# Configure Build Server

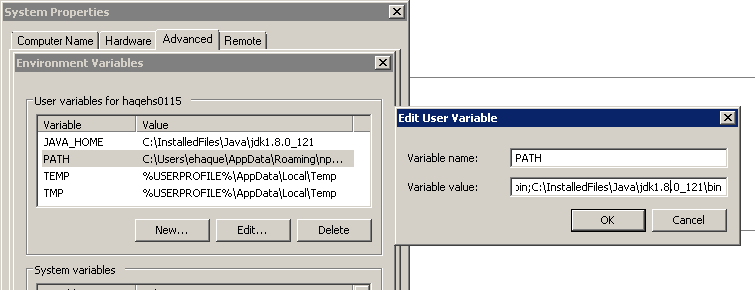
Follow the steps below to prepare the build server to build android and iOS packages

## JDK

Install and configure JDK [version: 1.8.x]

<https://jdk.java.net/java-se-ri/8-MR3>

- Set “JAVA\_HOME” environment variable to java root directory and include the bin directory in path variable



- Test from command line if java installed properly by following command

java --version

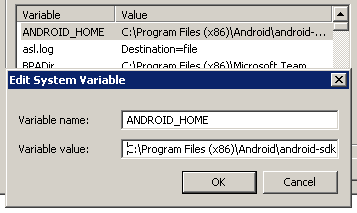


## Android SDK & Studio

Install and configure Android Studio and Android SDK. The latest version should be sufficient.

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

- Set "ANDROID\_HOME" in environment variable and include in path variable



Make sure you build and run a sample project and it works.

### Gradle

Install Gradle. The latest version will work.

**Make sure the path to the Gradle executable is in your environment variables**

<https://gradle.org/install/>

## Node & Modules

Install NodeJS Version 8.X. [Currently, the environment only works with Node 8]

- Install the most recent version of Node 8 from following location

<https://nodejs.org/dist/latest-v8.x/>

Note: make sure node installed properly using following two commands:

node -v

npm -v

**please note that we are currently using npm version 4.2.0 and node version 6.X.**

- Install cordova and ionic (specific versions)

**npm install -g** [cordova@8.0.0](mailto:cordova@8.0.0)

**npm install -g** [ionic@3.19.1](mailto:ionic@3.19.1)

**npm install -g cordova-android@6.4.0**

npm install -g [cordova-ios@4.5.4](mailto:cordova-ios@4.5.4)

Note: Please make sure the cordova and ionic versions matches as below:

cordova: 8.0.0

cordova-android: 6.4.0

cordova-ios: 4.5.4

ionic: 3.19.1

cordova version can be found by:

cordova -v

cordova platform version

ionic version can be found by:

ionic -info

- gulp using following command

**npm install -g gulp@3.9.1**

### Navigate to the student-cordova-wrapper\app folder

-Install all the packages required in gulp

**npm install gulp-rename gulp-replace yargs run-sequence del gulp-chmod child\_process fs**

or using

**npm install --save-dev**

- taco-cli using following command

**npm install -g taco-cli**

Now, you need to tell taco that you don’t want to give customer feedback. This needs to be done before running anything that needs taco, or it will hang.

Run: taco feedback, answer N for no, then let it time out (might take 30 seconds to a minute)

### Navigate to the student-cordova-wrapper\app\plugins\cordova-custom-config

Run npm install from that directory

Ensure you have a platform to build against. Check the platforms folder in the student-cordova-wrapper directory. For newly cloned repositories, this folder will be empty or non-existent.

Use the following command to create a build platform:

cordova platform add <ios|android>

# OSX

### Install, Configure remote build agent in MAC OSX and save the remote configuration in taco-cli to build ios package.

- Install remote build in host (MAC OSX)

**sudo npm install -g remotebuild**

- Run remote build agent in insecure mode and collect the host and port information

**remotebuild --secure false**

Note: A pin need to be generate to run remotebuild in secure mode. Pin can be generate using command “**remotebuild certificates generate**”

- Current Remote build server information:

Host: 10.200.124.60

Port: 3000

- Save remote build configuration in taco-cli using following command

**taco remote**

**or taco remote add ios**

To access MAC OSX remotely, VNC Viewer can be used:

MAC Mini Credentials are below:

Host: 10.200.124.60

Username: hadron

Password: Cec2017

Remote desktop client for MAC:

RealVNC (VNC Viewer)

URL: <https://www.realvnc.com/en/connect/download/viewer/>