Mobile App Development Assignment 1

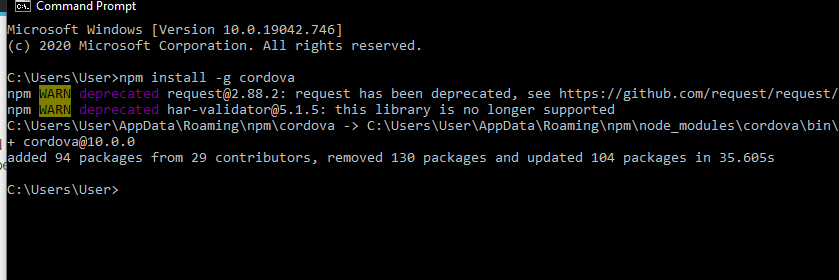
Maaz Malik

Z23385841

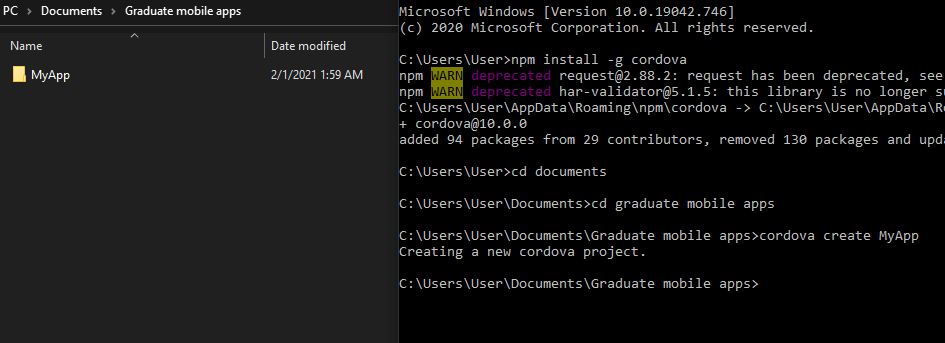
2/1/2021 5:59 p.m.

Professor Jaramillo

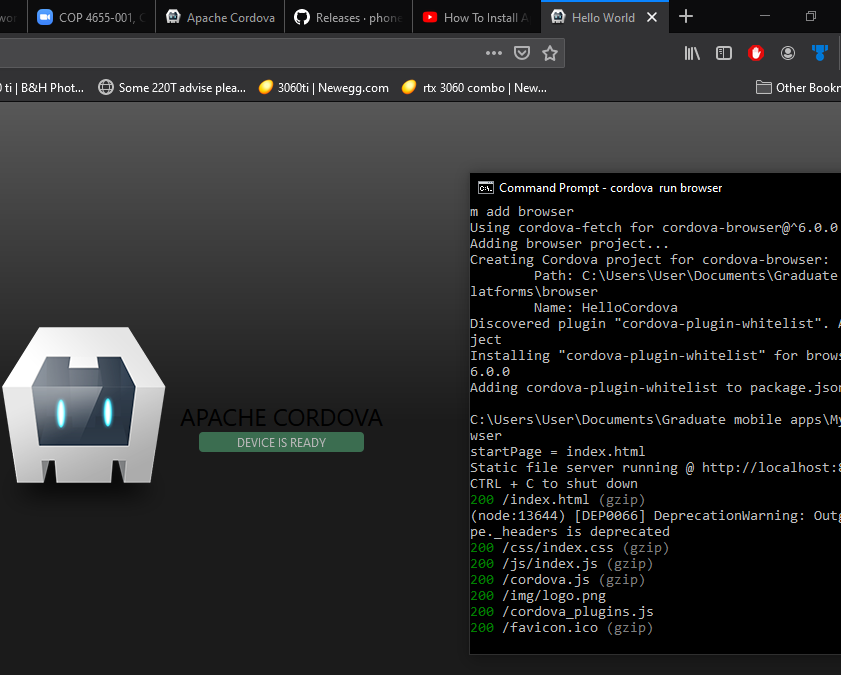
Objective 1



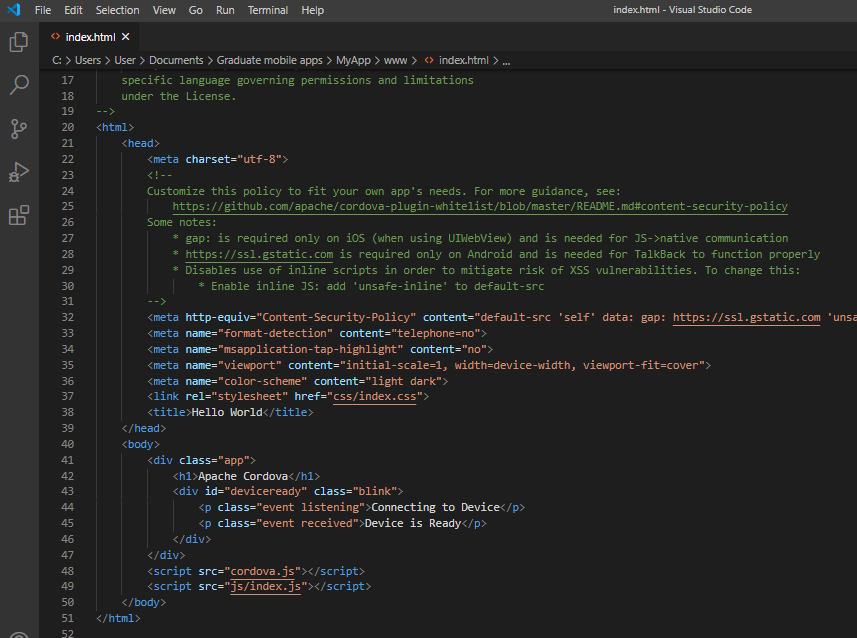
I have Node.JS and have downloaded apache cordova through the command prompt



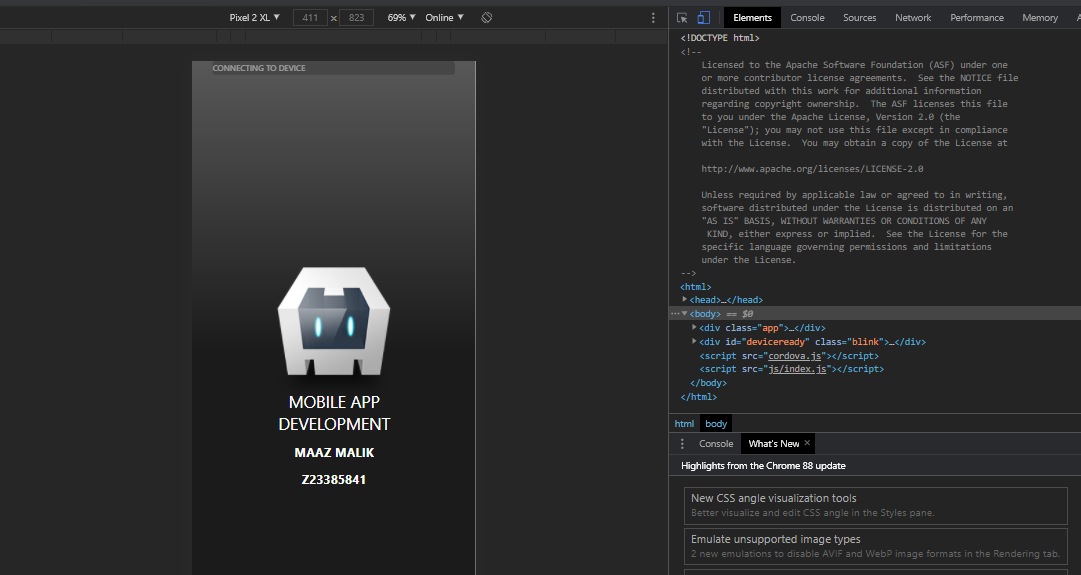
Created a new Cordova project in selected folder



I have created and installed the Hello world Cordova web application

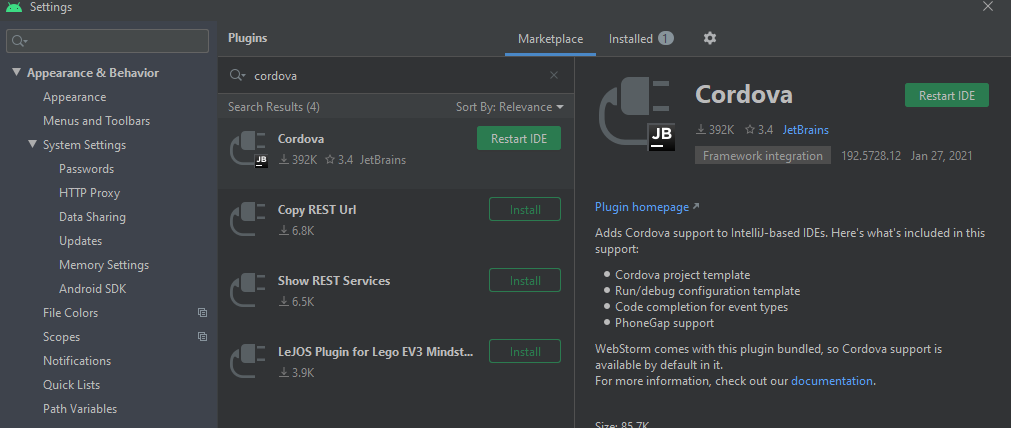


Visual Studio Code was already on my device. I just simply opened the index.html for hello world.

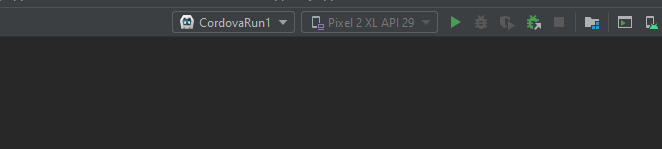


I have modified Index.html to display my name and Student Id. I then proceeded to load it onto google chrome and view it in mobile View mode.

**Objective 2:  Android Studio**

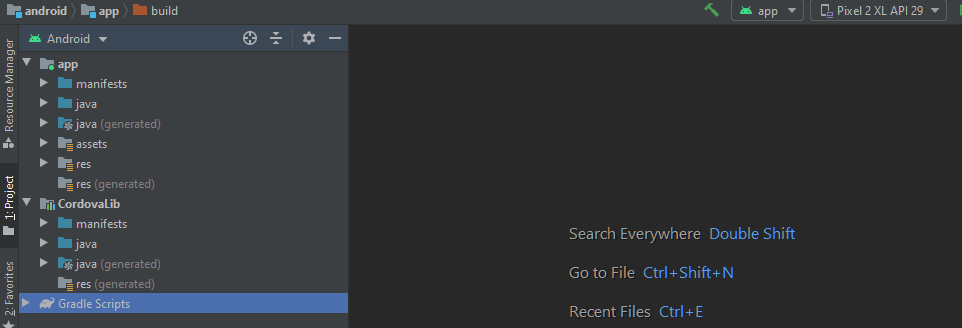


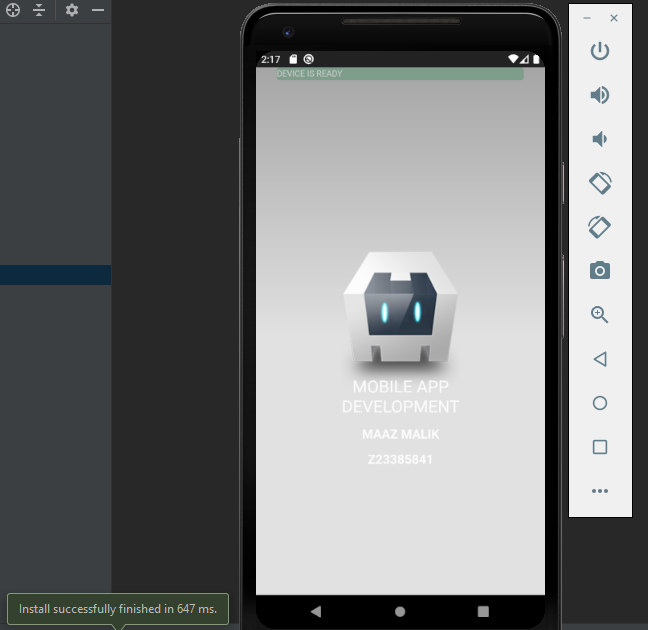
Android studio already installed. Went in and added the Cordova plugin.



Switch the platform to run on Cordova for the emulator.

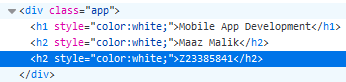
So I already had an emulator installed on my Android studio it was the Pixel 2 XL. I had a lot of trouble trying to run Cordova through android studio but after watching some youtube videos, I was able to have the platform created.





And Voila, I got the Cordova application to run through the Android Studio Emulator.

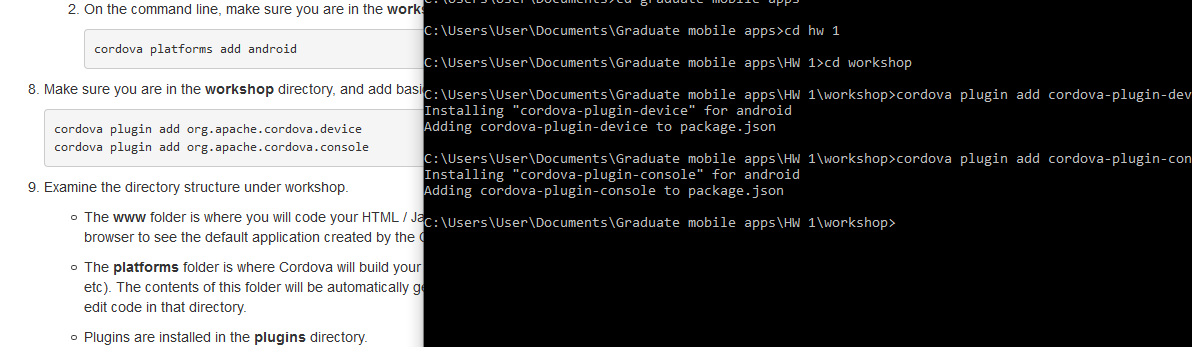
Now it’s a little hard to see but I manually readjusted the Letters to Display as white because on the browser version, the background was black, (see to the previous photo). I changed it in Visual Studio Code and I’ll paste the lines of code below but that’s why the android emulator looks like this.



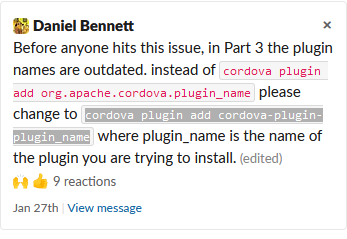
**Objective 3: Apache Cordova Tutorial on Android/iOS**

**Module 1**

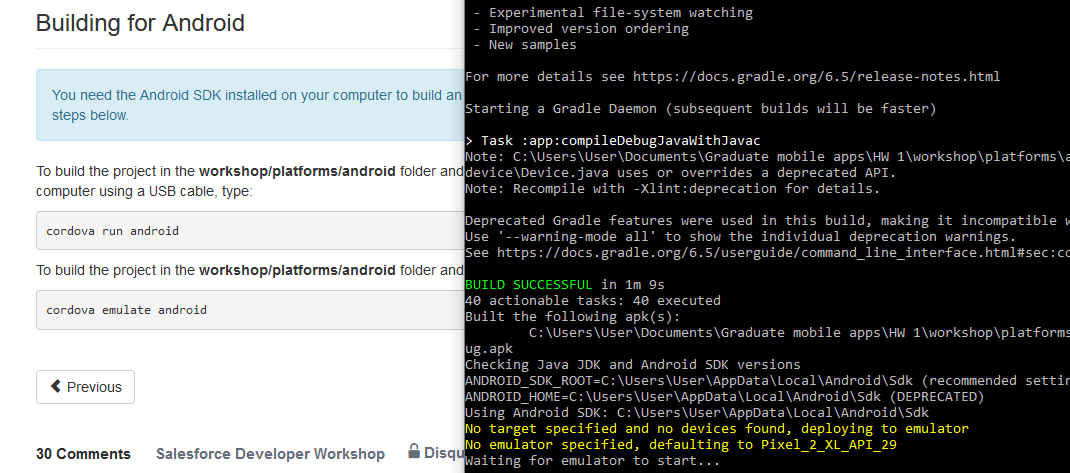
Module 1 was straight forward. I flew through all the steps except for the last step. Shown below is me accomplishing it.



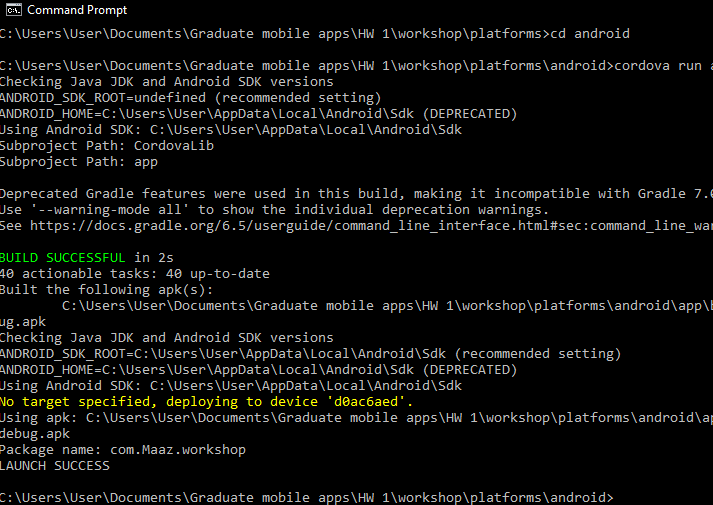
The only reason I was able to solve it Was because I went on the **SLACK** and saw Daniel’s explanation, shown below.



**Module 2**



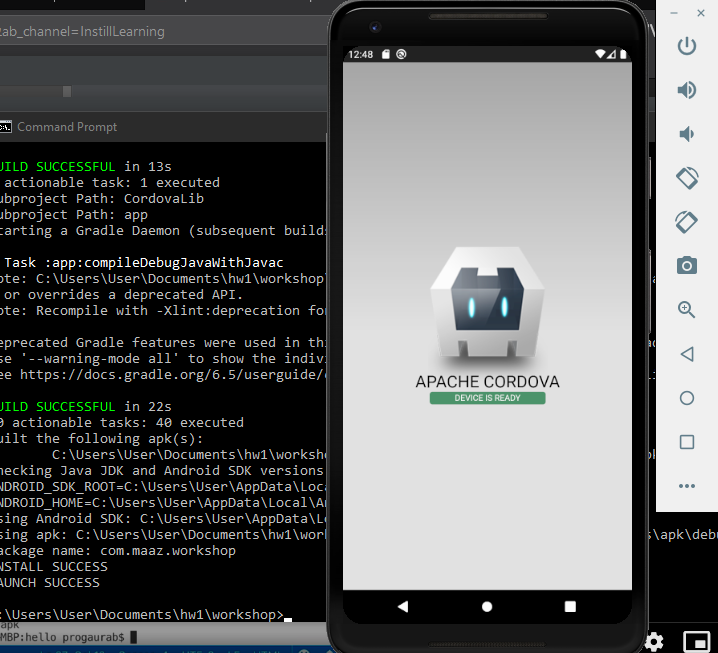
So when I first ran it, it ran successfully, problem was it was in the wrong folder. So It was stuck looking for the emulator.



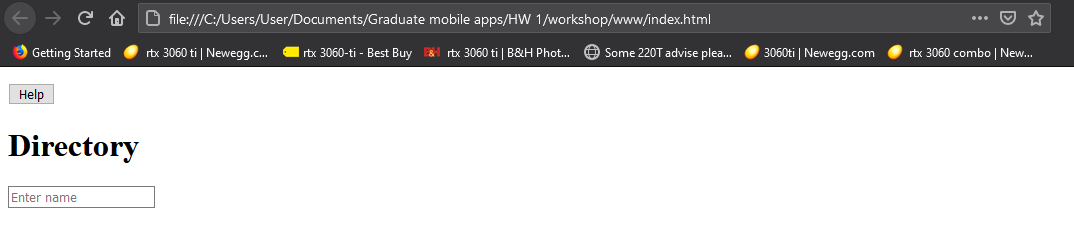
So I ran the command prompt with my Personal phone device plugged in and it ran successfully as u can see above. The Reason I did this is because Daniel Bennet the TA stated in the **Slack** to avoid using Android studio to launch cordova because gradle is weird on there. Below are screenshots from my personal android device. I also screenshotted the time as well.



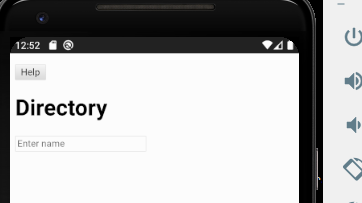
And after sometime I got it to work on the emulator as well.



**Module 3**

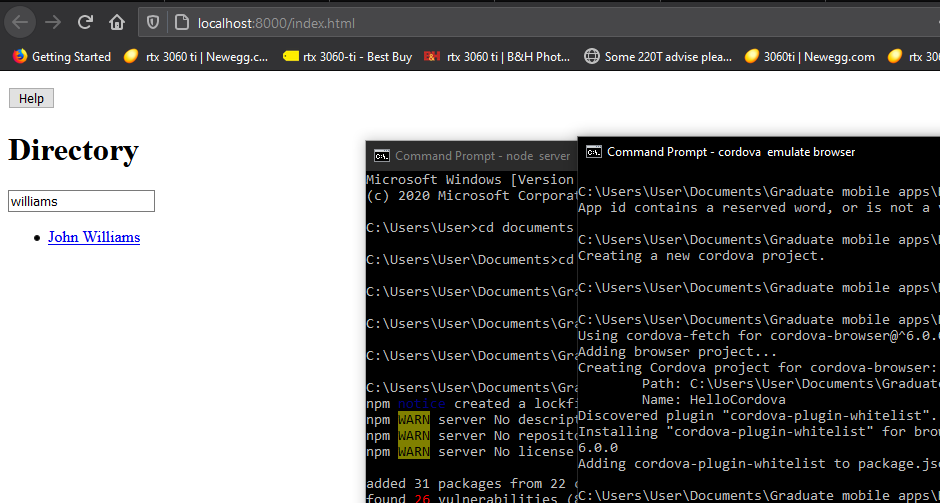


I have successfully replaced all the contents in the www directory

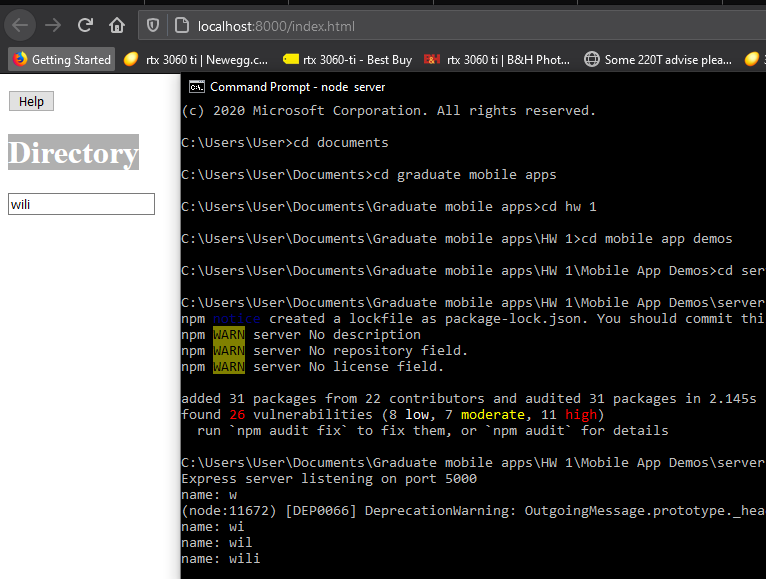


And here is the emulator.

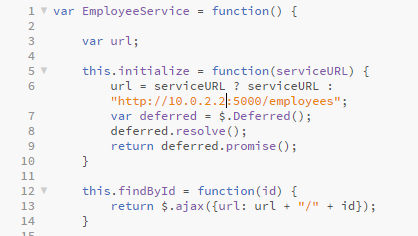
**Module 4**

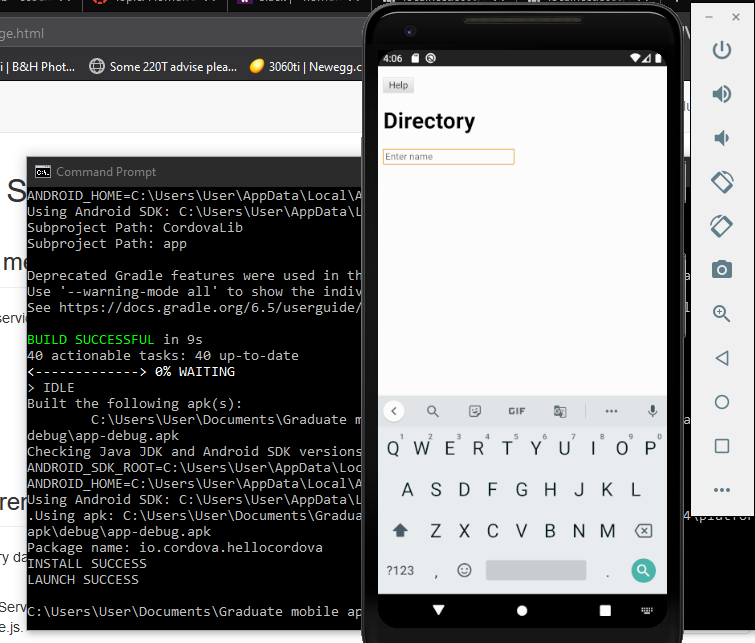


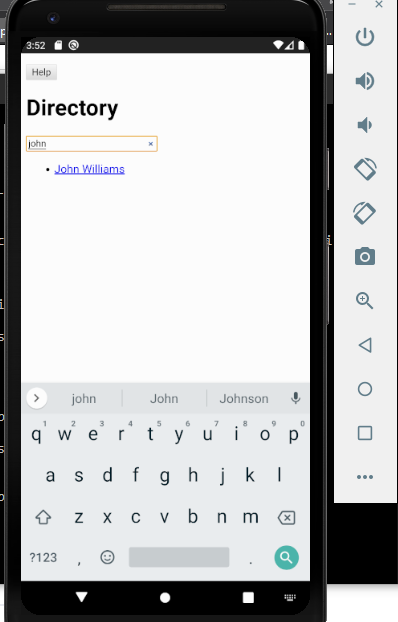
So far for module 4 I have followed the instructions and got the directory to search up an employees name.



I also followed the TA’s instructions and changed the index.html file in Brackets. I updated it to json so that now it displays the query.

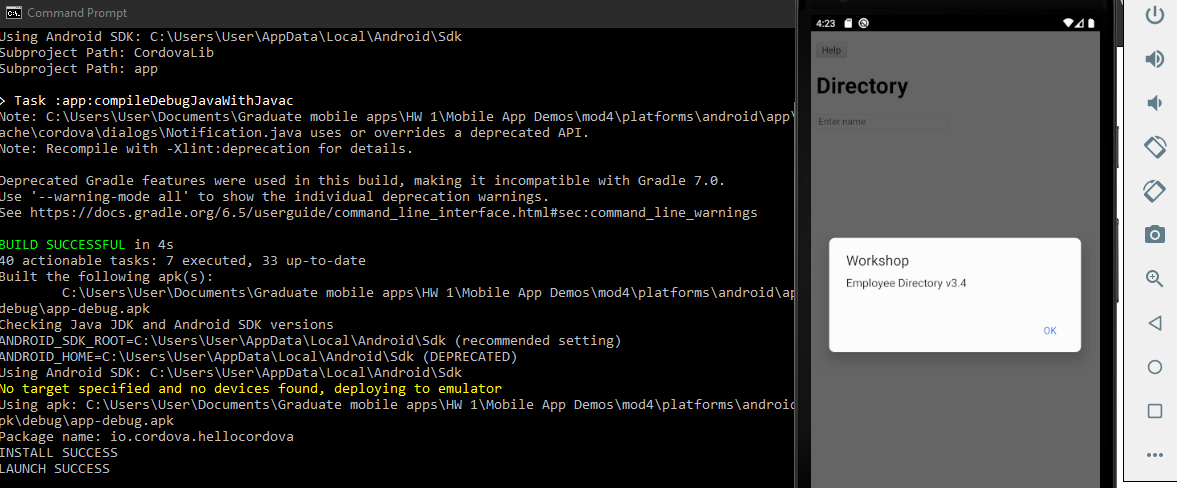


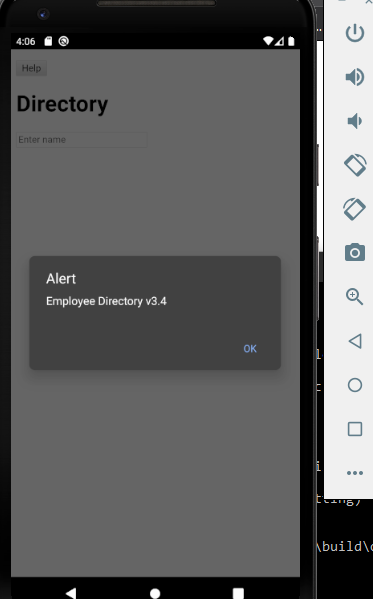




Following the TA’s instructions. I was able to successfully get it to launch on the emulator.

**Module 5**



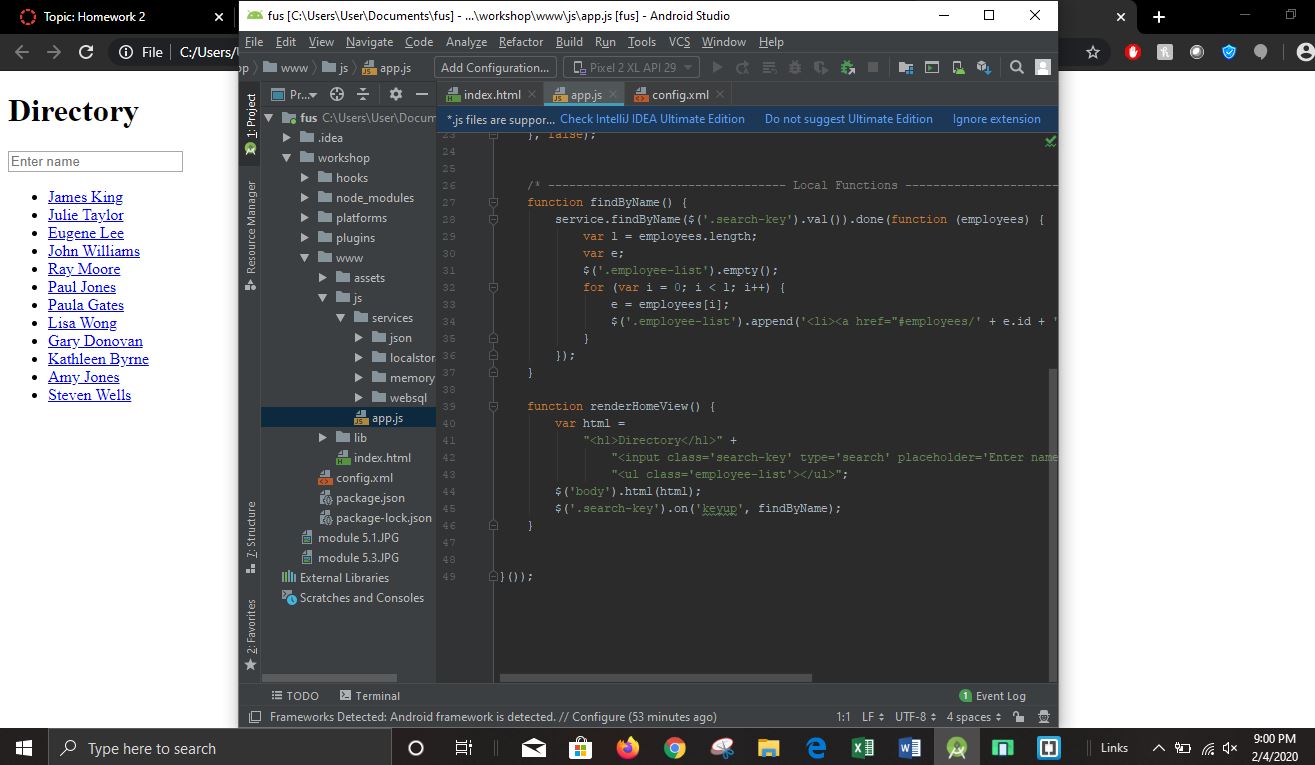


Followed the instructions and got the native notification to pop up on the android emulator.

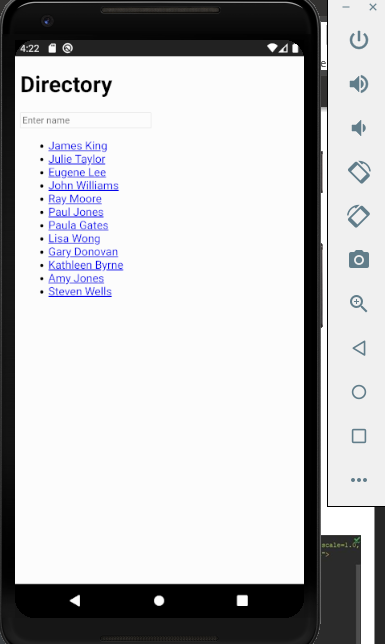
**Module 6**

(Module 6 was skipped since I am running this on android)

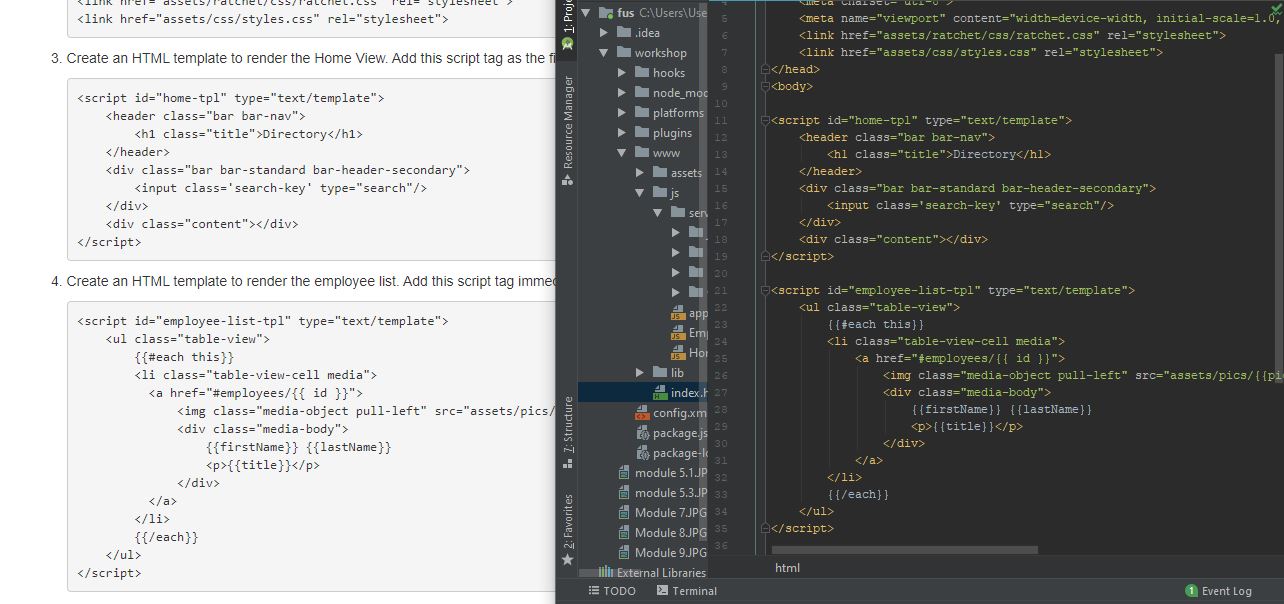
**Module 7**



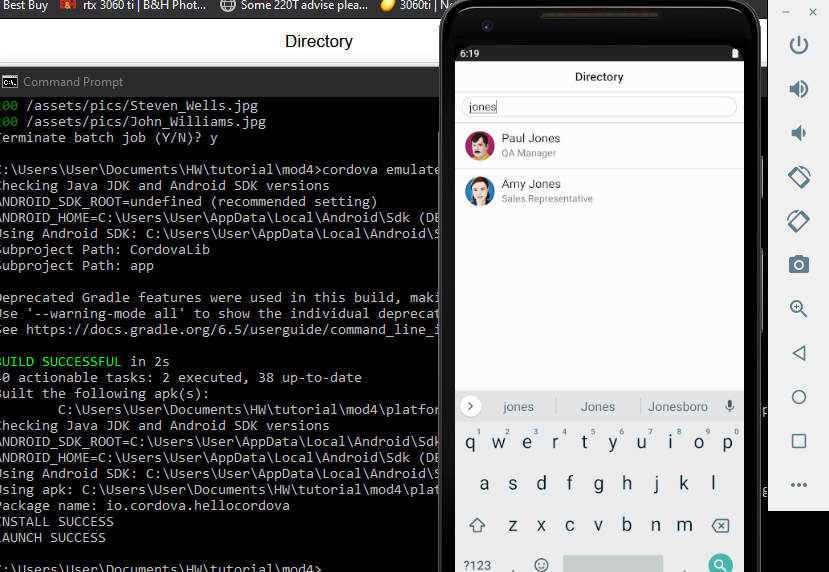
I followed the steps and added the code in via brackets and android studio. I ran it and displayed all the names on the home screen in the browser.



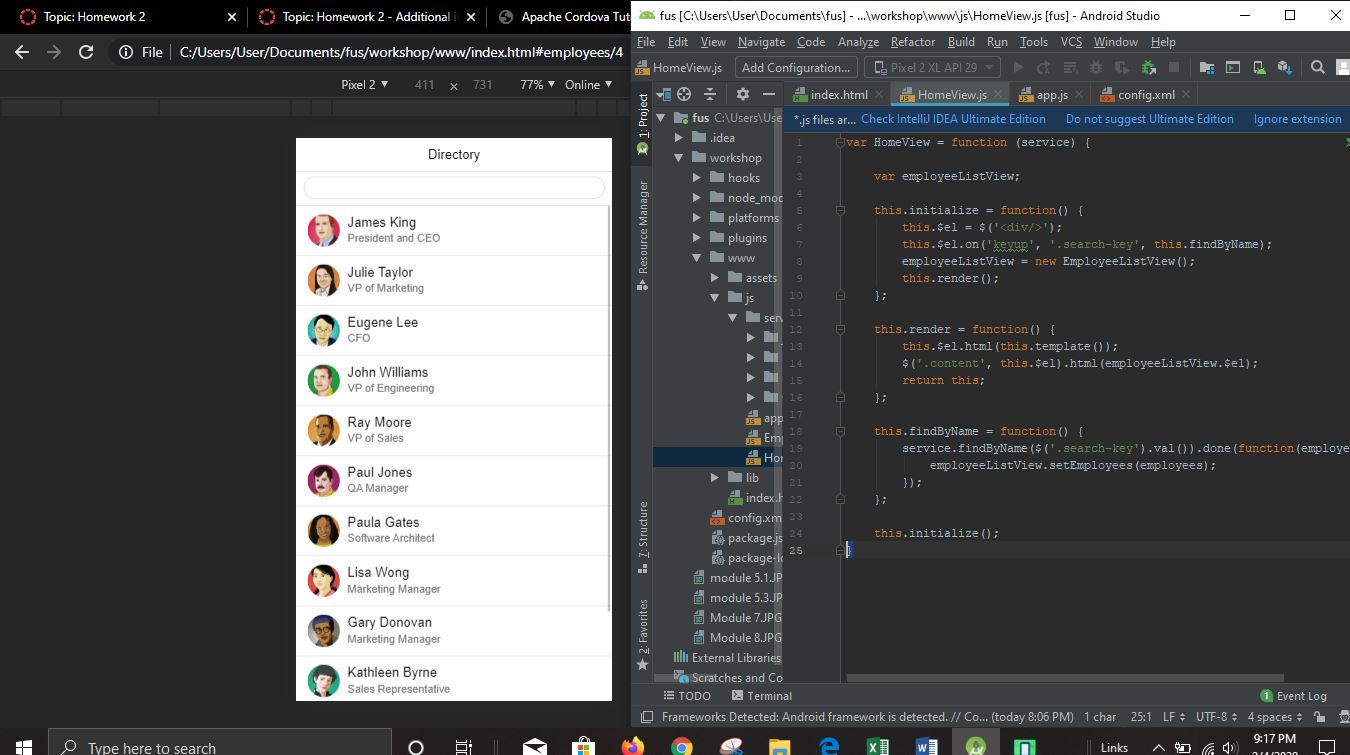
**Module 8**



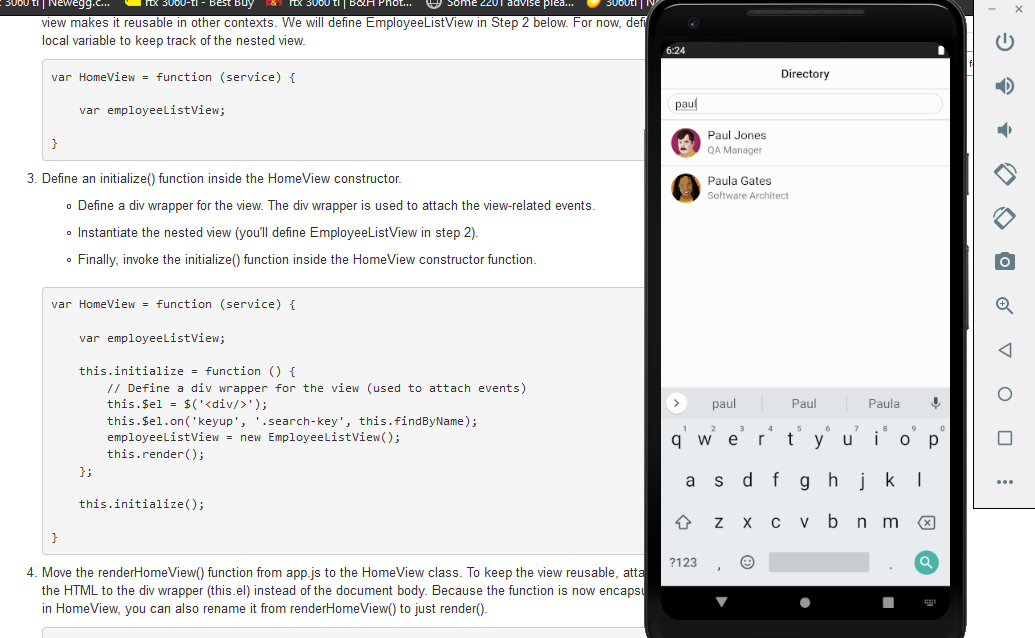
This template was used to display all the profiles of the employees while also displaying all their titles.



**Module 9**

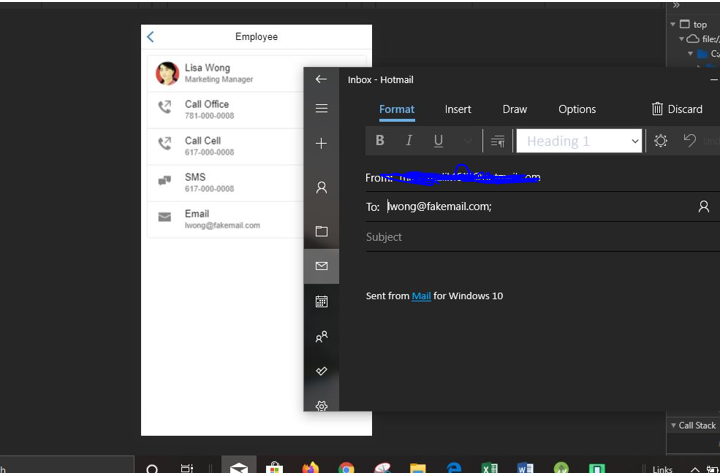


For this module, I used a div wrapper which is used to attach the view-related events. As shown in the photo I also created a new JS file called EmployeeListView. I had also started displaying the Web browser in mobile mode at this point.

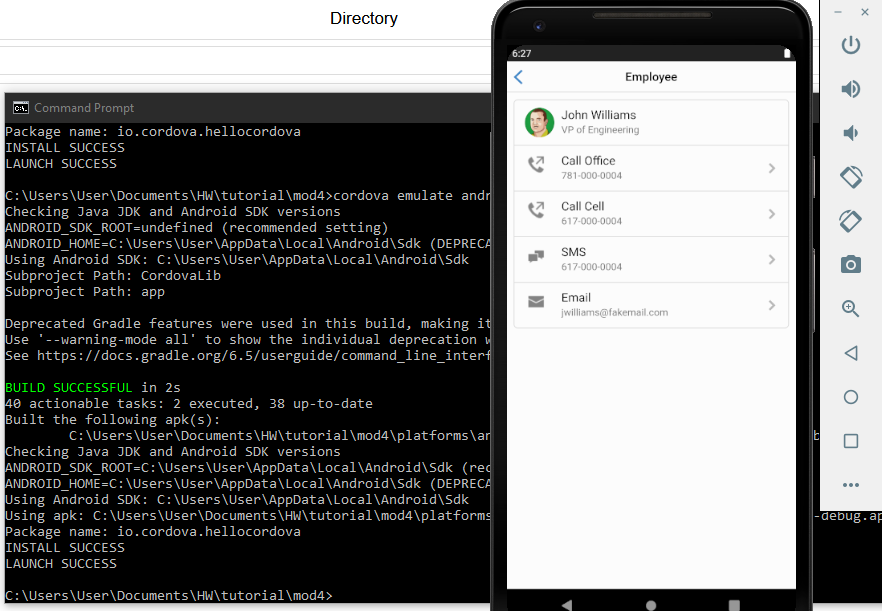


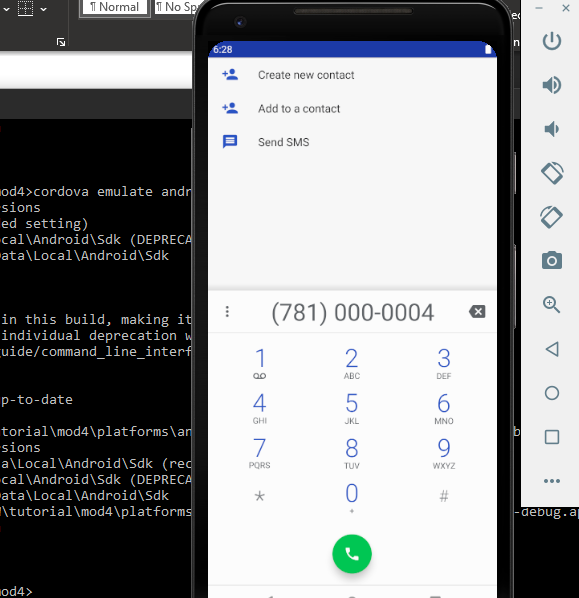
I then had it running on the android emulator through command prompt.

**Module 10**



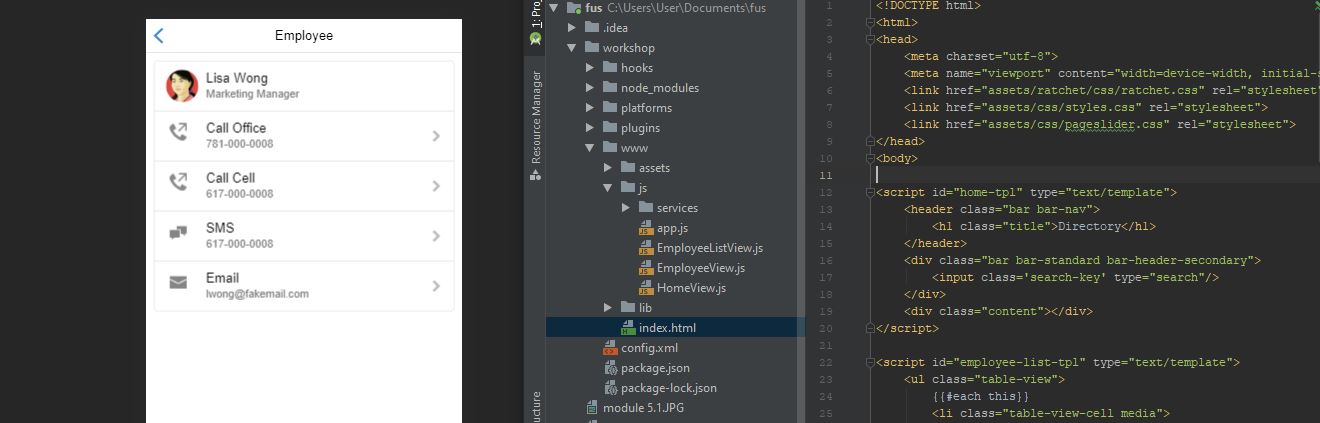
So Now, I started creating the view routes. I was able to create the contact information for all the employees. This information was backed up using the Employee View JS file which displays each employee’s personal information. As I tested out the email application it worked great and automatically opened up the built in Windows Email app when I clicked it.

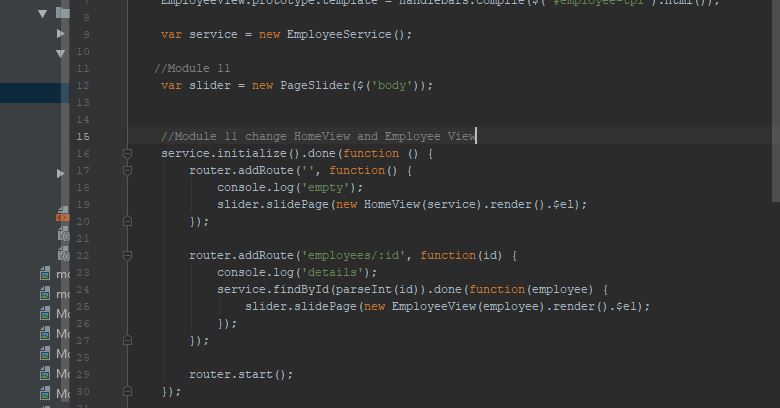




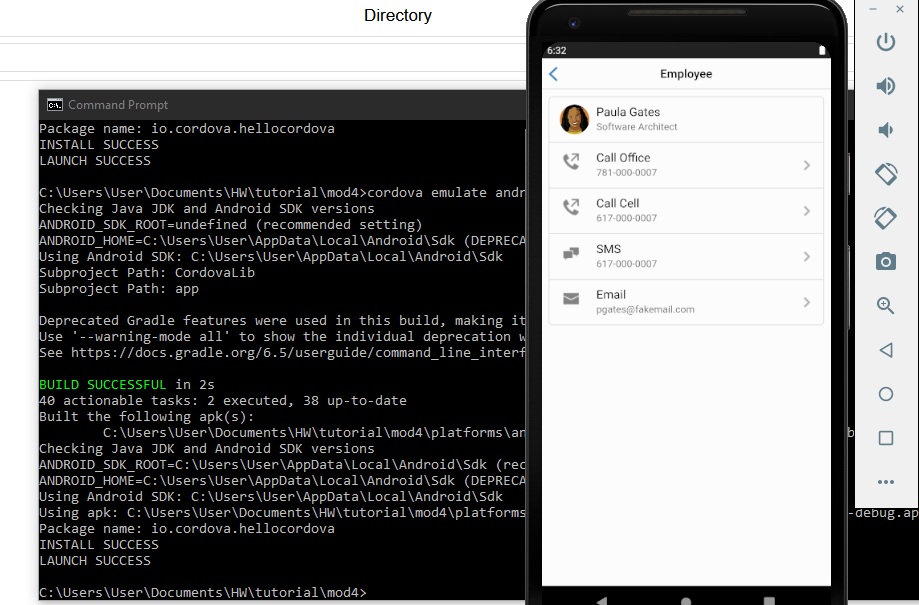
I then tested it on the android emulator. I have tested all the phone numbers, emails and text message buttons and they ALL worked. But the email required me to register a gmail since the emulator didn’t already have one. I’m not gonna bother because this is all test case and registering a gmail address for the emulator is pointless. But my test on the browser showed that it had the email already filled out.

**Module 11**



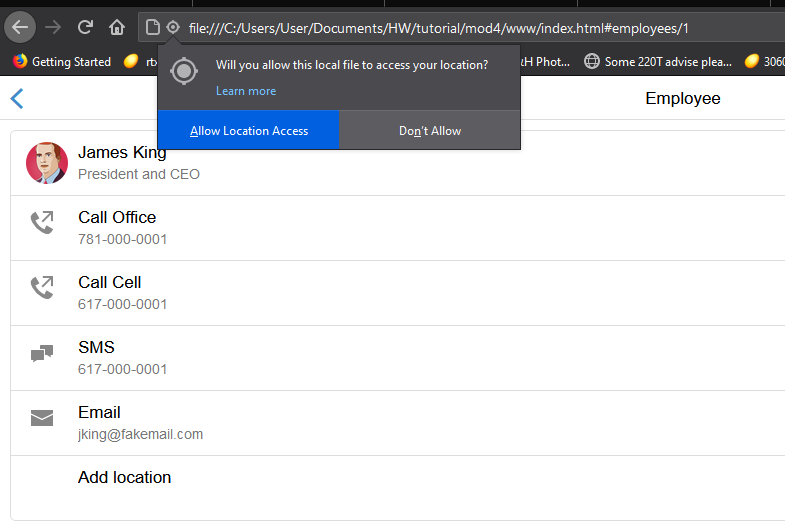


For me, this module was similar to adding bootstrap to a webpage in html. I was able to add the sliding animation to the Application.

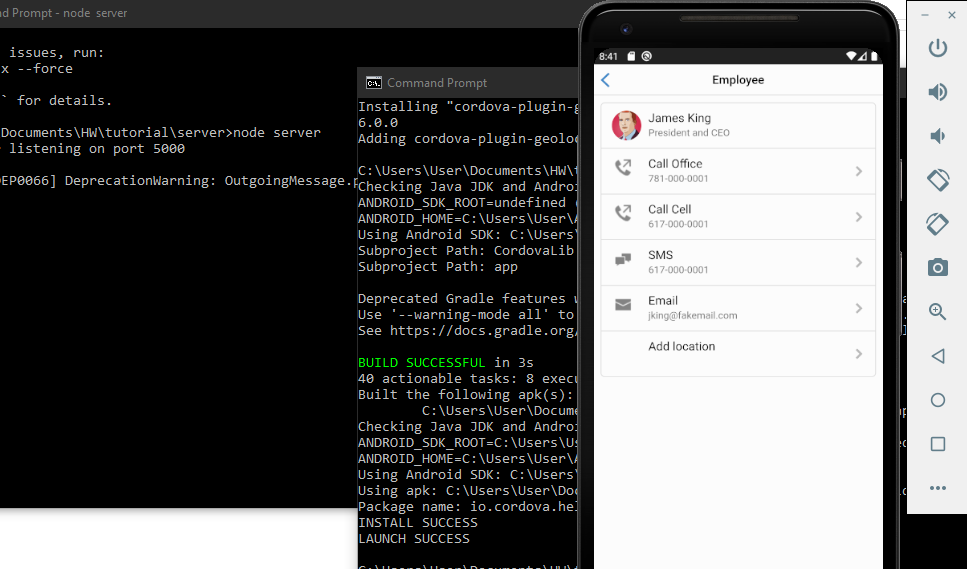


Since all this module did was add the sliding effect, there was no real way to show proof for it since I would need to record a video of it on the app. But here it is launched on the emulator.

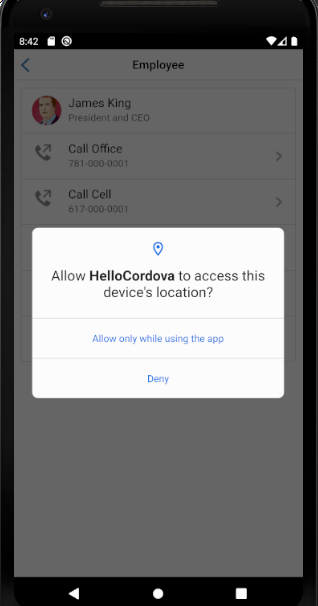
**Module 12**

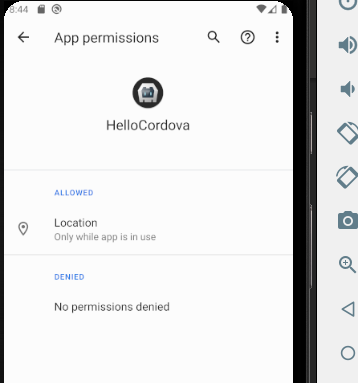


I followed the instructions given and first tested the application on the browser.

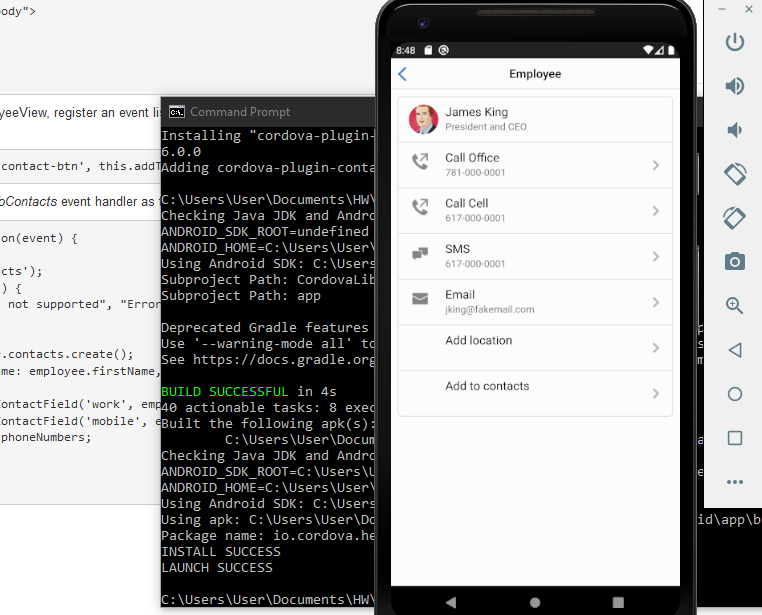


I was able to add the add location functionality to the app. I implemented it in Android studio but launched the app through commands.

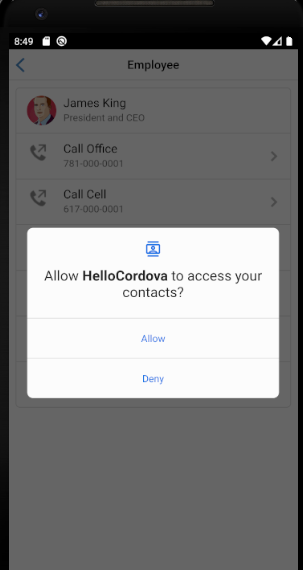
 here was the result after I pressed add location.

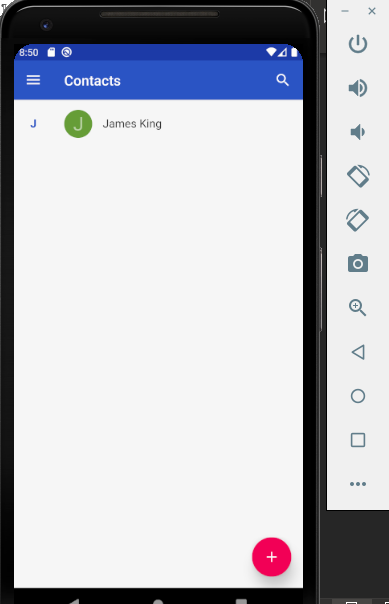
 I also took the liberty of going into the emulators settings to see if the Location permission had been granted and it had.

**Module 13**



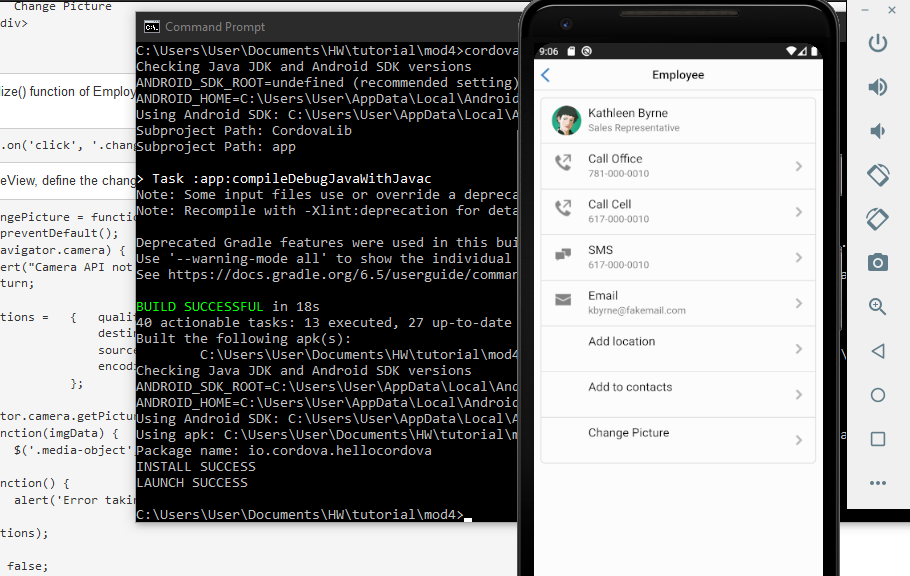
I followed the Exact same steps for Module 12 with Module 13 and easily go to this result.

The app then asked for permission to access contacts, which I allowed.

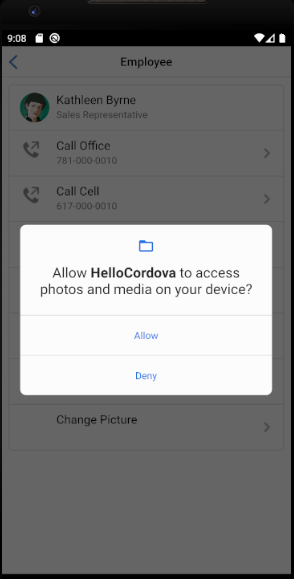
 And the Emulator has successfully added the Contact!

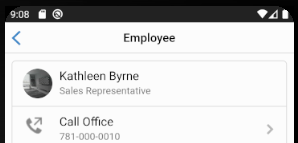
Unfortunately, I could not get the contacts to properly connect with the Android emulator’s contact list directory. I also ran into the same issue with **module 14** and the Camera API.

**Module 14**



Lastly, I repeated the steps from Module 12 and 13 for this since it was pretty much identical. I launched the emulator from the command prompt but this time I turned off the other command prompt that had the directory in the server.

 When I pressed Change picture, this is the popup

 I accessed the Emulator’s Camera and as you can see, Kathleen’s photo has been changed from what it originally was.

This HW assignment had me running tests in between browser and Android emulator but in the end, all Modules successfully ran on the Emulator. I even tested a few on my own personal device.