

Oracle Banking Digital Experience

Mobile Application Builder Guide – iOS
Release 19.2.0.0.0

Part No. F25153-01

December 2019

ORACLE®

Mobile Application Builder Guide – iOS
December 2019

Oracle Financial Services Software Limited
Oracle Park
Off Western Express Highway
Goregaon (East)
Mumbai, Maharashtra 400 063
India

Worldwide Inquiries:

Phone: +91 22 6718 3000

Fax: +91 22 6718 3001

www.oracle.com/financialservices/

Copyright © 2019, Oracle and/or its affiliates. All rights reserved.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are “commercial computer software” pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate failsafe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

This software or hardware and documentation may provide access to or information on content, products and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Table of Contents

1. Preface.....	4
1.1 Intended Audience	4
1.2 Documentation Accessibility	4
1.3 Access to Oracle Support	4
1.4 Structure.....	4
1.5 Related Information Sources.....	4
2. OBDX Servicing Application	5
2.1 Pre requisite	5
2.2 Create Project	5
2.3 Adding UI to workspace	5
2.4 Open project in Xcode.....	5
2.5 Generating Certificates for Development, Production and Push Notifications	8
3. Archive and Export.....	12
4. OBDX Authenticator Application	15
4.1 Authenticator UI (Follow any one step below)	15
4.2 Authenticator Application Workspace Setup	18
4.3 Building Authenticator Application.....	21

1. Preface

1.1 Intended Audience

This document is intended for the following audience:

- Customers
- Partners

1.2 Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=accandid=docacc>.

1.3 Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit

<http://www.oracle.com/pls/topic/lookup?ctx=accandid=info> or visit

<http://www.oracle.com/pls/topic/lookup?ctx=accandid=trs> if you are hearing impaired.

1.4 Structure

This manual is organized into the following categories:

Preface gives information on the intended audience. It also describes the overall structure of the User Manual.

The subsequent chapters describes following details:

- Configuration / Installation.

1.5 Related Information Sources

For more information on Oracle Banking Digital Experience Release 19.2.0.0.0, refer to the following documents:

- Oracle Banking Digital Experience Licensing Guide

2. OBDX Servicing Application

2.1 Pre requisite

- Download and Install node js as it is required to run npm and cordova commands.
- XCode to be download from Mac App Store.
- OBDX iOS App is supported only on versions 12 and 13.

2.2 Create Project

Ensure **Nodejs Version is >= 12 and Xcode version is >= 11.2**

1. Extract iOS workspace from installer and place in a folder.
2. The workspace by default contains framework for running on devices. Hence to run the application on simulator, delete and copy the 4 frameworks (OBDXExtensions.framework, OBDXFramework.framework, OBDXWatchFramework.framework, Cordova.framework) from installer/simulator to zigbank\platforms\ios directory.

2.3 Adding UI to workspace

Use any 1 option below

- a. Building un built UI (required in case of customizations)

Refer to User Interface Guide and then create a copy of index.html in the same folder and rename it to home.html.

Note: When copying to www, index.html already present in the workspace should be replaced)

- b. Using built UI (out of box shipped with installer which is same as built UI for internet) Hence take UI from main installer -

Copy folders/files from **OBDX_Installer/installables/ui/deploy** (build.fingerprint.js,build.txt,buildnum,components,cordovaScriptProperties.json,css-builder.js,css.js,extensions,flows,framework,images,index.html,jsdocs,lzn,manifest.json,metadata,normalize.js,oauthredirect.html,ojL10n.js,partials,resources,sw.3a7a a32d5c496a160ee6.js,text.js) to workspace (platforms/ios/www/)

Ensure webhelp folder is not copied.

2.4 Open project in Xcode

Open Xcode by clicking ZigBank.xcodeproj at zigbank/platforms/ios/

1. Adding URLs to app.plist (ZigBank/Resources)
 - a. FOR NONOAM (DB Authenticator setup)

SERVER_TYPE	NONOAM
KEY_SERVER_URL	https://mumaa012.in.oracle.com:18443/
WEB_URL	https://mumaa012.in.oracle.com:18443/

- b. OAM Setup (Refer to installer pre requisite documents for OAuth configurations)

SERVER_TYPE	OAUTH3
KEY_SERVER_URL	Eg. http://whf00bpp.in.oracle.com:17778
WEB_URL	Eg. http://whf00bpp.in.oracle:17777
KEY_OAM_URL	http://whf00ebe.in.oracle.com:15100/oauth2/rest/to ken
REDIRECT_URI	zigbank://oauthredirect
APP_CLIENT_ID	<Base64 of clientid:secret> of Mobile App client
APP_DOMAIN	OBDXMobileAppDomain
EXTENSION_CLIENT_I D	<Base64 of clientid:secret> of extensions
EXTENSION_DOMAIN	OBDXSiriDomain
WATCH_CLIENT_ID	<Base64 of clientid:secret> of wearables
WATCH_DOMAIN	OBDXWearDomain
SNAPSHOT_CLIENT_I D	<Base64 of clientid:secret> of snapshot
SNAPSHOT_DOMAIN	OBDXSnapshotDomain
LOGIN_SCOPE	OBDXMobileAppResServer.OBDXLoginScope
OFFLINE_SCOPE	OBDXMobileAppResServer.ValidateDeviceScope

c. IDCS Setup

SERVER_TYPE	IDCS
KEY_SERVER_URL	Eg. https://mumaa012.in.oracle.com:18443/ (This URL must be of OHS without webgate)
WEB_URL	Eg. https://mumaa012.in.oracle.com:18443/
KEY_OAUTH_PROVIDE R_URL	http://obdx- tenant01.identity.c9dev0.oc9qadev.com/oaut h2/v1/token
APP_CLIENT_ID	<Base64 of clientid:secret> of Mobile App client
WATCH_CLIENT_ID	<Base64 of clientid:secret> of wearables

SNAPSHOT_CLIENT_ID	<Base64 of clientid:secret> of snapshot
LOGIN_SCOPE	obdxLoginScope
OFFLINE_SCOPE	urn:opc:idm:__myscopes__ offline_access

2. Adding chatbot support to mobile application (Optional)

CHATBOT_ID	The tenant ID
CHATBOT_URL	The web socket URL for the ChatApp application in IBCS

3. Adding Bundle Identifiers

Bundle identifiers need to be added in the Info.plist of each of the frameworks along with the Signing Capabilities tab in Xcode. For example, the bundle identifier used is abc.def.ghi.jkl. The steps to be followed are,

- Right click on OBDXFramework.framework (in Xcode's Project Navigator) -> Show in Finder
- When the finder directory opens the right click OBDXFramework.framework -> Show package contents.
- Open Info.plist and set Bundle identifier as abc.def.ghi.jkl.OBDXFramework
- Repeat the steps for the other three frameworks as well, with the following values:
 - Bundle identifier for Cordova.framework : abc.def.ghi.jkl.Cordova
 - Bundle identifier for OBDXExtensions.framework : abc.def.ghi.jkl.OBDXExtensions
 - Bundle identifier for OBDXWatchFramework.framework : abc.def.ghi.jkl.OBDXWatchFramework

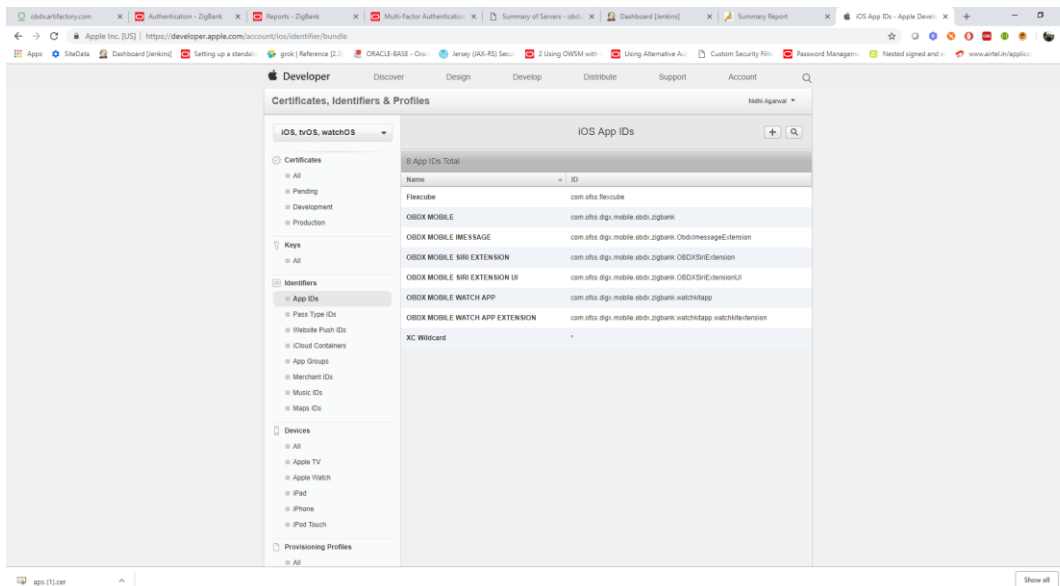
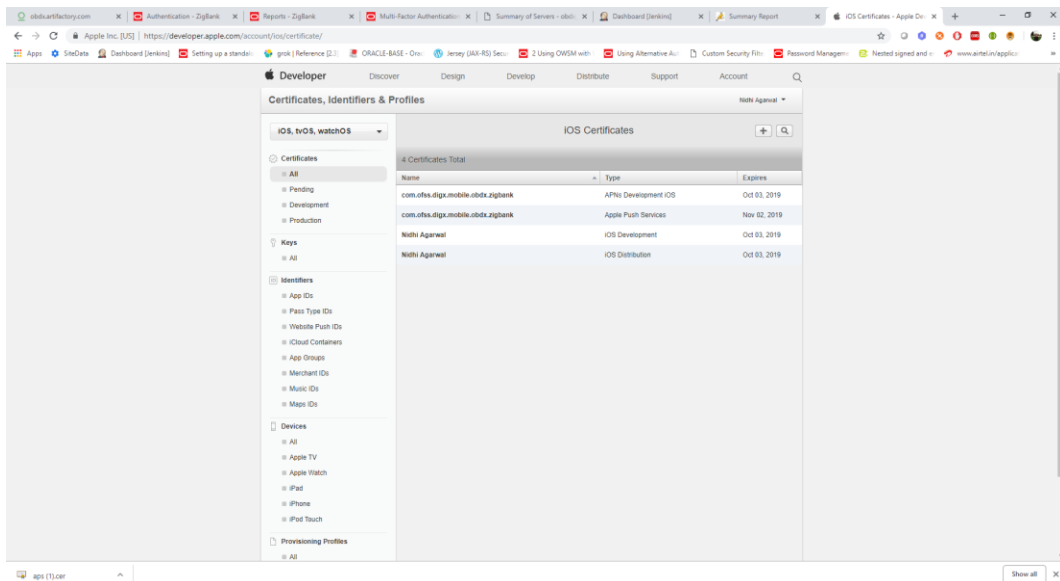
4. Bundle Versioning

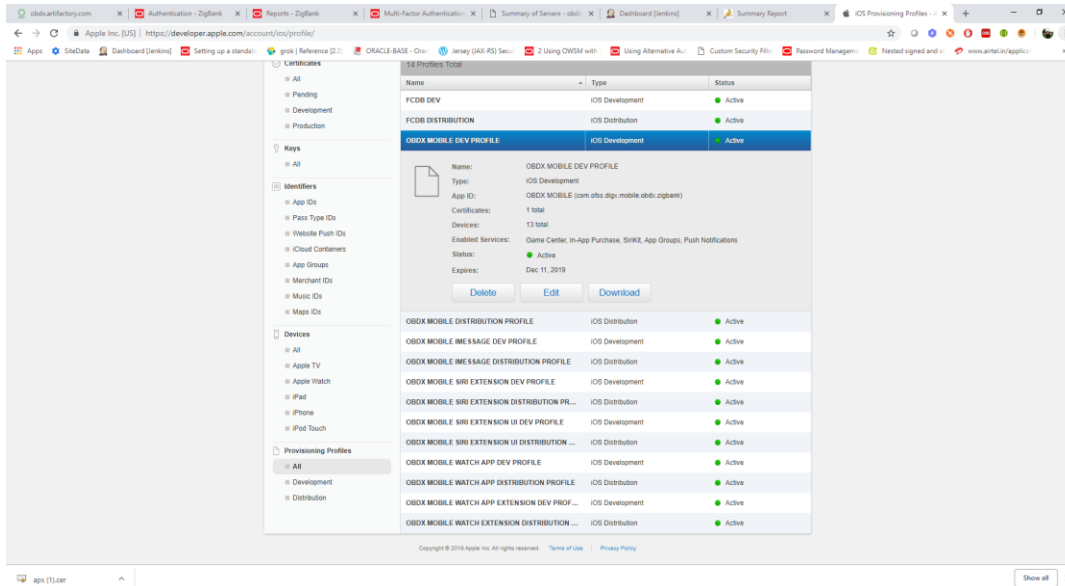
Proper bundle versioning should be done to prevent rejection from Apple's App Store. To achieve that the following steps need to be followed:

- Select "General" tab for "Zigbank" target and provide the values for Version and Build under "Identity" section. These values should be uniform across all the other targets and their Info.plists.
- Right click on OBDXFramework.framework (in Xcode's Project Navigator) -> Show in Finder
- When the finder directory opens the right click OBDXFramework.framework -> Show package contents.
- Open Info.plist and set Bundle Version with the same value as the Build Number and set the value of Bundle version string (short) as the Version Number.
- This needs to be repeated for all the other frameworks i.e. Cordova.framework, OBDXExtensions.framework and OBDXWatchFramework.framework.

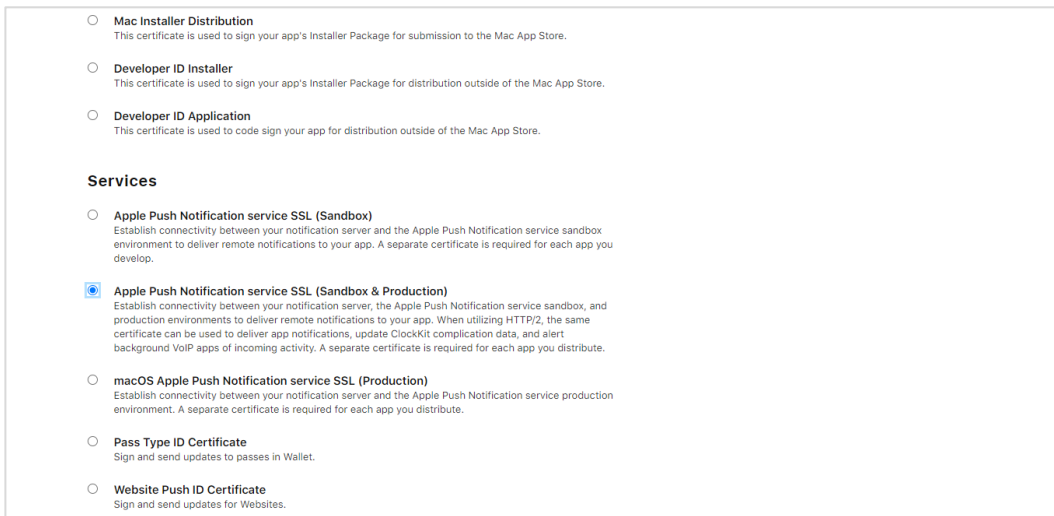
2.5 Generating Certificates for Development, Production and Push Notifications

Create all certificates (by uploading CSR for keychain utility), provisioning profiles and push certificates as shown below by login in developer console. For development add device UUIDs and add same to provisioning profiles. Add capabilities as shown below and ensure the bundle identifier matches the one of the application in Xcode

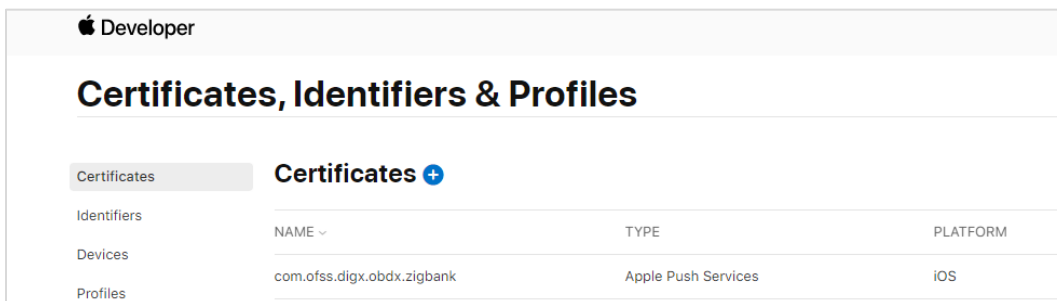




Ensure AppGroups capability is added to all profiles and for mobile profile SiriKit, App Groups, Push Notifications must be added.



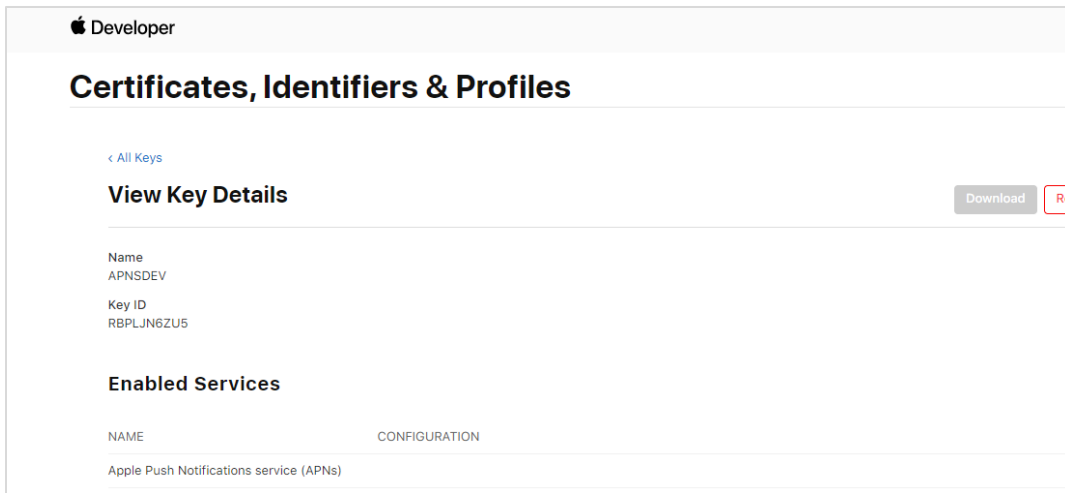
Note the certificate/bundle name



Note the Team ID from top right corner

Navigate to the “Keys” section and create APNS key

Note APNS key and download the .p8 file. Copy the .p8 to config/resources/mobile

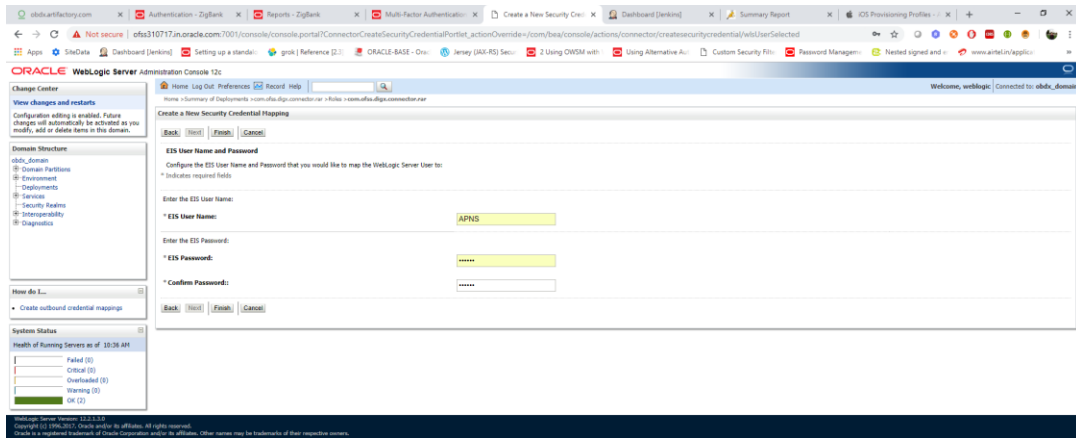


Update the password as shown below –

Sr. No.	Table	PROP_ID	CATEGORY_ID	PROP_VALUE	Purpose
1	DIGX_FW_CONFIG_ALL_B	APNS	DispatchDetails	<Password>	Provides key of .p8 certificate
2	DIGX_FW_CONFIG_ALL_B	APNSKeyStore	DispatchDetails	DATABASE or CONNECTOR	Specifies whether to pick certificate password from database or from connector. Default DB (No change)
3	DIGX_FW_CONFIG_ALL_B	Proxy	DispatchDetails	<protocol,proxy address>	Provides proxy address, if any, to be provided while connecting to APNS server. Delete row if proxy not required. Example: HTTP,148.50.60.8,80
4	DIGX_FW_CONFIG_ALL_B	CERT_TYPE	DispatchDetails	For dev push certs add row with value 'dev'	For prod push certificates this row is not required

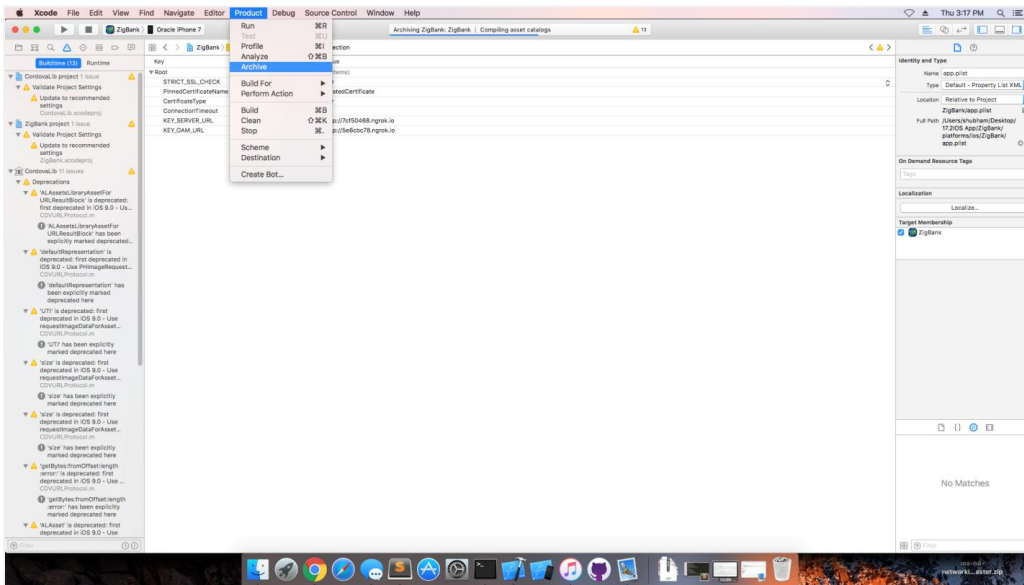
5	DIGX_FW_CONFIG_ALL_B	ios_cert_path	DispatchDetails	resources/mobile/AuthKey_RBPLJN6ZU5.p8	Update the certificate path/name if required. Should be relative to config directory
6	DIGX_FW_CONFIG_ALL_B	APNS_BUNDLE	DispatchDetails	Eg. com.offs.digx.obdx.zigbank	Certificate Name
7	DIGX_FW_CONFIG_ALL_B	APNS_TEAMID	DispatchDetails	Eg. 3NX1974C93	Team ID of Apple developer account

If CONNECTOR is selected in Step 2 update key as below

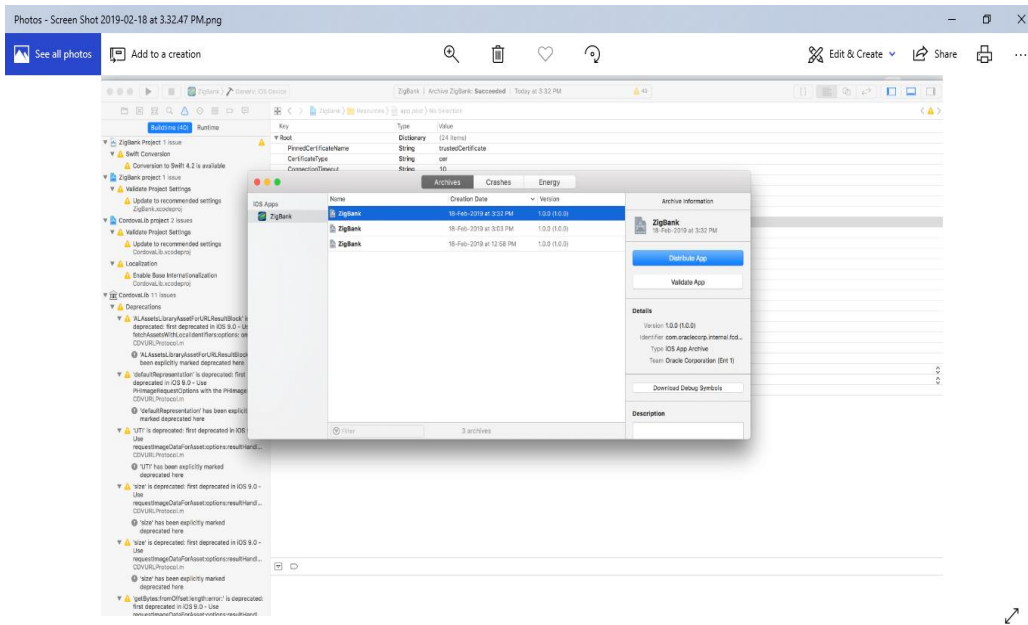


3. Archive and Export

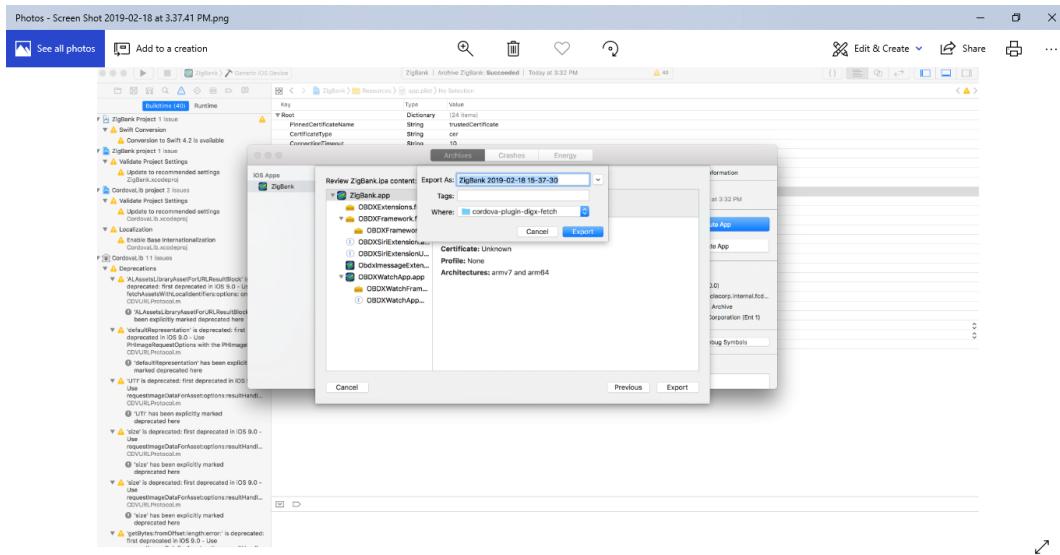
a. In the Menu bar click on **Product -> Archive (Select Generic iOS Device)**



b. After archiving has successfully completed. Following popup will appear



c. Click on **Distribute App** in the right pane of the popup -> select **the Method of Distribution -> Choose Provisioning Profile** according to the method of distribution -> select **Next -> Review the contents and click on Export -> Export** and generate the .ipa



To run the application on simulator copy & replace 4 frameworks (.framework files) from /simulator to zigbank/platforms/ios/

4. OBDX Authenticator Application

4.1 Authenticator UI (Follow any one step below)

4.1.1 Using built UI

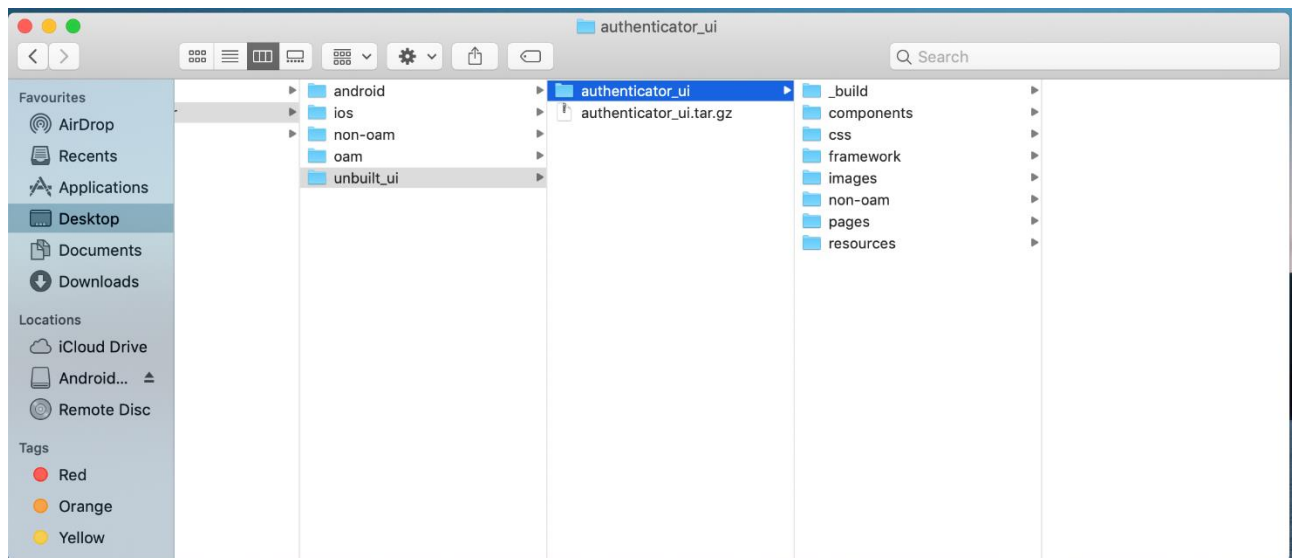
For Non-OAM - Unzip dist.tar.gz directory from OBDX_Patch_Mobile\authenticator\NON-OAM

For OAM - Unzip dist.tar.gz directory from OBDX_Patch_Mobile\authenticator\OAM

4.1.2 Building UI manually

1. Extract authenticator_ui.tar.gz from OBDX_Patch_Mobile\authenticator\unbuilt_ui.

The folder structure is as shown :



(a) OAM based Authentication

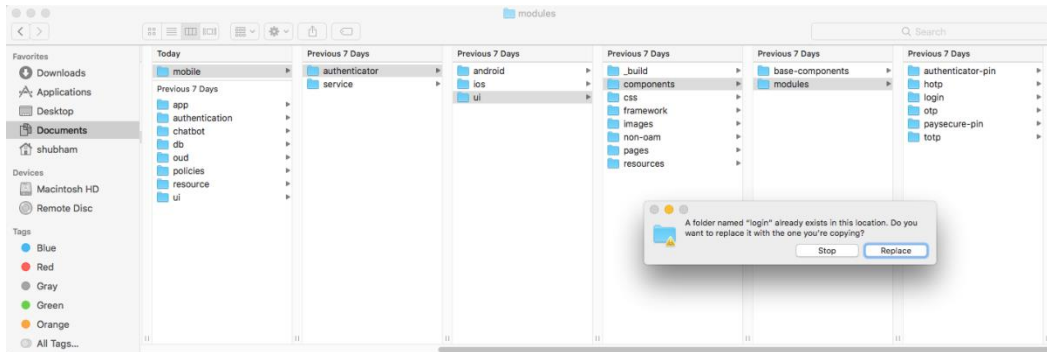
1. Open Terminal at “_build” level.
2. Run following command :

```
sudo npm install -g grunt-cli
sudo npm install
node render-requirejs/render-requirejs.js
grunt authenticator --verbose
```

3. After running above commands and getting result as “Done, without errors.” a new folder will be created at “_build” level with name as “dist”.

(b) NON-OAM Based Authentication

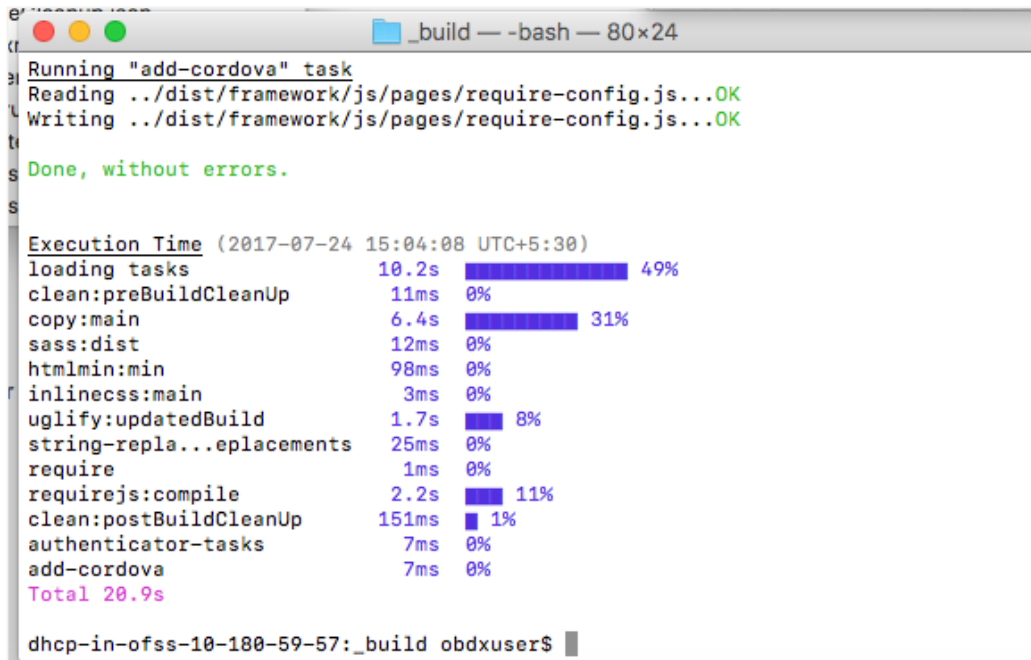
1. Copy “non-oam/login” folder and Replace it at location “components/modules/” [in ui folder] location. This will replace existing “login” folder.

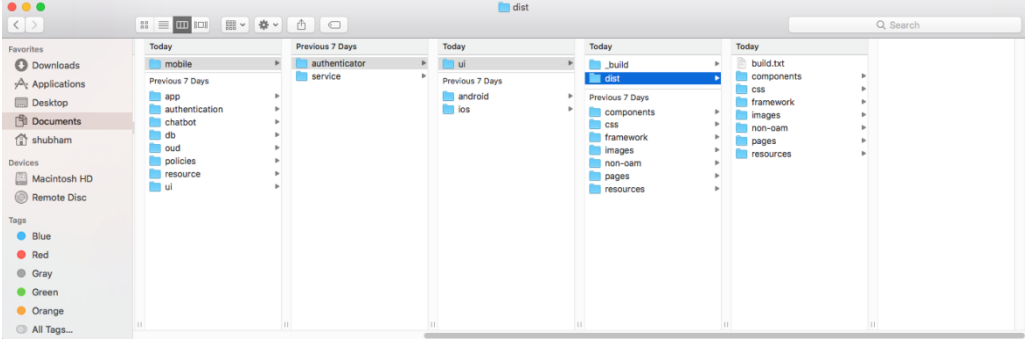


2. Open Terminal at “_build” level.
3. Run following command :

```
sudo npm install -g grunt-cli
sudo npm install
node render-requirejs/render-requirejs.js
grunt authenticator --verbose
```

4. After run command, a folder named “dist” is created at “_build” folder level with name as “dist”.





4.2 Authenticator Application Workspace Setup

1. Unzip and navigate to iOS workspace as shipped in installer.
 - a. Copy & Paste the cordova and CordovaLib as given below.
Run Following commands on terminal
 1. Install cordova using the command


```
sudo npm install -g cordova
```
 - a. If you face the above error then set proxy using following commands on command line.


```
npm config set proxy <provide your proxy value here>
npm config set https-proxy <provide your proxy value here>
```
 2. Create project using following command

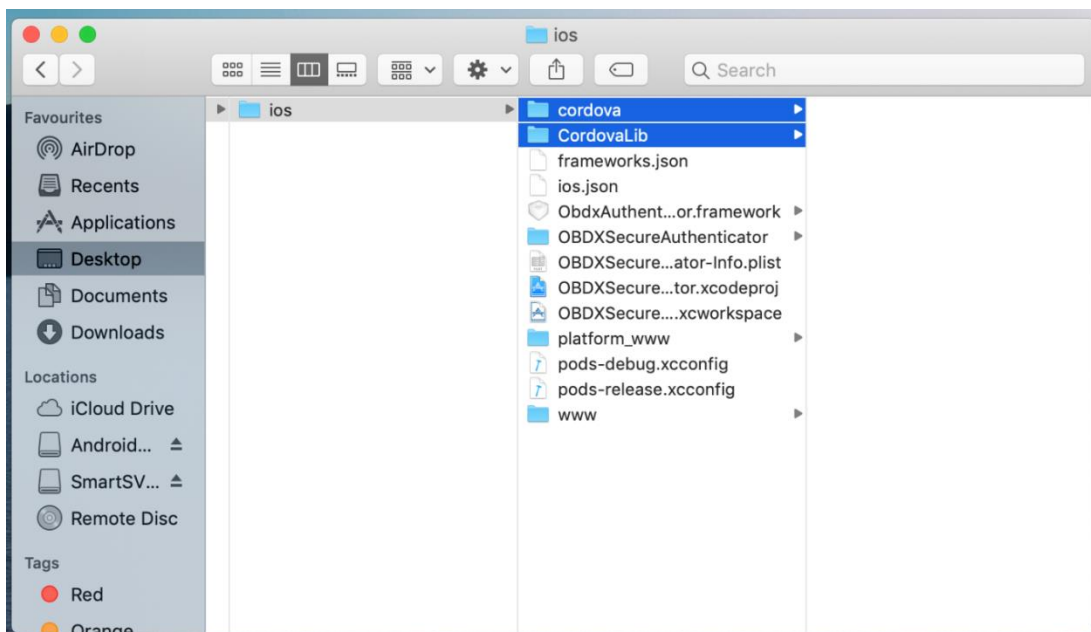

```
cordova create <directory name> <package name / app identifier> <project name>
```

Eg : `cordova create zigbank com.ofss.zigbank ZigBank`
 3. All subsequent commands need to be run within the project's directory


```
cd <directory name>
```

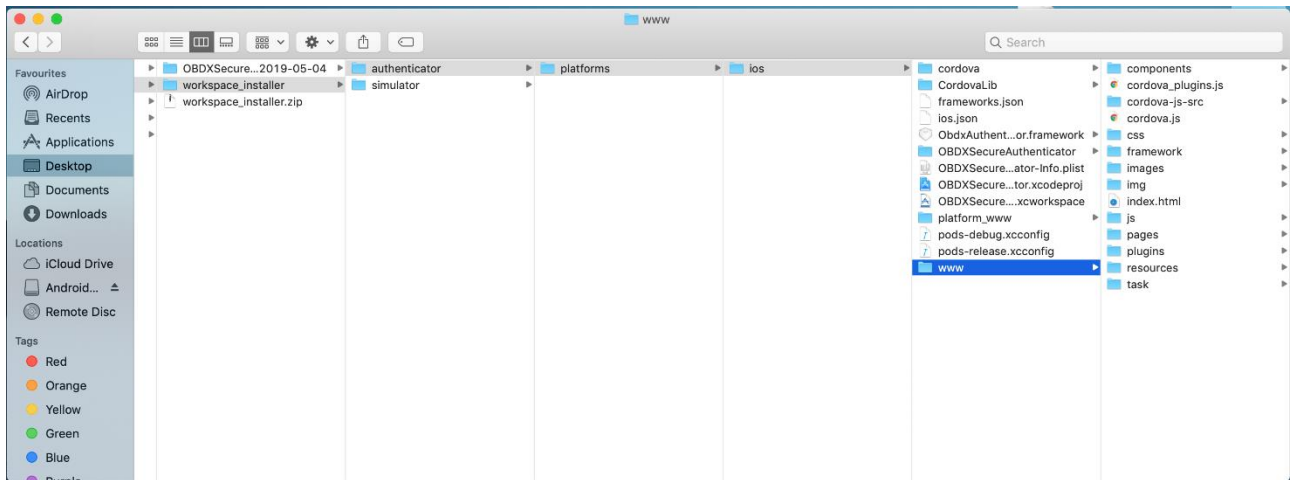
Eg: `cd zigbank`
 4. Add platform ios to the project using following command


```
cordova platform add ios@4.x.x
```

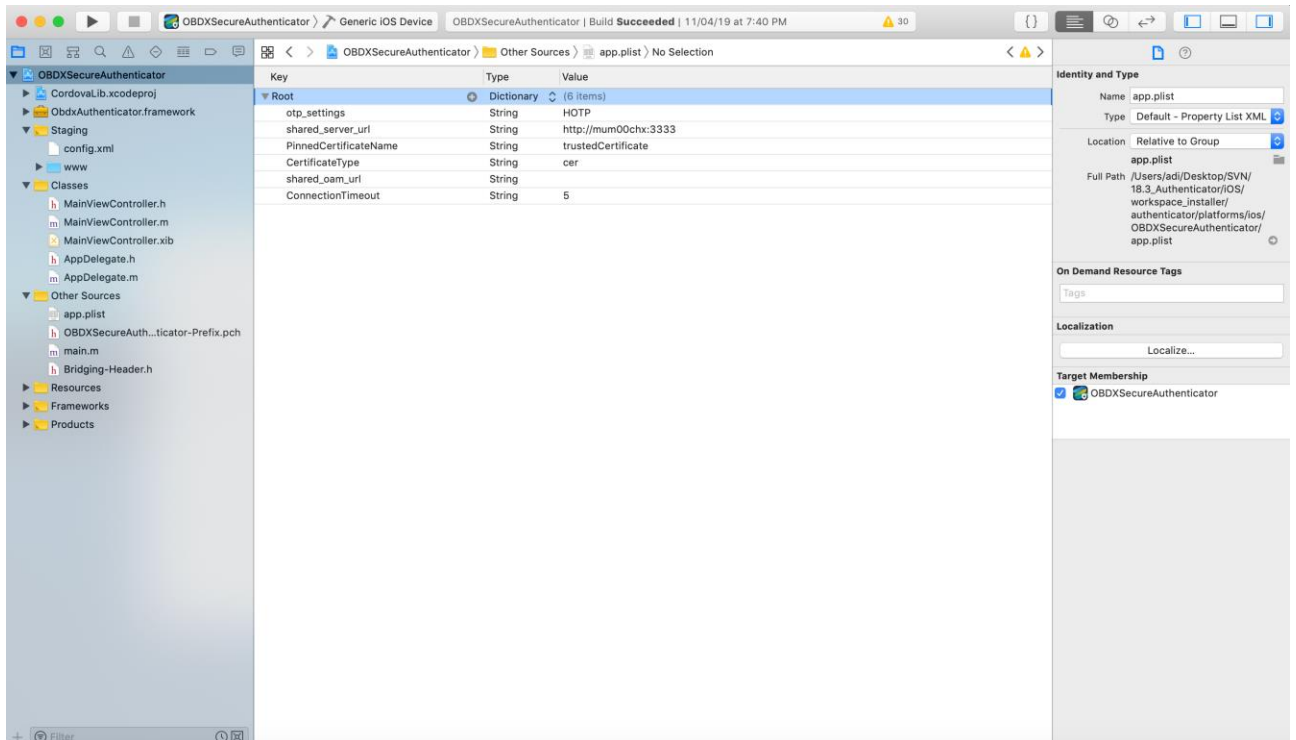


2. Open the workspace as shown below and find and replace the following generated UI files from “*ui/dist*” folder :
 - components

- css
- framework
- images
- pages
- resources



3. Double click on OBDXSecureAuthenticator.xcodeproj to open the project in Xcode



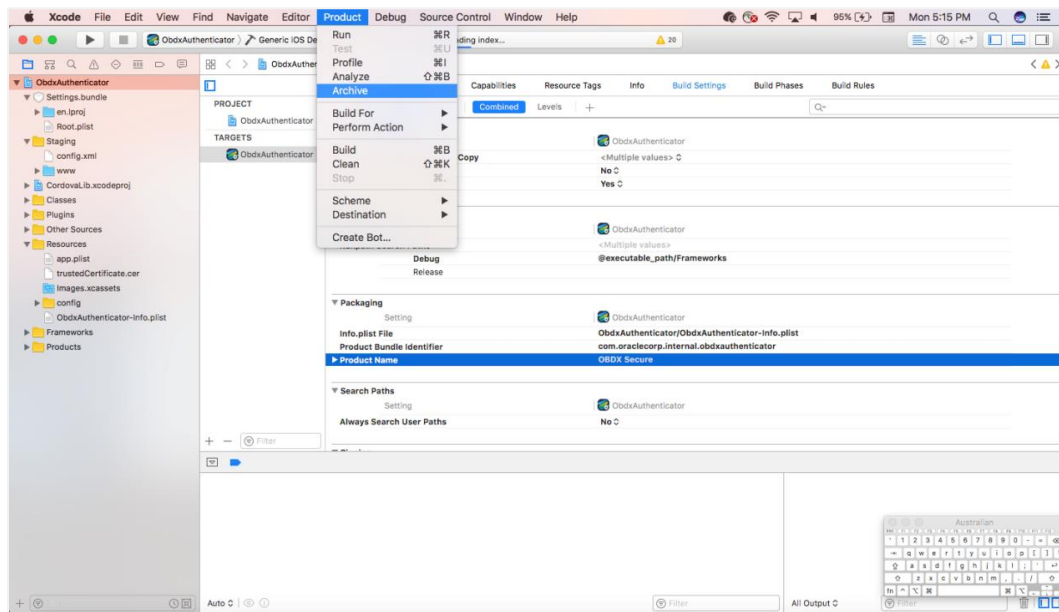
Update HOTP or TOTP in above screenshots and update the server URL.

4. The application can be archived using steps in Section 4.3 for running on device

5. To run the application on simulator, copy & replace the framework from simulator/ObdxAuthenticator.framework to /authenticator/platforms/ios/

4.3 Building Authenticator Application

1. Set the simulator to *Generic iOS* device. Then go to *Product -> Archive*.



2. Choose your Archive and then click “Export”. .ipa file will be generated

