

Apple Build

01/10/2024 1:50 pm EST

Log into "mac mini" as "bold mac mini". I was only able to use "VNC Viewer" to connect. MRemoteNG would not connect for me.

Close Xcode.

Open Terminal.

```
cd "/Volumes/Macintosh HD/Applications/BoldNetMobile"
```

Use git to retrieve your branch from DevOps. You may have to setup SSH keys in order to access DevOps from the mac. <https://docs.microsoft.com/en-us/azure/devops/repos/git/use-ssh-keys-to-authenticate?view=azure-devops>.

Update cordova - npm install -g cordova.

To build BoldNet Mobile run `sudo ./BuildBoldNetMobile.sh release` (for Legacy Mobile run `sudo ./BuildLegacySecurityMobile.sh release`).

This will copy the appropriate files in place to build BoldNet Mobile.

Clean up from the last cordova build. NOTE: The clean may fail. This is typically OK. Building in Xcode creates other directories that cordova clean won't clean up because it didn't create the files.

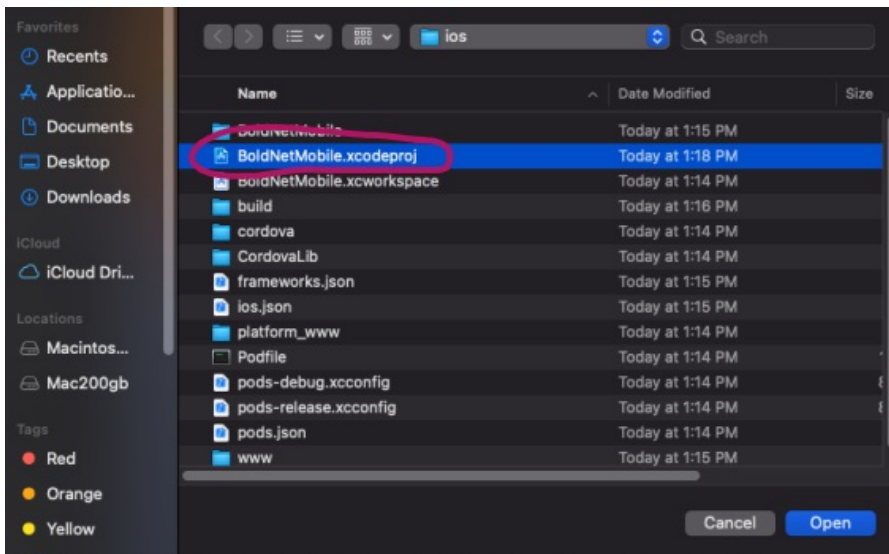
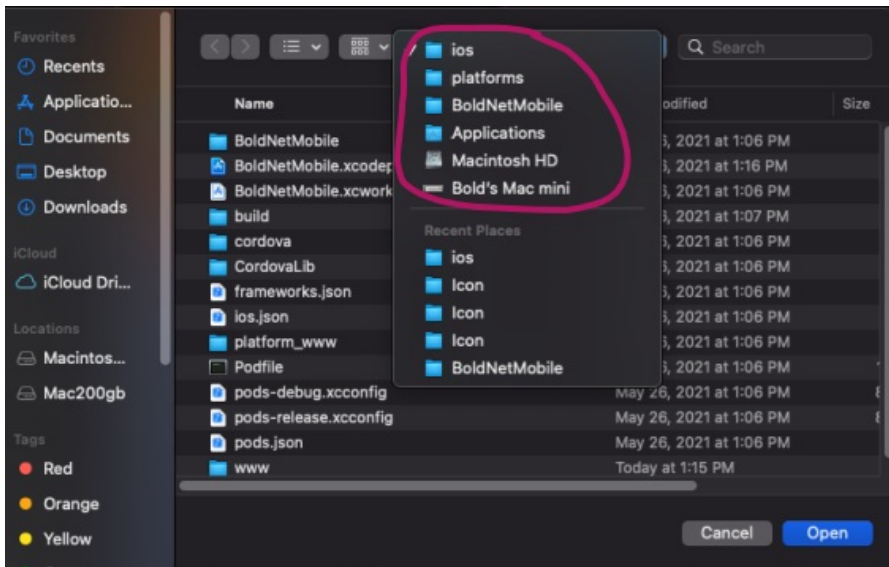
Remove the cordova ios platform.

Add the cordova ios platform. This reinstalls the plugins found in the config files.

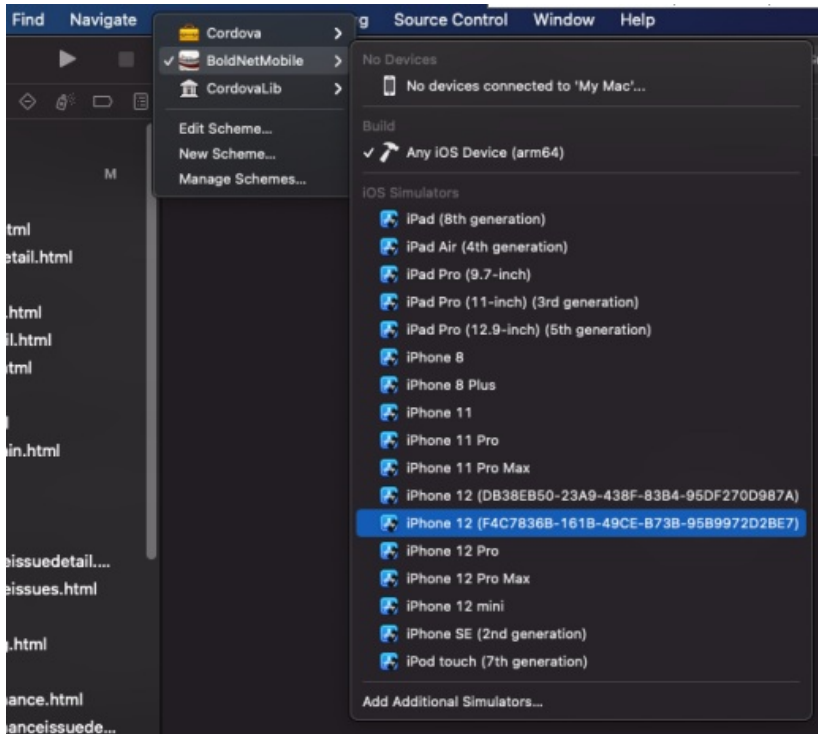
Build a release build of the ios platform.

You will now have an app you can load in Xcode.

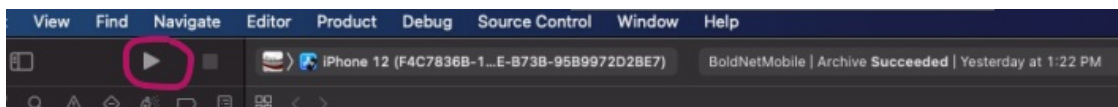
To test and get an app into the app store, start Xcode. Load the BoldNet Mobile application.



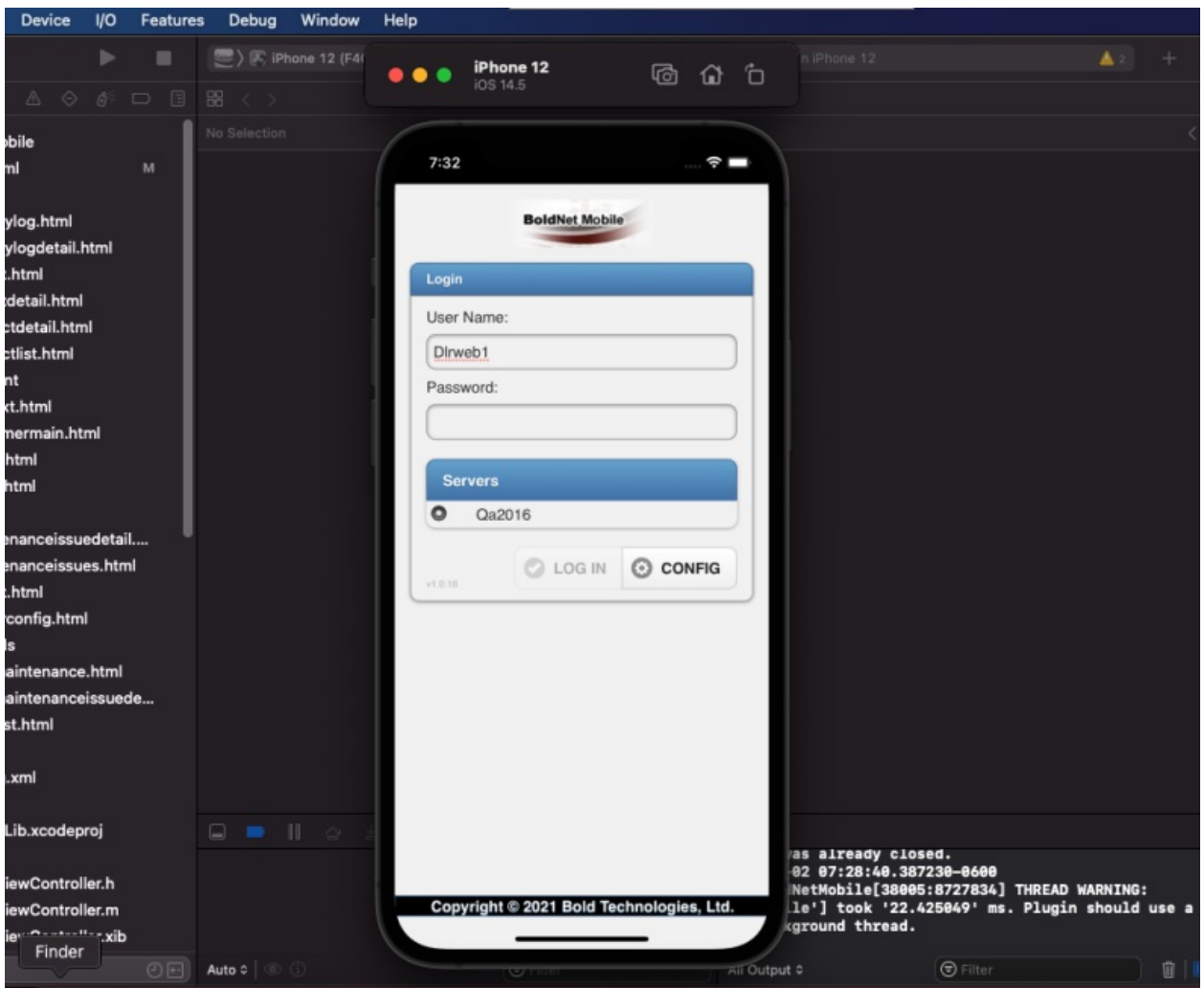
To run in a simulator, click on the application and pick the device you want to run on.



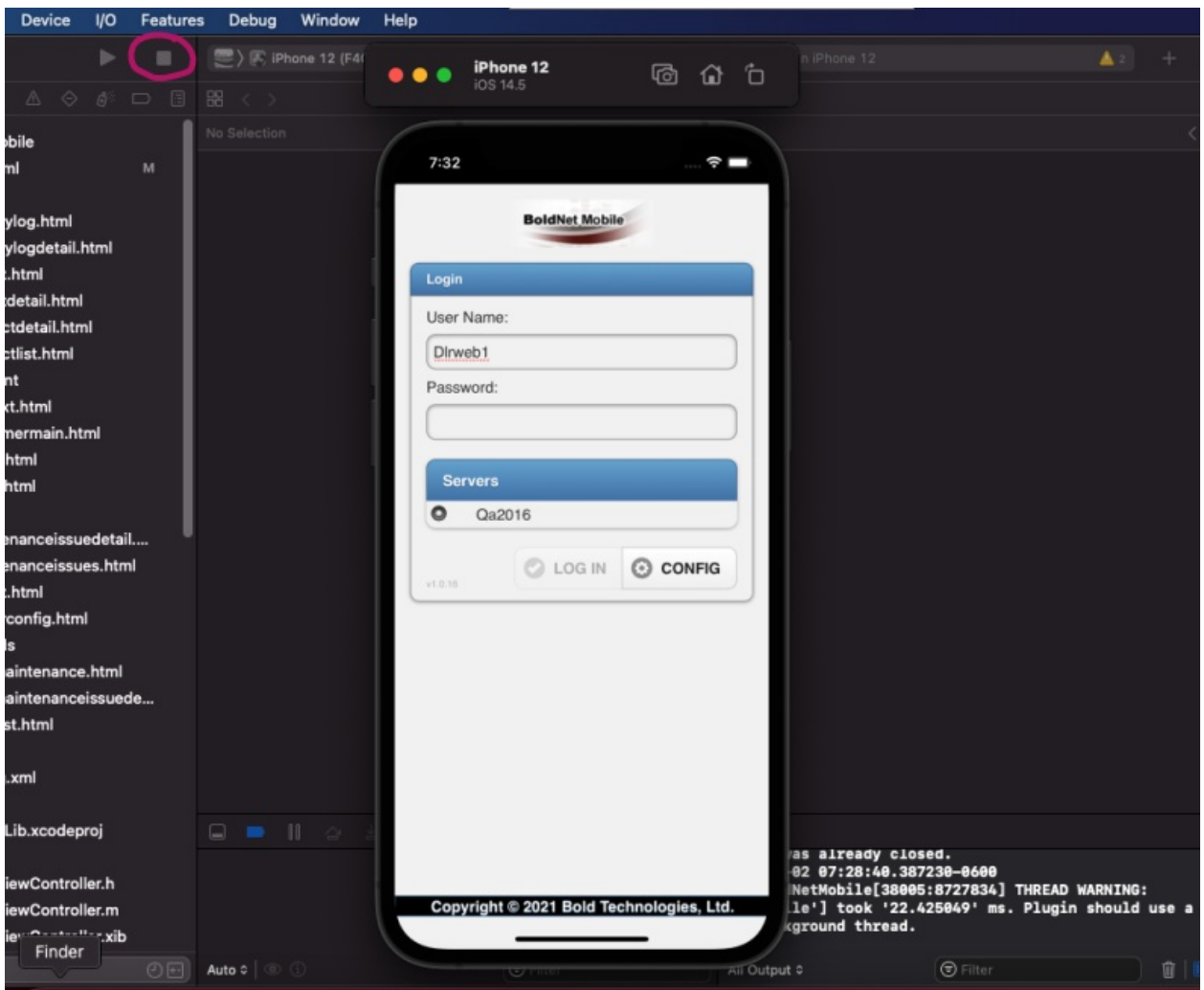
Click on the run button.



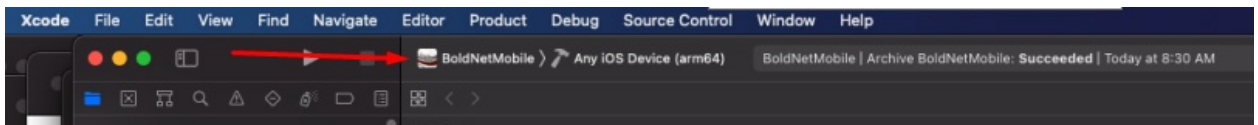
Important! Running in the simulator for a device for the first time takes about an hour for it to spin up. Subsequent runs on the same device only takes minutes.



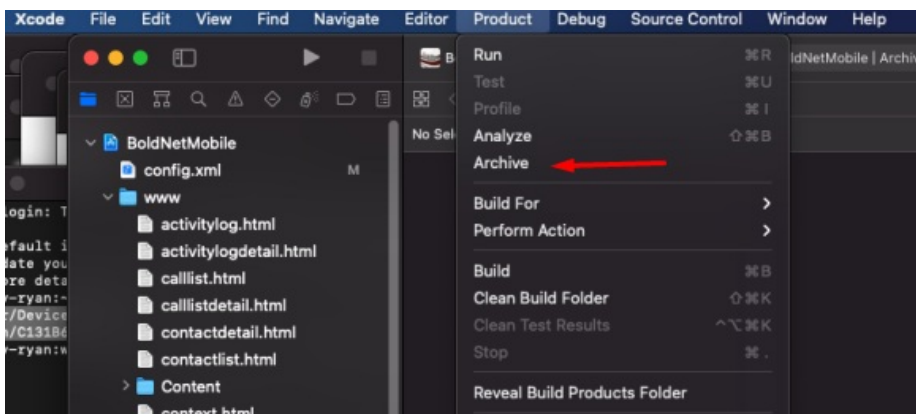
Click the stop button to stop debugging.



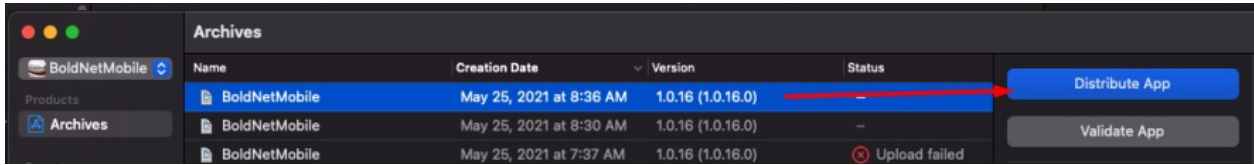
To build and release to the app store, make sure you select the app and choose device "Any iOS Device (arm64)".



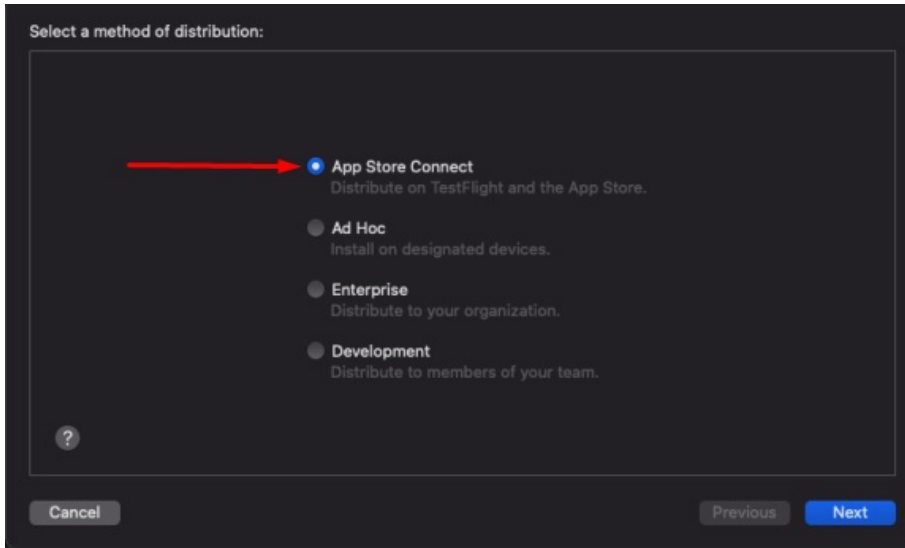
Click the Product menu and click Archive.



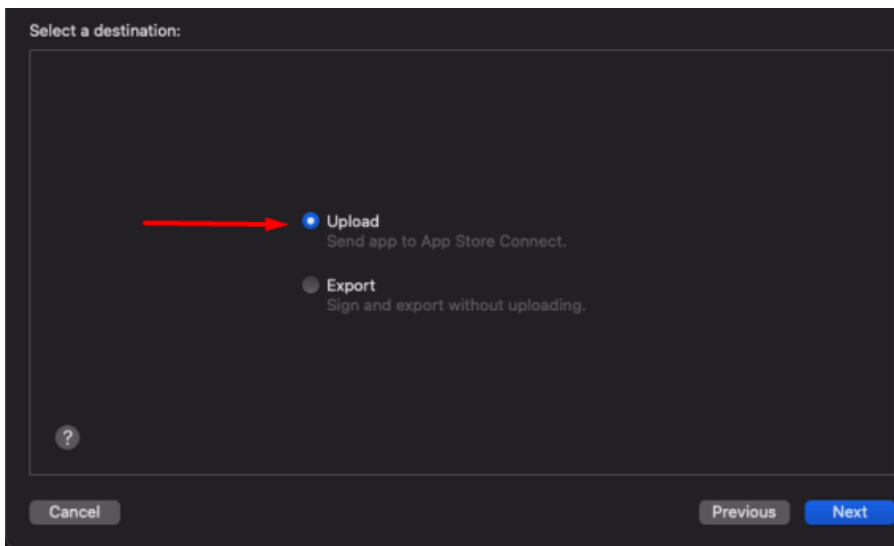
When it's done building, you will see a list of your builds. You can then distribute the app.



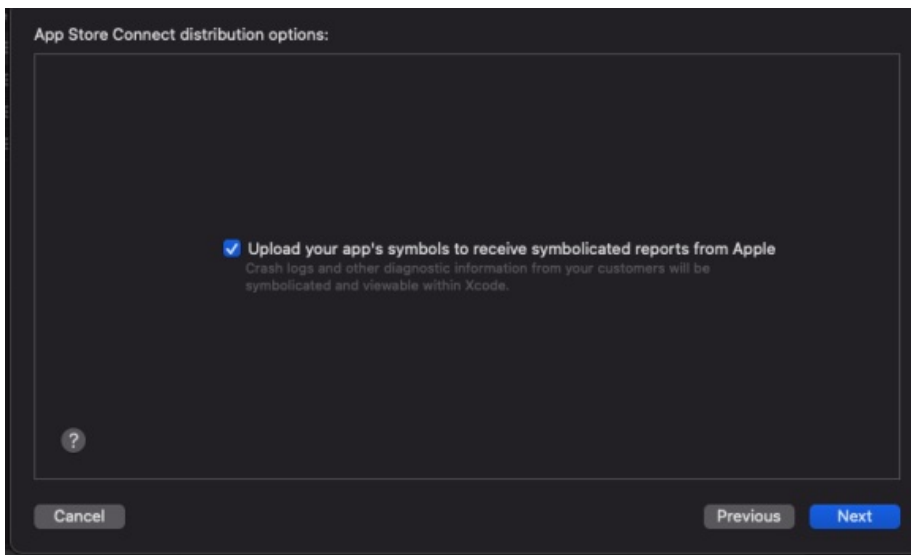
Once you've clicked on the "Distribute App" button, select "App Store Connect" and click Next.



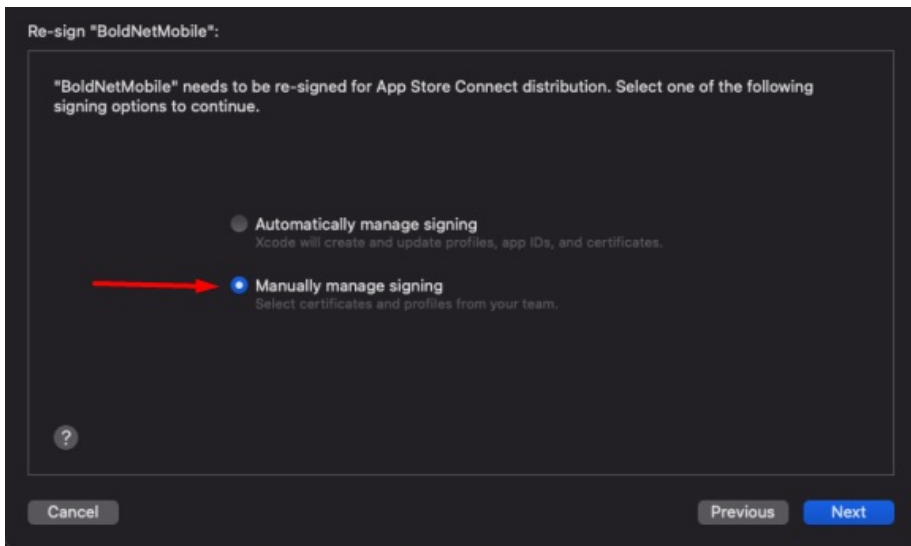
Select "Upload". Click Next.



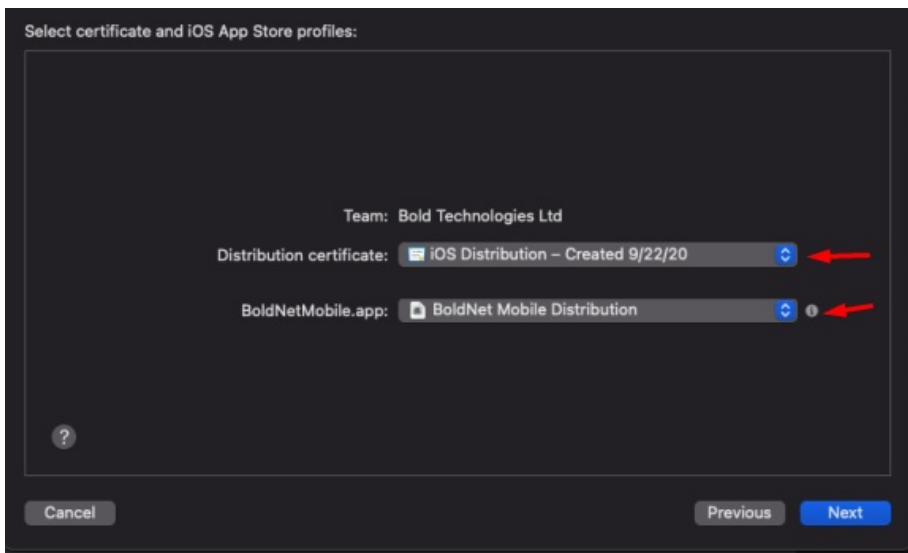
Check "Upload your app's symbols to receive symbolicated reports from Apple". Click Next.



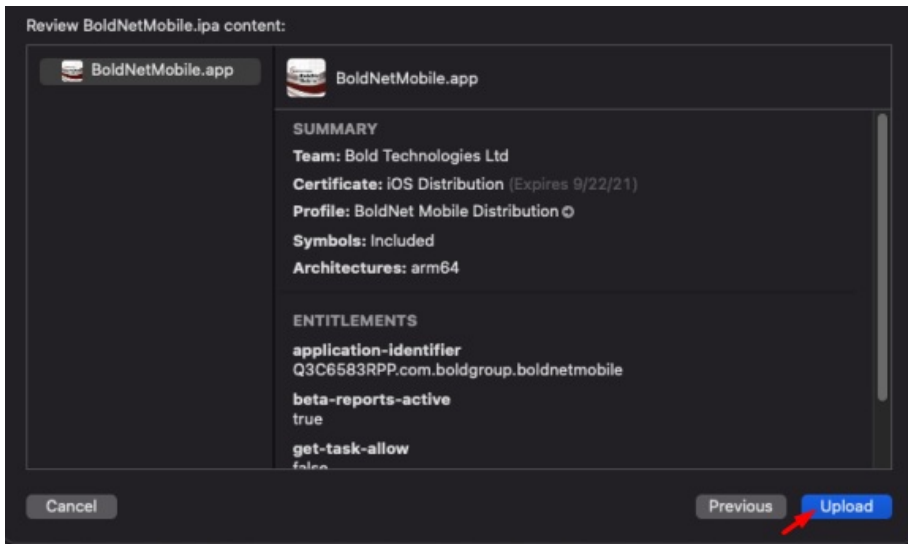
Click on "Manually manage signing". Click Next.



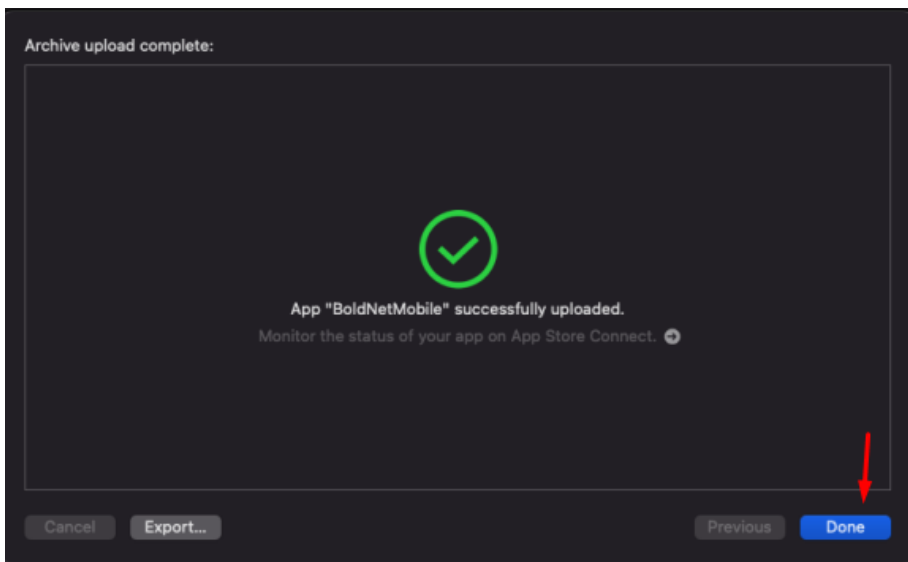
Pick the certificate and provisioning profile associated with the app you're publishing. Since Apple's certificates and provisioning profiles expire, they might not be named the same as the picture below.



If the information on the next screen is correct, press Upload.



Apple does a check of the app and will display a screen with any errors. Apple is notorious for changing the number of images and the dimensions of the images it requires. If the upload was successful, you'll see a screen like this.



Click Done.

Next you will need to go into the app store and set it up in testflight.