**Mobile Automation using emulator**

**Index:**

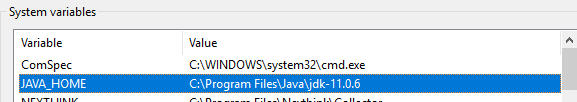
1. **Install JDK**
2. **Update System Environment variables**
3. **Install Eclipse IDE**
4. **Install Appium**
5. **Install Android Studio/SDK Manager**
6. **Create Android Virtual Device(Emulator) and Install Mobile App in Emulator**
7. **Inspect objects using Appium Inspector**
8. **Import mobile automation project in Eclipse using git url**
9. **Perform Test execution and validate the test results**
10. **Mobile app Issues**
11. **Install JDK (JDK version 7 or above:**

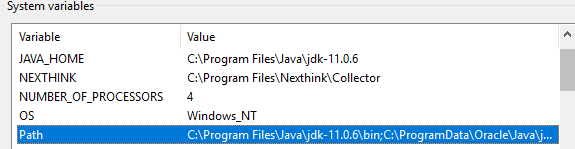
<https://www.oracle.com/java/technologies/javase-downloads.html>

1. **Update System Environment variables:**

JAVA\_HOME: jdk path

Path: jdk bin path





1. **Install Eclipse IDE:**

<http://www.eclipse.org/downloads/>

1. **Install Appium (Desktop version will be better)**

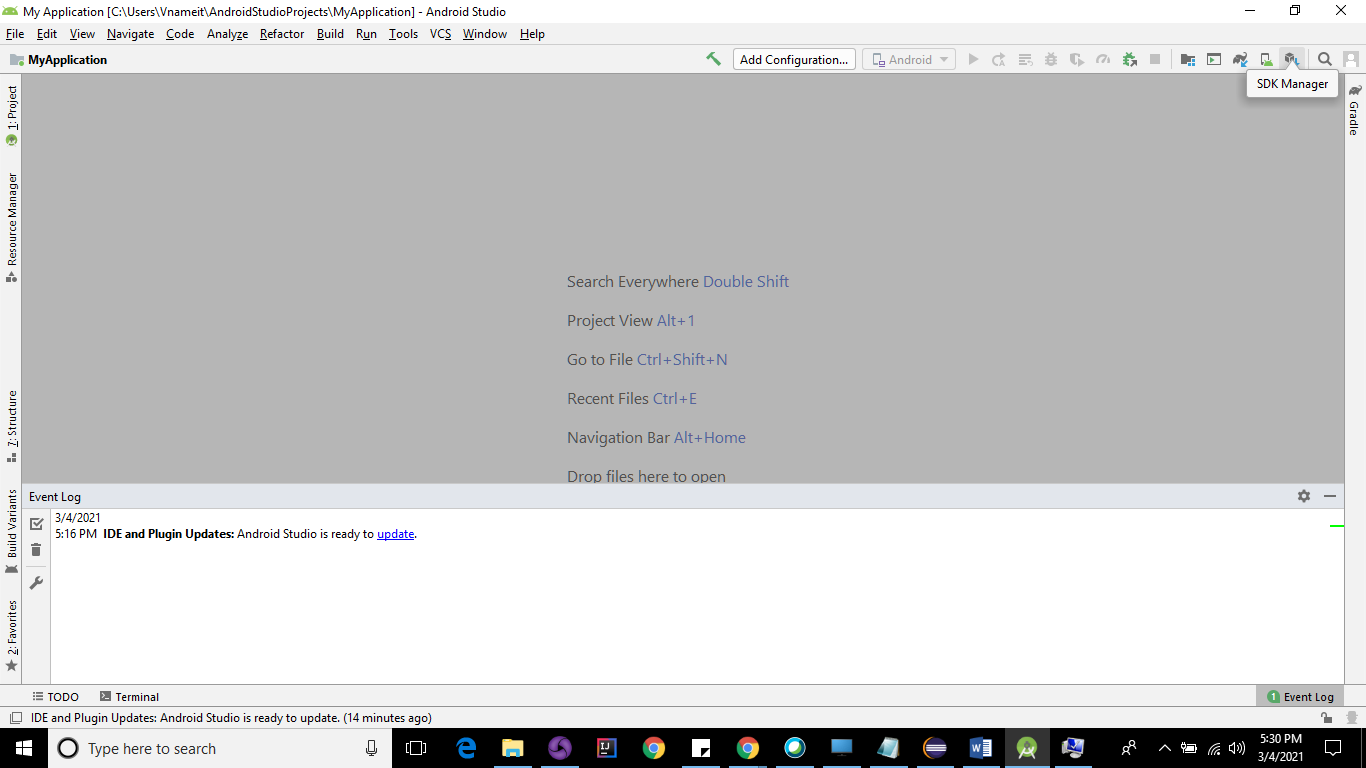
<https://appium.io/downloads.html>

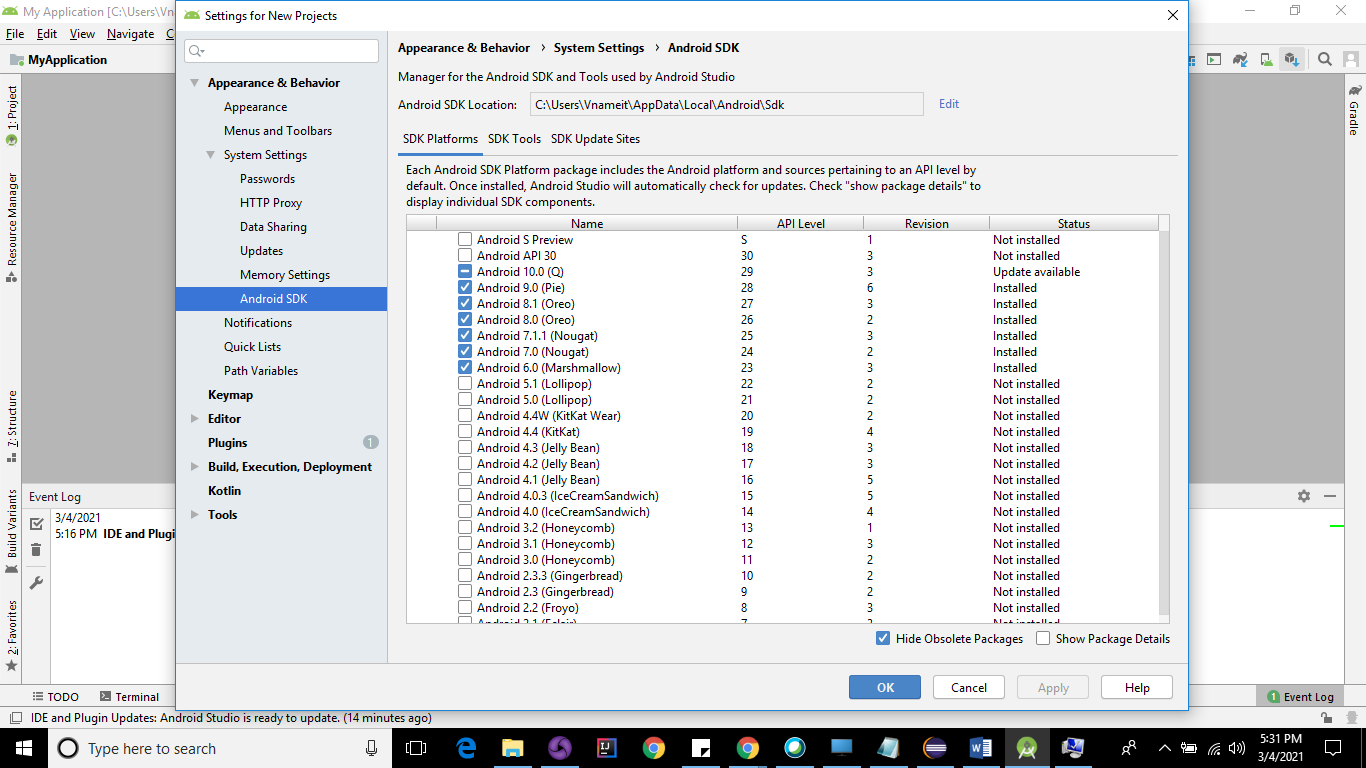
1. **Install Android Studio/Android SDK Manager:**

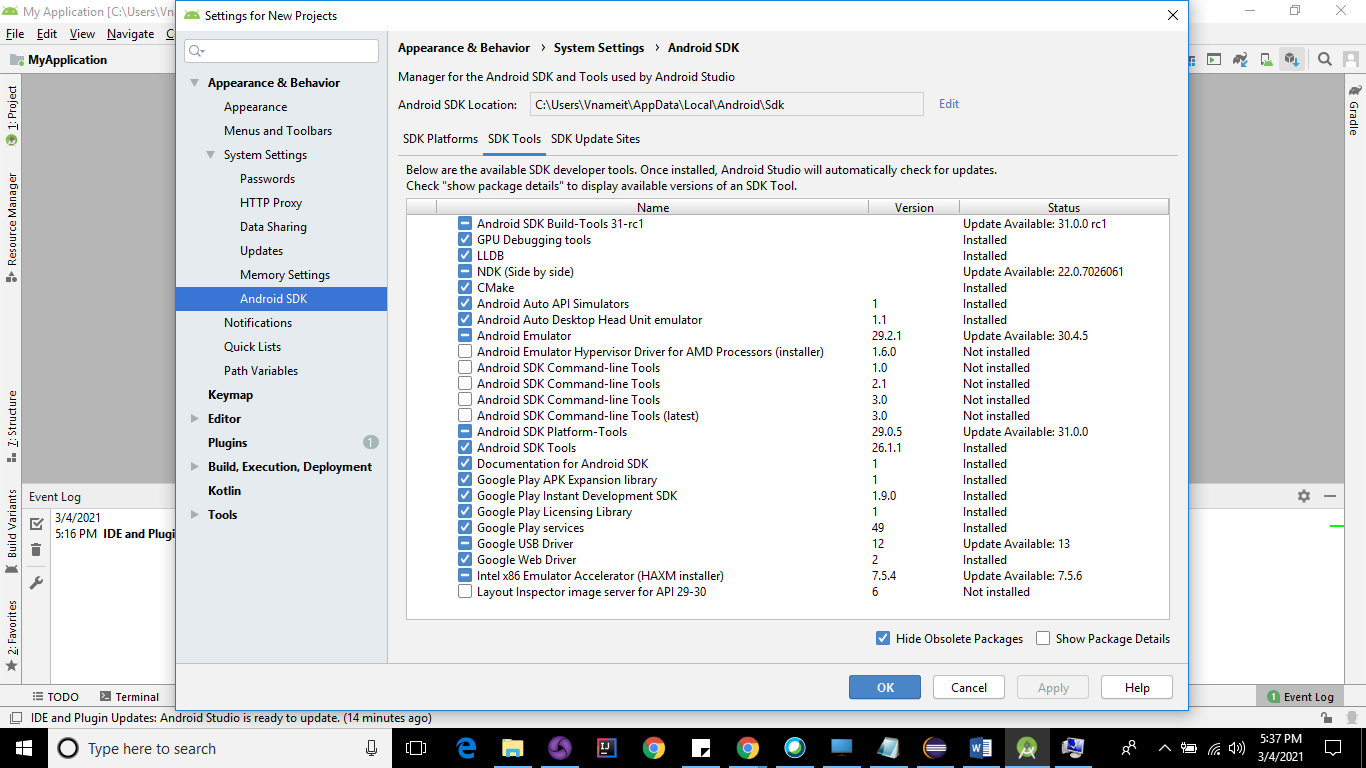
<https://developer.android.com/studio>

1. After successful Installation of Android Studio, select SDK Manager and verify whether the required SDK platforms installed or not. If any required platforms missed during installation, select the required platforms and install the same.
2. Also ensure required SDK tools got installed.
3. After successful installation of required SDK platforms and tools, update the below mentioned environment variables

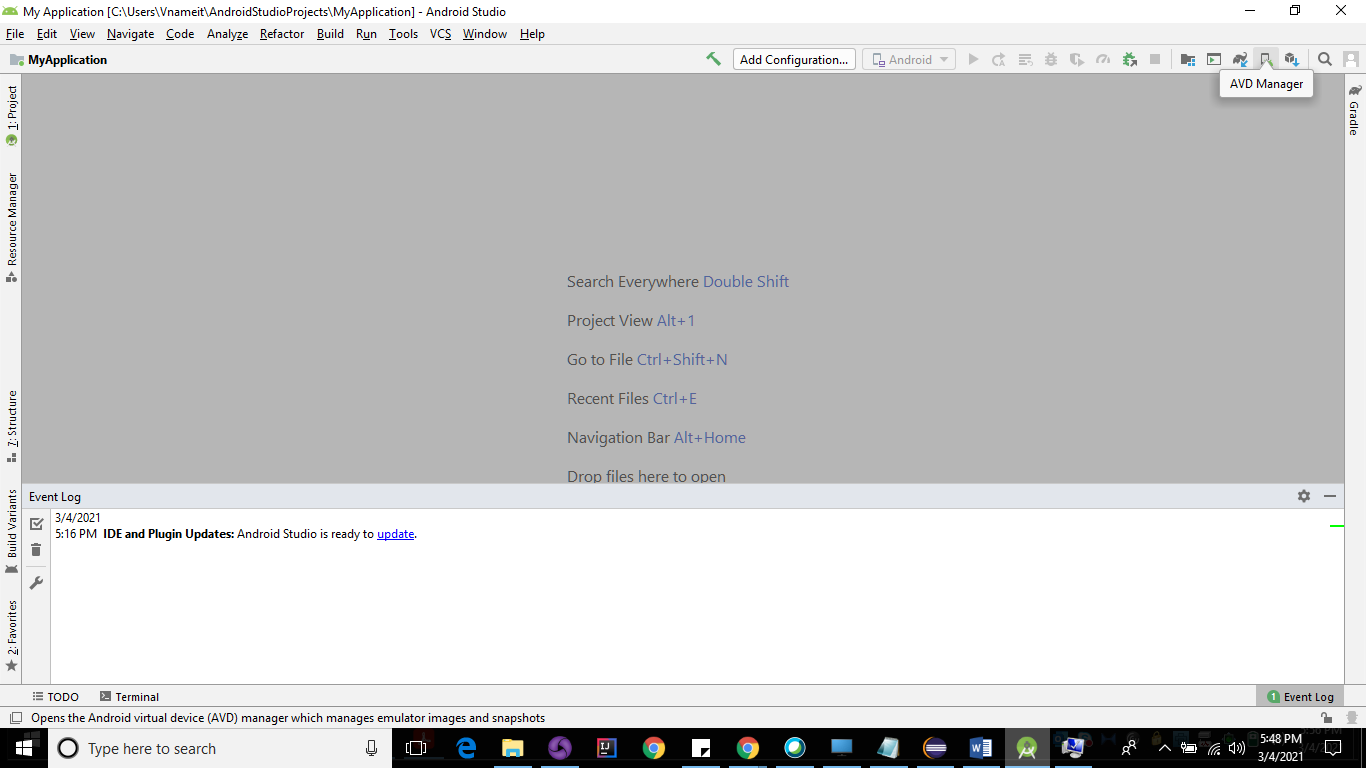
ANDROID\_HOME: Add Android SDK path  
Path: Add Android SDK platform-tools and tools path

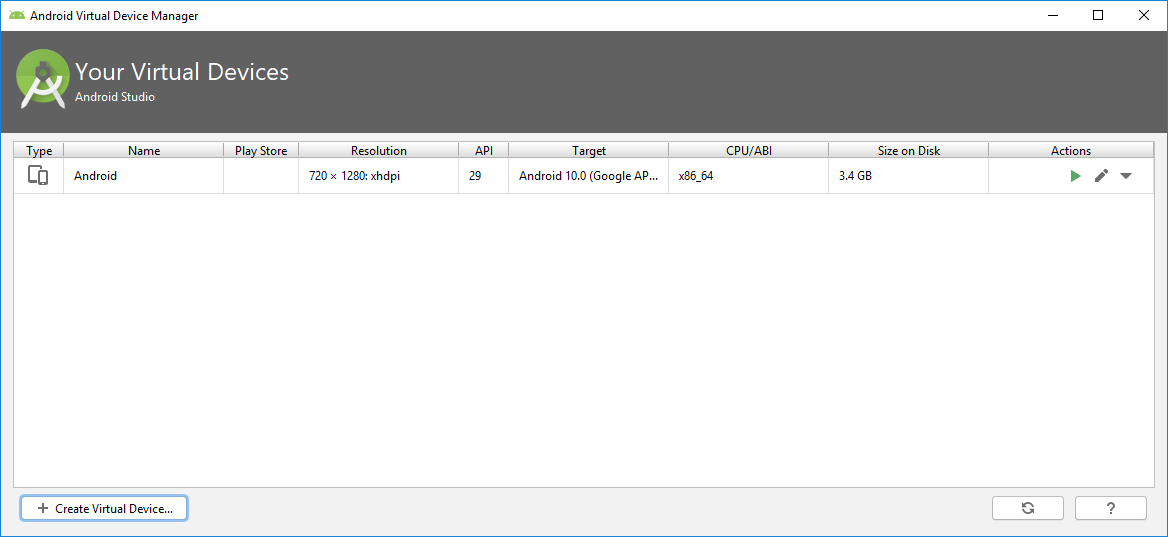


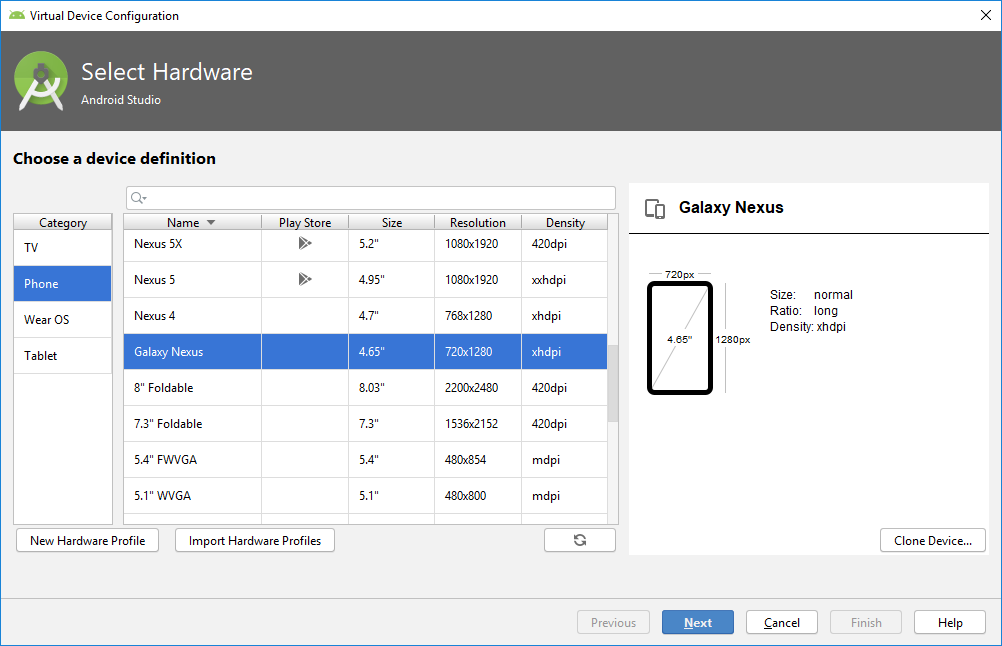


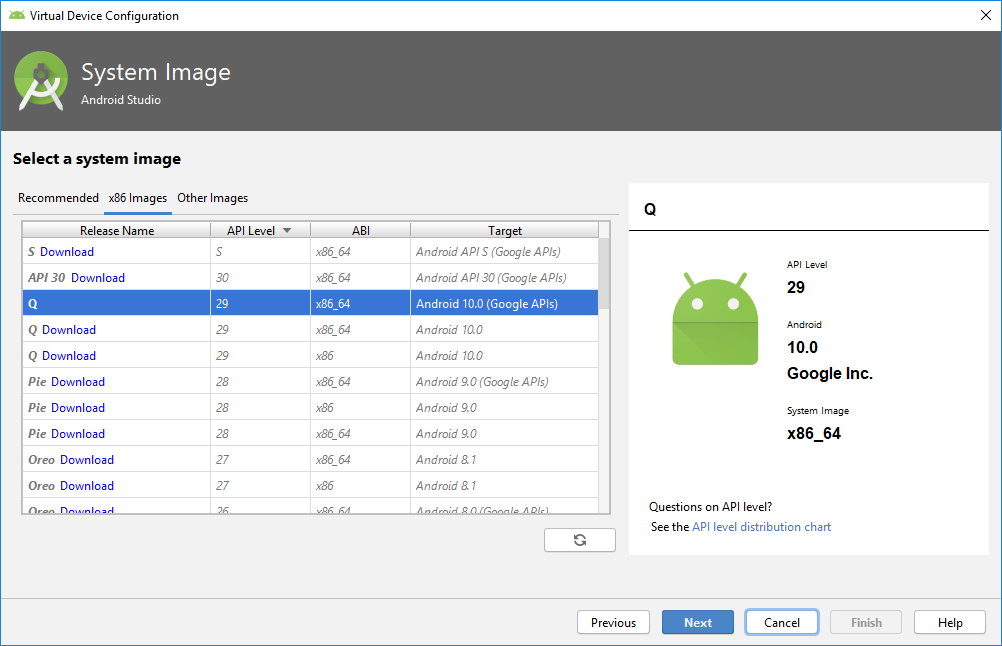


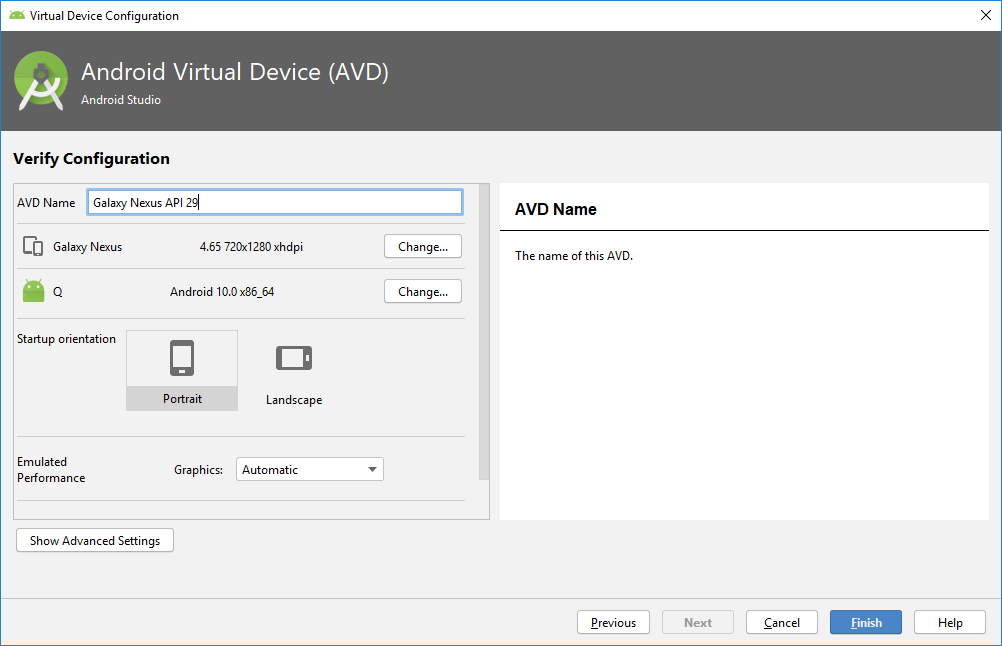
1. **Create Android Virtual Device (Emulator) and Install mobile app in the emulator:**
2. Click on AVD Manager option from Android Studio or use Search for the option to open AVD Manager
3. Click on Create Virtual Device button in AVD Manager Window
4. Select the Phone Hardware
5. Select a System Image for emulator
6. Click on ‘Show Advanced Settings’ in verify configuration window and update the RAM, Internal storage and other required configuration.
7. Click on Finish button and verify selected android emulator got created successfully.
8. Click on the dropdown icon from the selected Android emulator and click on ‘Cold Boot Now’ option.
9. Click on Ok button in case any ADB Binary warning window pop’ed up and wait for the selected emulator screen get displayed (normally emulator screen loading will take some time, around 5 to 10 mins).
10. Once the emulator launched successfully, verify the device id using ‘**adb devices**’ command in command prompt
11. To Install the mobile app in emulator, first download the apk file (for Android) and keep it under Android sdk -> platform tools folder
12. Once the apk file placed under the platform tools folder, go to command prompt window and install the app using ‘ **adb** **install** apk path’
13. Verify whether Success message is displayed in the command prompt and mobile app successfully installed in the emulator

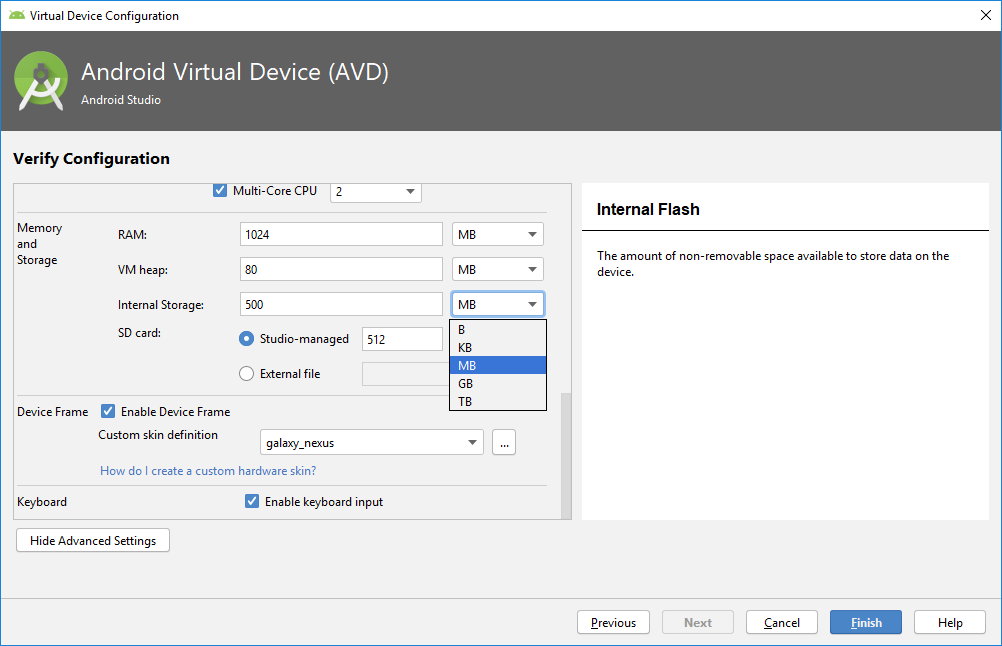


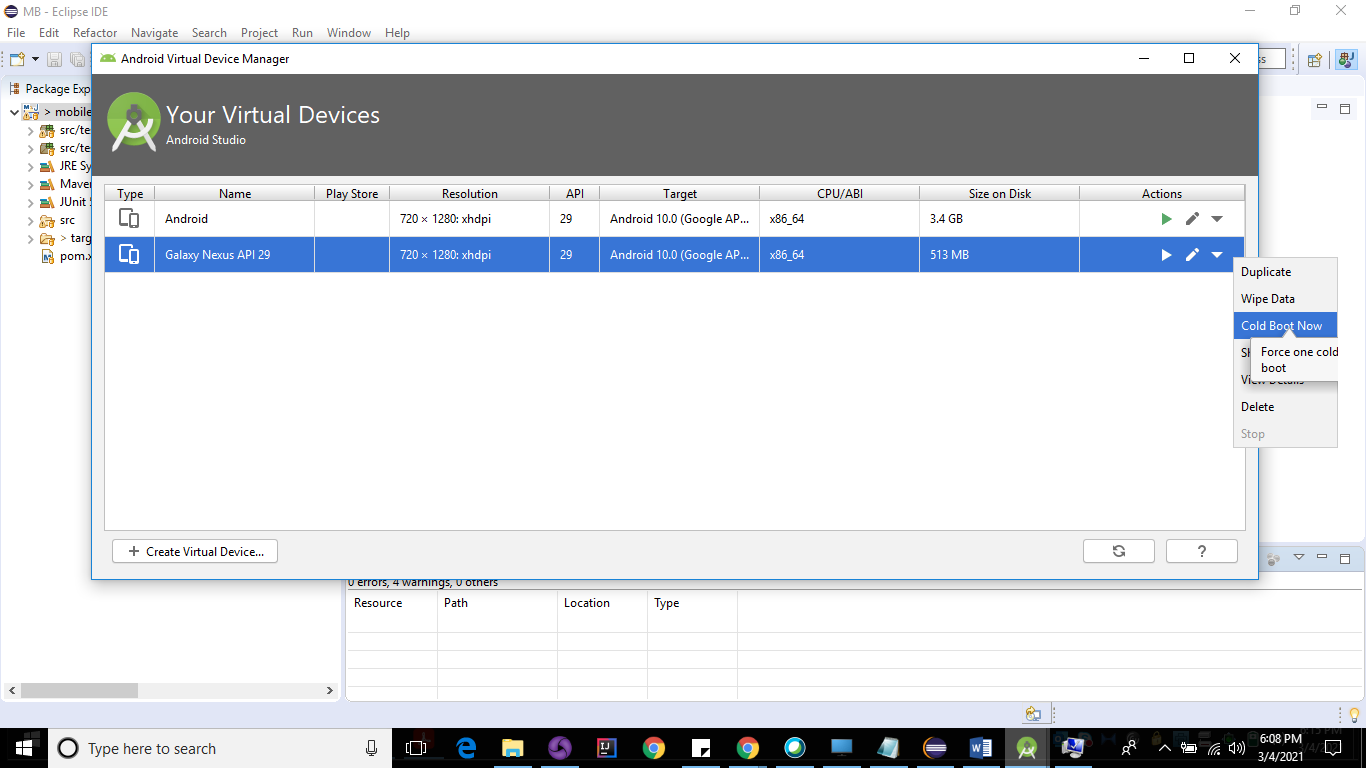


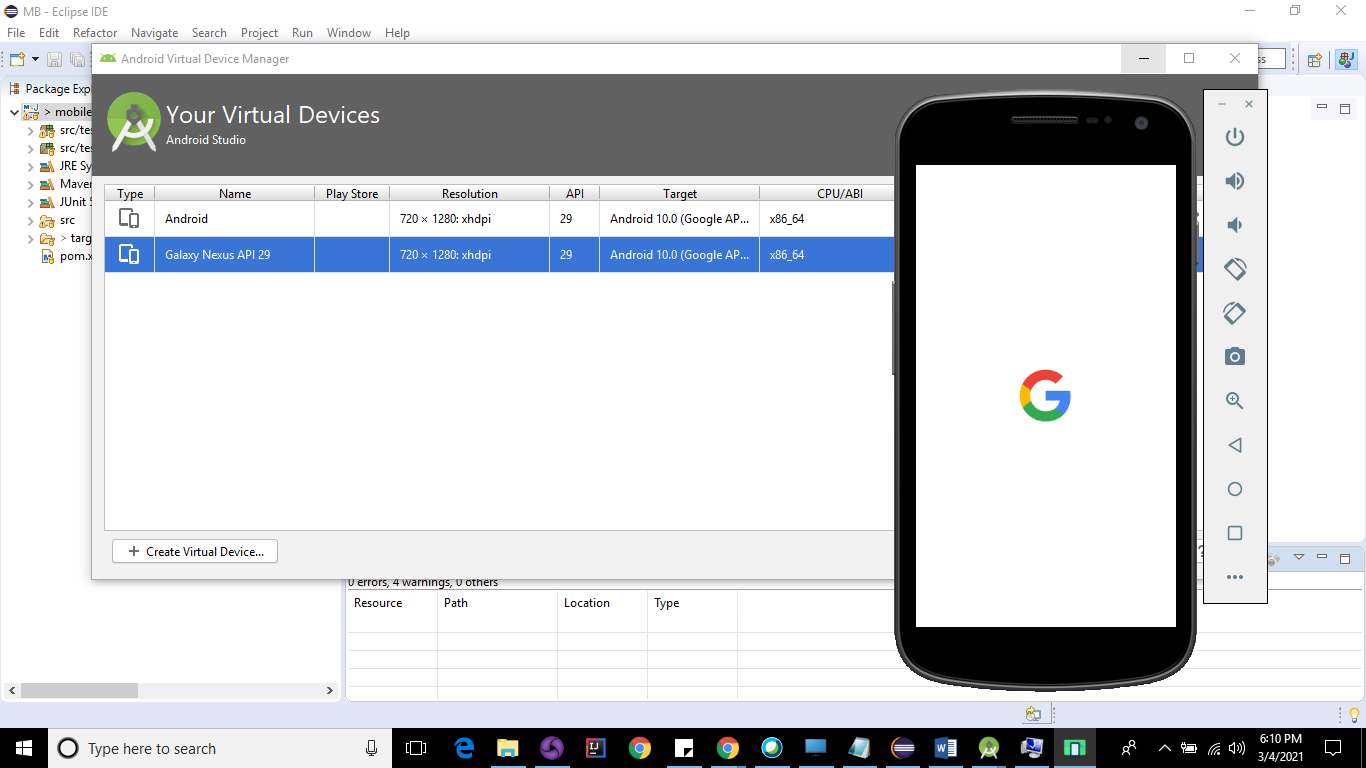


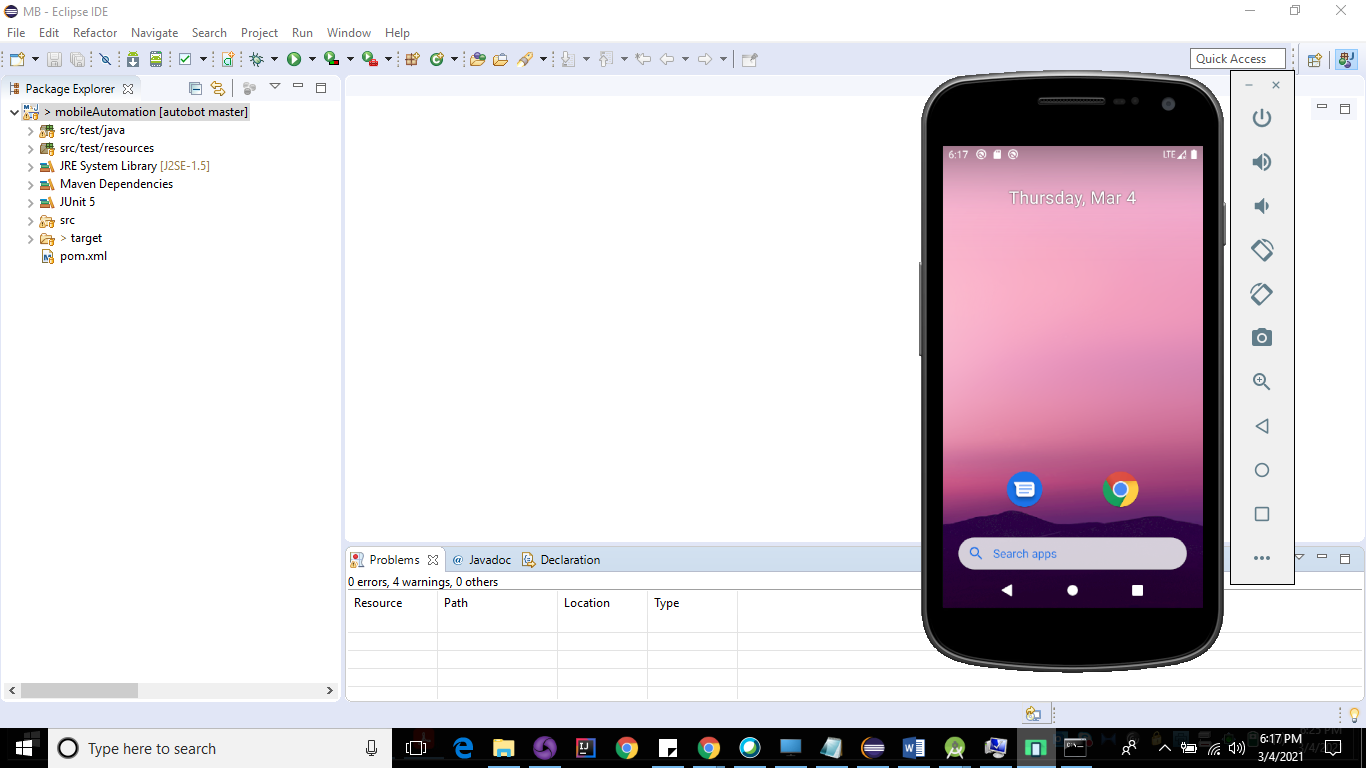


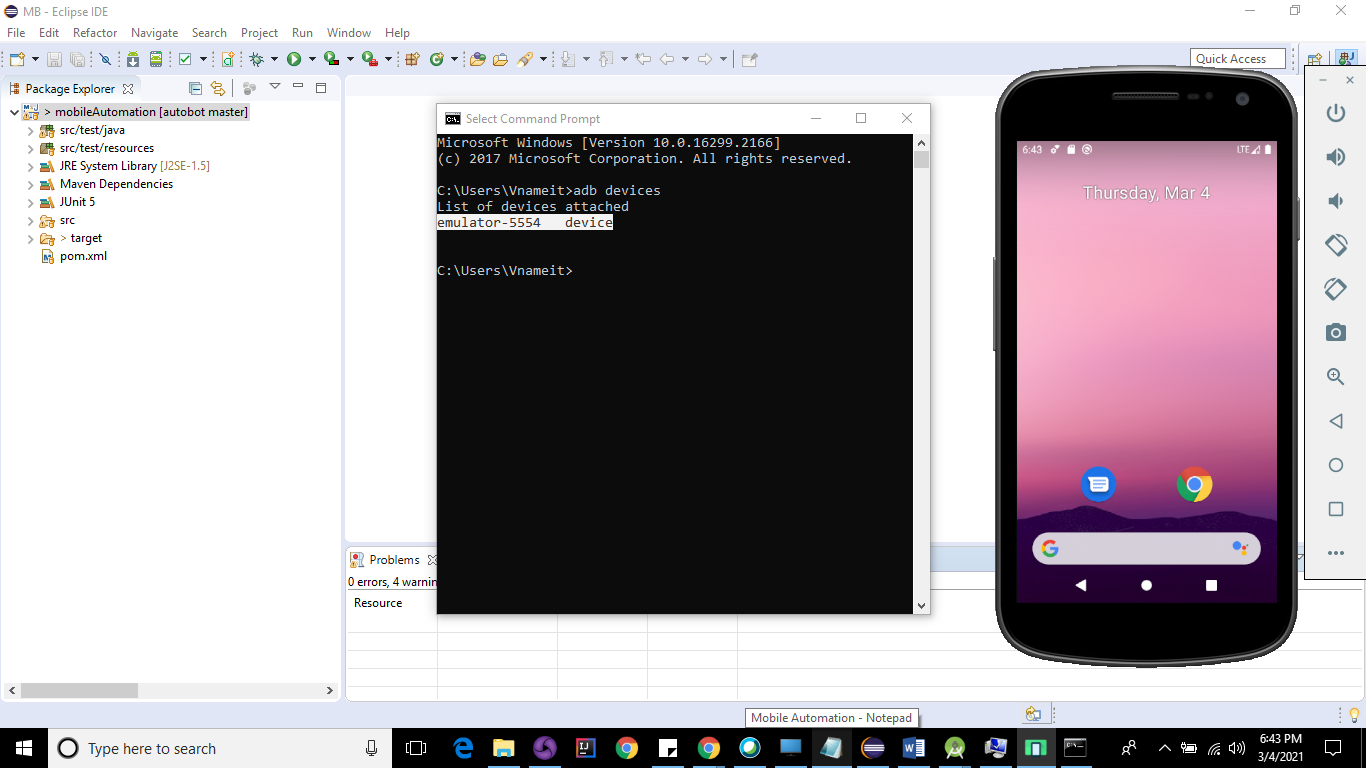


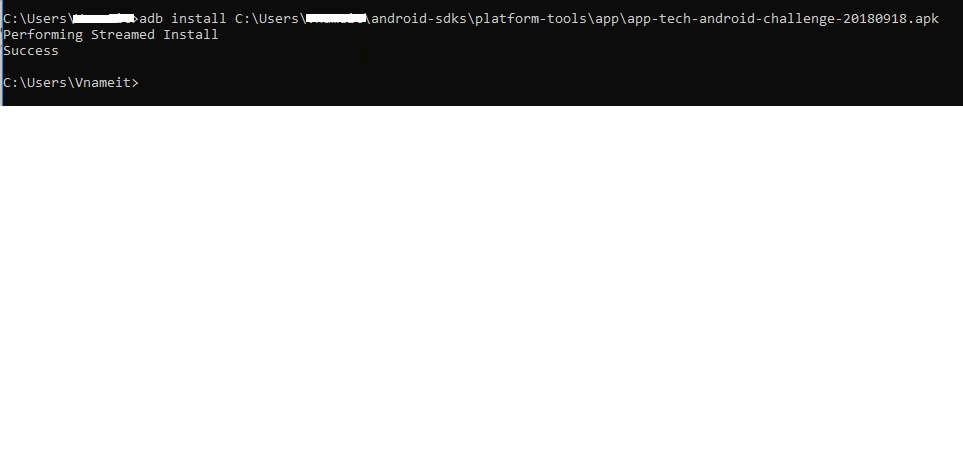


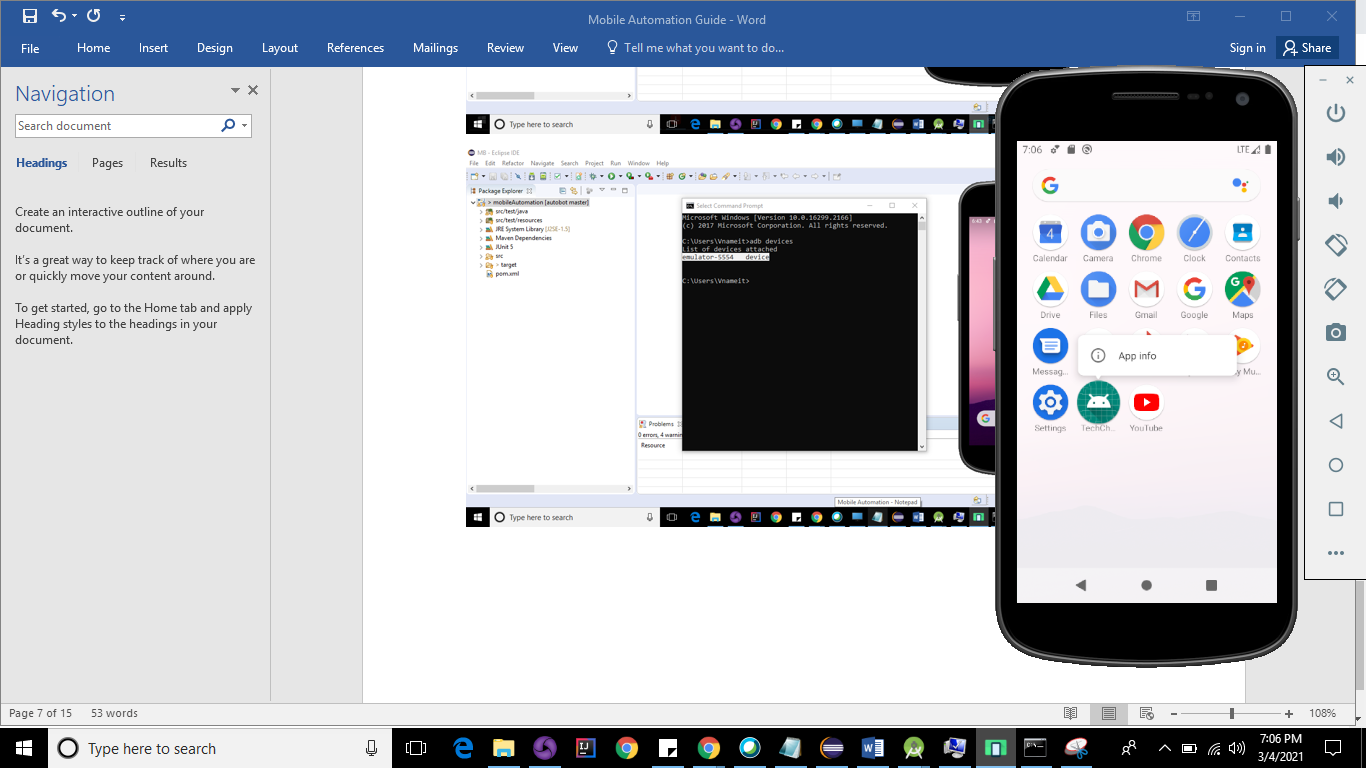








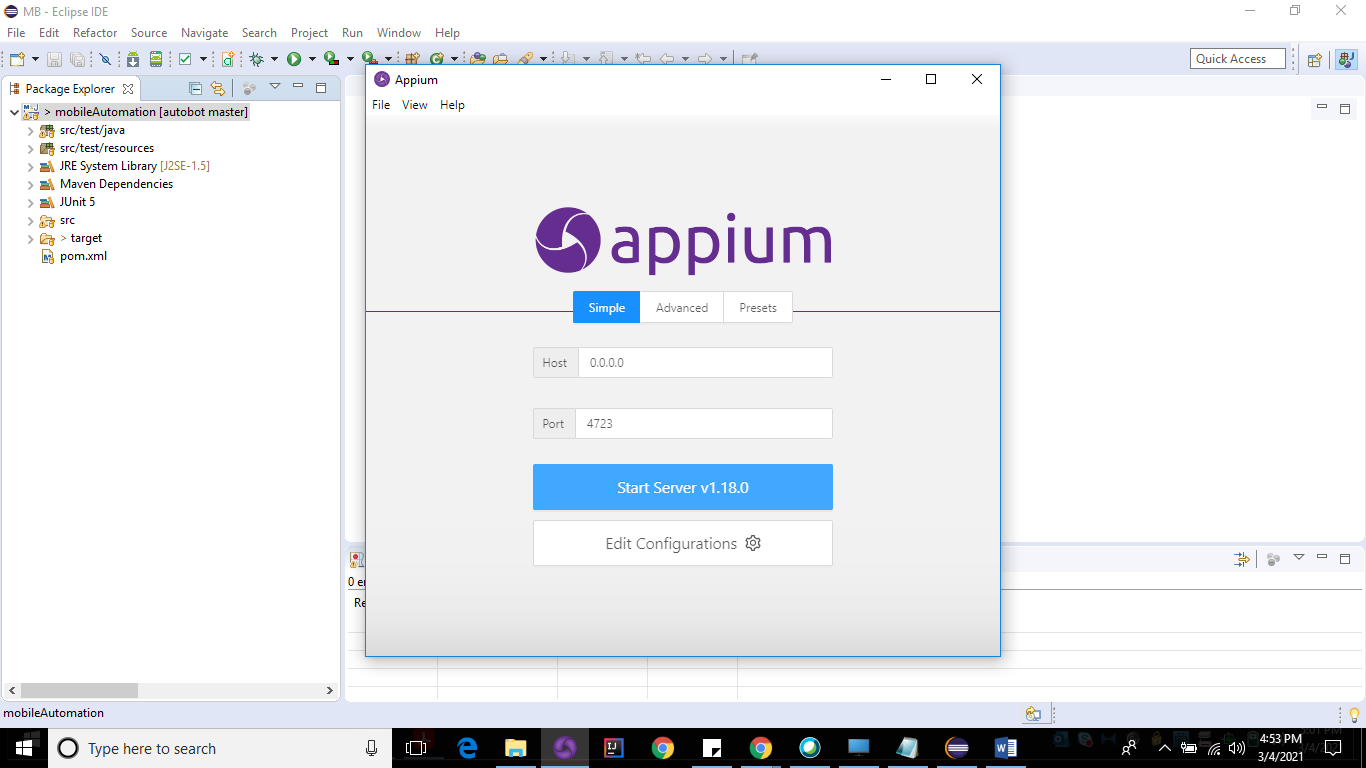
****

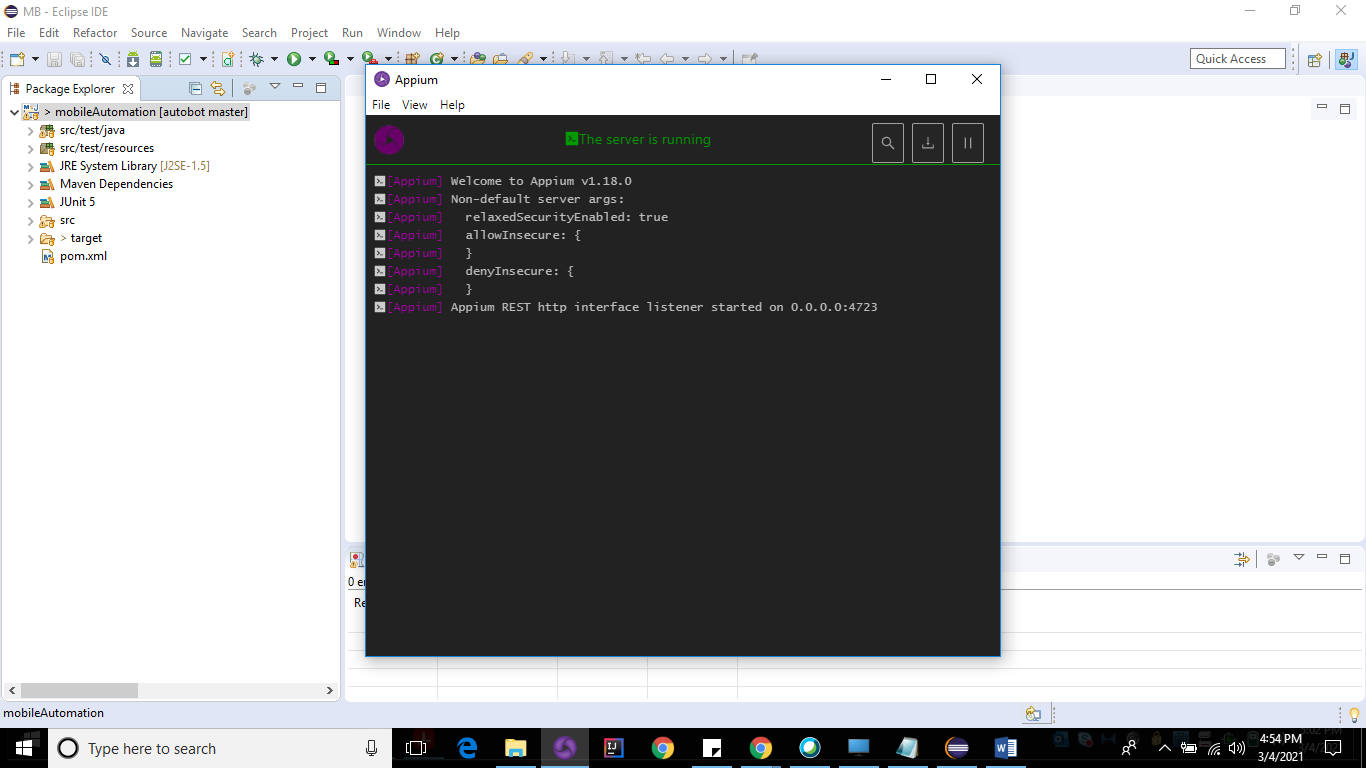


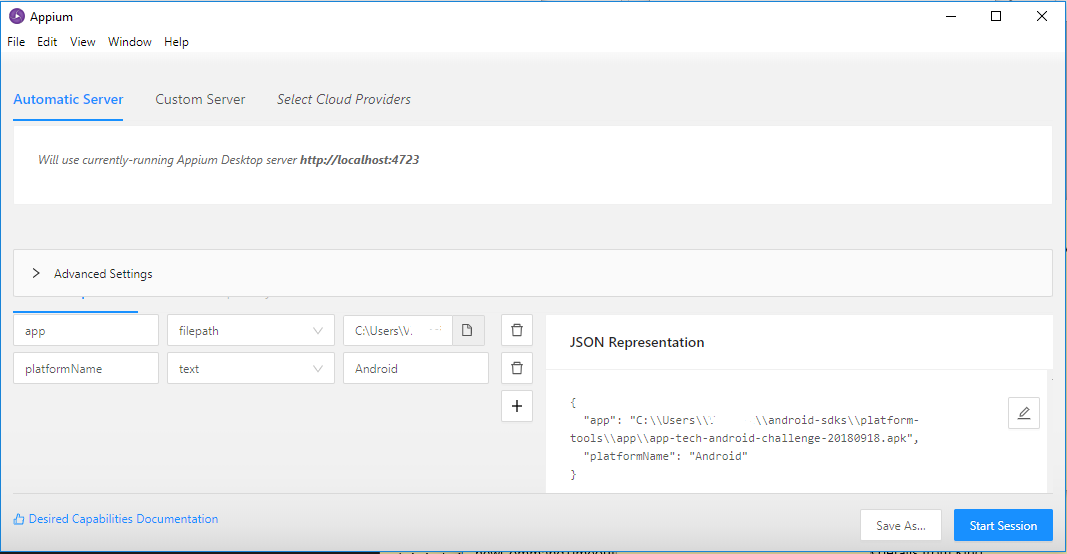
1. **Start Appium Server and Inspect mobile objects using Appium Inspector**
2. We can use the default host IP (0.0.0.0) and port (4723)
3. Click on Search Button in Appium
4. Add the below mentioned capabilities under Automatic Server tab and click on Start session

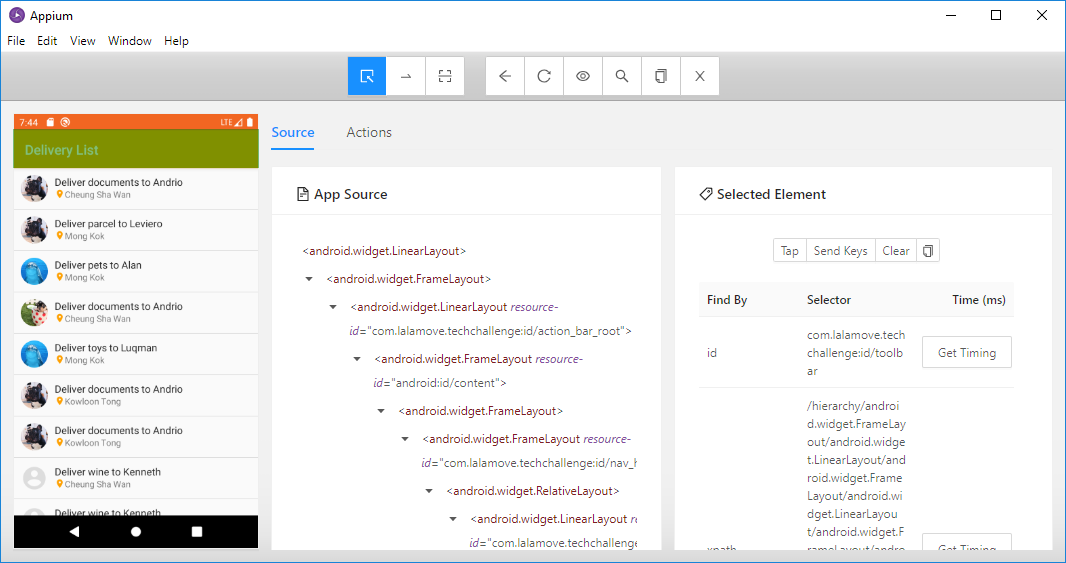
* **app** – apk file path
* **platformName** – Android/iOS
* **deviceName** – Device name (Ex: Samsung)

1. Identify the objects using unique id or xpath

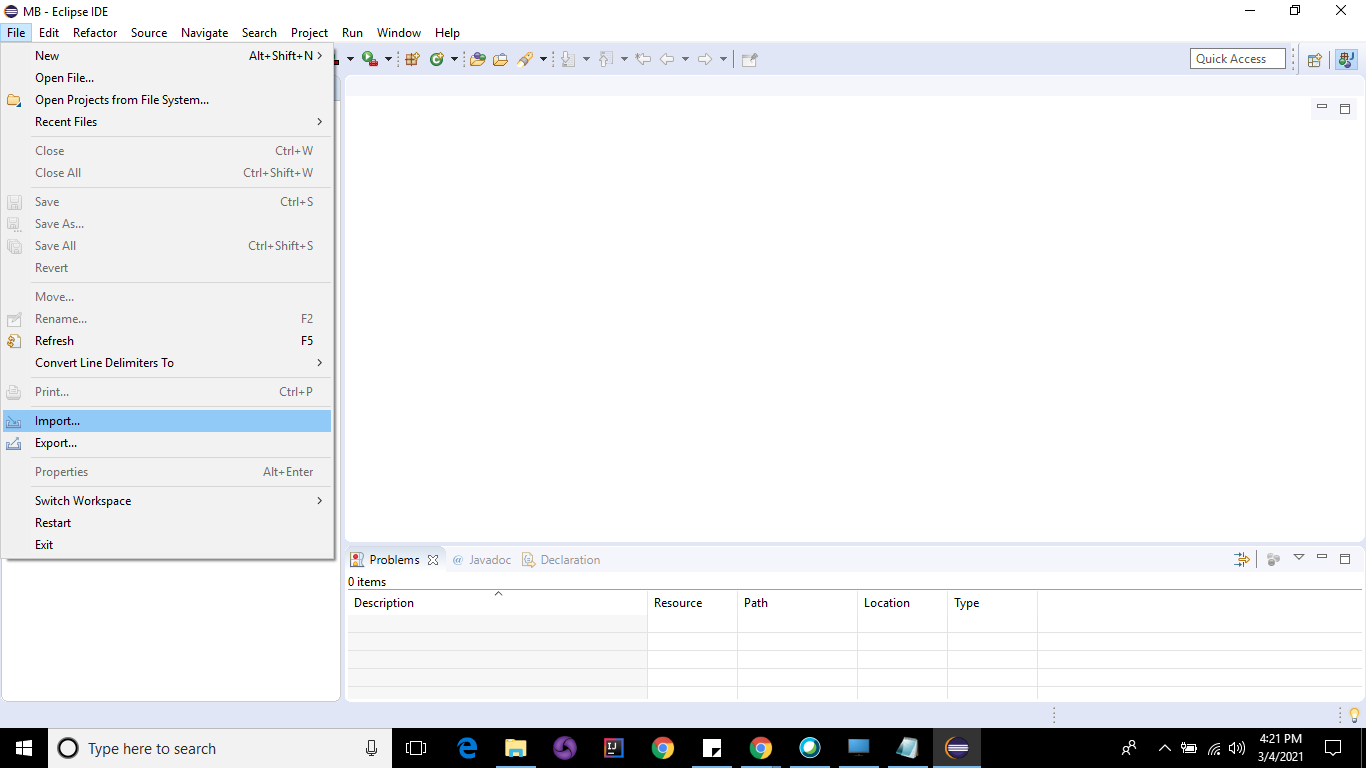


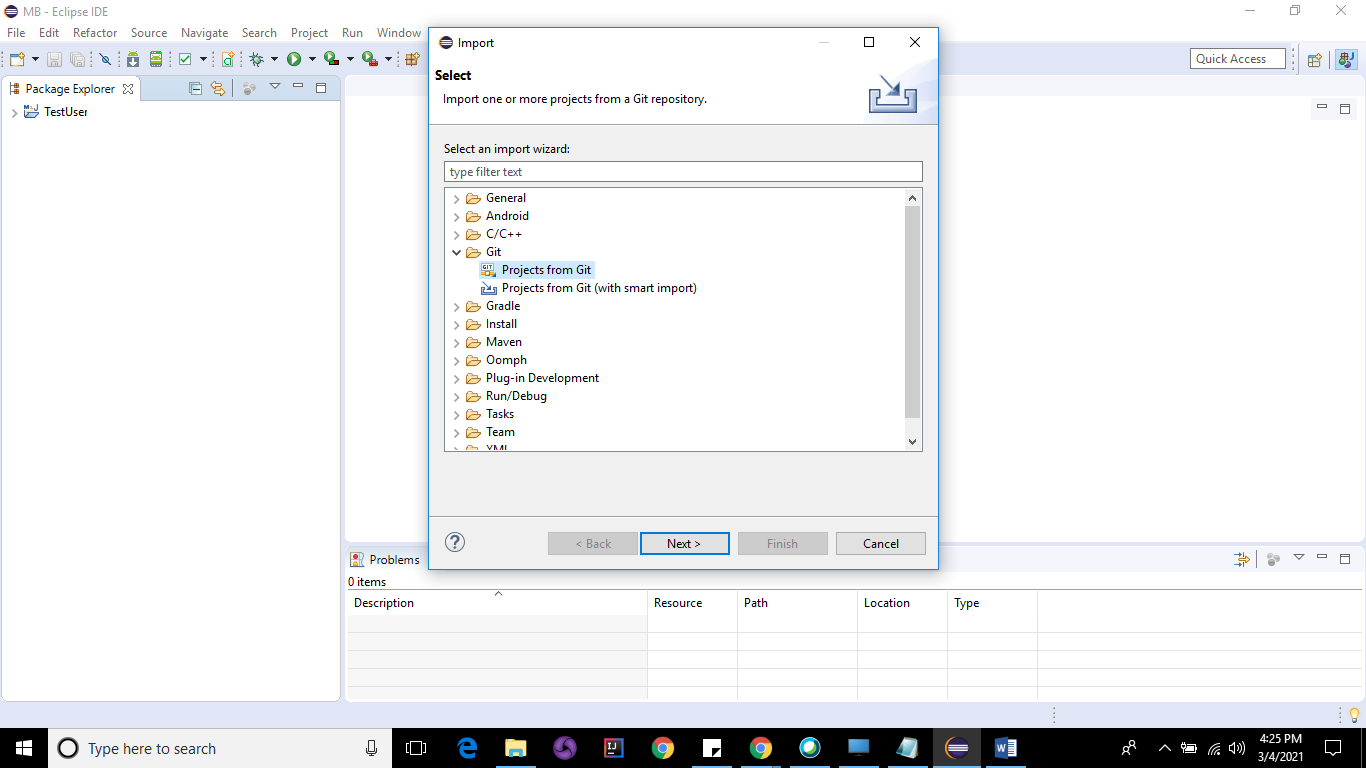


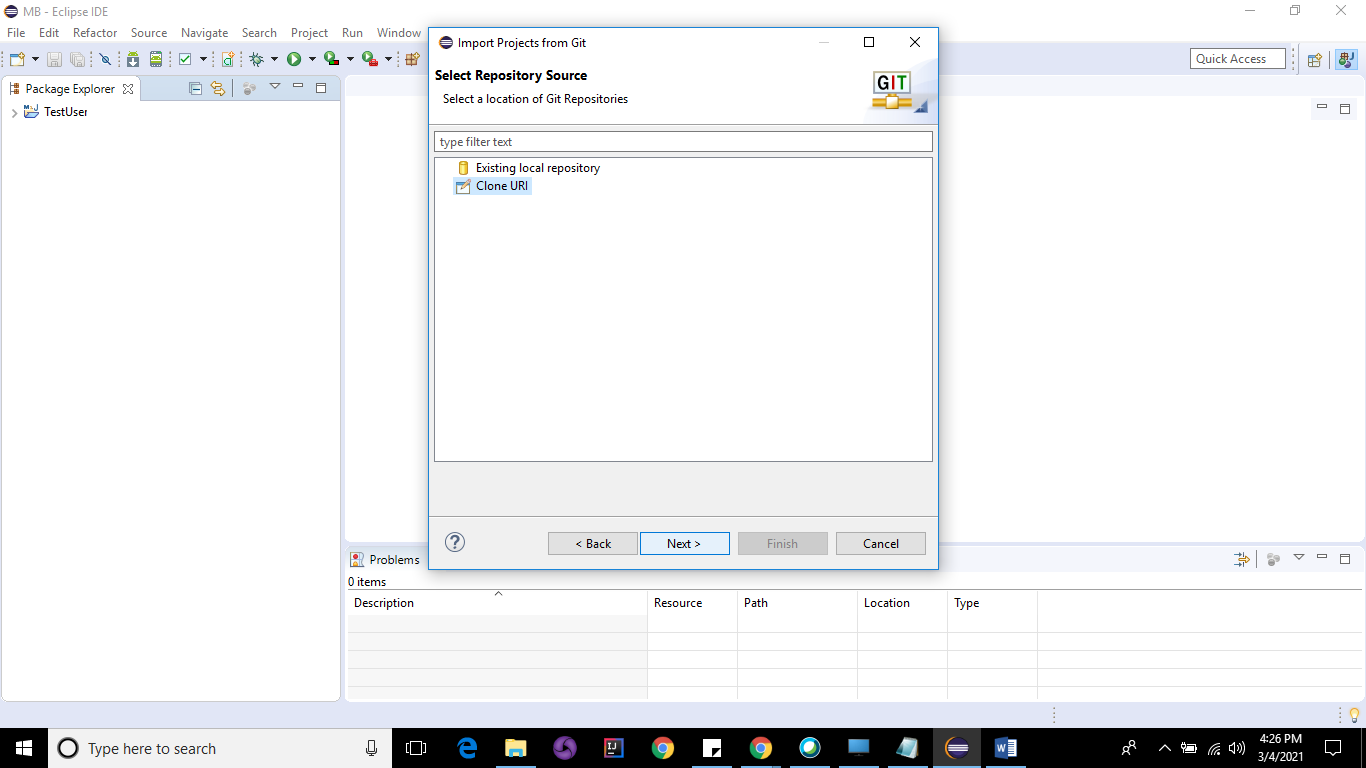


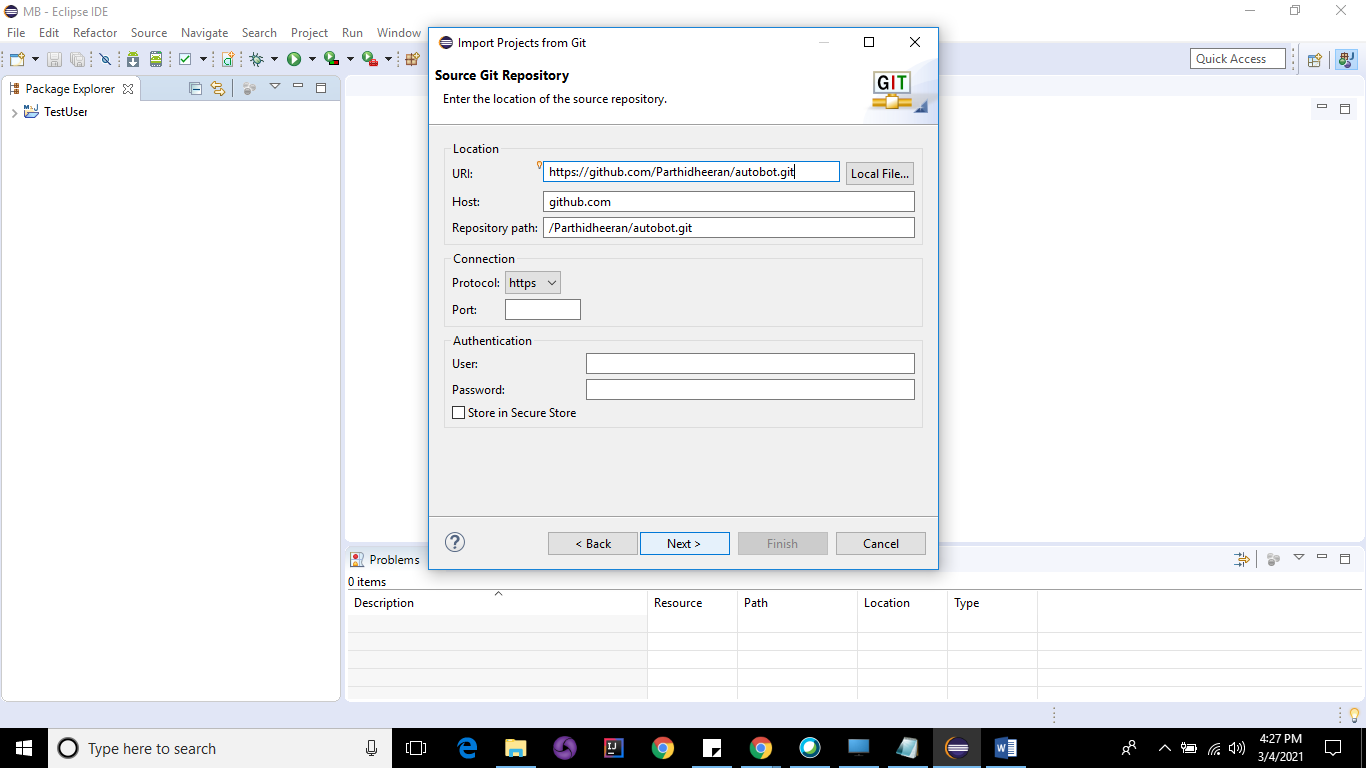


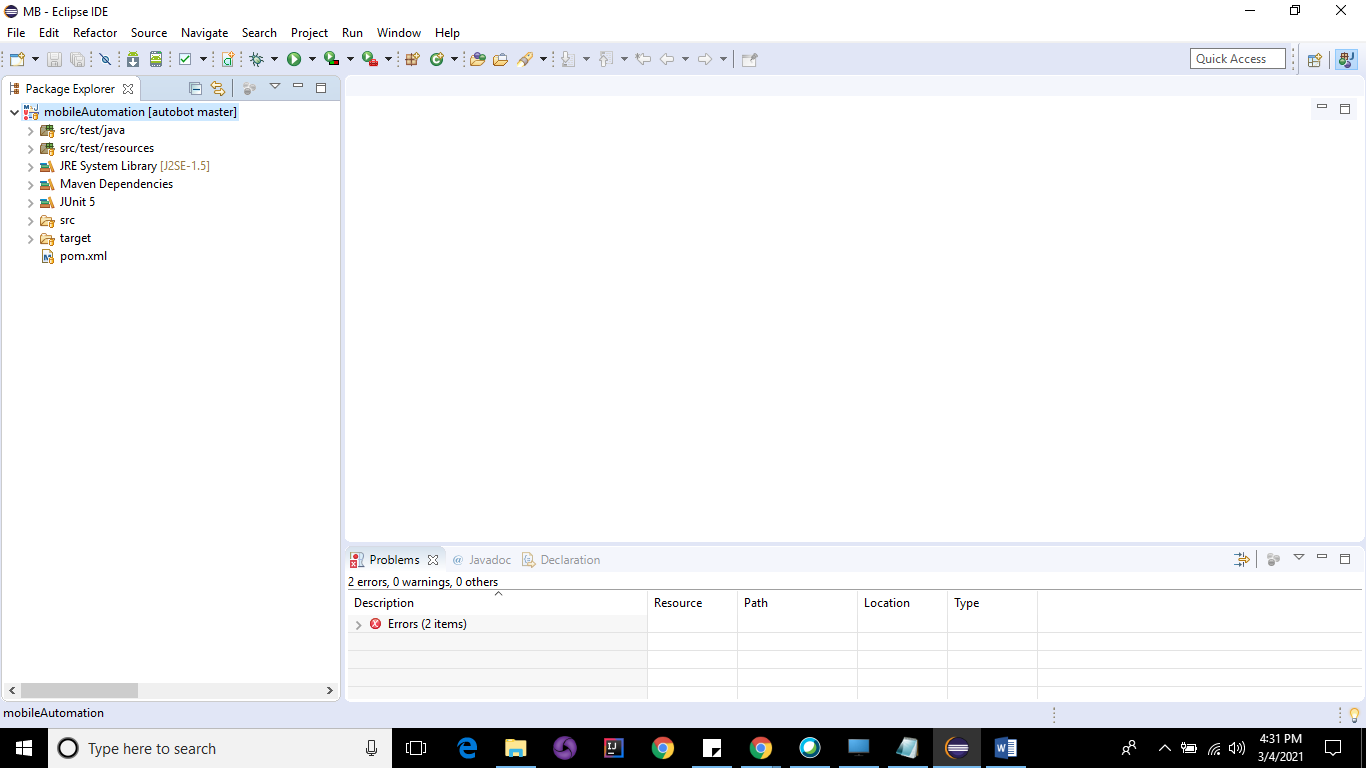
1. **Steps to Import Project from git for Test Execution:**
2. Mobile Automation git url:https://github.com/Parthidheeran/autobot.git



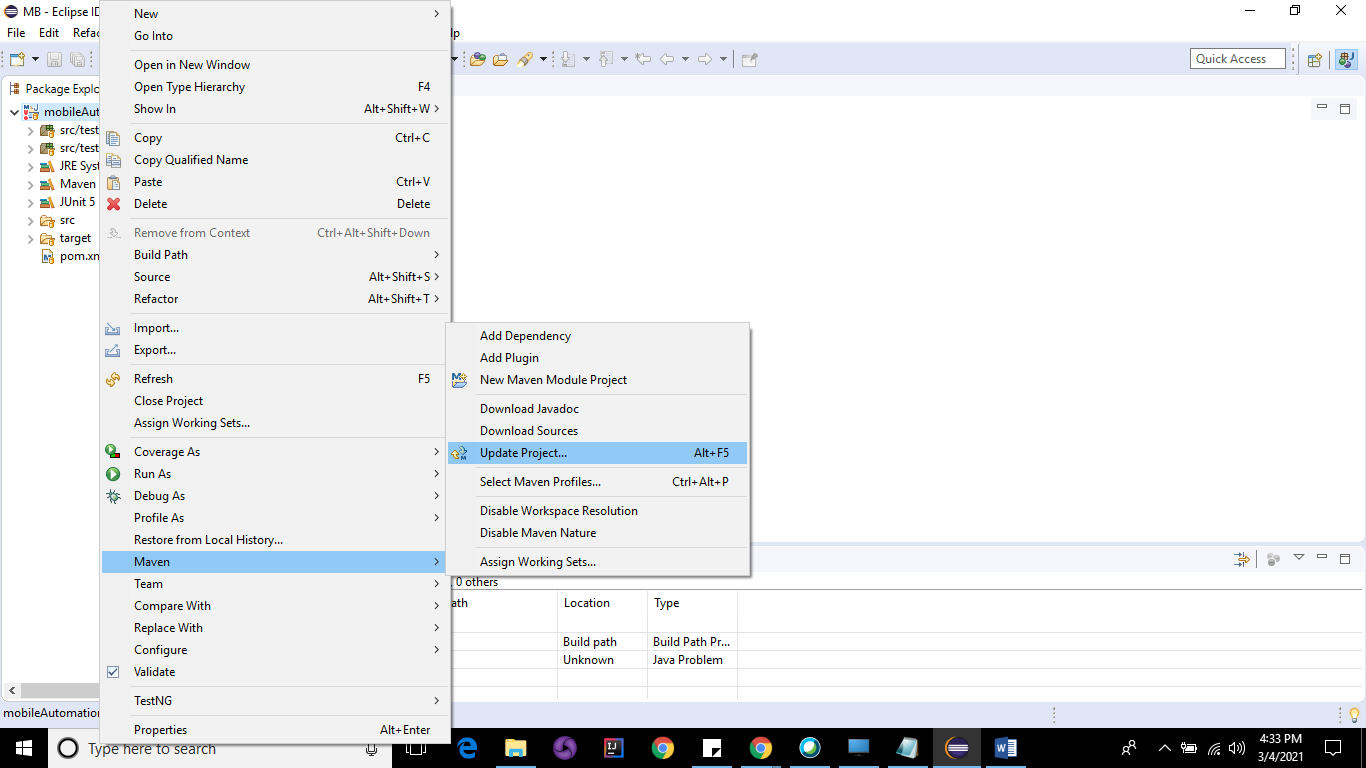


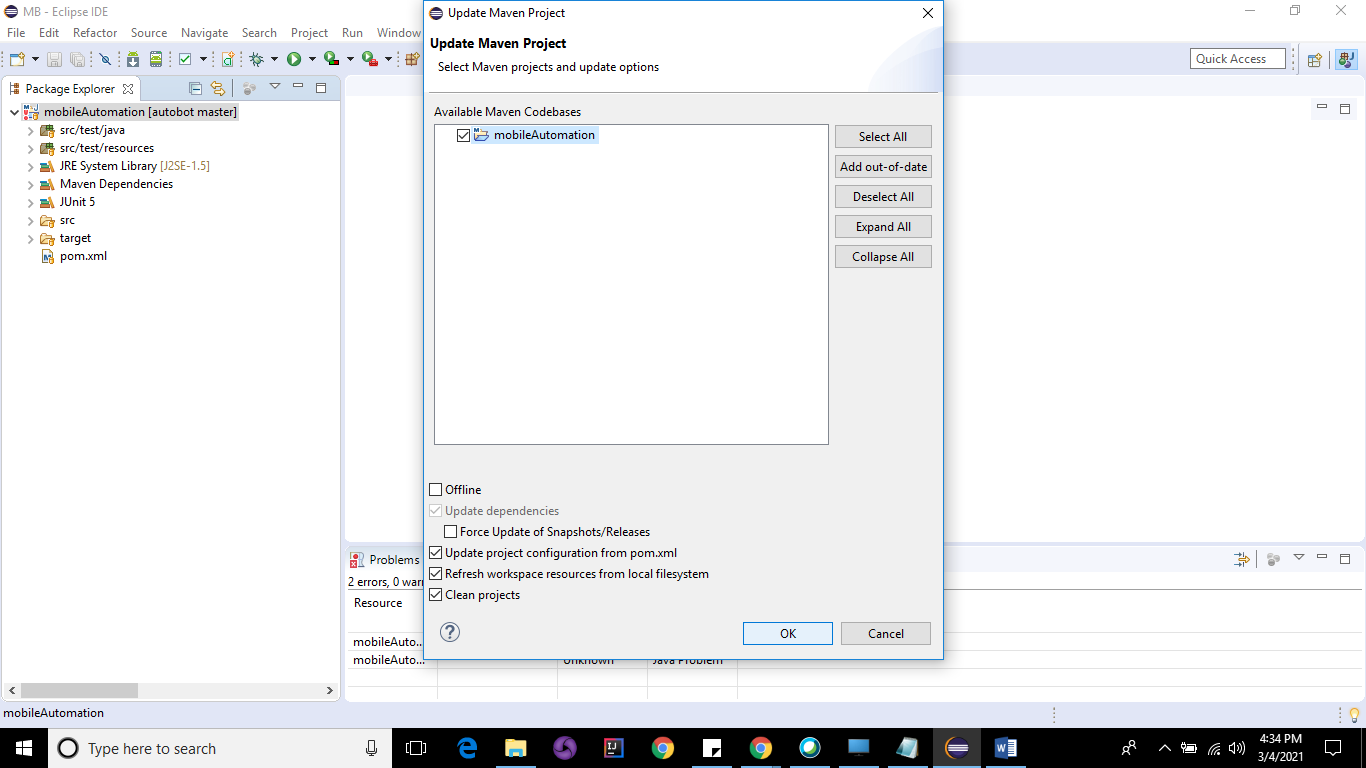


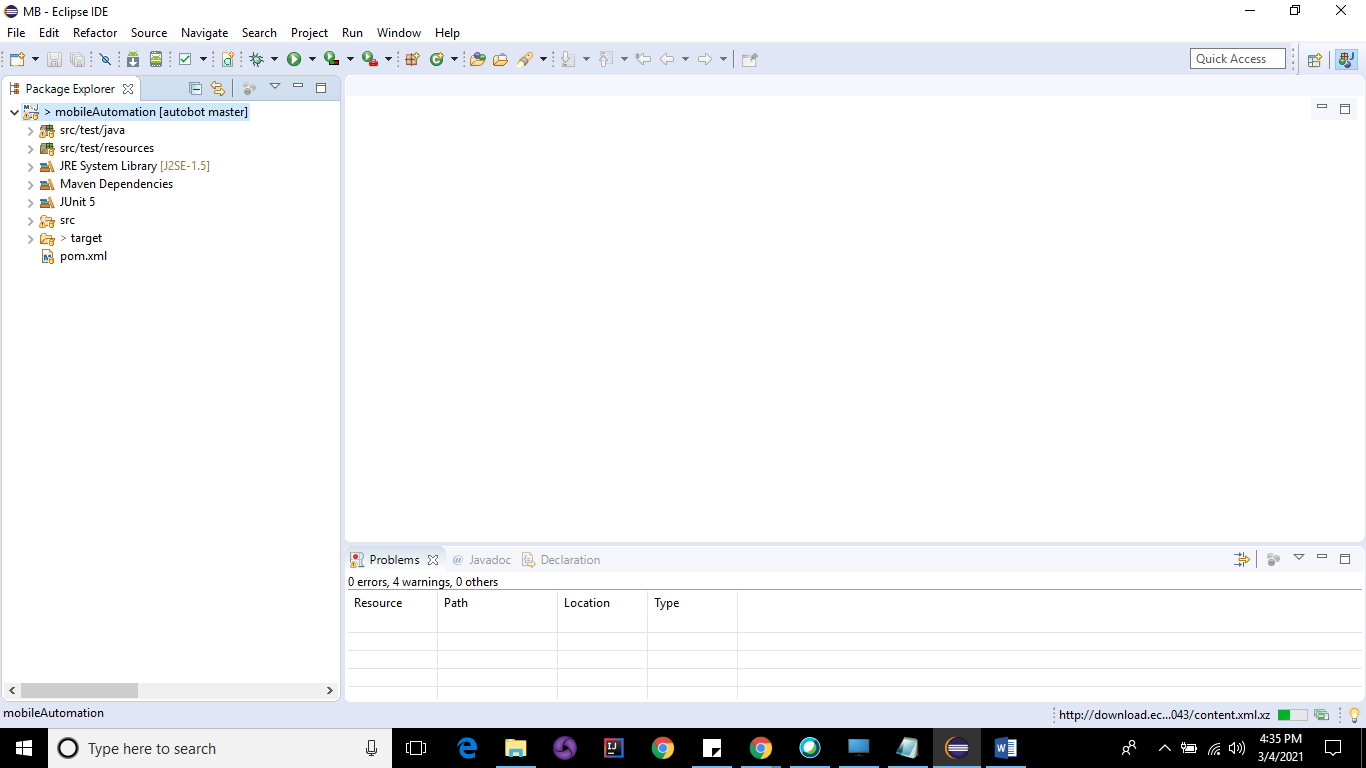




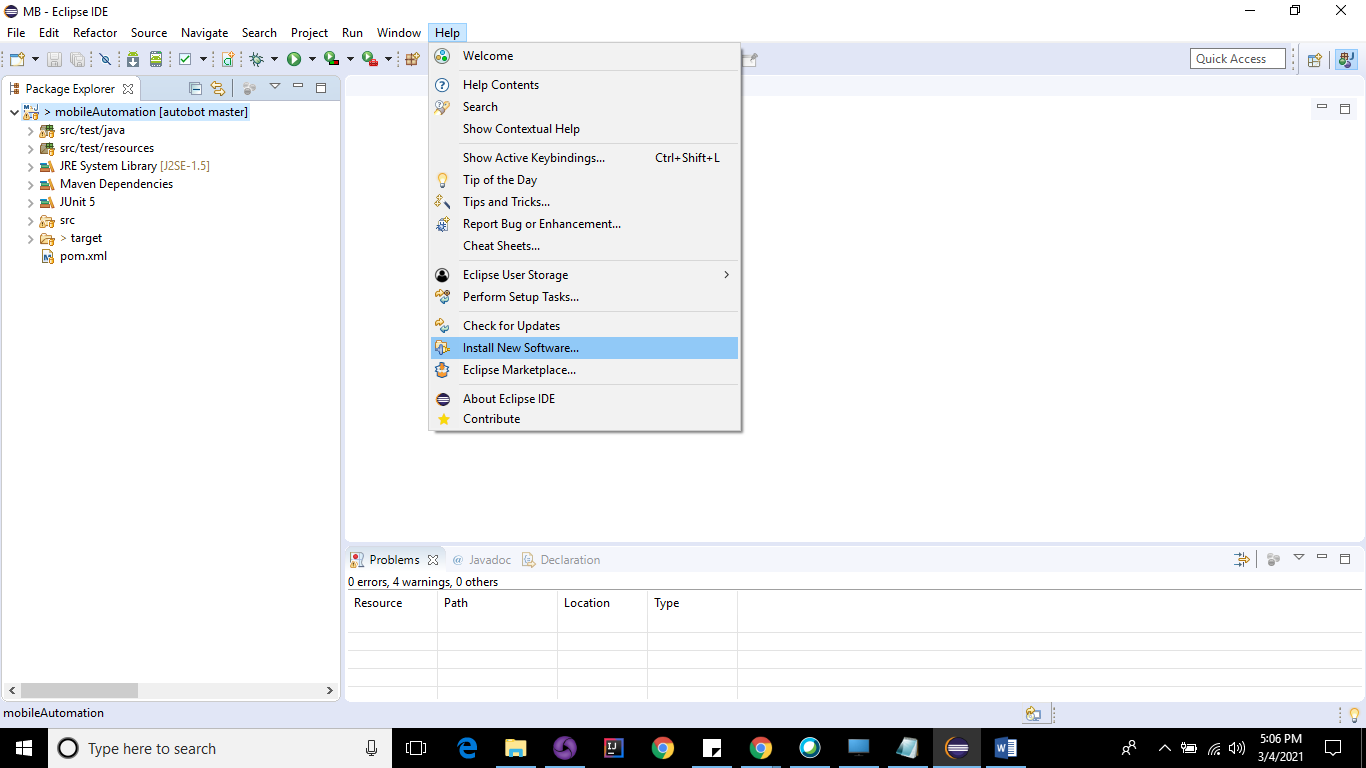
1. Steps to clear import errors –> Update Maven project



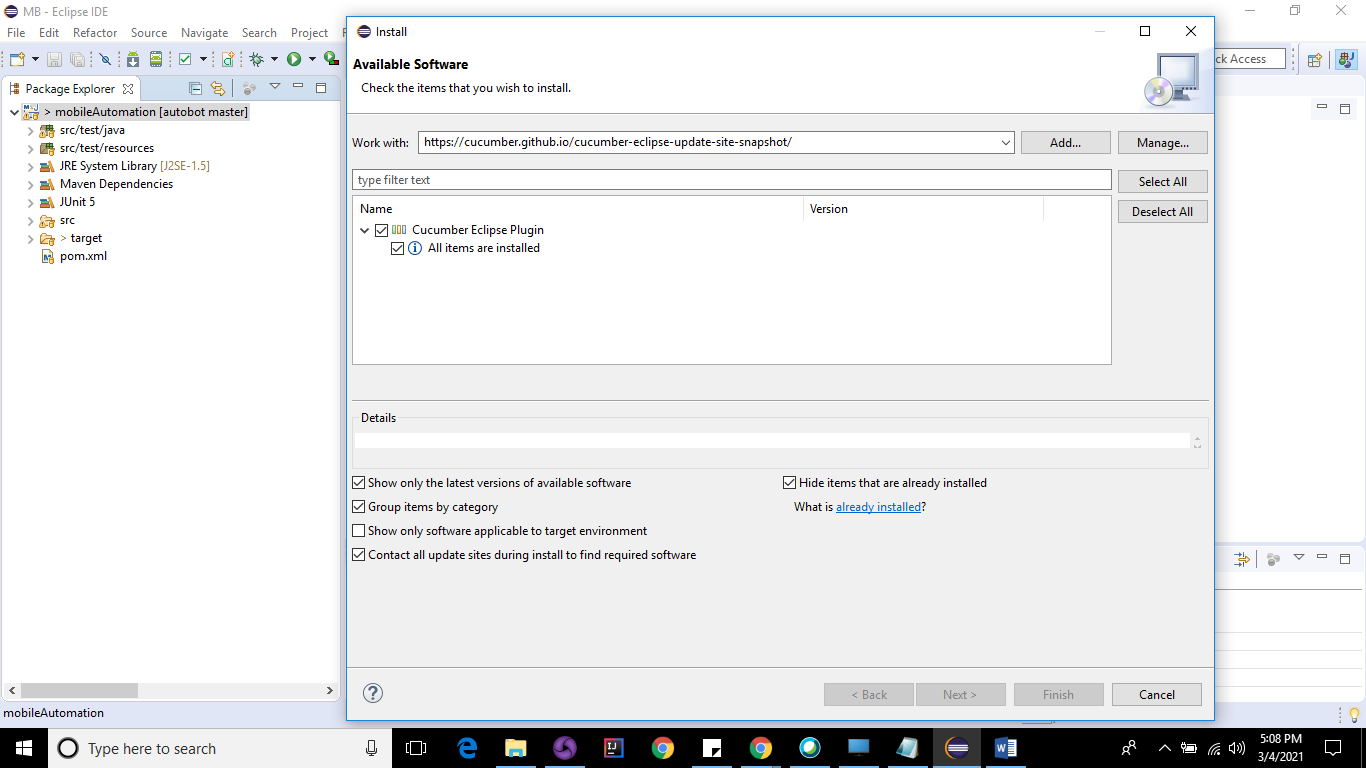




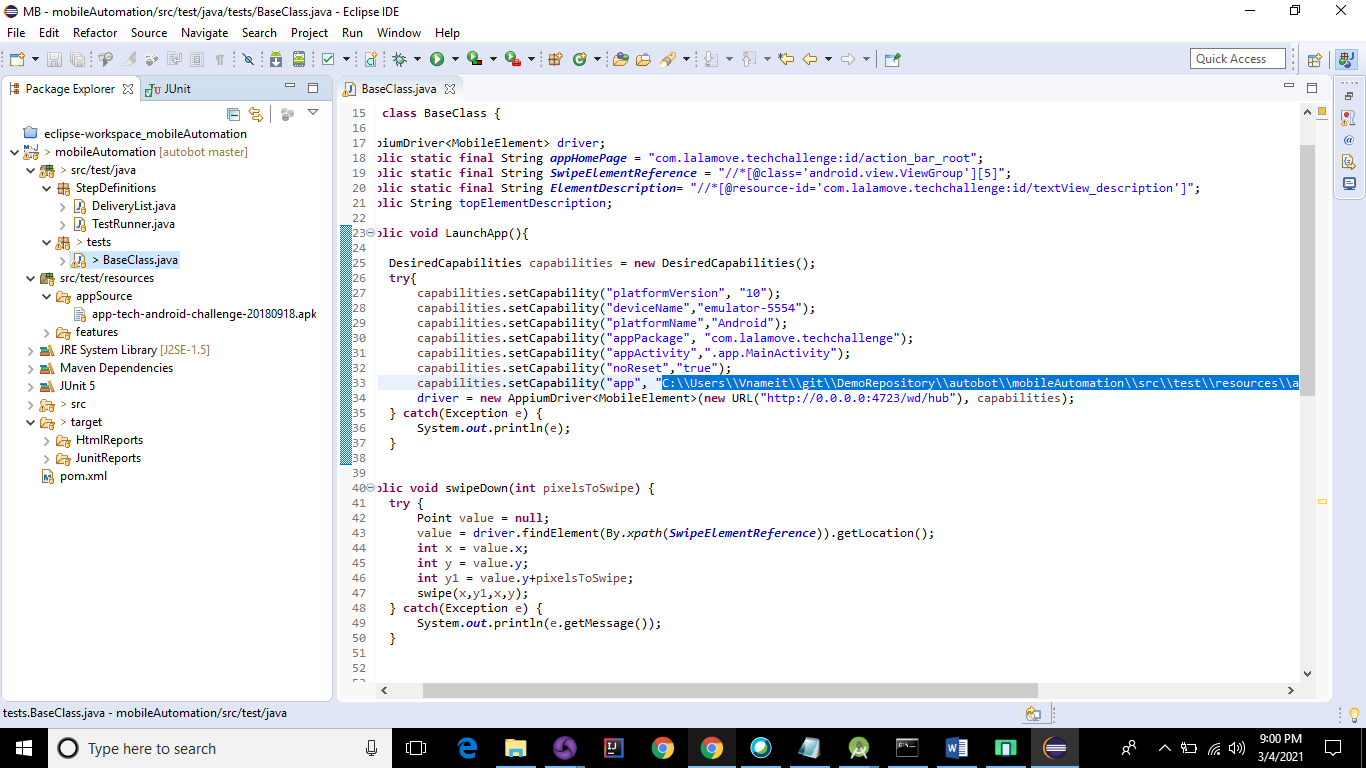
1. Install Cucumber and AVD Plugins using Install New Software/Marketplace option:



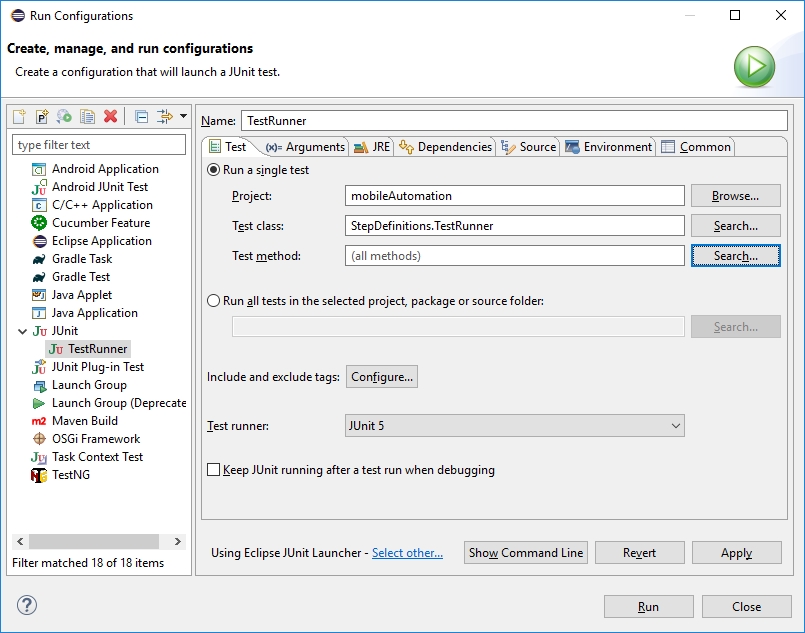
Cucumber url: <https://cucumber.github.io/cucumber-eclipse-update-site-snapshot/>



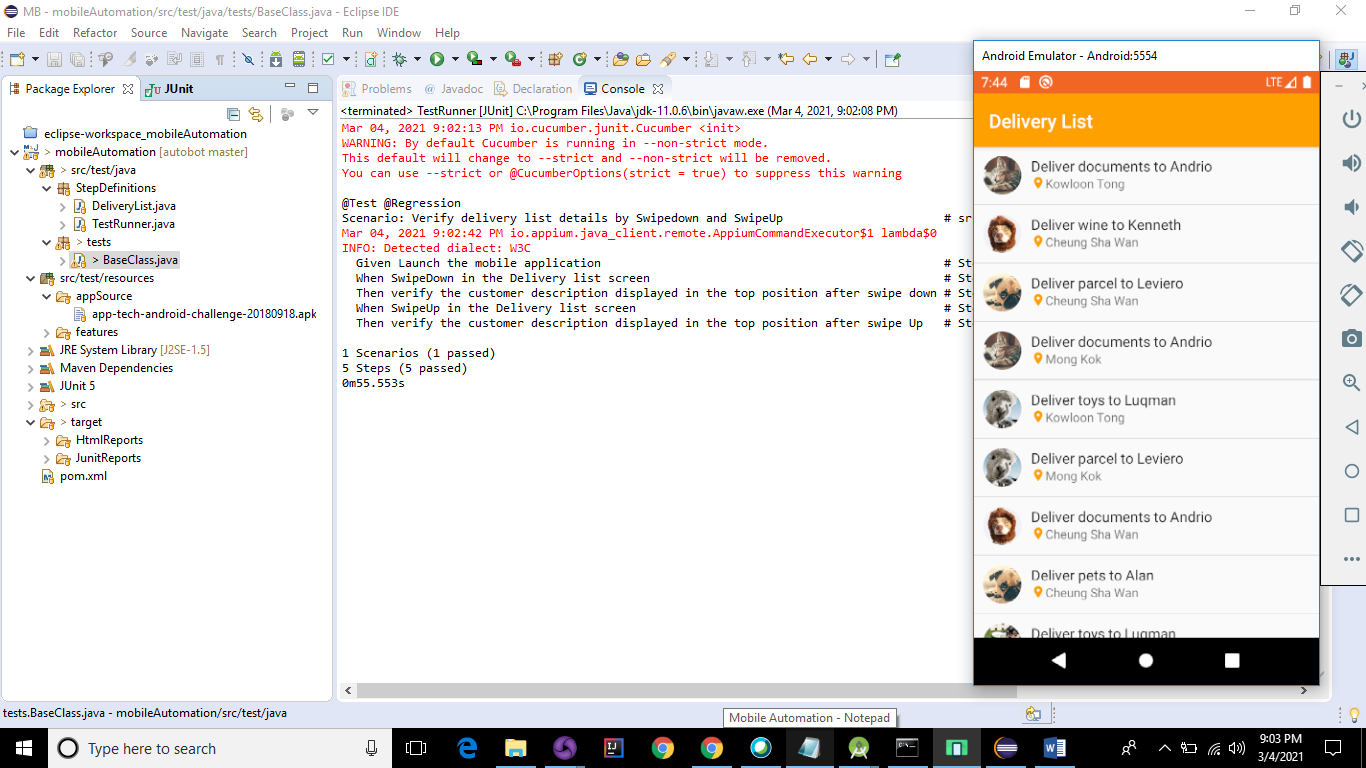
1. **Perform Test Execution:**
2. Update the app source path (mistakenly provided obsolete path of the file) in the BaseClass.java file



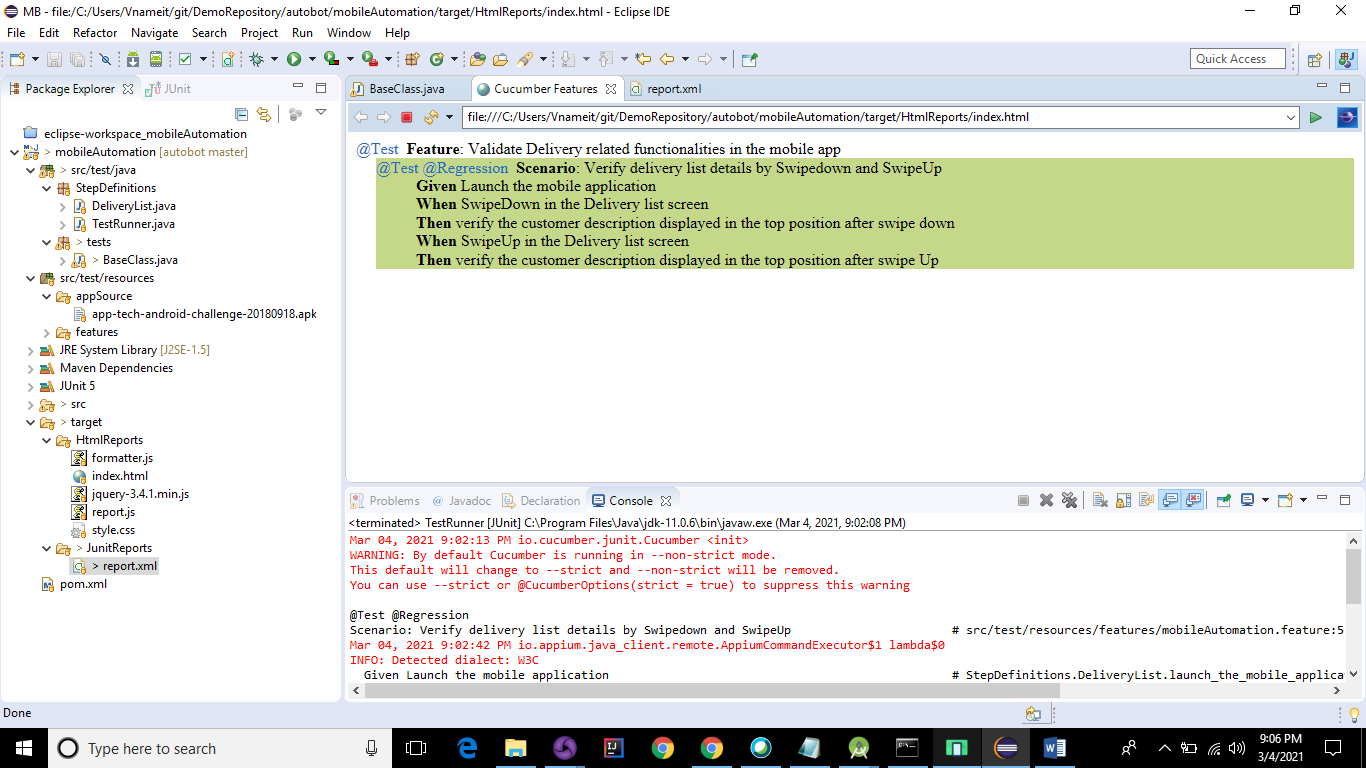
1. Configure Junit test to perform test execution

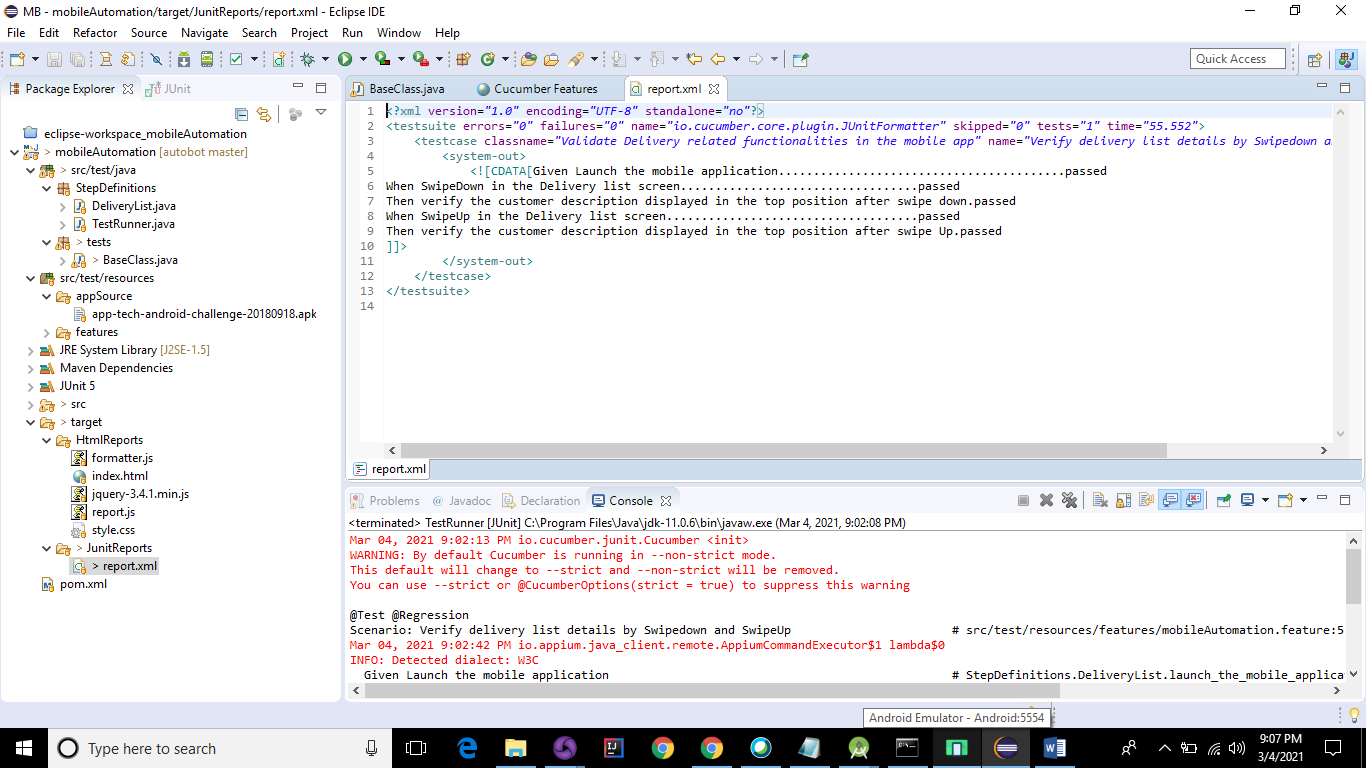


1. Once click on Run button, verify the successful test run in Eclipse console and also in Emulator



1. After successful test execution, HTML and Junit test results will be generated under target folder. (Configured TestRunner to generate to generate html and Junit, can be updated to generate json and other test result format).





1. **Mobile app Issues:**
2. Able to perform app launch and scroll through the Delivery list screen, while click on anyone of the delivery item to view Delivery details, mobile app is closing with below mentioned error message. Tried manually in the emulator, getting same error message, hence unable to automate other steps.

