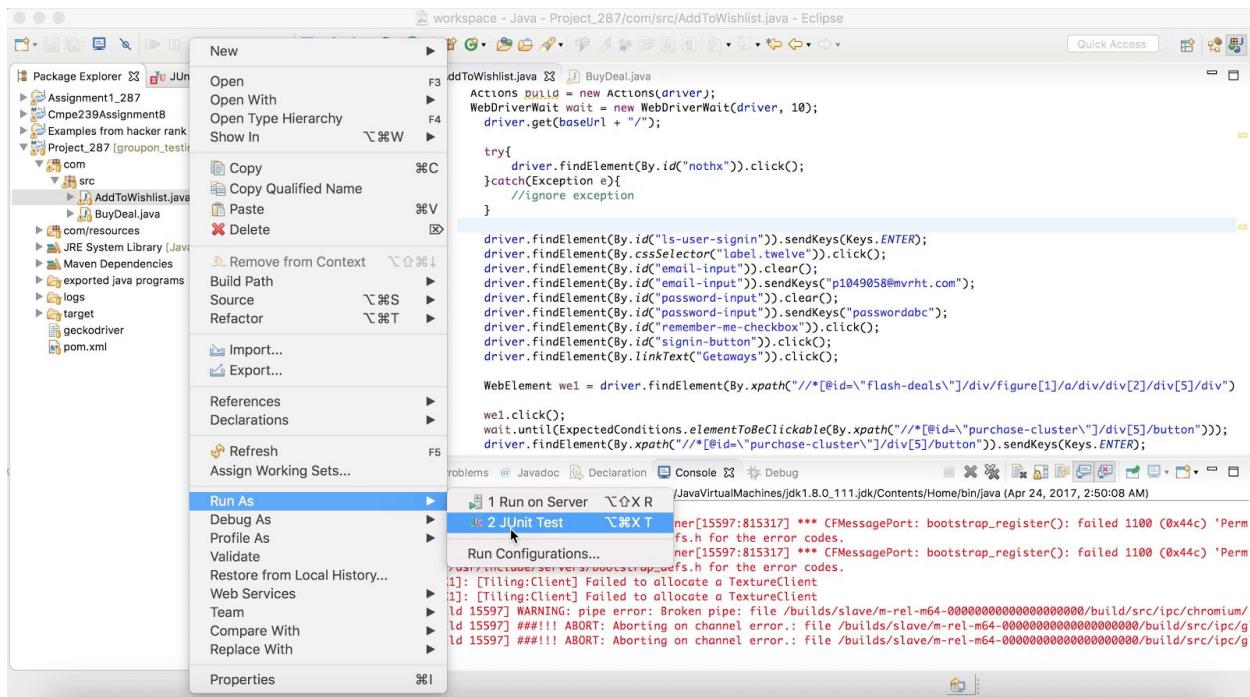
- Open eclipse IDE and install a maven plugin in eclipse.
- Create a new maven project in eclipse.
- Download geckodriver file.
- Place the geckodriver file in the project folder besides pom.xml file.
- Because we have implemented “Add to Wishlist” and “Buy Deal” end-to-end test automation flow, create two java files under the “src” folder: AddToWishlist.java and BuyDeal.java with code given in below in corresponding files.



3.1.2 End-To-End Test Automations

3.1.2.1 Add to Wishlist

1. Run the Add To Wishlist flow by selecting JUnit test in the Run As options.



```

workspace - Java - Project_287/com/src/AddToWishlist.java - Eclipse
Quick Access

Package Explorer JUnit JUnit
src.AddToWishlist - testBuyDeal
Runs: 1/1 Errors: 0 Failures: 0
src.AddToWishlist [Runner: JUnit 4]
testBuyDeal

AddToWishlist.java
Actions build = new Actions(driver);
WebDriverWait wait = new WebDriverWait(driver, 10);
driver.get(baseUrl + "/");
try{
    driver.findElement(By.id("nothx")).click();
} catch(Exception e){
    //Ignore exception
}

driver.findElement(By.id("ls-user-signin")).sendKeys(Keys.ENTER);
driver.findElement(By.cssSelector("label.twelve")).click();
driver.findElement(By.id("email-input")).clear();
driver.findElement(By.id("email-input")).sendKeys("p1049058@mvrht.com");
driver.findElement(By.id("password-input")).clear();
driver.findElement(By.id("password-input")).sendKeys("passwordabc");
driver.findElement(By.id("remember-me-checkbox")).click();
driver.findElement(By.id("signin-button")).click();
driver.findElement(By.linkText("Getaways")).click();

WebElement we1 = driver.findElement(By.xpath("//*[@id='flash-deals']/div/figure[1]/a/div/div[2]/div[5]/div"));
we1.click();
wait.until(ExpectedConditions.elementToBeClickable(By.xpath("//*[@id='purchase-cluster']/div[5]/button")));
driver.findElement(By.xpath("//*[@id='purchase-cluster']/div[5]/button")).sendKeys(Keys.ENTER);

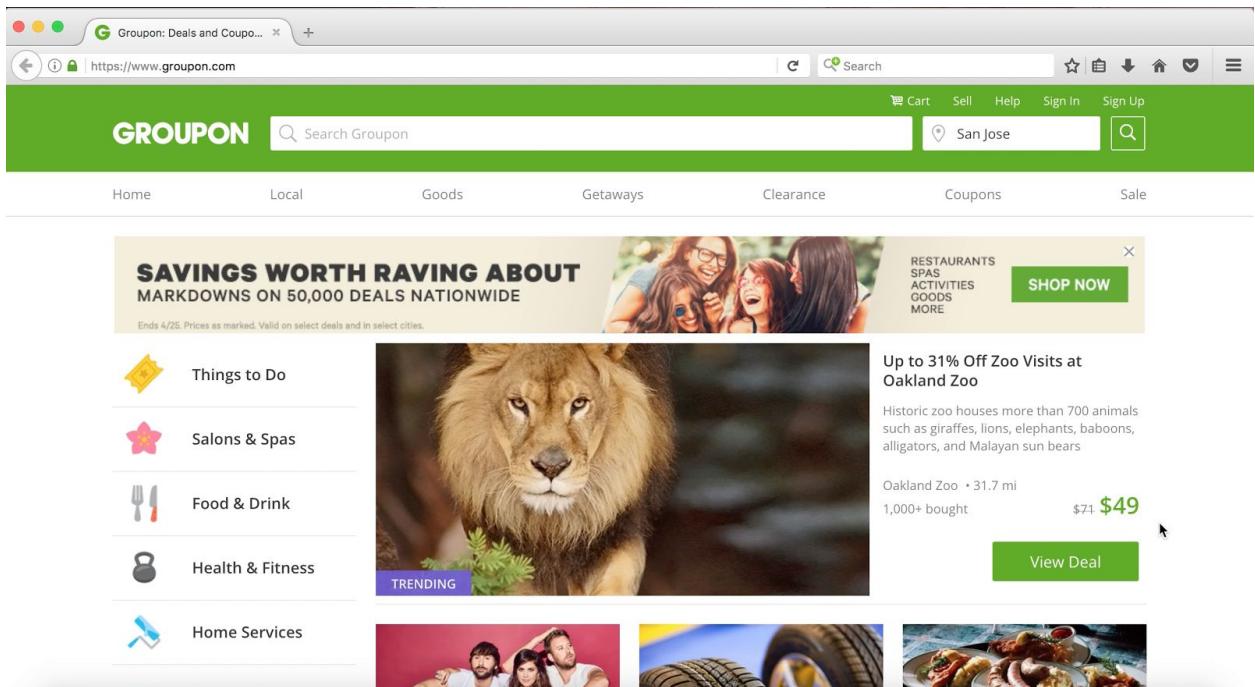
```

Problems @ Javadoc Declaration Console Debug

AddToWishlist [JUnit] /Library/Java/JavaVirtualMachines/jdk1.8.0_111.jdk/Contents/Home/bin/java (Apr 24, 2017, 2:54:02 AM)

1493027645842 geckodriver INFO Listening on 127.0.0.1:36100
1493027646743 mozprofile::profile INFO Using profile path /var/folders/hd/clhv5x251fn7s1gf7p7tfj1m0000gn/T/rust_mozprofile
1493027646745 geckodriver::marionette INFO Starting browser /Applications/Firefox.app/Contents/MacOS/firefox-bin
1493027646749 geckodriver::marionette INFO Connecting to Marionette on localhost:54506
1493027649411 Marionette INFO Listening on port 54506

- The automation script opens the "www.groupon.com" website in the browser.

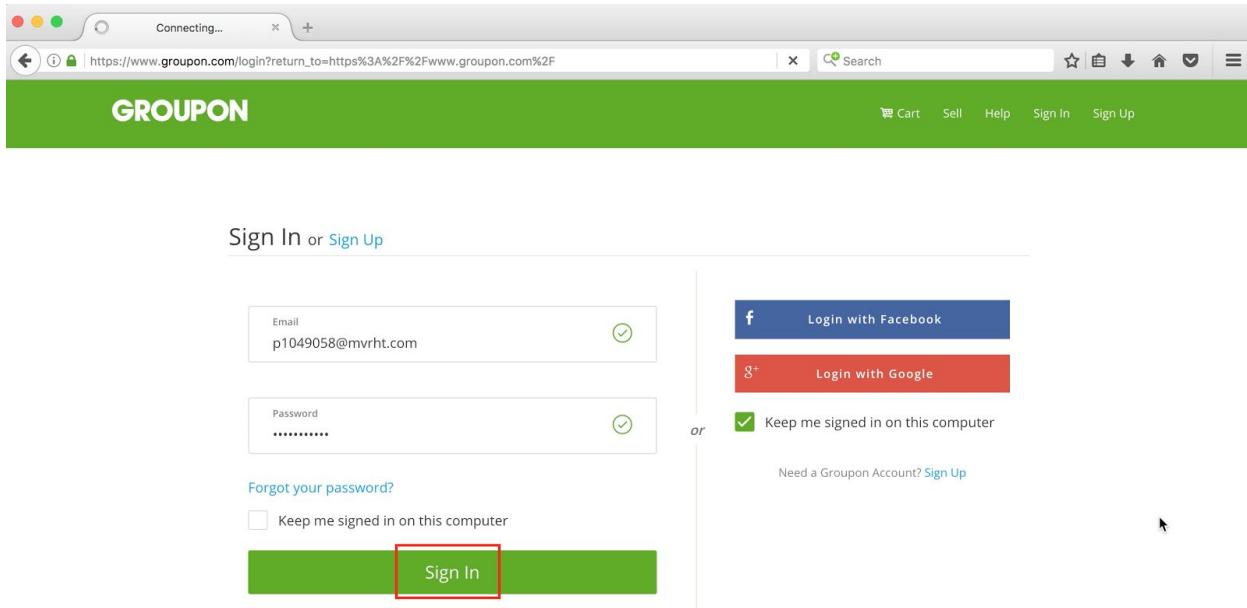


- The automation script clicks on the "Sign In" button in the Groupon web application to open the Login page.

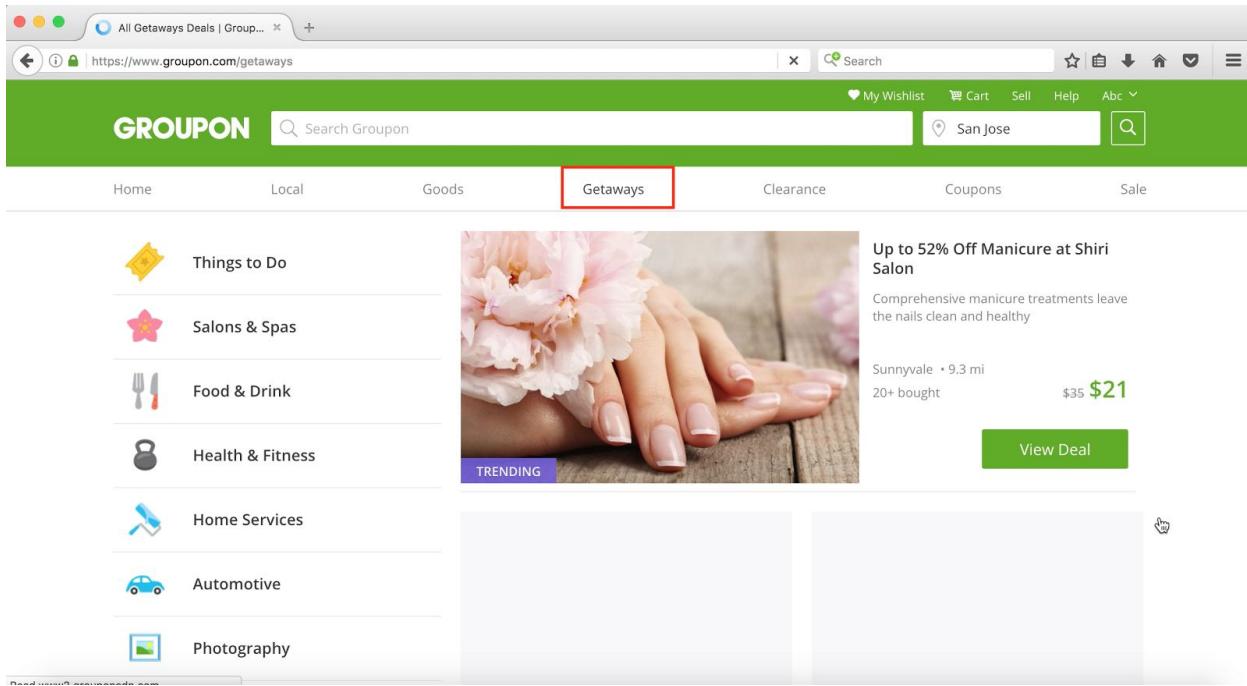
The screenshot shows the Groupon homepage. At the top, there is a navigation bar with links for 'Cart', 'Sell', 'Help', 'Sign In' (which is highlighted with a red box), and 'Sign Up'. Below the navigation bar is a search bar with the placeholder 'Search Groupon' and a location dropdown set to 'San Jose'. The main content area features a banner with the text 'SAVINGS WORTH RAVING ABOUT' and 'MARKDOWNS ON 50,000 DEALS NATIONWIDE'. It includes a photo of four people laughing and a call-to-action button 'SHOP NOW'. To the left, there is a sidebar with categories: 'Things to Do', 'Salons & Spas', 'Food & Drink', 'Health & Fitness', and 'Home Services'. A large image of a lion's face is prominently displayed. On the right, there is a deal for 'Up to 31% Off Zoo Visits at Oakland Zoo' with a price of '\$71 \$49' and a 'View Deal' button.

The screenshot shows the Groupon login page. The top navigation bar is identical to the homepage, with 'Sign In' and 'Sign Up' being the most prominent links. The main content area features a 'Sign In or Sign Up' header. Below this are fields for 'Email' and 'Password', both with accompanying icons. There is also a link 'Forgot your password?' and a checkbox for 'Keep me signed in on this computer'. To the right, there are social media login options for 'Login with Facebook' and 'Login with Google', along with a checkbox for 'Keep me signed in on this computer'. A note at the bottom says 'Need a Groupon Account? [Sign Up](#)'.

4. Email and password are given and the "Sign In" button is clicked to Login into the user's account.



5. The automation code Logs in successfully into the user's account. Then, "Getaways" button is clicked.



6. The Getaways page is opened. In this page, "View Deal" button is clicked to view a particular deal.

Discover 1,000s of Top-Rated Getaways

Destination Dates Search Getaways

Browse Flash Deals

All Destinations

- USA & Canada
- Northeast
- Southeast
- Midwest
- Mountain
- Pacific
- Southwest
- Canada
- Mexico, the Caribbean & Latin America

All-Inclusive Dominican Vacation
4, 6, or 7 Night All-Inclusive Vista Sol Punt...
Punta Cana, Dominican Republic
80+ bought \$839 **\$599**

View Deal

7. A particular deal is displayed. In this page, “Add to wishlist” button is clicked by the automation script to add that deal to the user’s wishlist.

4, 6, or 7 Night All-Inclusive Vista Sol Punta Cana Trip with Nonstop Air from Vacation Express - Punta Cana, Dominican Republic

4, 6, or 7 Night Vista Sol Punta Cana Trip with Nonstop Airfare. Price per Person Based on Double Occupancy.

FROM **\$599**

Buy!

LIVE CHAT

OPTIONS

- Select Departure City
- Select Number of Nights
- Select Dates

Give as a Gift

33,500+ viewed today

Add to WISHLIST

8. To check whether the deal is added to the wish list, the automation code clicks the “My Wishlist” button on the slightly top right corner. From the below screenshot, the deal is added in the user’s wish list.

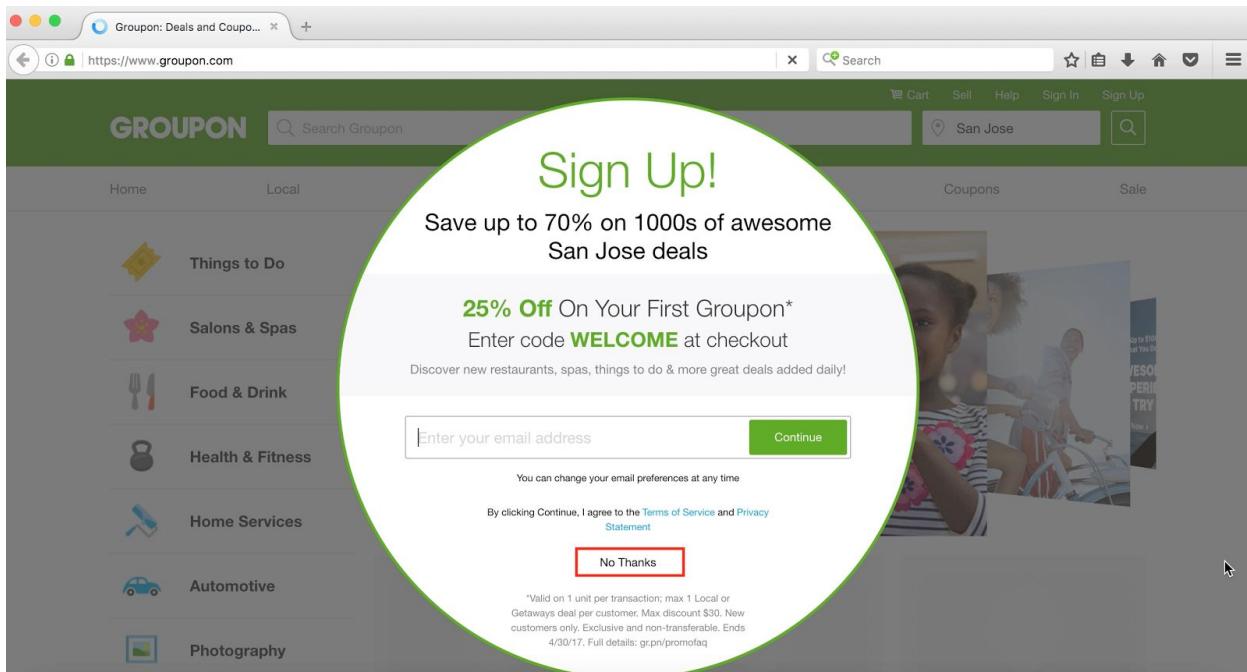
The screenshot shows the Groupon website's wishlist section. At the top, there's a navigation bar with links for Home, Local, Goods, Getaways, Clearance, Coupons, and Sale. A search bar is also present. The main content area is titled "My Wishlist" and displays a deal for "Option 1: Miami (MIA): 4 Nights, Depart 5/11, 6/1, 6/8". The deal includes a small image of a beach, a price of \$839.00, and a discounted price of \$599.00. It also mentions "Ends in 9 days" and "4, 6, or 7 Night All-Inclusive Vista Sol Punta Cana Trip with Nonstop Air from Vacation Express". There are "Remove" and "Sort by: Date Added | Expiration Date" buttons. Below the wishlist, there are sections for Company information, Work with Groupon, More links, and social media follow options. There are also promotional boxes for the Groupon mobile app and Merchant services.

3.1.2.2 Buy Deal

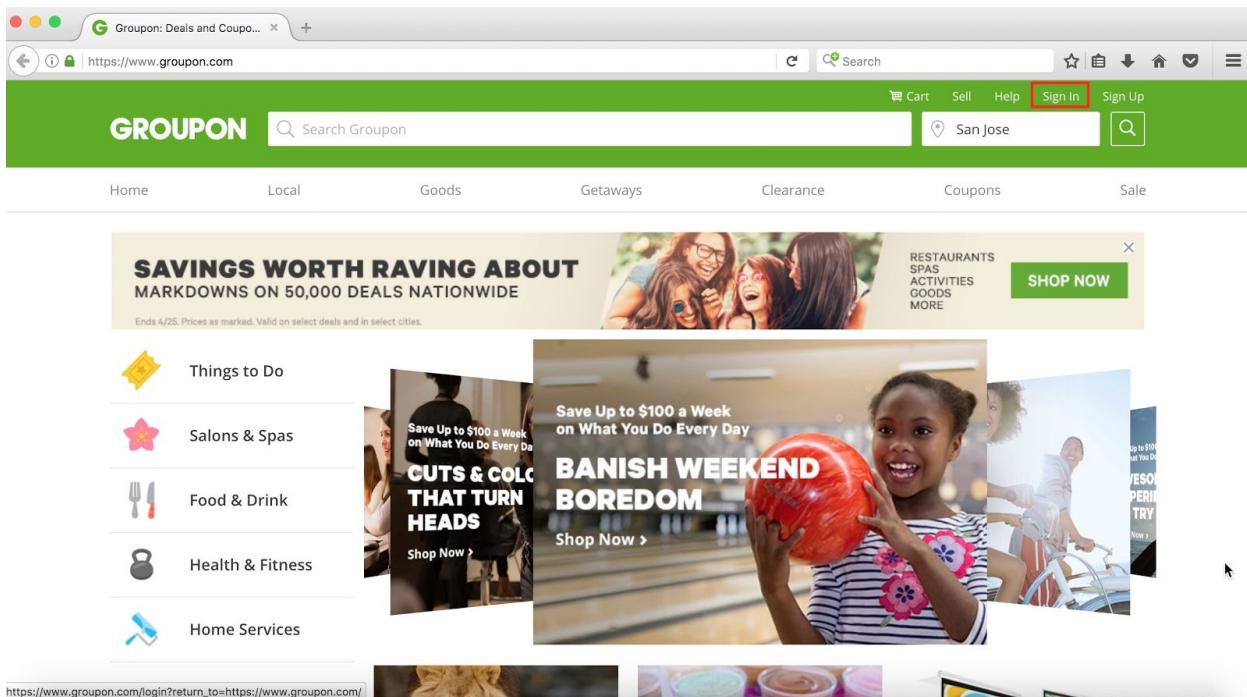
1. Run the Buy Deal flow by selecting JUnit test in the Run As options.

The screenshot shows the Eclipse IDE interface with the "BuyDeal.java" file selected in the Package Explorer. A context menu is open over the file, with the "Run As" option highlighted. The "JUnit Test" option under "Run As" is also highlighted. The status bar at the bottom indicates "1 Run on Server" and "JUnit Test". The Eclipse toolbar and various tool tabs are visible along the top and right edges of the interface.

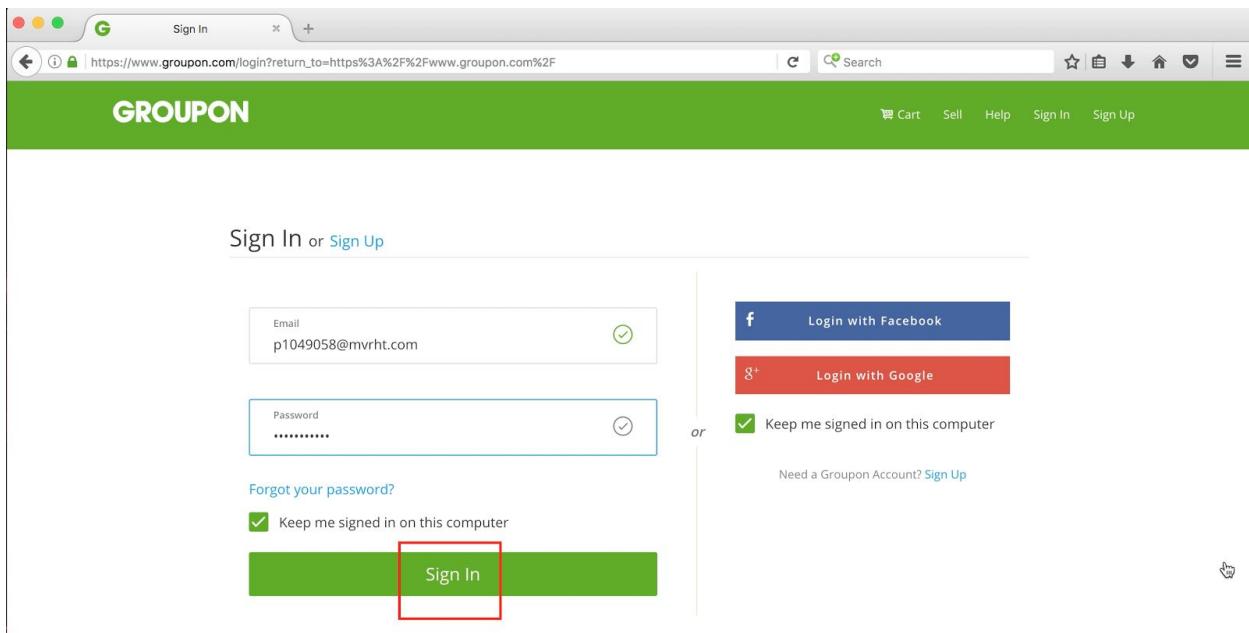
2. The automation script opens the “www.groupon.com” in the browser. The automation script clicks on the “No Thanks” button.



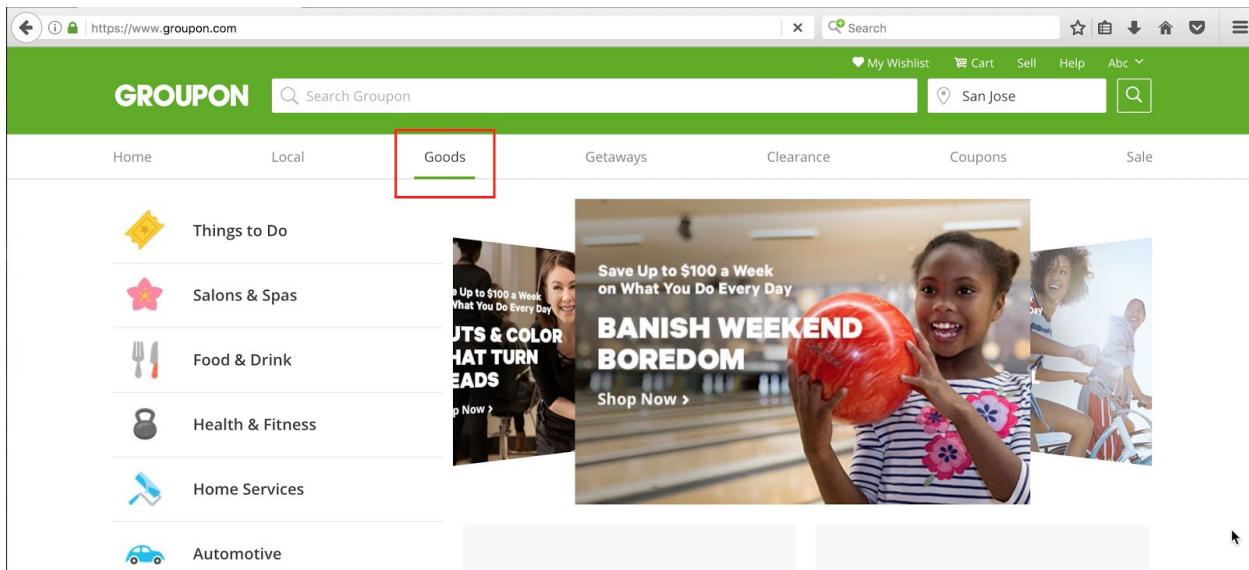
3. The automation script clicks the “Sign In” button in the Groupon web application to open the Login page.



4. Email and password are given and the "Sign In" button is clicked to Login into the user's account.



5. The automation code Logs in successfully into the user's account. Then, "Goods" button is clicked.



6. The automation script then clicks on the "For the Home" options from the categories. Then, the script selects the "Art" option to see particular deals.

For the Home - Deals & Co... https://www.groupon.com/goods/for-the-home

GROUPON Search Groupon Zip Code, Neighborhood

Home Local Goods Getaways Clearance Coupons Sale

FREE Shipping with \$34.99 Purchase | FREE Returns

Goods > For the Home

For the Home

Sort by Relevance

Categories

- < All Goods
- For the Home (52568)**

Discount Types

- Clearance

Brands

- Trademark Fine Art (19947)



Natural Crystal Himalayan Rock Salt Lamps

Rock salt lamps made with natural Himalayan salt crystals help cleanse the air, minimizing irritation caused by airborne...

10,000+ bought \$49.99 **\$22.99**
★ ★ ★ ★ (1,193) Sale Ends 4/25

View Deal

For the Home - Deals & Co... https://www.groupon.com/goods/for-the-home

GROUPON Search Groupon San Jose

Home Local Goods Getaways Clearance Coupons Sale

SAVINGS WORTH RAVING ABOUT
MARKDOWNS ON 100S OF GOODS

Ends 4/25. Prices as marked. Valid on select deals and in select cities.

Home Electronics Health & Beauty More **SHOP NOW**

FREE Shipping with \$34.99 Purchase | FREE Returns

Goods > For the Home

For the Home

Sort by Customer Ratings

Sort by Relevance

Categories

- < All Goods
- For the Home (52568)**
- Art (23187)**
- Kitchen & Dining (7128)



Natural Crystal Himalayan Rock Salt Lamps

Rock salt lamps made with natural Himalayan salt crystals help cleanse the air, minimizing irritation caused by airborne...

10,000+ bought \$49.99 **\$22.99**
★ ★ ★ ★ (1,193) Sale Ends 4/25

7. The automation script clicks on a particular deal's "View Deal" to view that deal.

SAVINGS WORTH RAVING ABOUT
MARKDOWNS ON 100S OF GOODS

Ends 4/25. Prices as marked. Valid on select deals and in select cities.

FREE Shipping with \$34.99 Purchase | FREE Returns

Goods > For the Home > Art

Art

Categories

- All Goods
- For the Home (5256)
- Art (23187)
- Prints & Decals (21720)
- Photography (874)

Sort by Customer Ratings Sort by Relevance

Carrie Schmitt 'Illumination' Matted Birch Framed Art

Carrie Schmitt 'Illumination' Matted Birch Framed Art

\$89.99 **\$24.99**

<https://www.groupon.com/deals/gg-cm-carrie-schmitt-illumination-matted-birch-framed-art>

- The deal page will get opened. The automation script then select a particular size “11 X 11” and clicks on the “Buy” button. So, the “Buy” button will be disabled and its text will get changed to “Processing... Please wait”.

Carrie Schmitt 'Illumination' Matted Birch Framed Art

FROM
\$24.99

Discount 72%

Select Size-FreeText

11 x 11

Buy

The screenshot shows a web browser window for Groupon's cart page at <https://www.groupon.com/cart>. The top navigation bar includes links for Home, Local, Goods, Getaways, Clearance, Coupons, and Sale. A promotional banner for "SAVINGS WORTH RAVING ABOUT" is visible, along with a search bar and a location dropdown set to San Jose. The main content area displays a product: "Carrie Schmitt 'Illumination' Matted Birch Framed Art". The product image is a colorful floral painting. Pricing information shows a原价 of \$89.99 and a discounted price of \$24.99. A "Discount 72%" badge is present. Below the price, there are dropdown menus for "11 x 11" and "Black". A green button at the bottom right of the product card says "Processing... Buy please wait". The status bar at the bottom left indicates "Waiting for www.groupon.com...".

9. The cart page will get opened. Then, the automation script clicks on the “Proceed to Checkout” button.

The screenshot shows the Groupon Checkout page at <https://www.groupon.com/cart>. The top navigation bar is identical to the previous screenshot. The main content area displays a summary of the cart: "You have 1 item in your cart!". The "Your Cart" section shows the same product as before: "Carrie Schmitt 'Illumination' Matted Birch Framed Art, Black, 11 x 11" with a price of \$24.99. The "Cart Summary" section provides a breakdown of costs: Subtotal (1 item) \$24.99, Estimated Shipping See in Checkout, Taxes & Fees See in Checkout, and a total Cart Subtotal of \$24.99. A large green button labeled "Proceed To Checkout" is highlighted with a red border. Below the cart summary, there is a "Recently Viewed Deals" section with three items: "All-Inclusive Dominican", "Japan Vacation with", and "Great Wolf Lodge in".

10. Then, the Checkout page will get opened.

The screenshot shows a Groupon checkout page titled "Your Purchase". At the top, there's a navigation bar with links for "Sell", "Help", "Sign In", and "Sign Up". Below the navigation, a green header bar says "GROUPON". The main content area has a heading "We need some information" and a sub-instruction "Please fill in your shipping and billing information before finalizing your order". On the left, a "Payment Method" section offers "PayPal" (unchecked) and "Credit Card" (checked). The "Credit Card" section includes fields for "Name on Card" (empty), "Card Number" (empty), "Expiration Date" (empty), and "Security Code" (empty). To the right, a "Secure Transaction" box contains an "Order Summary" table:

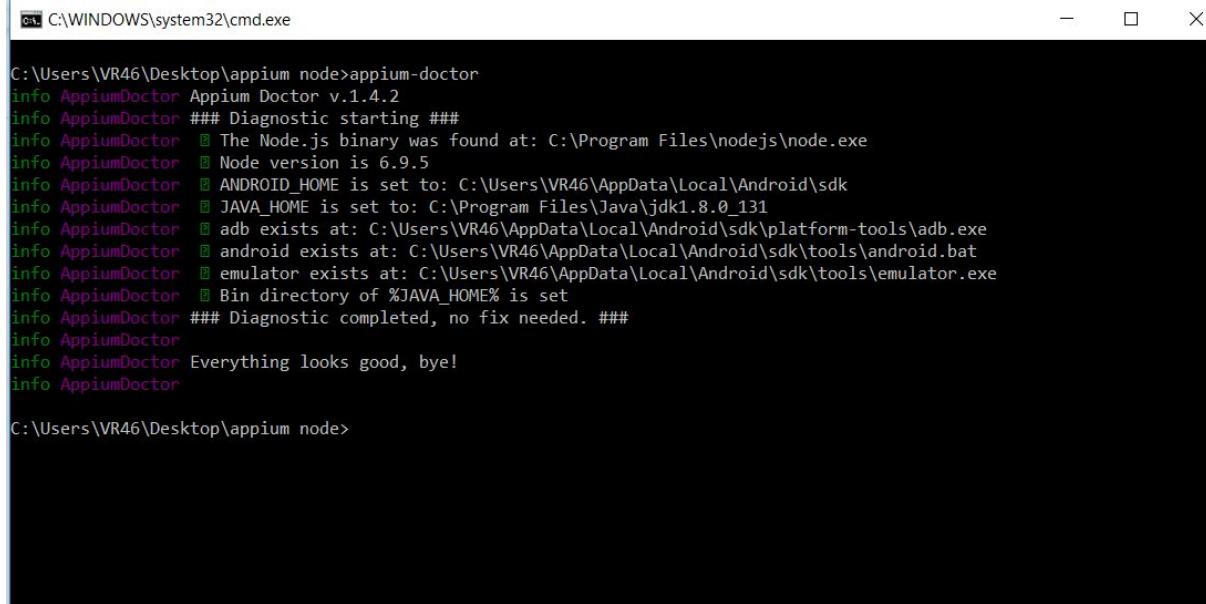
Order Summary	
Subtotal	\$24.99
Shipping & Handling	\$3.99
Promo or Gift Code	
Order Total:	\$28.98

Below the summary, a note states: "By clicking below I accept the current [Terms of Use](#) and [Privacy Statement](#)". A large green "Place Order" button is at the bottom.

3.2 Platform: Android

3.2.1 Steps to Run Appium

- Install Appium Server and one of the client (Java, Node.Js, Ruby etc).
- The Appium desktop can be downloaded which comes with the Appium server prepacked as a windows desktop application.
- Check that all the Appium dependencies exist on your machine using the Appium NPM module “appium-doctor”. This is a pre-setup step that shows you any errors that you can avoid during setup.
- Run the following command:
 - `> npm install -g "appium-doctor"`
 - `> appium-doctor`
- If the screen shows all the dependencies as below, the system is ready for appium testing :

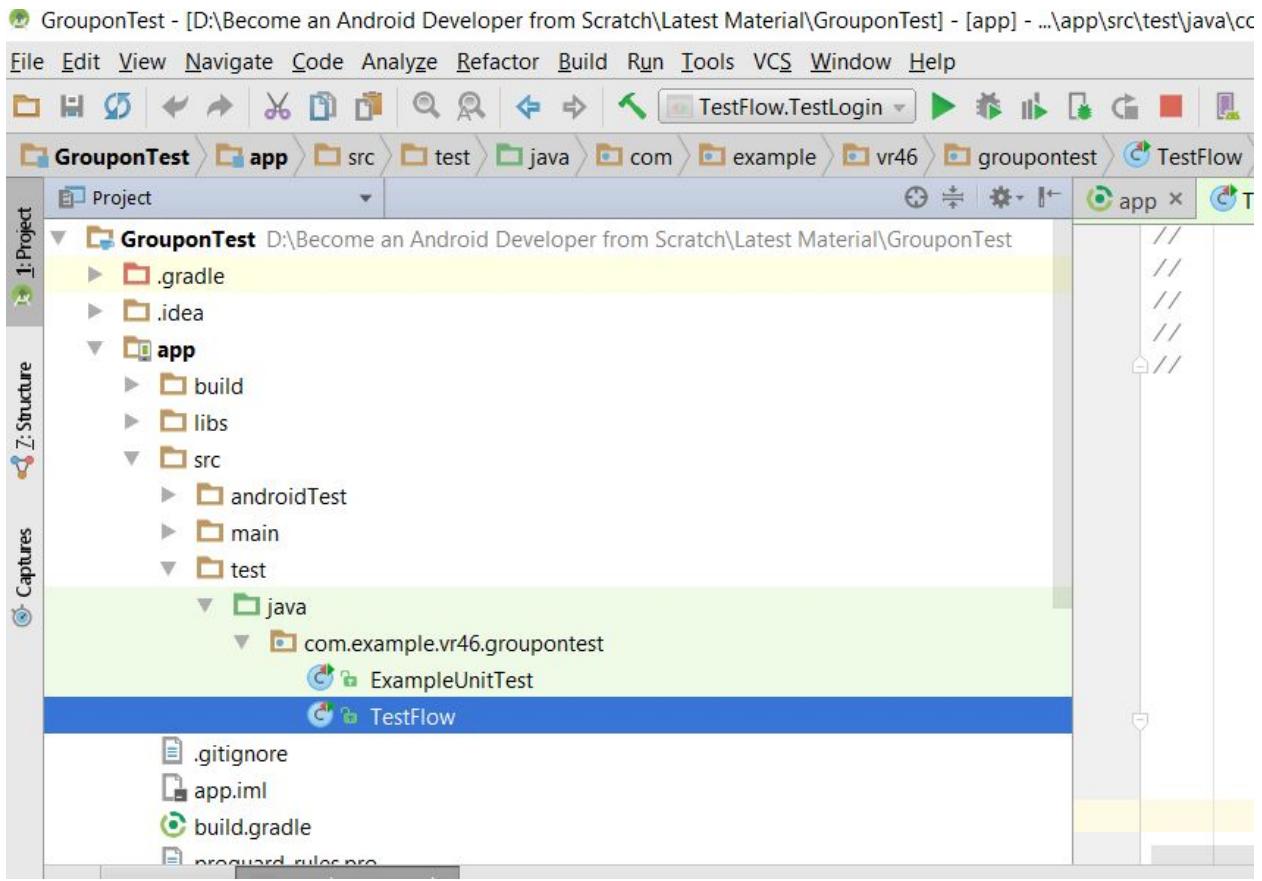


```
C:\WINDOWS\system32\cmd.exe
C:\Users\VR46\Desktop\appium node>appium-doctor
info AppiumDoctor Appium Doctor v.1.4.2
info AppiumDoctor ### Diagnostic starting ####
info AppiumDoctor  The Node.js binary was found at: C:\Program Files\nodejs\node.exe
info AppiumDoctor  Node version is 6.9.5
info AppiumDoctor  ANDROID_HOME is set to: C:\Users\VR46\AppData\Local\Android\sdk
info AppiumDoctor  JAVA_HOME is set to: C:\Program Files\Java\jdk1.8.0_131
info AppiumDoctor  adb exists at: C:\Users\VR46\AppData\Local\Android\sdk\platform-tools\adb.exe
info AppiumDoctor  android exists at: C:\Users\VR46\AppData\Local\Android\sdk\tools\android.bat
info AppiumDoctor  emulator exists at: C:\Users\VR46\AppData\Local\Android\sdk\tools\emulator.exe
info AppiumDoctor  Bin directory of %JAVA_HOME% is set
info AppiumDoctor  ### Diagnostic completed, no fix needed. ####
info AppiumDoctor
info AppiumDoctor Everything looks good, bye!
info AppiumDoctor

C:\Users\VR46\Desktop\appium node>
```

- Once Appium is set up, you need to create a client environment and load the appium libraries. Appium framework can be used using a variety of programming languages like Node.js, Java, Ruby etc.
- We will be testing using Java, hence we need to download the Java client libraries for appium. For this, we will use the Gradle Build tool already embedded in Android Studio. Hence, Use the SRC/Test folder of android project for testing appium flows.
- Following steps are given to setup the Appium environment:

1. Create a new Android Project



- Organize the project hierarchy such that your flow tests are written in src/test folder. All the Jars for Java Appium framework need to be added to gradle build file.

```

compile group: 'org.seleniumhq.selenium', name: 'selenium-java', version: '2.53.0'

compile group: 'org.seleniumhq.selenium', name: 'selenium-chrome-driver', version: '2.53.1'

compile group: 'org.seleniumhq.selenium', name: 'selenium-firefox-driver', version: '2.53.1'

compile group: 'org.seleniumhq.selenium', name: 'selenium-remote-driver', version: '2.53.1'

```

- Before we run any tests, we need to configure the UI capabilities of the emulator



```

    WebDriver driver;

    @Before
    public void setUp() throws MalformedURLException {
        // Created object of DesiredCapabilities class.
        DesiredCapabilities capabilities = new DesiredCapabilities();

        capabilities.setCapability("deviceName", "new");

        capabilities.setCapability(CapabilityType.BROWSER_NAME, "Android");

        capabilities.setCapability("platformName", "Android");

        capabilities.setCapability("appPackage", "com.groupon");

        capabilities.setCapability("appActivity", "com.groupon.activity.TodaysDeal");

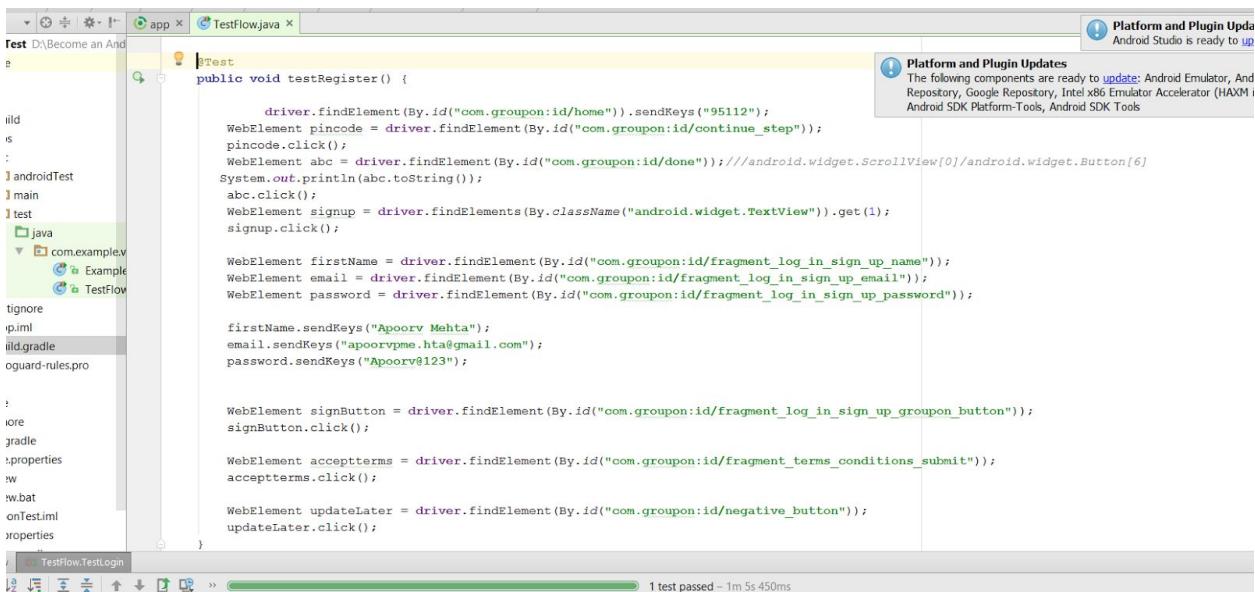
        driver = new RemoteWebDriver(new URL("http://127.0.0.1:4723/wd/hub"), capabilities);
        driver.manage().timeouts().implicitlyWait(15, TimeUnit.SECONDS);
    }
}

```

3.2.2 End-To-End Test Automations

3.2.2.1 Signup using Email address

1. Below is the screenshot of the code for the Signup end-to-end flow.



```

Test D:\Become an And
e
:
] androidTest
] main
] test
[ java
  com.example.v
    Example
    TestFlow
ignore
ip.gradle
ilid.gradle
oguard-rules.pro

;
ore
gradle
tproperties
w
w.bat
onTestiml
xproperties
}

public void testRegister() {
    driver.findElement(By.id("com.groupon:id/home")).sendKeys("95112");
    WebElement pincode = driver.findElement(By.id("com.groupon:id/continue_step"));
    pincode.click();
    WebElement abc = driver.findElement(By.id("com.groupon:id/done")); // android.widget.ScrollView[0]/android.widget.Button[6]
    System.out.println(abc.toString());
    abc.click();
    WebElement signup = driver.findElements(By.className("android.widget.TextView")).get(1);
    signup.click();

    WebElement firstName = driver.findElement(By.id("com.groupon:id/fragment_log_in_sign_up_name"));
    WebElement email = driver.findElement(By.id("com.groupon:id/fragment_log_in_sign_up_email"));
    WebElement password = driver.findElement(By.id("com.groupon:id/fragment_log_in_sign_up_password"));

    firstName.sendKeys("Apoorv Mehta");
    email.sendKeys("apoorvpme.htha@gmail.com");
    password.sendKeys("Apoorv@123");

    WebElement signButton = driver.findElement(By.id("com.groupon:id/fragment_log_in_sign_up_groupon_button"));
    signButton.click();

    WebElement acceptterms = driver.findElement(By.id("com.groupon:id/fragment_terms_conditions_submit"));
    acceptterms.click();

    WebElement updateLater = driver.findElement(By.id("com.groupon:id/negative_button"));
    updateLater.click();
}

```

Platform and Plugin Updates
The following components are ready to update: Android Emulator, And Repository, Google Repository, Intel x86 Emulator Accelerator (HAXM) and Android SDK Platform-Tools, Android SDK Tools

2. Run the testRegister() method



```

    }
}

@Test
public void testRegister() {
    // Run 'testRegister()' Ctrl+Shift+F10
    // Debug 'testRegister()'
    // Run 'testRegister()' with Coverage
    // TE/CLR button to clear result text box before running test.

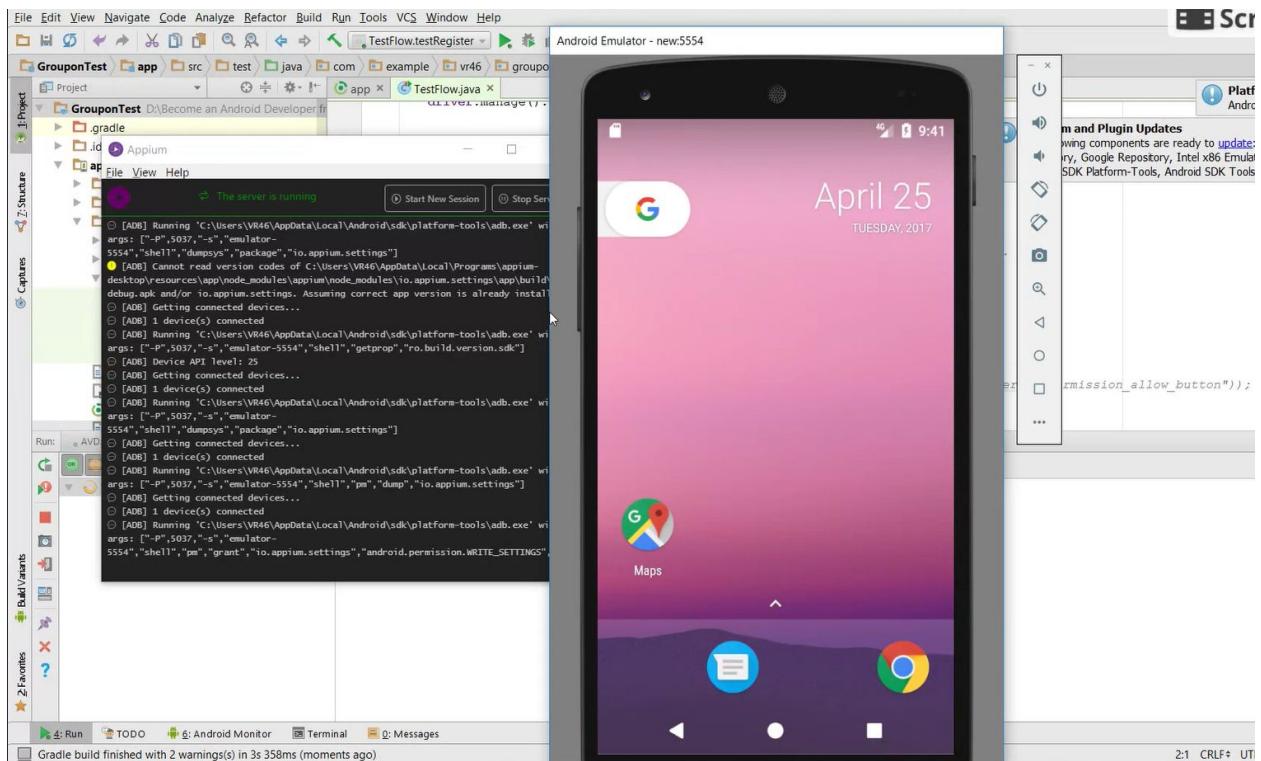
    // Click on number 2 button.
    driver.findElement(By.id("digit_7")).click();

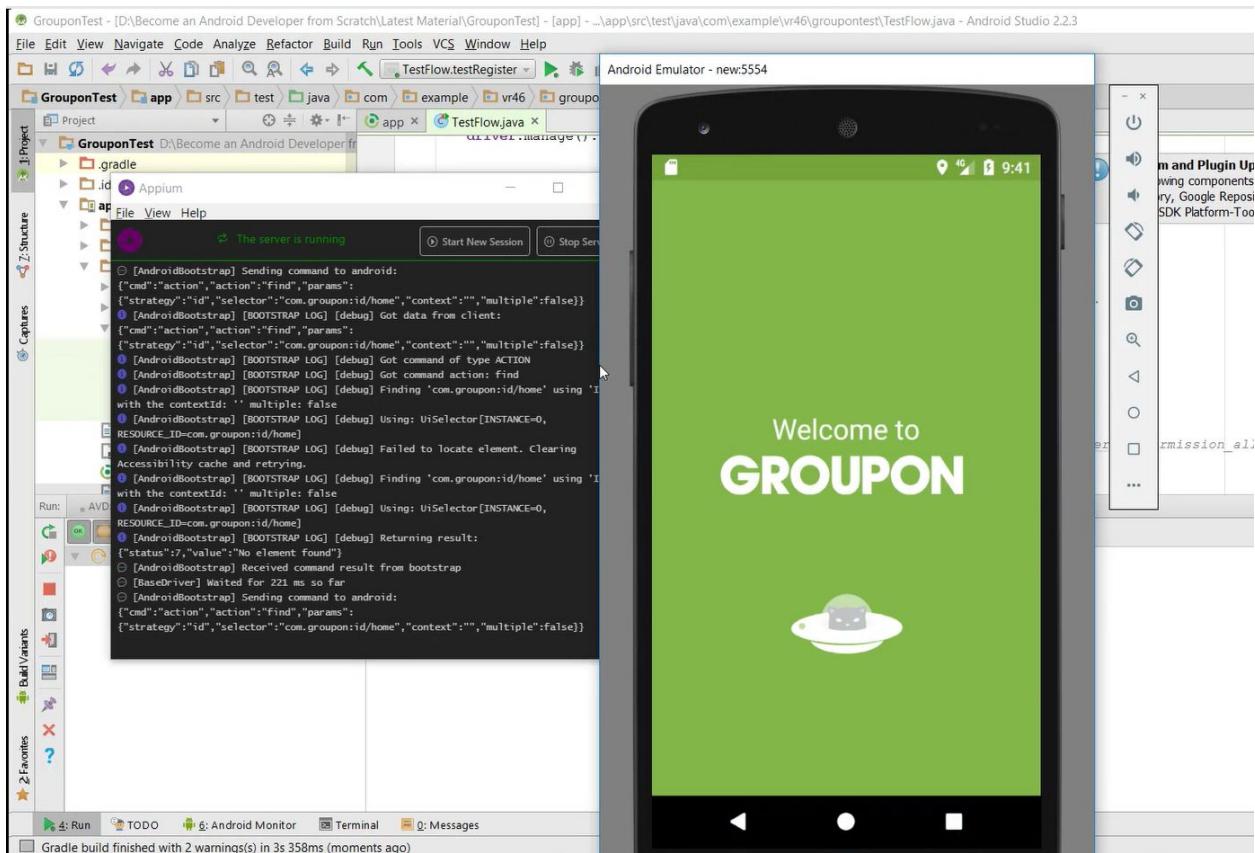
    driver.manage().timeouts().implicitlyWait(30, TimeUnit.SECONDS);

    /* WebElement myButton = driver.findElement(By.id("com.android.packageinstaller:id/permission_allow_button"));
    if(myButton != null){
        myButton.click();
    }
}

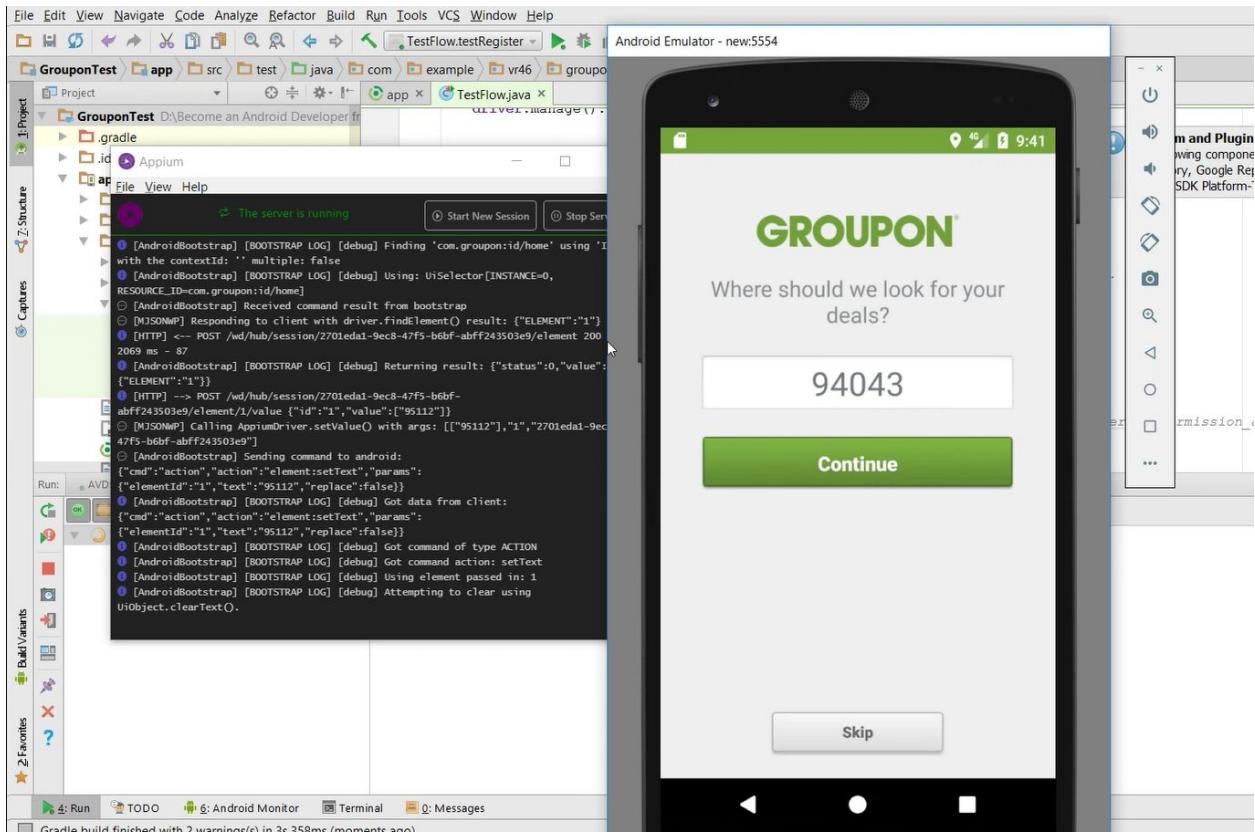
```

3. The package name as well as the activity name are given in the automation script. So, the automation script will open the Groupon application.

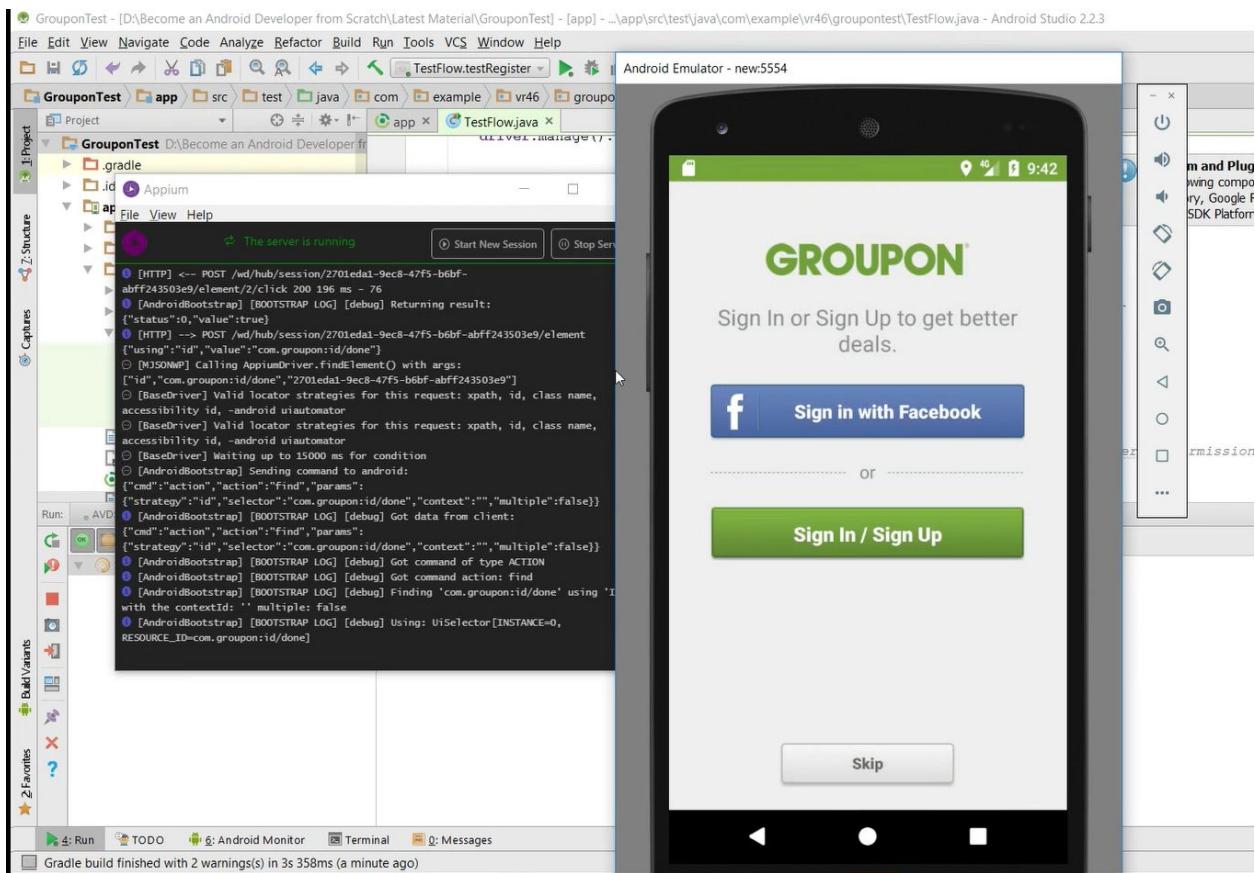




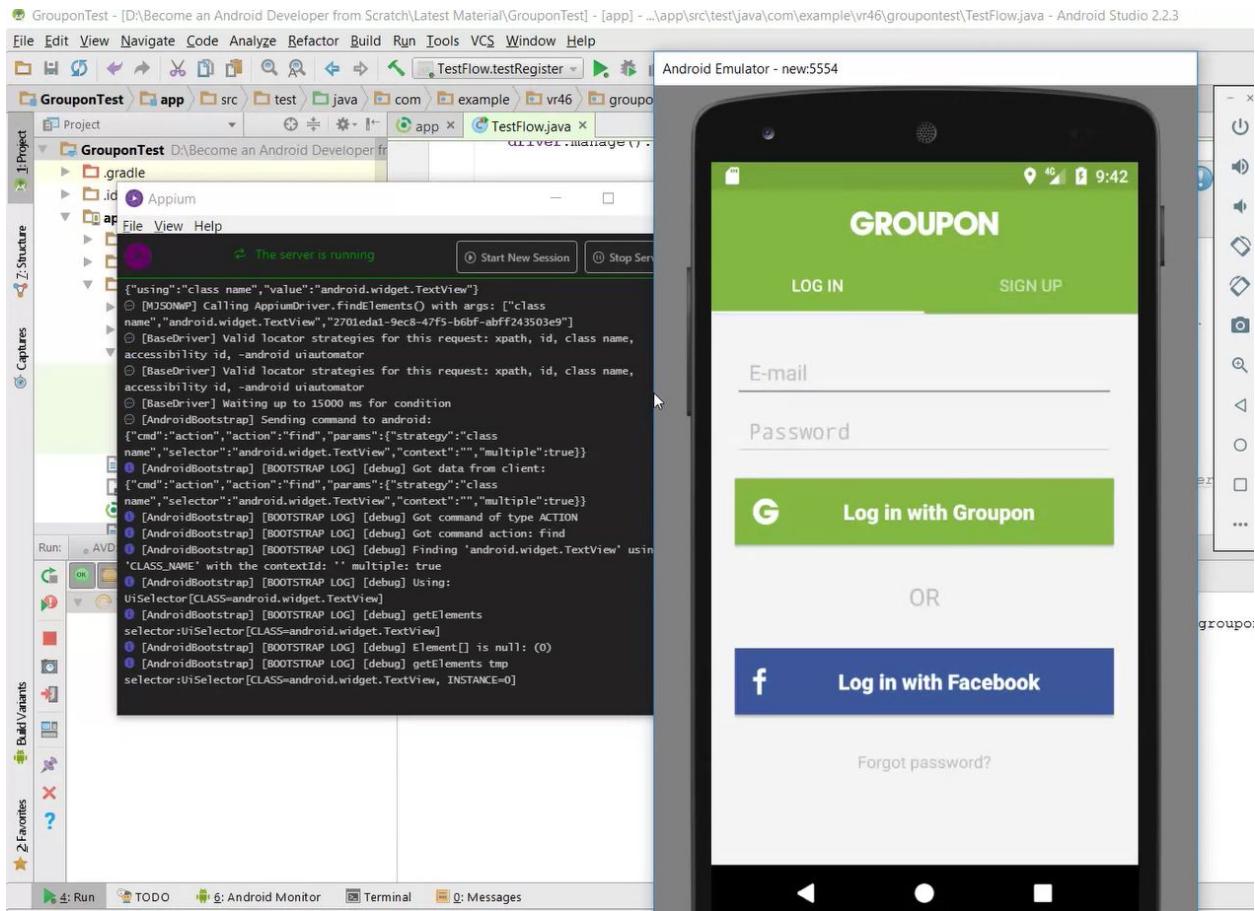
4. The script gives the pin code and click on “Continue” button.

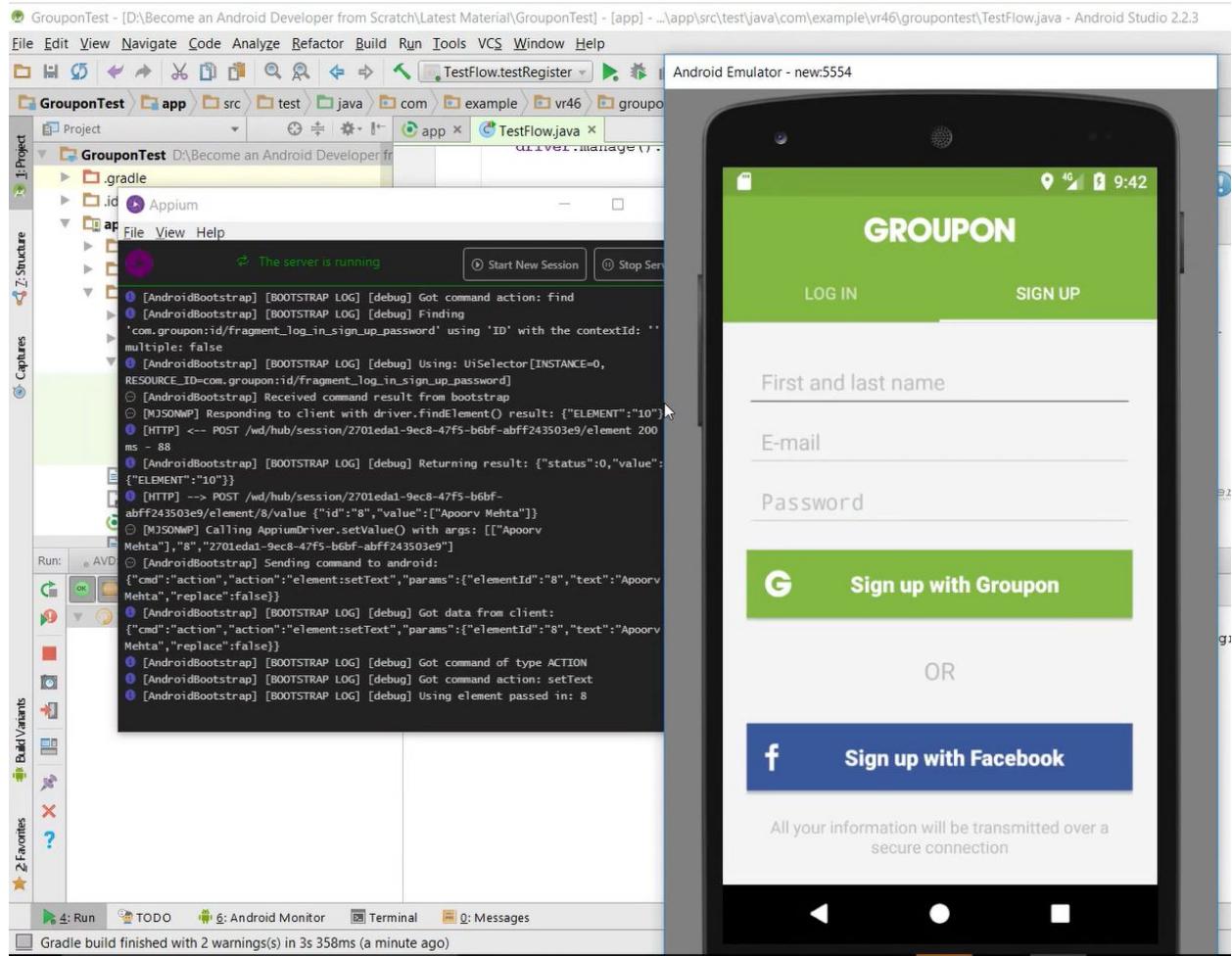


5. The Groupon application displays several options to login or signup. So, the script will select the “Sign In/ Sign Up” option.

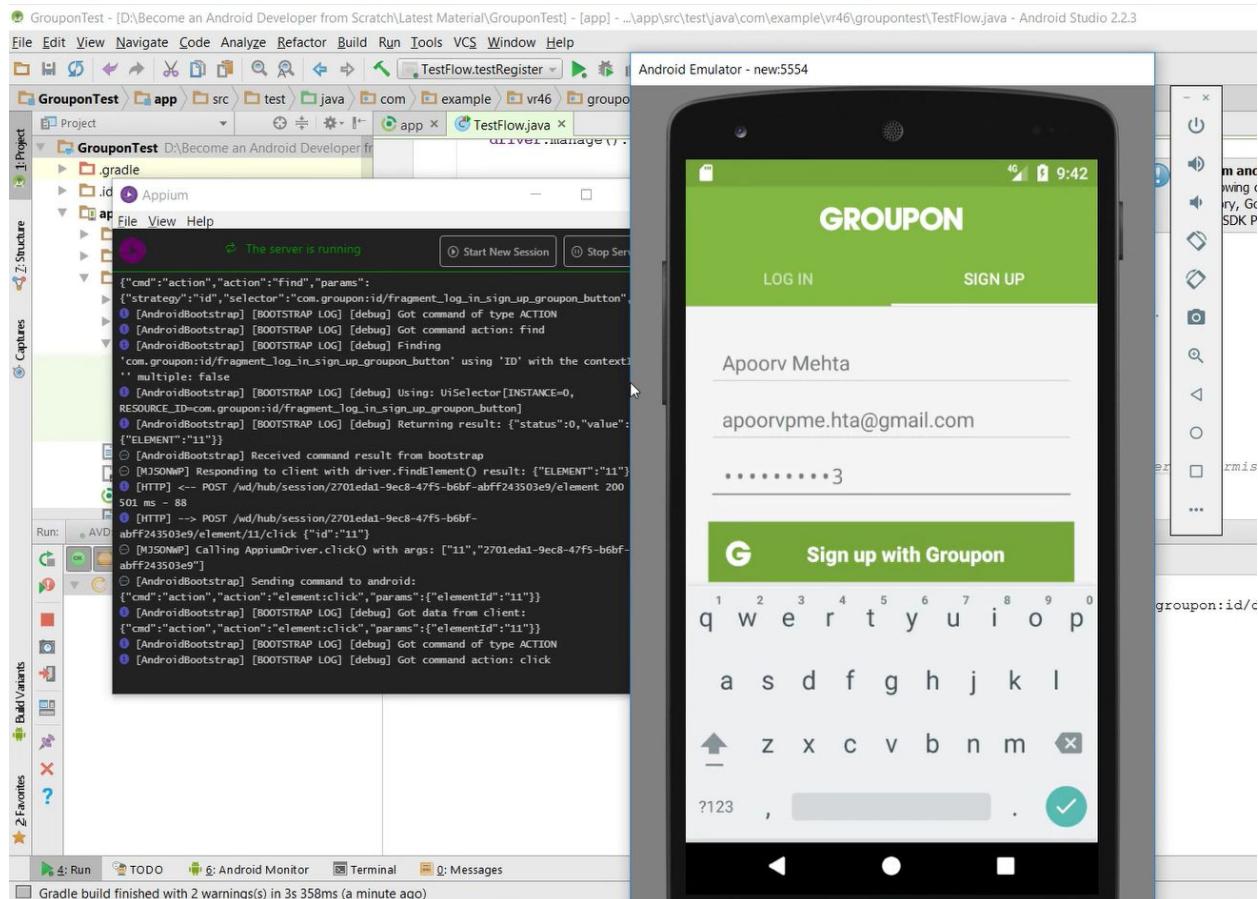


6. The Sign Up/ Login page gets opens. The script selects the Sign Up tab in the application.

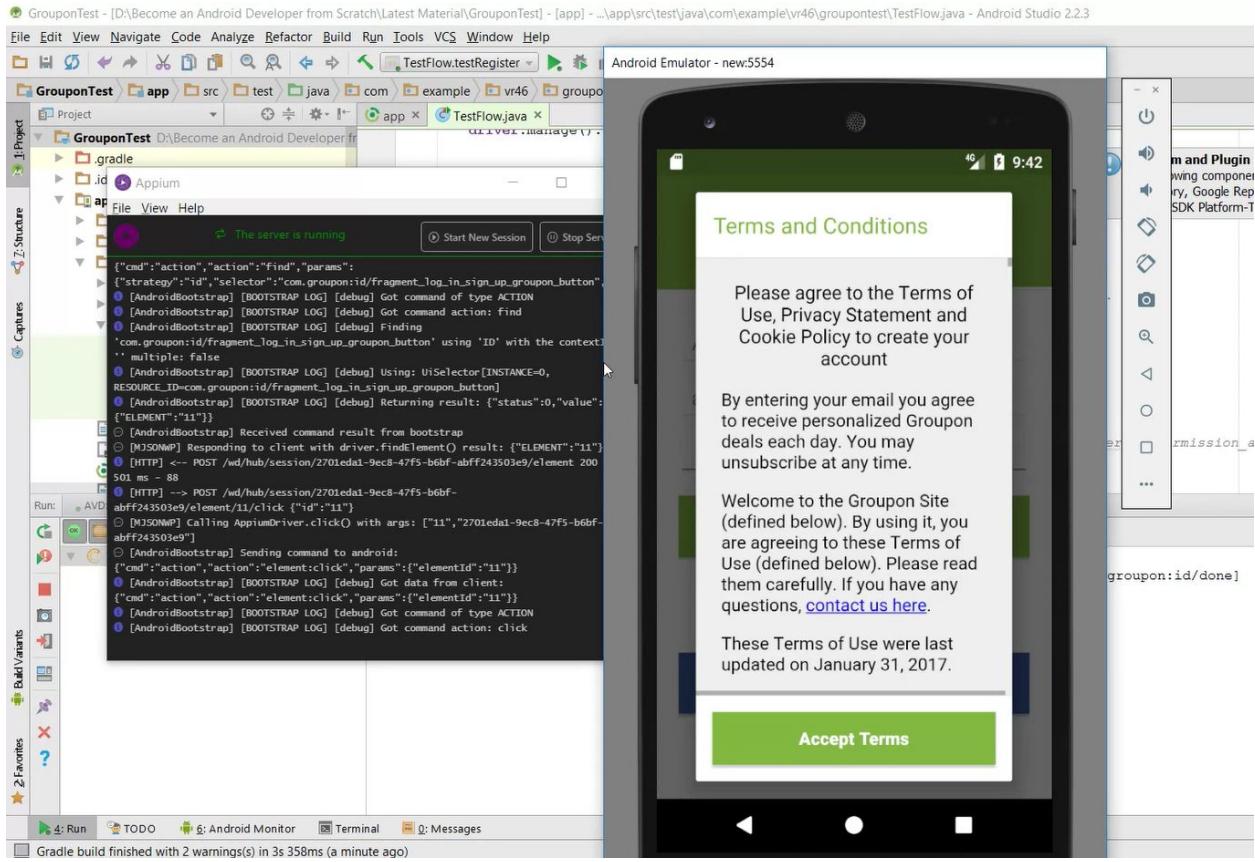




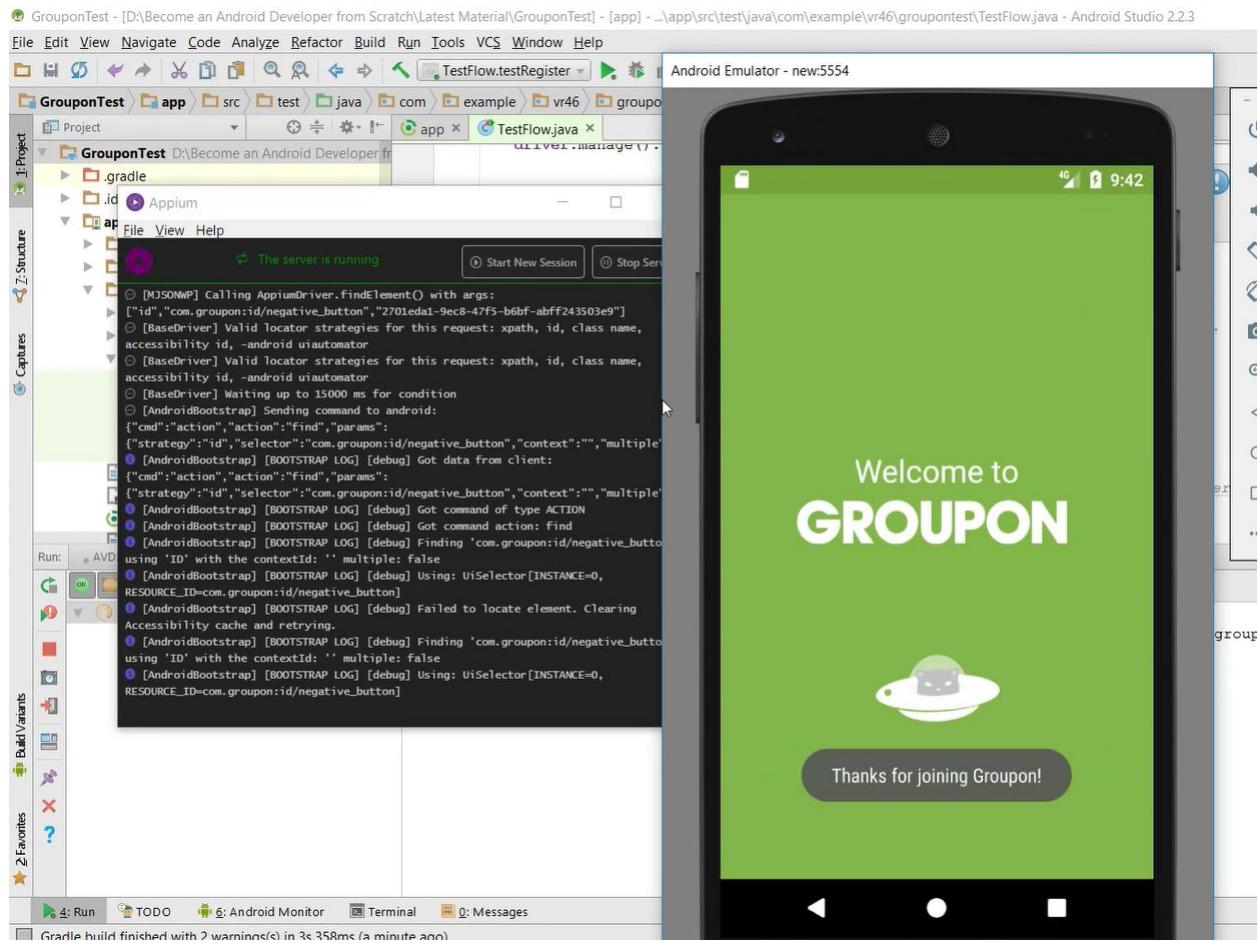
7. The script fills up the needed details like Name, E-mail and Password and clicks on the “Sign Up with Groupon” button.



8. Terms and Conditions page gets open. The script clicks on “Accept Terms” button.



9. So, welcome screen gets opened.



3.2.2.2 Login and see deals in specific pincode

1. Below is the screenshot of the code for the end-to-end flow.

```

public void testRegister() {
    driver.findElement(By.id("com.groupon:id/home")).sendKeys("95112");
    WebElement pincode = driver.findElement(By.id("com.groupon:id/continue_step"));
    pincode.click();
    WebElement abc = driver.findElement(By.id("com.groupon:id/done")); // android.widget.ScrollView[0]/android.widget.Button[6]
    System.out.println(abc.toString());
    abc.click();
    WebElement signup = driver.findElements(By.className("android.widget.TextView")).get(1);
    signup.click();

    WebElement firstName = driver.findElement(By.id("com.groupon:id/fragment_log_in_sign_up_name"));
    WebElement email = driver.findElement(By.id("com.groupon:id/fragment_log_in_sign_up_email"));
    WebElement password = driver.findElement(By.id("com.groupon:id/fragment_log_in_sign_up_password"));

    firstName.sendKeys("Apoorv Mehta");
    email.sendKeys("apoorvpme.ha@gmail.com");
    password.sendKeys("Apoorv@123");

    WebElement signButton = driver.findElement(By.id("com.groupon:id/fragment_log_in_sign_up_groupon_button"));
    signButton.click();

    WebElement acceptterms = driver.findElement(By.id("com.groupon:id/fragment_terms_conditions_submit"));
    acceptterms.click();

    WebElement updateLater = driver.findElement(By.id("com.groupon:id/negative_button"));
    updateLater.click();
}

```

1 test passed – 1m 5s 450ms

2. Run the TestLogin() method

```

driver.manage().timeouts().implicitlyWait(15, TimeUnit.SECONDS);

@Test
public void testLogin() {
    driver.findElement(By.id("com.groupon:id/email")).sendKeys("apoorvpme.ha@gmail.com");
    driver.findElement(By.id("com.groupon:id/password")).sendKeys("Apoorv@123");
    driver.findElement(By.id("com.groupon:id/sign_in_button")).click();
}

@Test
public void testRegister() {
    driver.findElement(By.id("com.groupon:id/home")).sendKeys("95112");
    WebElement pincode = driver.findElement(By.id("com.groupon:id/continue_step"));
    pincode.click();
    WebElement abc = driver.findElement(By.id("com.groupon:id/done")); // android.widget.ScrollView[0]/android.widget.Button[6]
    System.out.println(abc.toString());
    abc.click();
    WebElement signup = driver.findElements(By.className("android.widget.TextView")).get(1);
    signup.click();

    WebElement firstName = driver.findElement(By.id("com.groupon:id/fragment_log_in_sign_up_name"));
    WebElement email = driver.findElement(By.id("com.groupon:id/fragment_log_in_sign_up_email"));
    WebElement password = driver.findElement(By.id("com.groupon:id/fragment_log_in_sign_up_password"));

    firstName.sendKeys("Apoorv Mehta");
    email.sendKeys("apoorvpme.ha@gmail.com");
    password.sendKeys("Apoorv@123");

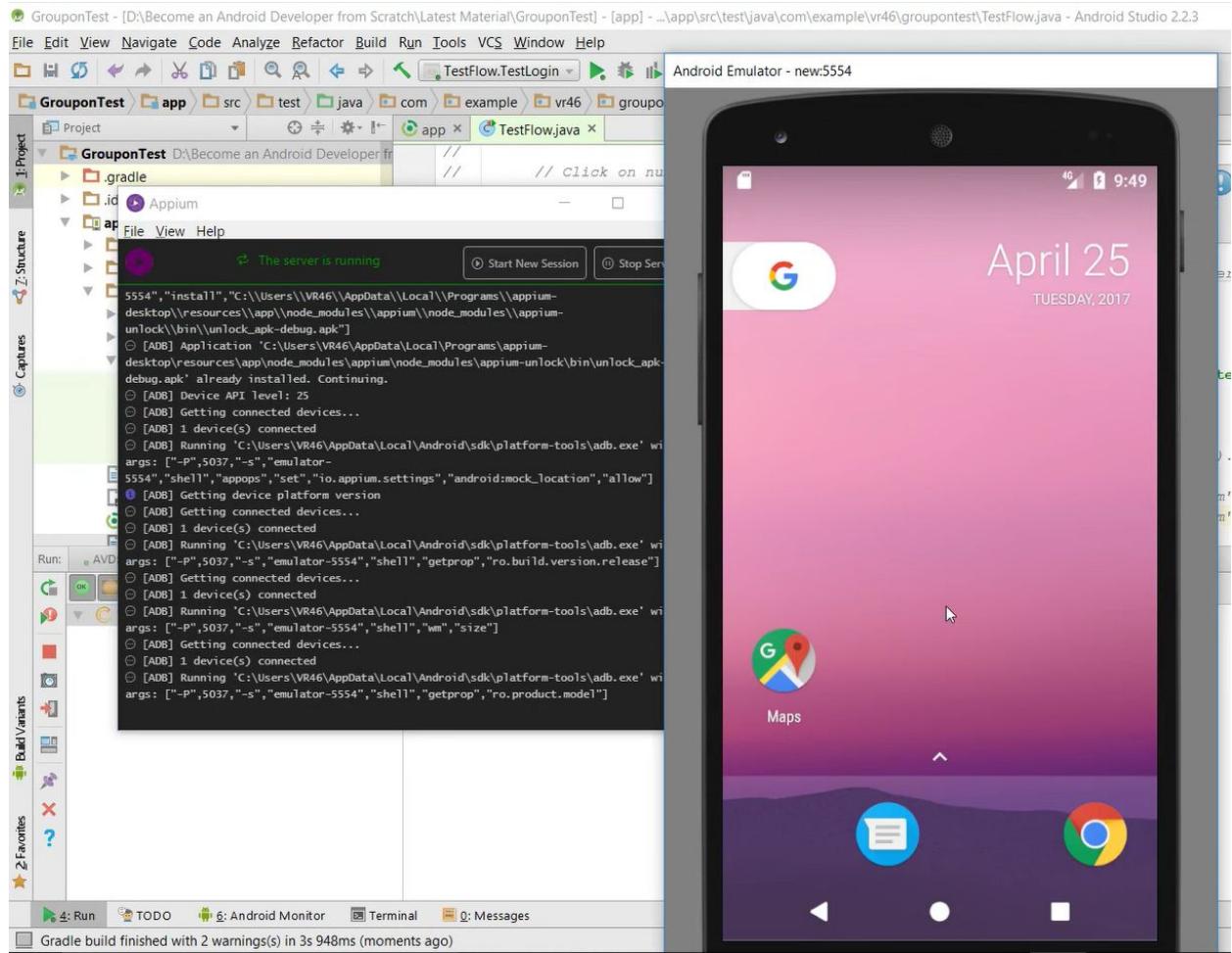
    WebElement signButton = driver.findElement(By.id("com.groupon:id/fragment_log_in_sign_up_groupon_button"));
    signButton.click();

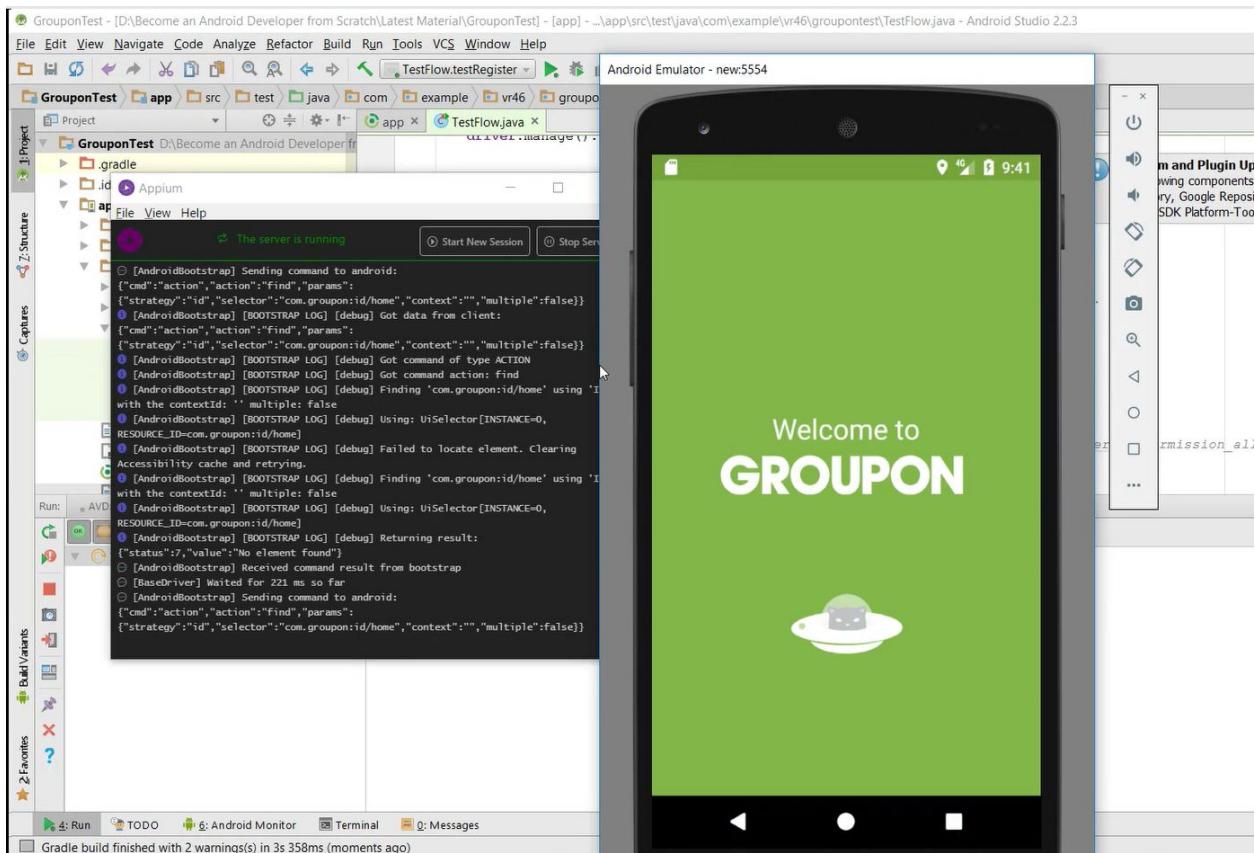
    WebElement acceptterms = driver.findElement(By.id("com.groupon:id/fragment_terms_conditions_submit"));
    acceptterms.click();

    WebElement updateLater = driver.findElement(By.id("com.groupon:id/negative_button"));
    updateLater.click();
}

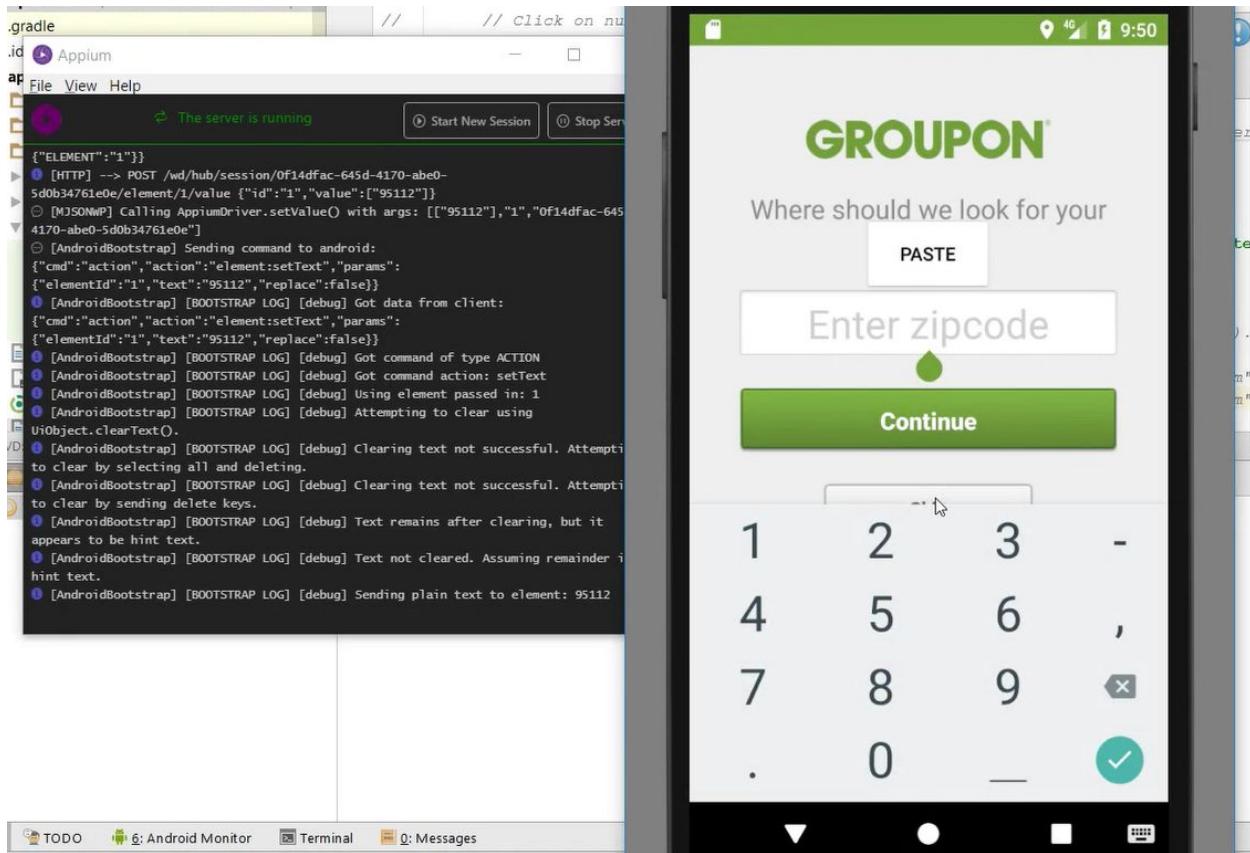
```

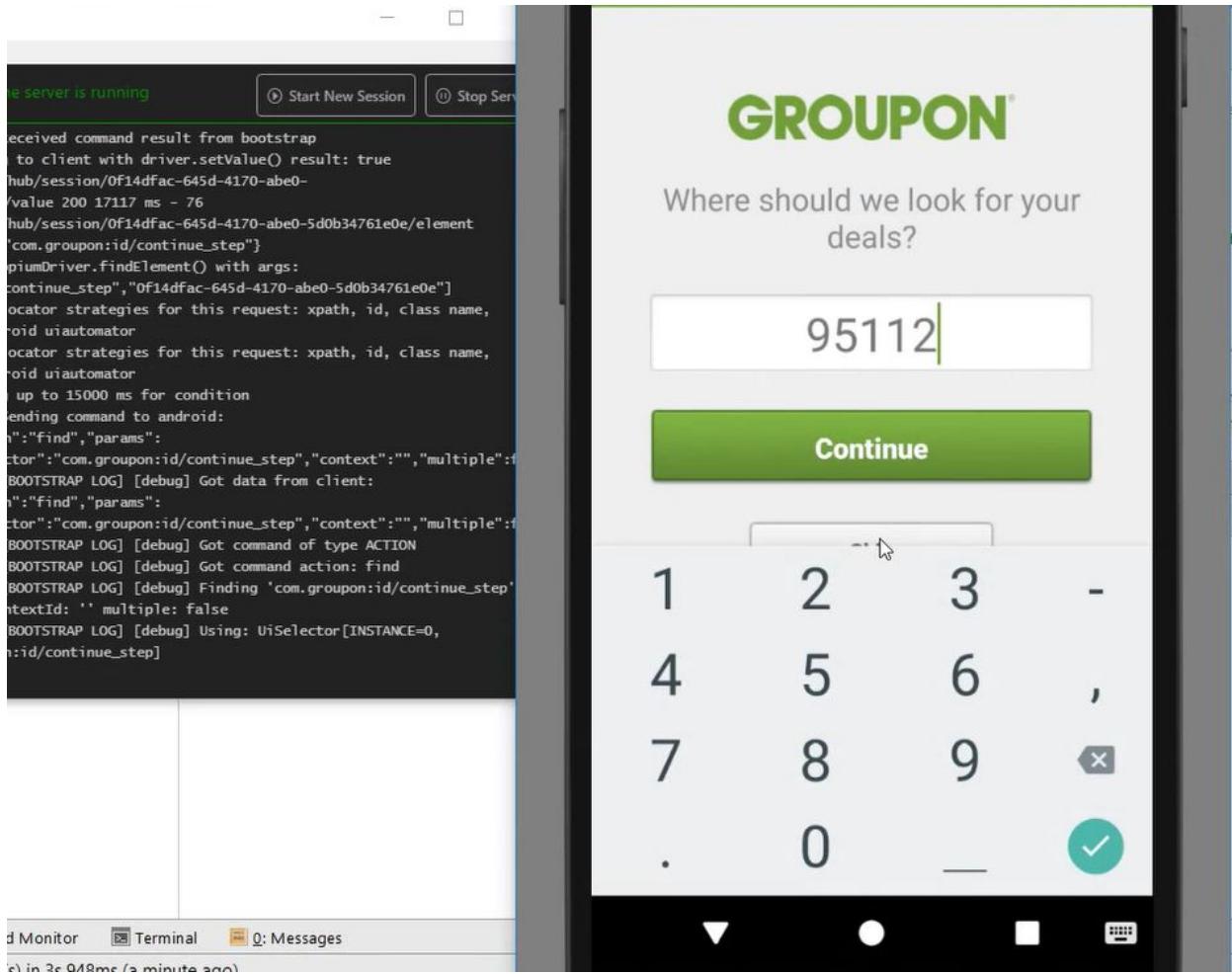
3. The package name as well as the activity name are given in the automation script. So, the automation script will open the Groupon application.



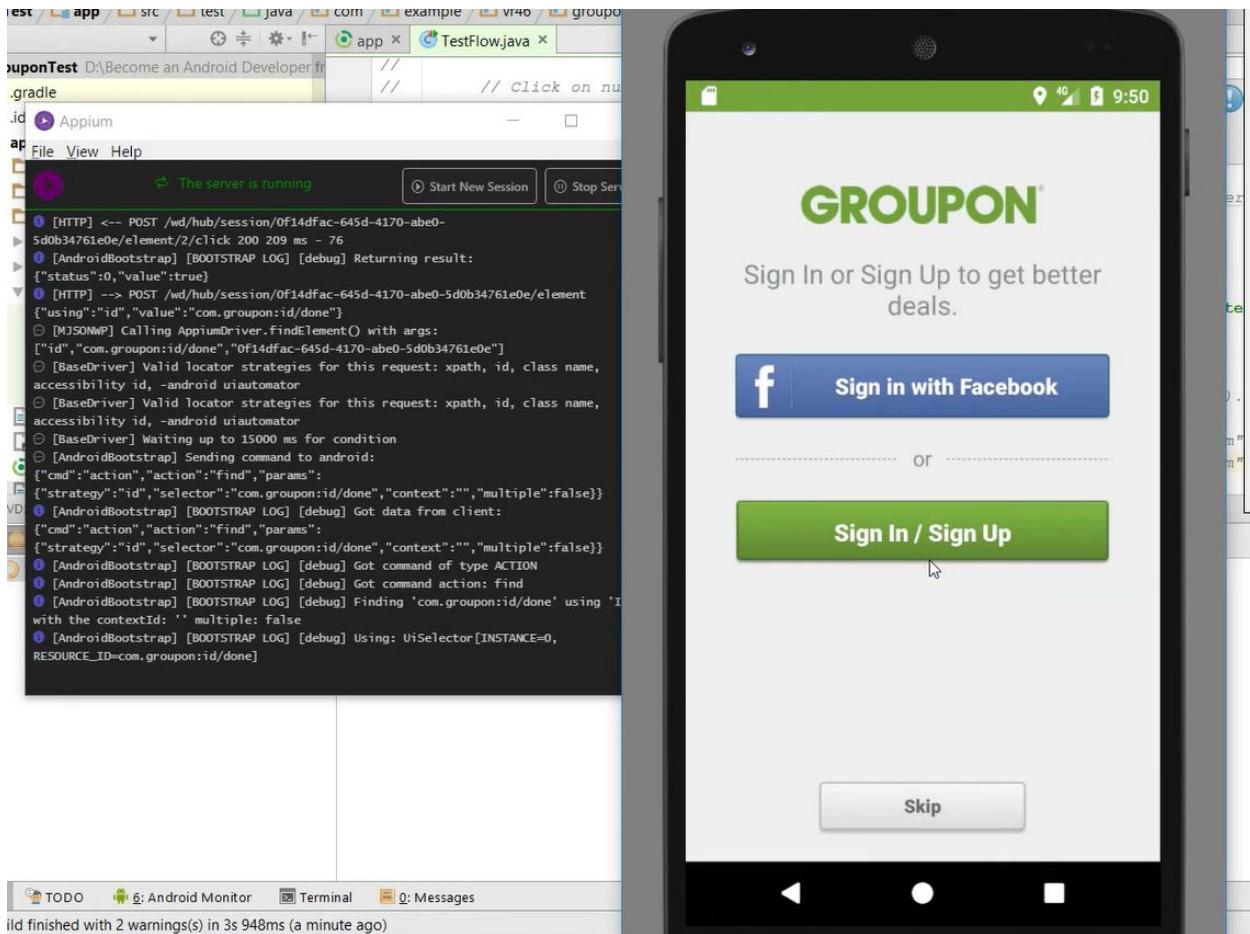


4. The script gives the pin code and click on “Continue” button.

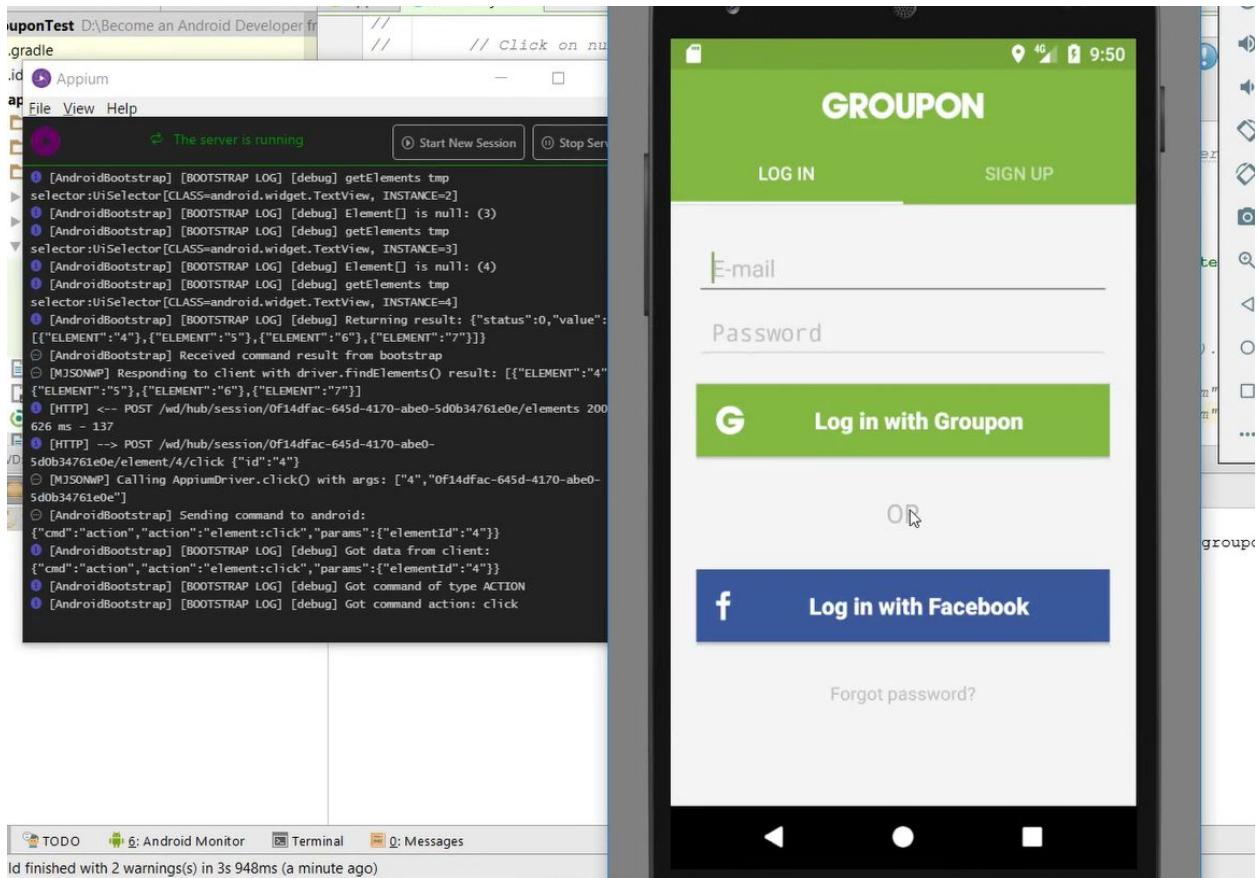




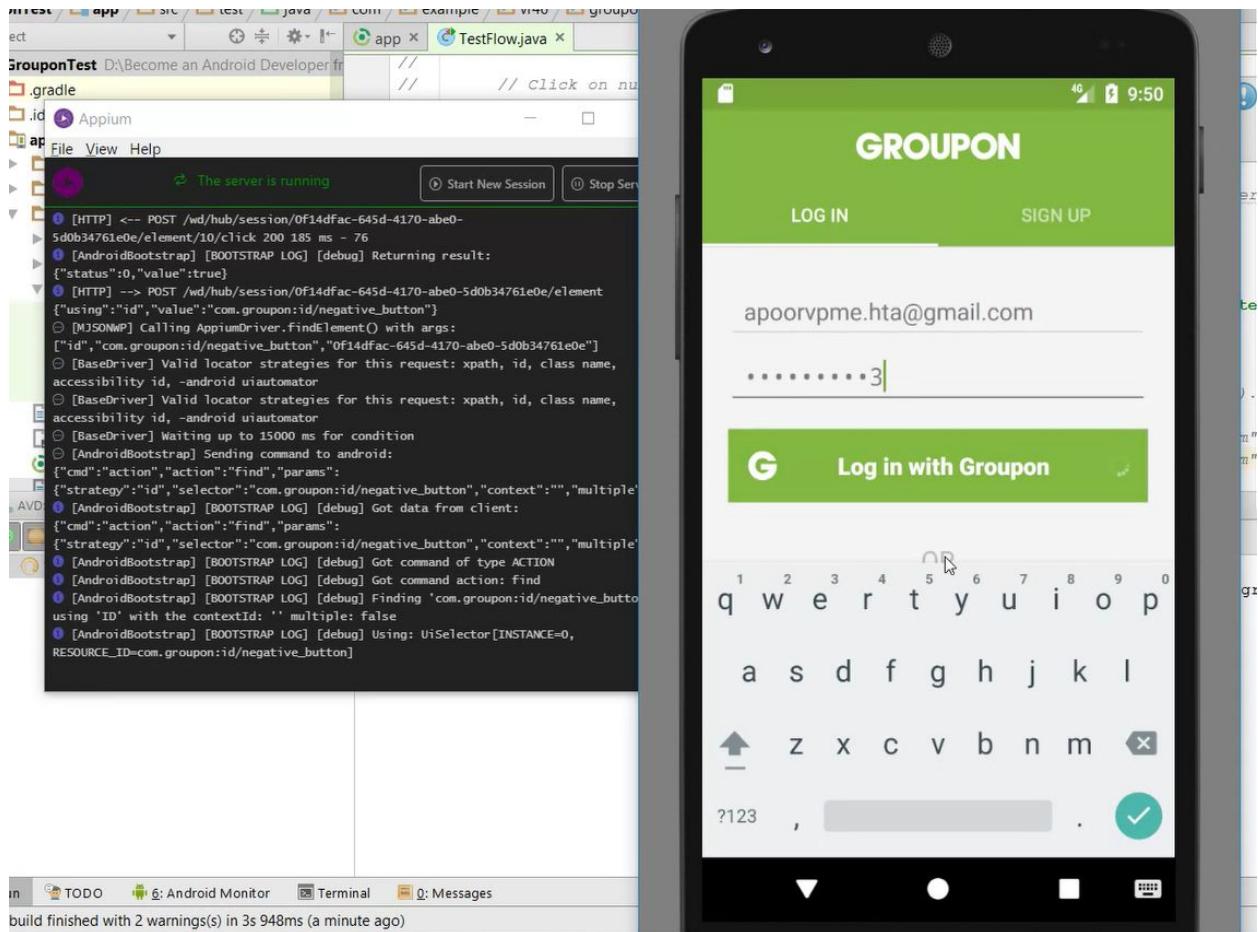
5. The Groupon application displays several options to login or signup. So, the script will select the “Sign In/ Sign Up” option.



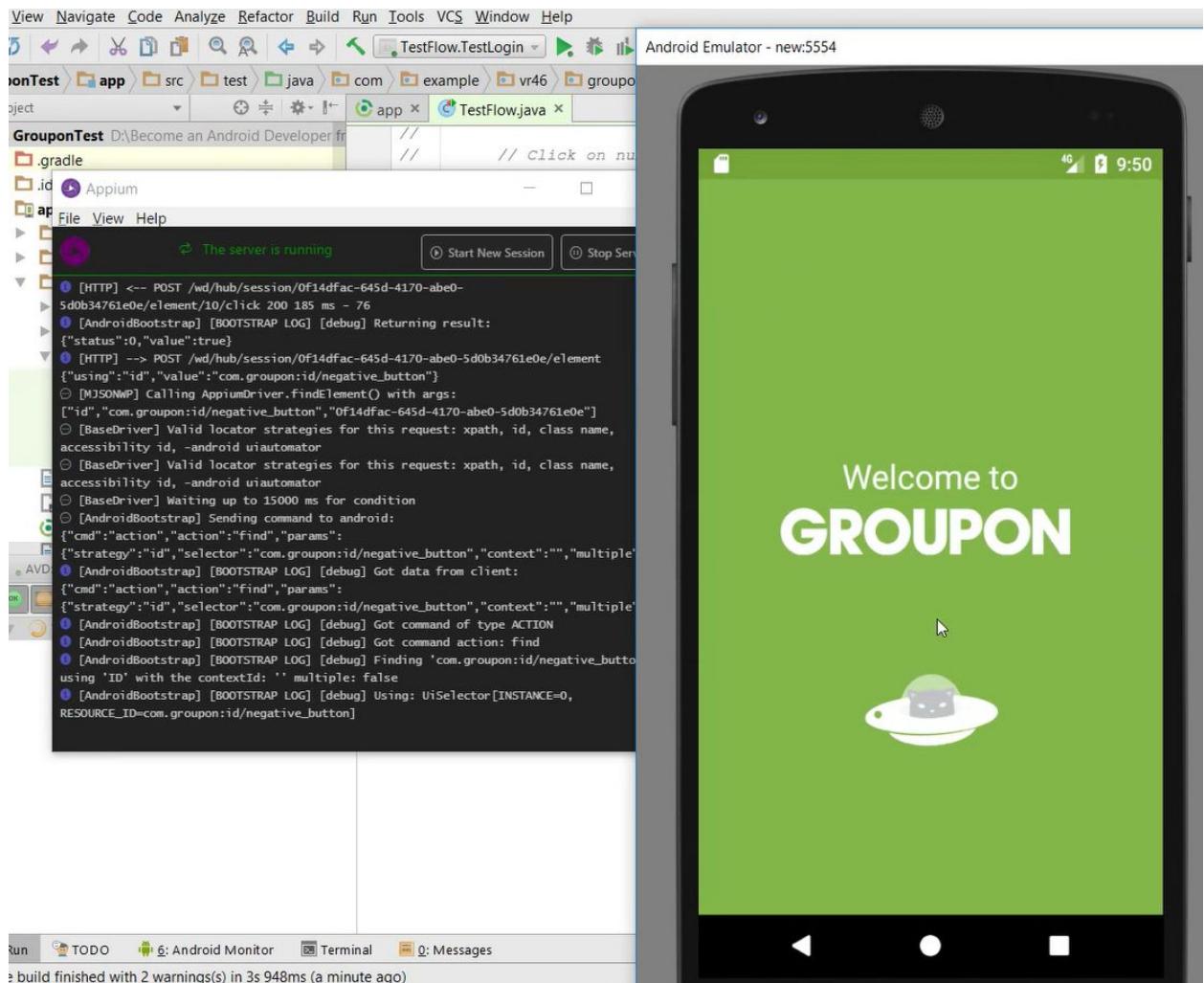
6. The Sign Up/ Login page gets opens. The page has Login tab selected as a default option.



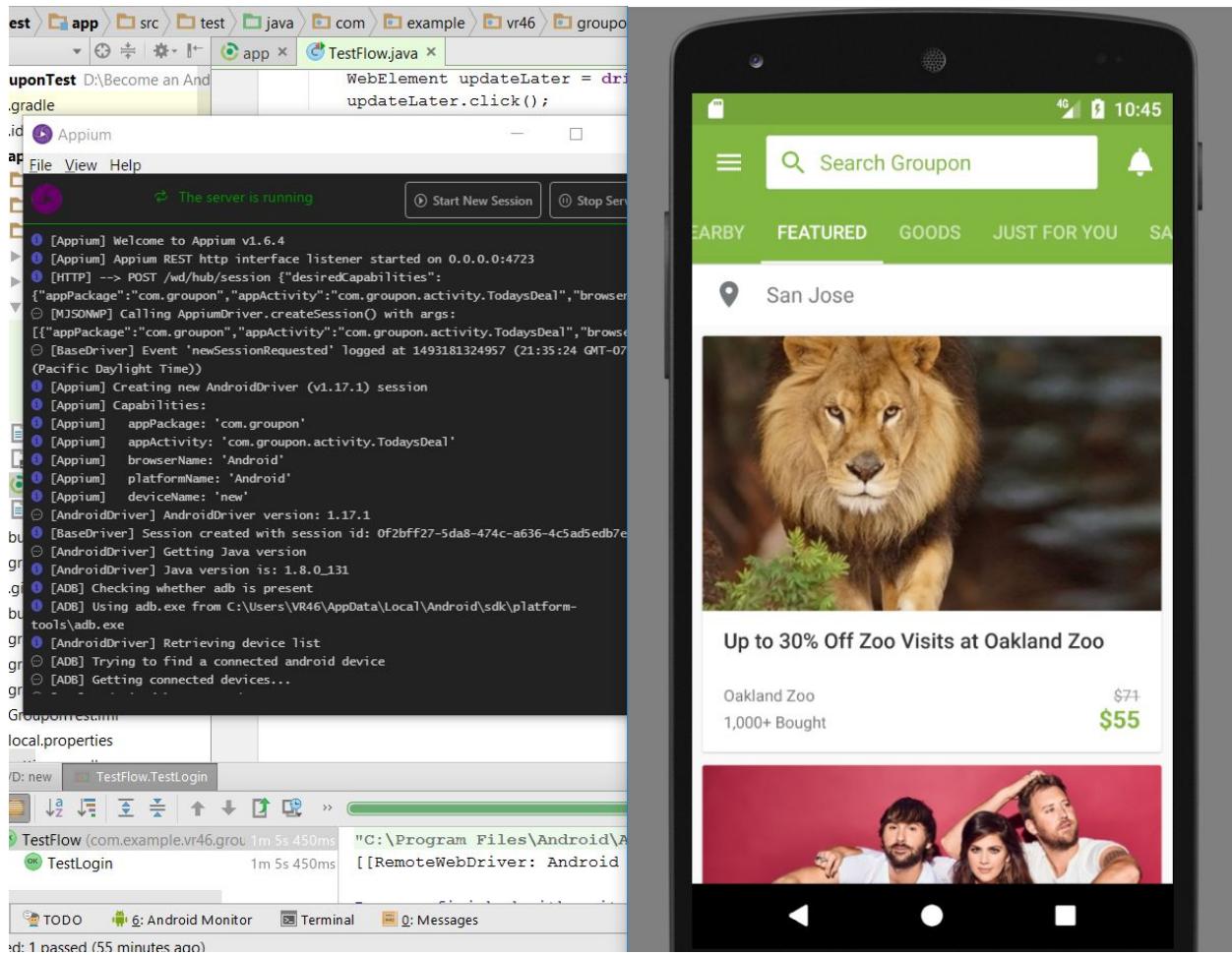
7. The script fills up the needed credentials to Login like E-mail and Password and clicks on the “Log in with Groupon” button.



8. The Groupon application will check the credentials and if the credentials are correct, Welcome activity will be shown.



9. Then, the user account gets open.



4. Test Automation Deliverables

Deliverable	Expected delivery date	Assigned To
Setup Selenium environment	April 13, 2017	Darshil, Parvez, Siddharth
Setup Appium environment	April 13, 2017	Apoorv, Shachi
End-to-end test automation flow design(Web)	April 18, 2017	Apoorv, Darshil, Parvez, Shachi, Siddharth
End-to-end test automation flow design(Android)	April 18, 2017	Apoorv, Darshil, Parvez, Shachi, Siddharth
End-to-end test automation flow implementation(Web)	April 20, 2017	Darshil, Parvez, Siddharth
End-to-end test automation flow implementation(Android)	April 20, 2017	Apoorv, Shachi
End-to-end test automation flow execution(Web)	April 24, 2017	Darshil, Parvez, Siddharth
End-to-end test automation flow execution(Android)	April 24, 2017	Apoorv, Shachi

5. Test Automation Summary

The project is categorized with four individual end-to-end test automation flows across 2 platforms. Following are the end-to-end automation flows:

- Add to Wish list
- Buy Deal
- Signup using email address
- Login and see deals in specific pincode

Over and above the functions in the above modules, the team has done the end-to-end test automation for the Groupon application.

6. Team Members

Apoory Mehta
Darshil Saraiya
Parvez Patel
Shachi Shah
Siddharth Daftari

6.1 Roles and Responsibilities

QA lead: Apoorv Mehta

Platform/ Task	End-to-end Test Automation Flow Name	Environment Setup	End-to-end test automation flow design	End-to-end test automation flow implementation	End-to-end test automation flow execution
Web	Add to Wish list	Darshil, Parvez, Siddharth	Darshil, Parvez, Siddharth	Darshil, Parvez, Siddharth	Darshil, Parvez, Siddharth
	Buy Deal		Darshil, Parvez, Siddharth	Darshil, Parvez, Siddharth	Darshil, Parvez, Siddharth
Android	Signup using email address	Apoorv, Shachi	Apoorv, Shachi	Apoorv, Shachi	Apoorv, Shachi
	Login and see deals in specific pincode		Apoorv, Shachi	Apoorv, Shachi	Apoorv, Shachi

7. Conclusion

The project was a very good learning exercise for test automation. Functionally Groupon is a very well crafted multi platform app that works seamlessly across devices.

The main components and their functionalities were automated thoroughly and were found working as expected.