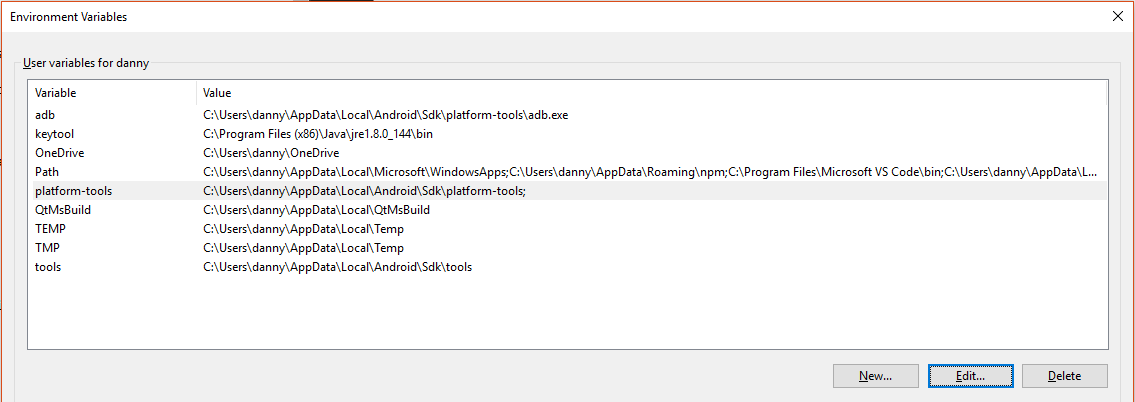
**Possible Errors:**

Environmental variables:

If any of the commands in this document do not work. Go to your windows Environmental Variables, and make sure that the following items are in there, and with the correct path.

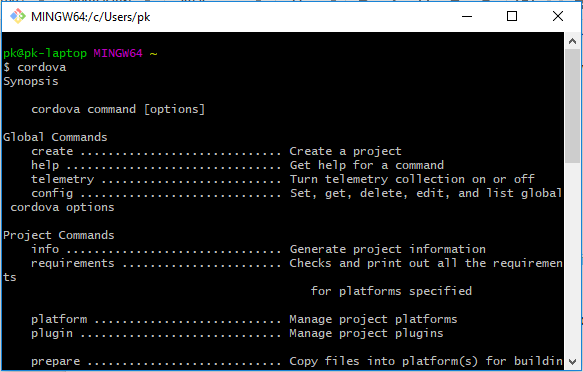
Platform-tools: C:\Users\user\AppData\Local\Android\Sdk\platform-tools

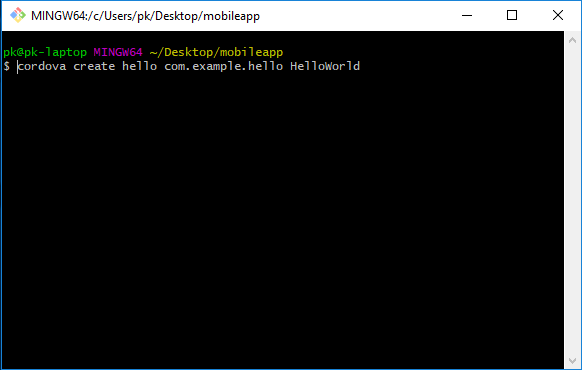
Tools: C:\Users\user\AppData\Local\Android\Sdk\tools  


**How to install Apache Cordova**  
Cordova Documentation: <https://cordova.apache.org/docs/en/latest/guide/cli/index.html>  
  
You will need to install node.js to invoke “npm” : <https://nodejs.org/en/>

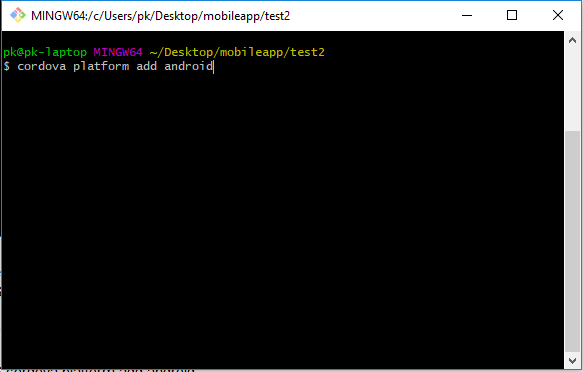
Install the cordova module using npm utility of Node.js. The cordova module will automatically be downloaded by the npm utility.

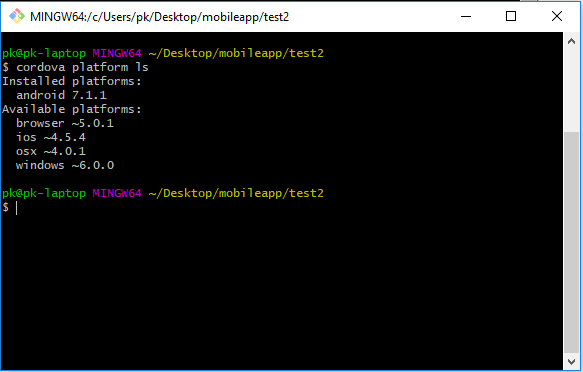
**OS X and Linux:**  
 $ sudo npm install -g cordova  
  
**Windows:**  
 C:\>npm install -g cordova  
  
The **-g** flag above tells npm to install cordova globally. Otherwise it will be installed in the node\_modules subdirectory of the current working directory.  
  
Following installation, you should be able to run cordova on the command line with no arguments and it should print help text.



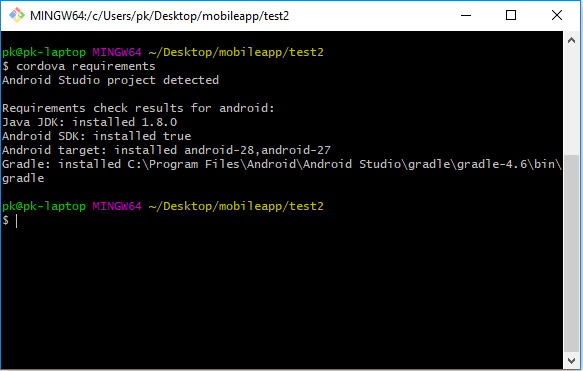
**Creating The App**  
Go to a directory where you would want to maintain your source code, and create a cordova project:  
  
 $ cordova create hello com.example.hello HelloWorld  
  
  
After creating your cordova app, transverse into the created directory:  
 $ cd hello

Once in the directory, you will now have to add your platform (we used the android platform):  
 $ cordova platform add android



You can also check your current set of platforms:  
 $ cordova platform ls  
  


Now we need to install the dependencies for building the application. To build and run apps, you need to install SDKs for each platform you wish to target.  
  
To check if you satisfy requirements for building the platform:  
 $ cordova requirements



**Plugins**

Only plugin required is the “Camera” plugin, version 4.0.3 or higher.

1. Navigate to project directory
2. Enter command: cordova plugin add cordova-plugin-camera

**NOTE\***  
Java JDK version 1.8.0 is a must, any other versions may not work (you can try 1.9.1 if you can't find version 1.8.0 online).  
Android SDK is installed through Android Studio  
Android Target- android-28, and android-27, is installed through Android Studio.

After all the dependencies are satisfied you should be able to build the app now.  
  
Run the following command to build the project for *all* platforms:  
 $ cordova build  
  
You can optionally limit the scope of each build to specific platforms - 'android' in this case:

$ cordova build android

**Running Cordova Mobile Application on WAMP or XAMMP (For easier debugging)**Install WAMP or XAMMP, make sure your apache and mysql services are running.

If using **WAMP**:

Transfer the api files and all the files in the “www” folder into the “www” folder.  
  
If using **XAMMP**:

Transfer the api files and all the files in the “www” folder into “htdocs” folder.

**Setting up localhost SMTP for email service:**  
  
SMTP EMAIL:

advicebee123@gmail.com

Advicebee1

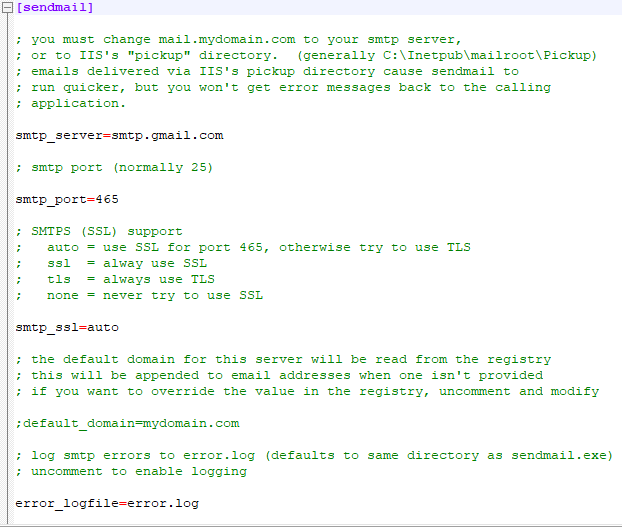
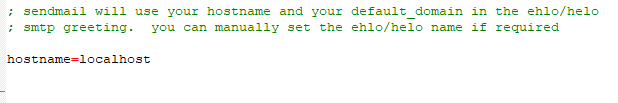
MODERATOR EMAIL:

advicebeemod@gmail.com

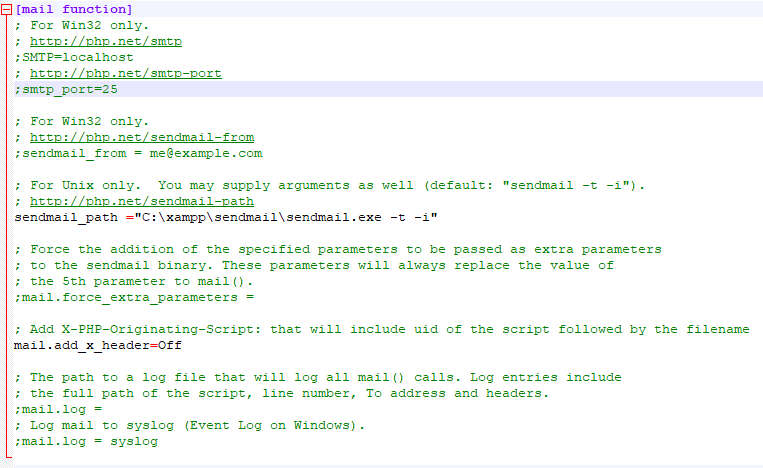
Advicebee1  
  
**XAMMP**:  
You should have sendmail already in your XAMMP folder, if not download sendmail and for WAMP you will need to download sendmail and place it into your WAMP folder.

Access the sendmail folder and open up the sendmail.ini file

**Set smtp\_server:**   
 smtp\_server=smtp.gmail.com  
**Set smtp\_port:**  
 smtp\_port=465  
**Set hostname:**   
 hostname=localhost

  
  
  
After that go into your php.ini file in WAMP or XAMMP and set the pathing to where send mail is residing.

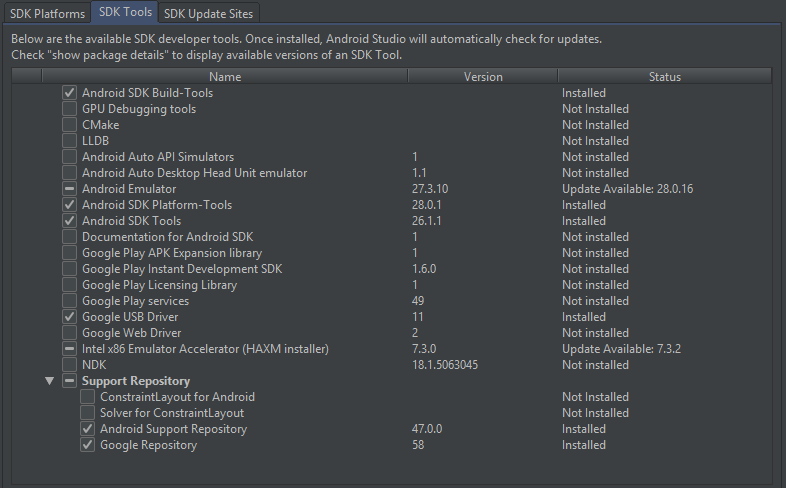
sendmail\_path ="C:\xampp\sendmail\sendmail.exe -t -i"



**\*NOTE**  
WAMP and XAMMP php.ini files are in different locations, and sendmail should come with XAMMP and for WAMP you will need to download sendmail.

**Running Cordova Mobile Application How to Get/use Emulator**

1. Install Android studio for emulator (you will not be using it as a text editor or use Java.)
   1. Go to <https://developer.android.com/studio/> and install android studio
   2. Go to Tools bar
   3. Click on “SDK manager”
   4. Click on “SDK tools”
   5. Check off “Android emulator”, “Android SDK Platform-tools”, “Android SDK Tools”, and “Google USB Driver”



1. Download the android emulator of your choice. For this project we used “Nexus 5X API 24”
   1. Click on Tools
   2. Click on “AVD Manager”
   3. Click “Create new virtual device”
   4. Select Device you wish to emulate to, for optimal performance choose “Nexus 5X API 24”
   5. Click “next” and choose “Nougat” for system image.
   6. Choose “Portrait mode” and click finish.

**You are now ready to use the emulator to run the project. Please use a command prompt (or another method) to run the following commands:**

1. Cordova build android
2. Cordova emulate <Emulator name> | If emulator name is not specified, a default choice will be selected for you.
3. App is now running! If error occurs and app doesn’t auto start, please close emulator and rebuild the app.
4. Inside the emulator, you may need to go to the PlayStore and update “Google chrome.” android apps use web browser apps to run and may need to be up to date.

**Running Cordova Mobile Application on a device:**

Before moving forward, please make sure you have an android device that follows the requirements specification form created for this application.

1. Open Android Studio
   1. Go to “Tools” tab
   2. Click on “SDK Manager”
   3. If you have not already, download “Google USB driver”
2. Enable Developer options on your device. This can be different on each device.
   1. Go to Settings.
   2. Go to “Software information”
   3. Click on “Build number” until developer mode is enabled. “Developer options” should now be available in the settings
   4. Click on “Developer options”
   5. Enable “USB debugging”
   6. Look for the “Revoke USB debugging authorizations” button, you might need it if an error appears.
3. Connect your device using the USB cable provided with the phone.
   1. Allow the device to be seen by your PC.
   2. Using the command prompt, enter the following command: adb device
   3. You will now see if the device is authorized or unauthorized. If unauthorized, go to the “Revoke USB debugging authorizations” mentioned earlier. Click on it, you should see a pop up after replugging the device that allows you to reconnect and make the device authorized.
   4. If device is authorized, navigate to project directory and enter the following command: cordova run.
   5. If you followed the instructions and no errors have occurred, the app should automatically download and run on your device.