**Mobile Automation Framework - VCM**

Framework : WebDriverIO

Language: JavaScript

IDE: VS Code

Integration : Cucumber  
Other tools & technologies involved: Node.JS, Appium, Android Studio

* Apps : contains apk files
* Config : configurations files related to each type of platform like android or iOS
* Logs : log files
* Node\_modules : all the required node modules and dependencies
* Reports : Execution repots (Allure, HTML, JSON)
* Src folder: sub-folders inside it are :

1. data : contains input data files
2. featureFiles : Feature files translated from JIRA manual tests
3. helpers : re-usable common functions/methods
4. Logs : log files
5. pageObjects : pageObject files based on the application flow
6. screens : application screens which will hold the object repository of the elements with locator’s
7. stepDefinition : contains stepDefinition files

* Package.json : contains list of components that need to be installed and npm install will refer this file to install
* README : Instructions on how to use this framework

A screenshot of a computer

Description automatically generated

**Prerequisites:**

1. Node.js ( To install Node.js for windows [click here](https://www.qafox.com/webdriverio-install-node-js-and-configure/))
2. Set Node Home path in system variable
3. Create new NPM project
4. Download and import project into Visual Studio code
5. Understand the importance of package.json in Node projects
6. Install WebDriver CLI dependencies
7. Generate configuration file to store WebDriver IO settings
8. Create Cucumber spec/Test file to write first automation program

Reference directory: <https://webdriver.io/docs/gettingstarted>

Helpful points:

* Install appium of particular version: npm install -g [appium@1.19.1](mailto:appium@1.19.1)
* Uninstall appium- npm uninstall -g appium
* For making appium compatible to the machine : npm i @wdio/appium-service --save-dev
* Start appium server at particular port: appium -p 4726
* Appium doctor commands: npm install appium-doctor -g
* appium-doctor –-android
* When feature file is not redirecting to step definition:

VSCode file>Preferences>Settings> Type cucumber> scroll to bottom> autocomplete>

Add this under cucumberAutoComplete:

    "cucumberautocomplete.steps": [

        "/\*\*/stepDefinitions/\*\*/\*.js"

**Appium :**



**Appium Inspector:**

A screenshot of a computer

Description automatically generated

**A screenshot of a computer screen

Description automatically generated**

**Step Definition files:**

* A screenshot of a computer screen

  Description automatically generated The feature file contains all the scenario with the required steps and the filename needs to be in ““feturename“.feature” format.
* All the required screen objects for a step or screen will be captured in “xxxx.screen.js“ file.
* With captured screen objects for the give page all action related to the step will be written in the “xxx.page.js” file.