**Luna Installation on a Mac (Creating a new ionic project)**

You can install the summer term Luna app, using the Luna ionic project created during that term, following the previous directions. These directions will create a new ionic project in a new directory. You may want to do this if there is an ionic upgrade, or you want to change the app name.

1. From the directory where you will create the Luna home directory, execute from a Mac terminal:

**npm install -g cordova ionic** (Installs ionic)

1. From this directory execute:

**ionic start Luna tabs**

Do not make directory Luna. This command makes the directory Luna You will be asked:

"Link this app to your Ionic Dashboard to use tools like Ionic View?"

Ionic sells development tools (Ionic View) that are not needed to build and maintain Luna. Answer no.

The app name this will be displayed on the iPhone or iPad under the icon is taken from line 3 of the file Config.xml in the created Luna directory:

<name>MyApp</name>

The default is MyApp. **Change it to Luna NOW**. Once the following steps are taken, ionic won't let you change it without creating errors.

1. Copy source files files for Luna

**pages directory:** Delete the created pages directory for the sample app, replace with the Luna pages directory from the directory: LunaShip/Luna\_app/src/pages

**app directory:** Ionic has two files that define the pages:

Luna/src/app/app.module.ts

Luna/src/app.component.ts

Replace these with the ones in LunaShip/Luna\_app/src/app

1. Replace the **theme file** in the theme directory: Luna/src/themevariables.scss

with the one from LunaShip/Luna\_app/src/theme

1. Run the following commands from a Mac terminal in the ionic project home directory (the Luna directory you created in step 2):

**ionic cordova plugin add cordova-plugin-http** You get the message:

The plugin @ionic/cli-plugin-cordova is not installed. Would you like to install it and continue? (Y/n) answer Y

1. Add local storage by running the following commands:**npm install @ionic/storage@2.0.0 --save --save-exact**

**npm uninstall --save @ionic/storage**

**npm install --save @ionic/storage**

1. Install momentJS for the Calendar by running the following command:

**npm install --save moment**

1. Install the md5 hashing protocol by running the following command:

**npm install ts-md5**

1. Create the ios support:

**cordova platform add ios**

1. Replace **app icons**, copy from:

LunaShip/Luna/resources/ios/icon directory

to the created Luna icon directory

Note: Creating the ionic app icons is explained [here](http://blog.ionic.io/automating-icons-and-splash-screens/).

1. Install cordova HTTP support:

**ionic cordova plugin add cordova-plugin-http**

1. Copy **node\_modules** directories from: LunaShip/Luna/node\_modules/@ionic-native/:

http

keyboard

sql-lite

**Building and Executing Luna**

**On the Xcode simulator:**

The three commands to execute from the terminal in the Luna home directory, that build, then executes Xcode to test:

1. **ionic serve**

brings up a browser window with the Luna login screen. If syntax errors, they are displayed.

ctl-c in the terminal to enter next command.

1. **cordova build ios DO NOT HAVE DEVICE PLUGGED INTO MAC**
2. **open platforms/ios/Luna.xcodeproj**

Commmand 3 starts Xcode with the Luna Xcode project. Choose the simulator to run (upper left by Luna name, pull down list). Then press

run icon (to its left, left "arrow")

"Build Succeeded" window appears, simulator starts, displays Luna login page.

**These three commands are run whenever you modify the app code.**

**On a device (iPhone, iPad)**

1. Plug the device into the Mac. Its name should appear in Xcode device window (click pulldown to right of Luna icon). Click it.
2. Press Xcode start build icon.You should see "build succeeded" window, and request to update key chain.
3. On the device, from the settings app, press device management, and allow this app developer access. The Luna icon should appear ona screen.

**Note:**

We got "signing" errors, and "provisioning" errors when we first tested devices from Xcode. To eliminate errors like "Failed to create provisioning profile", or signing errors, we had to:

Show the project navigator on the left pane: view-->navigators-->show project navigator Click on Luna (top in navigator pane)

in identity: replace bundle name with something else (qwert)

in signing: click automatically manage signing

Team: should display your Apple developer ID

Your Apple developer ID can be displayed: Xcode-->Preferences-->accounts