

CompteEstBon

This project is a simple singleView app providing a way to solve quickly *The total is right*

The project is configured to be tested on jailbroken iDevices with AppSync.

To get it working

NB : if you are a registered iOS developer, see the last paragraph

To get it working on your Jailbroken device, follow the steps below (for Xcode >=6 and iOS >= 7):

- Install AppSync Unified on device from cydia.angelxwind.net repo
- Open

/Applications/Xcode.app/Contents/Developer/Platforms/iPhoneOS.platform/Developer/SDKs/iPhoneOS.sdk/SDKSettings.plist

- Change value of parameter **AD_HOC_CODE_SIGNING_ALLOWED** to **YES**
- Change value of parameter **CODE_SIGNING_REQUIRED** to **NO**
- Set “Code Signing Identity” parameter in “Build settings” (Both “Target” and “Project”) to “Ad Hoc Code Sign”
- Add new “Property List” file to your project with name “Entitlements.plist”
- Add “Can be debugged” boolean parameter with value “YES” to Entitlements.plist
- Add “application-identifier” string parameter with value key copied from “Bundle identifier” in target’s “Info” tab.
- Set “Code Signing Entitlements” parameter in “Build settings” (Both “Target” and “Project”) to “Entitlements.plist”

Now you can use your jailbroken device as a testing device !

If you want it to work on any jailbroken iPhone :

- Once it’s compiled, right-click on *Products>CompteEstBon.app* in the project hierarchy and choose “Show in Finder”
- Copy the .app anywhere

- set permission of the executable *CompteEstBon.app/CompteEstBon* to 755 :

```
$ chmod 755 CompteEstBon.app/CompteEstBon
```

- ldid sign the executable :

```
$ ldid -S CompteEstBon.app/CompteEstBon
```

- ssh it to your device and respring :

```
iphone$ killall SpringBoard
```

or

```
iphone$ uicache
```

If you are a registered iPhone Developer :

- Set “Code Signing Identity” parameter in “Build settings” (Both “Target” and “Project”) to “iOS Developer”
- Make sure a matching codesigning identity is set in “General”, section “Identity” (Both “Target” and “Project”)

The scripts used are using the following tools. Make sure to install them beforehand

NB : scripts are using terminal-notifier to notify the completion of pushing. See <https://github.com/julienXX/terminal-notifier> and install by using :

```
$ brew install terminal-notifier
```

NB : The conversion from .md to .pdf needs pandoc, which depends on a Latex distribution. To install pandoc :

```
$ brew install pandoc
```