Native app- java-android

Objective c,cyst-ios

Hybrid app

Native app-high maintainence

HTML,CSS,JS-

Embedded browser in an app-webgloo

Ionic framework-cordova framework(hybrid) – building &

Develop an app in cloud – goal

Ionic pro in cloud

Ionic framework – Hello world –

**IONIC**

**IONIC -framework to mobile apps with one codebase.**

**HTML,CSS,JS**

**Ionic cloud-to push and deploy**

**Nodejs- proj to build cmd line tools,web service and js libraries**

**Ionic cli(command line interpretors tools) – used to start,test and run ionic apps**

**Ionic pro-serve as a remote origin-use git to track code..this doesnot have those features**

Day today changes -in git

Ionic services-git push ionic master

After building and testing an ionic app –

1. **Hooks – scripts run before and after building& testing the app.**
2. Node modules – place where nodejs/npm stores all js dependencies (ionic,angular2)
3. Platform-is where cordova generates platform specific code.ios – objective c
4. Plugin – where cordova stores native plugin code(eg:Bluetooth)
5. Resources – stores icon,splash screens
6. Src- where app code runs
7. www-is static output for ionic app.root of web server.
8. Edit.config -for better code staandards.
9. Git ignore – example node module which should not be stored in git. So ignored.
10. Config – config file for cordova.
11. Ionic.config-config for ionic.
12. Package.json – node dependencies.
13. Tsconfig-config for typescript..ionic provides default typescript.
14. Tslint-linters tell u about code which is not formatted(formatttig codE) -when working with large group helps u maintain same format.

**SRC**

1. App-runs initial code.its like a tree structure where there is a root at the top .
2. Assests- stores images,icons tat gets bundled during build process.
3. Pages –
4. Themes- to change color and all
5. Decoration.ts – external modules for typing
6. Index.html – root file which gets put into www.starting point for app in visual stand point which kicks off the app
7. Manifest.json – identifies app as standalone app -progressive webapp
8. Service worker.js – for progressive webapp.

Ionic – app which runs in a shell.

**IONIC PRO**

News feed -on initial download

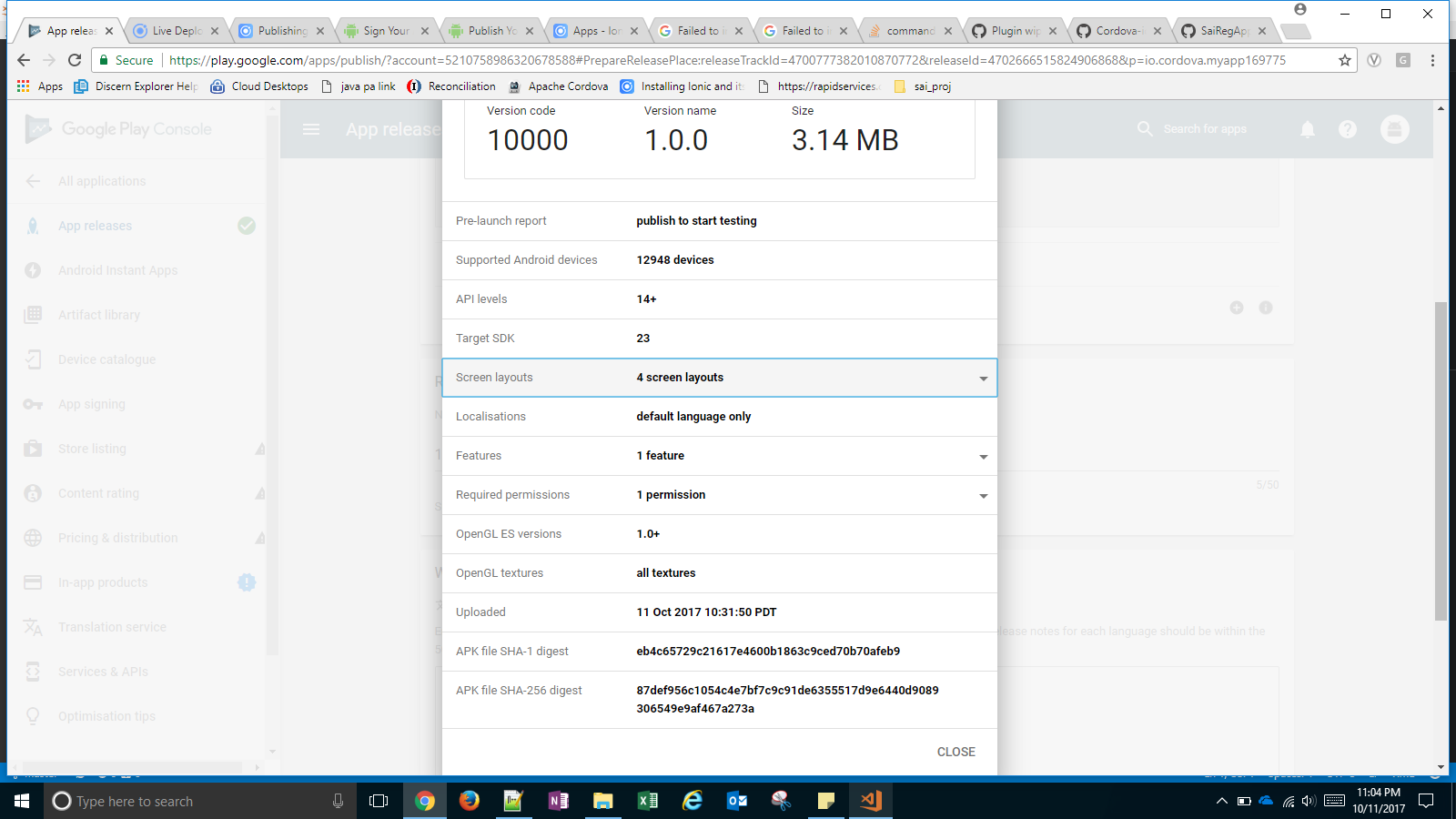
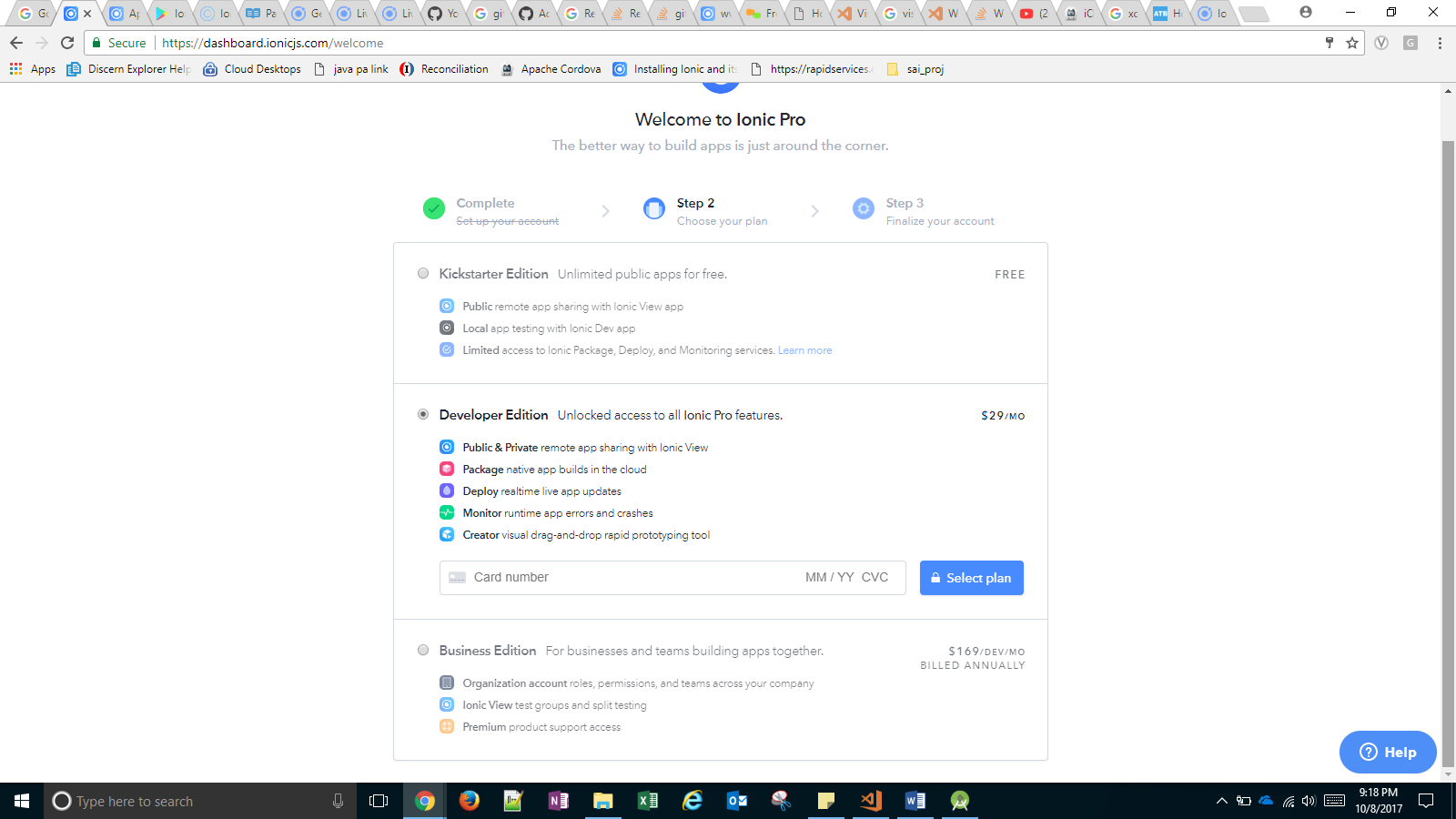
Link/news -based on samite

Extension(plugin)

Login – PHONE NUMBER,NAME- WE SEND A OTP AND ACCEPT – just for registration.

Want more – Samiti details

Publishing a



**STEPS:**

1. cordova plugin add cordova-plugin-ionic --save --variable APP\_ID**=**"YOUR\_APP\_ID" --variable CHANNEL\_NAME**=**"YOUR\_CHANNEL\_NAME" --variable UPDATE\_METHOD**=**"auto"
2. git
3. <http://ionicframework.com/docs/v1/guide/publishing.html>

* **Since we have a key, directly sign the app**
* **Upload as a release in play store**
* **Download form play store**

1. **New change(live update)**
   1. **Change in file**
   2. **Git status**
   3. **Git add -p**
   4. **Git push**
   5. **Git push ionic master**
   6. **Ionic serve ---lab -> to check changes**
   7. **Go to dashboard, wait till the build finishes.**
   8. **Click deploy for the recent build for the master**
   9. **Click Deploy**
   10. **Wait till process is complete.**
   11. **Open the app and the new change will be reflected**
   * **cordova plugin add cordova-plugin-ionic --save --variable APP\_ID="11352576" --variable CHANNEL\_NAME="Master" --variable UPDATE\_METHOD="auto"**
   * **jarsigner -verbose -sigalg SHA1withRSA -digestalg SHA1 -keystore my-release-key.keystore android-release-unsigned.apk alias\_name**
   * **./zipalign -v 4 android-release-unsigned.apk saireg\_1.apk**