**Luna Installation on a Mac**

These directions install the app, created during the summer term 2017. A new Luna ionic project can be created. You may want to do this if there is a major ionic upgrade, or to change the name of the product (Luna) that appears below the icon on the device. See those instructions.

1. Execute from a Mac terminal:

**sudo npm install -g cordova ionic**

This installs ionic. "sudo" runs the command as the system administrator.

1. Copy the Luna\_app directory from the LunaShip directory to where you want to install the Luna appproject. The Luna\_app directory is the home directory to execute the commands to build and execute the

Luna app using Xcode to test the app on its simulator, or an iPhone, or iPad device connected to your mac.

1. Run the following commands from a Mac terminal in the ionic project home directory (the Luna directoryyou created in step 2): **sudo 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 app local storage support by running the following commands:

**sudo npm install @ionic/storage@2.0.0 --save --save-exact**

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

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

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

**sudo npm install --save moment**

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

**sudo npm install ts-md5**

1. Create the ios support:

**sudo cordova platform add ios**

1. Install cordova HTTP support:

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

**Building and Executing Luna**

**Using the Xcode simulator as the device:**

The three commands to execute from the terminal in the Luna home directory (luna\_app), 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 A DEVICE PLUGGED INTO THE 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.**

**Using Xcode to install the app on a device (iPhone, iPad)**

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

The Luna icon should appear on a 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 do on Xcode:

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 (the string "paulp" works)

in signing: click automatically manage signing

Team: should display your Apple developer ID

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

Apple developer IDs are free, and you must have one to develop an app for Apple devices.

See these [instructions](http://www.cs.uky.edu/~paulp/luna/appledeveloper.html) on becoming an Apple developer.