iOS Application Distribution Guide

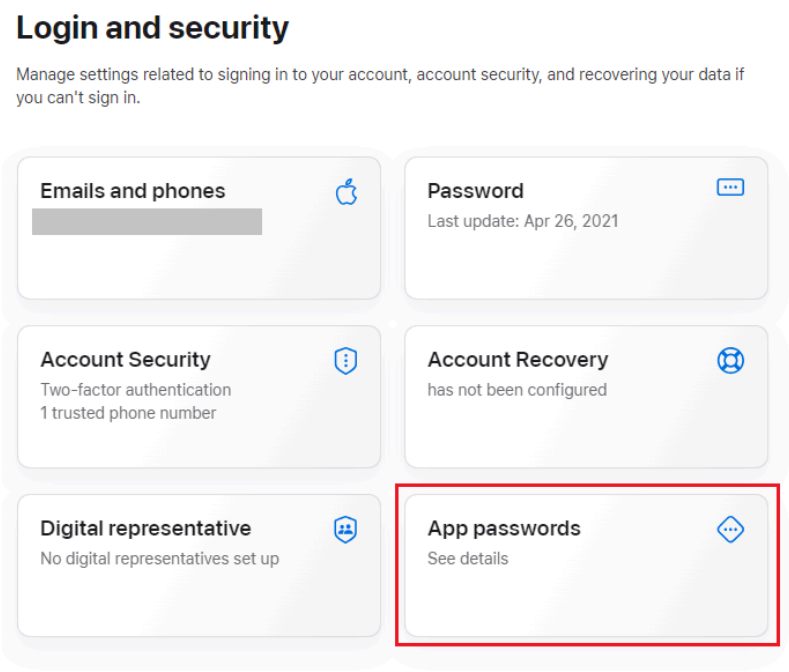
Data Collection for Signature

It is necessary to have a paid membership with Apple in order to distribute iOS applications. Depending on the membership, different types of developer account can be obtained.

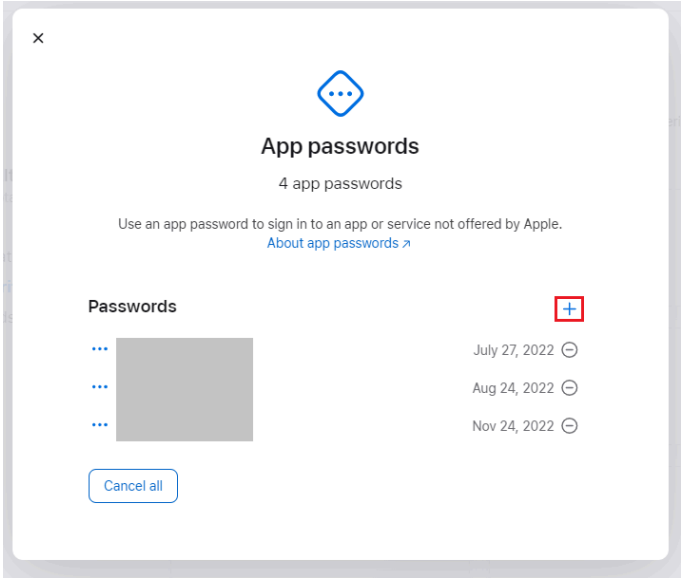
To create a new Apple developer account, go to https://developer.apple.com/account. As a result of the subscription, the registered email is used as the Apple ID identifier to access the developer account control panel and the Apple team identifier is obtained in the membership detail section.

To generate the password for Apple services it is necessary to access https://appleid.apple.com/sign-in and perform the following steps:

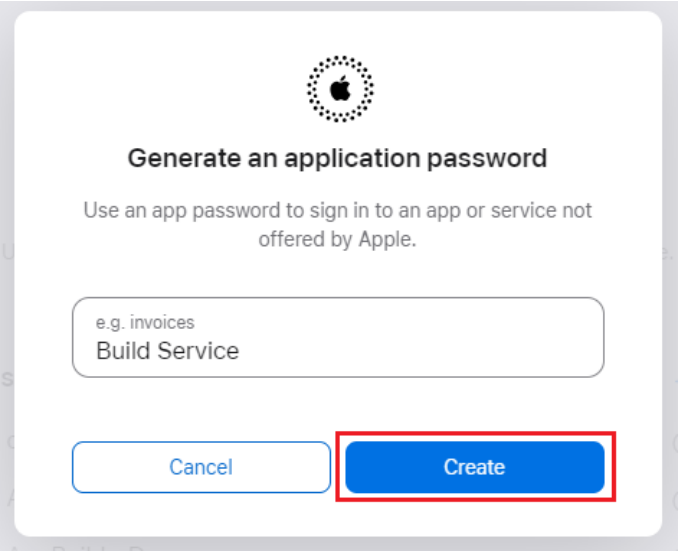
1. Enter “Application passwords”.



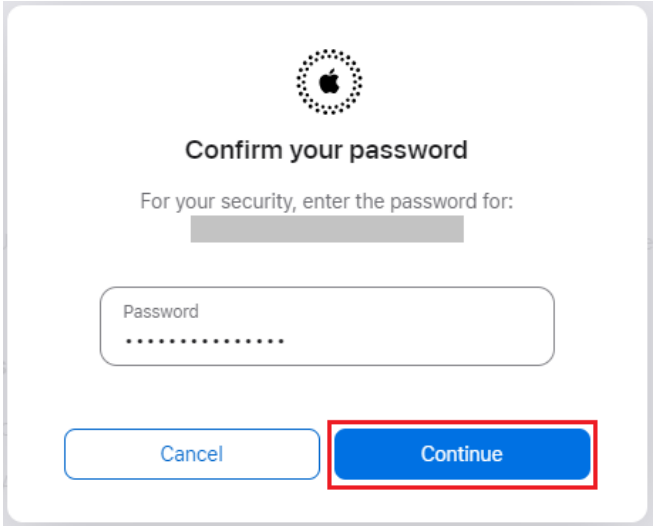
2. Create a new password.



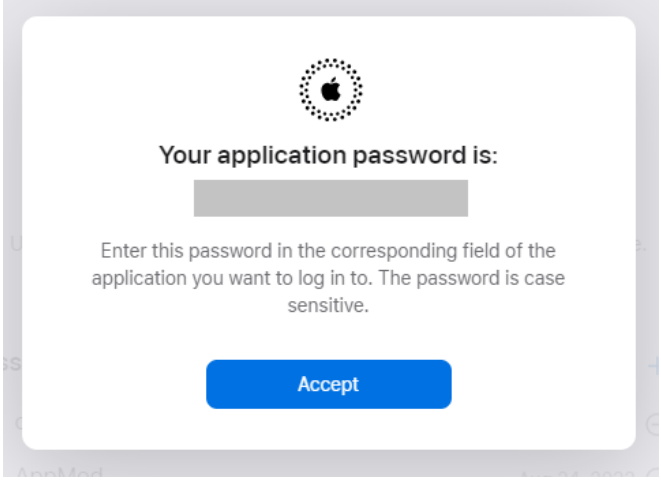
3. Give the password a descriptive name.



4. Enter the password associated with the Apple ID of the developer account.



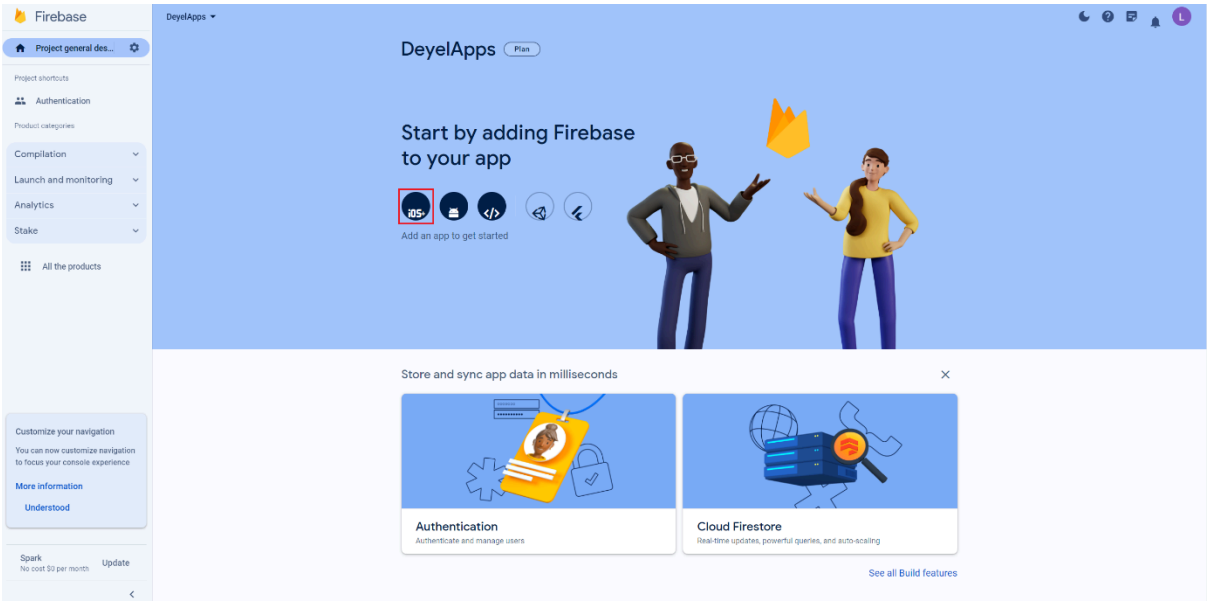
5. The developer receives the password for Apple services, which must be remembered as it cannot be retrieved later.



Collection of Google Services File

This configuration file allows integrating Google services, especially those that are part of Firebase and that are used for the use of “push” notifications in mobile applications. To obtain this file, the following steps must be followed:

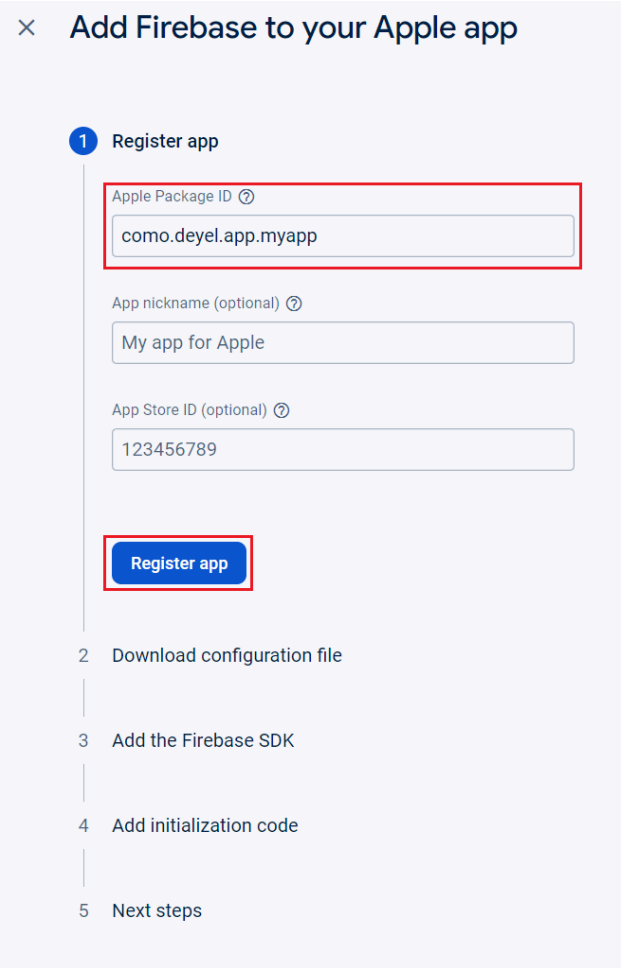
1. Login to https://console.firebase.google.com/, create a new project (enable Google Analytics) and select the iOS logo to create a new application within the project.



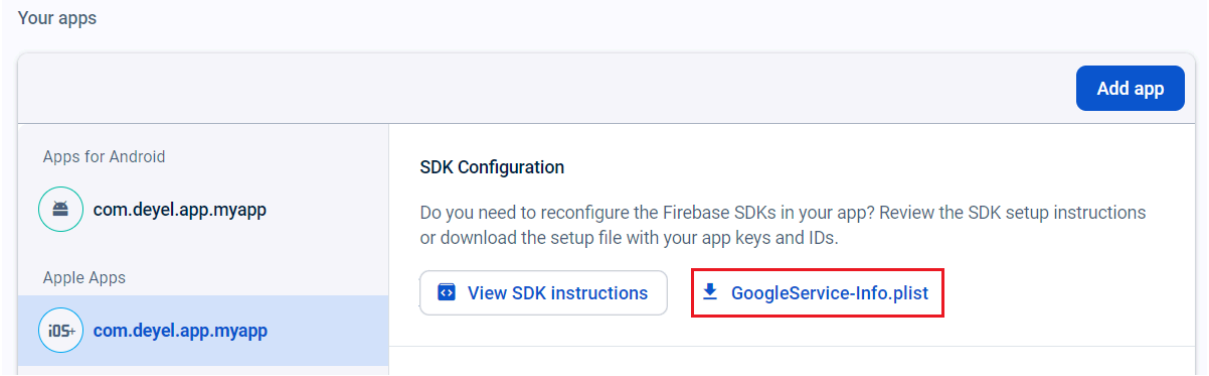
2. Within the creation of a new application in the project created previously, perform the following steps:

    a. Get the mobile application identifier from the [properties panel of the applications modeler of Deyel].

    b. Enter the identifier in the iOS package name and click on the “Register app” button.



3. Once the application is created, download the “GoogleServices-info.plist” file in the project configuration.



Creation of the Certificate and Provisioning Profile

Certificates identify who has signed an application or wants to use a service. They have a code signature to assure the user that an application comes from a trusted source and has not been modified since the certificate was requested. It is possible to create development and distribution certificates. Development certificates are used to run the application on test devices, and distribution certificates are used to distribute it to a limited group of devices or upload it to “App Store Connect”.

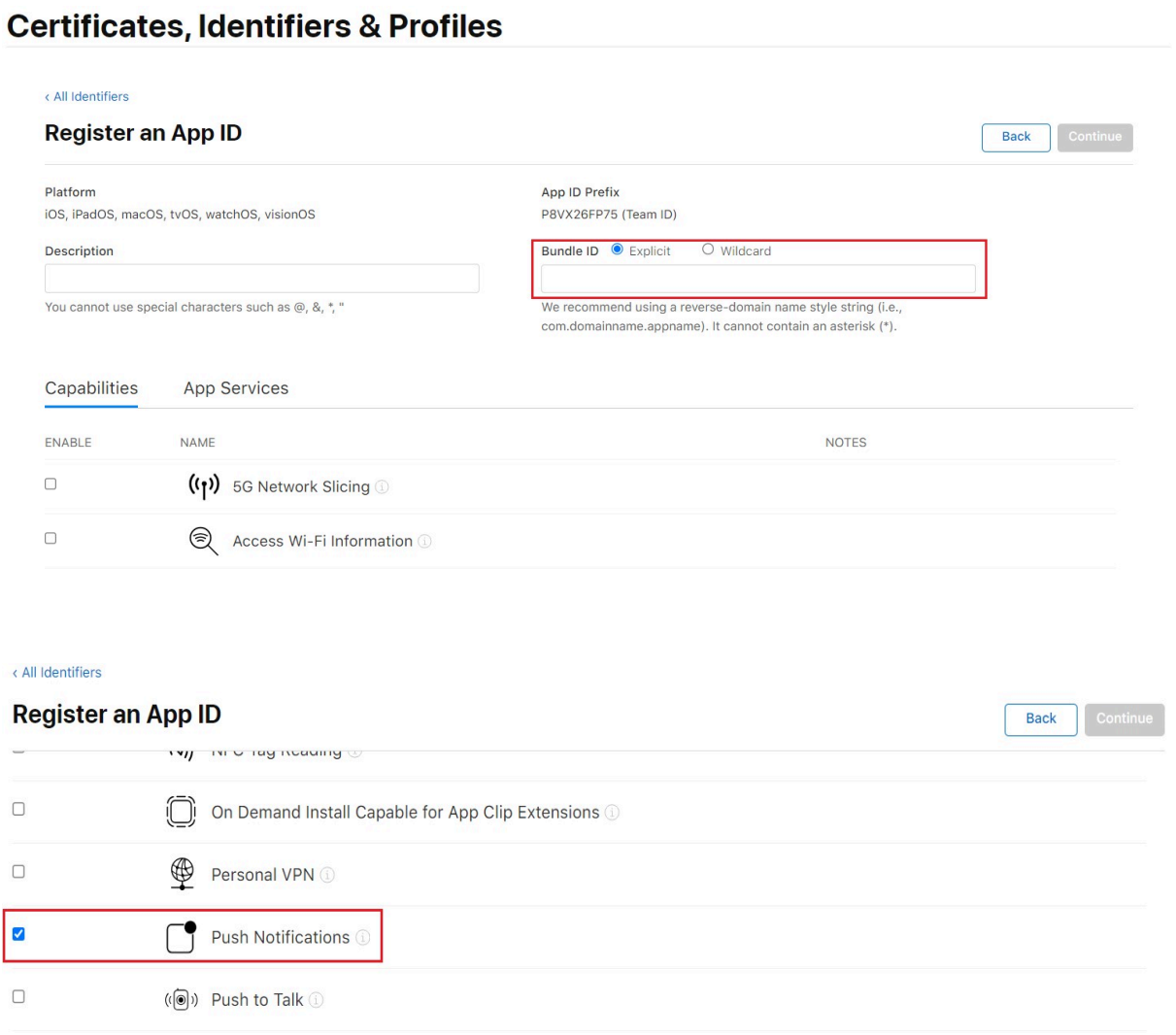
A provisioning profile authorizes the application to use certain services and ensures that the developer is known, that is, develops, uploads, or distributes the application. A provisioning profile contains a unique application identifier and a distribution certificate.

For more information on how to obtain an iOS provisioning profile and signing certificate, go to certificate management or how to create profiles and distribution certificates.

Certificate Creation

To generate a distribution certificate in the Apple developer panel,the following steps must be performed:

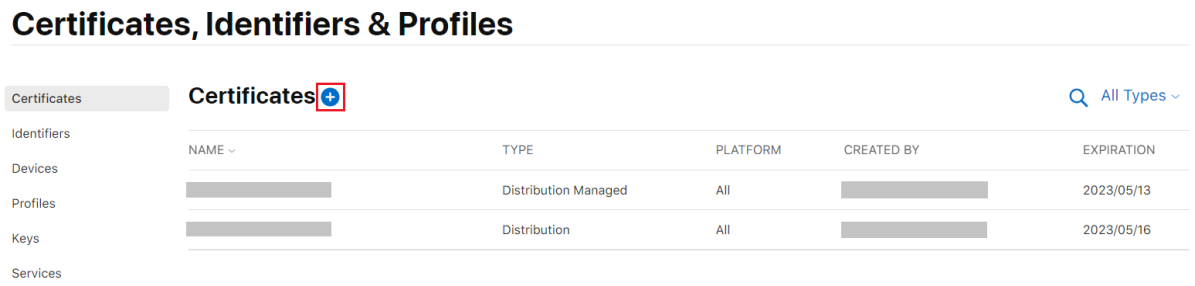
1. Register the Deyel mobile application identifier.
    a. In the Apple Developer Center, select the option “Certificates, IDs” & Profiles” and then go to section “Identifiers”.
    b. Enter in the bundle ID field the value of the “Application Identifier” property obtained from the properties panel of the Deyel application modeler.
    c. Select the “Push Notifications” item in the “Capabilities” section.



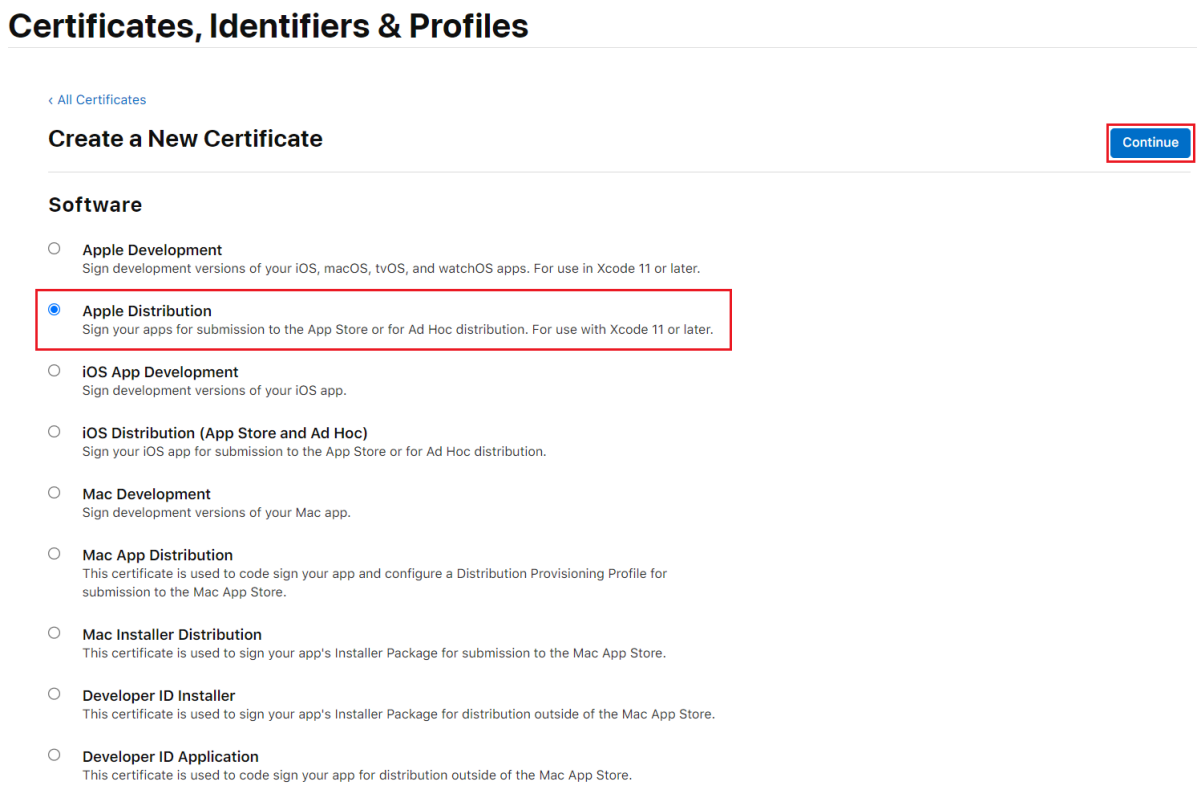
2. Before creating the Apple certificate, a certificate signing request file must be generated. This file can be created on Mac OSX using the “Keychain Access” application and on other operating systems using OpenSSL in the command line interface.
    a. Generate a private key with the following command.
    openssl genrsa -out keyname.key 2048
    b. Generate a file with “.csr” extension in the current working directory by executing the following command.
    openssl req -new -key keyname.key -out CertificateSigningRequest.certSigningRequest

    The questions requested by the command must be answered and a password should be defined when requested. This entered password should be remembered as it will be used in the configuration of the “Certificate Password” property in the properties of the Deyel environment.

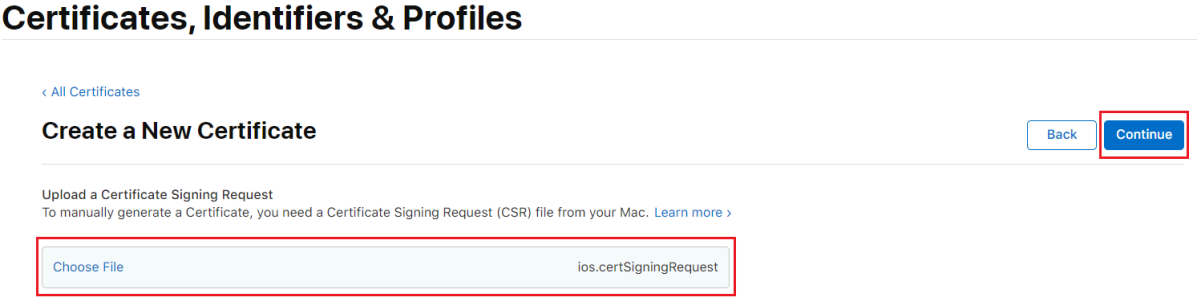
3. Click on “Create a new certificate” in the [Apple developer panel](#).



4. Select the “Apple Distribution” option.



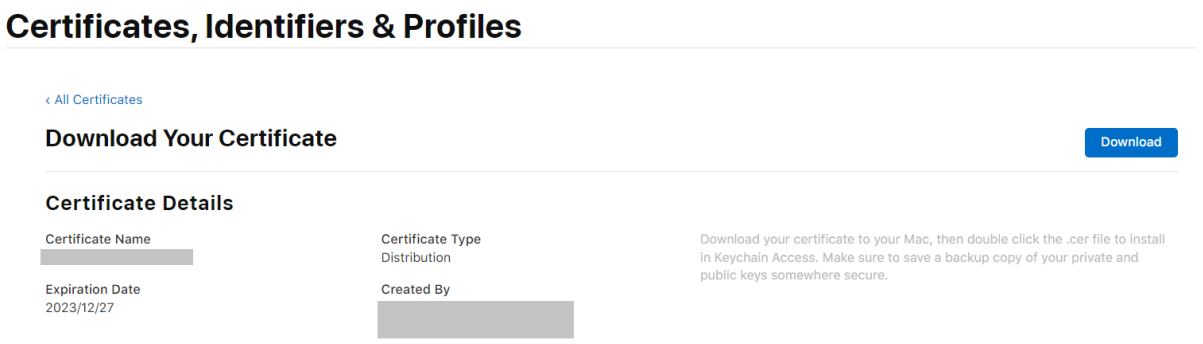
5. Upload the file with the extension “.certSigningRequest” created in step 2.



6. By clicking the “Continue” button. Apple signs the file with the “.certSigningRequest” extension and the signed certificate is available for download.

7. If a message saying that the certificate signing request is pending approval is received, it means that it does not have the required access rights.

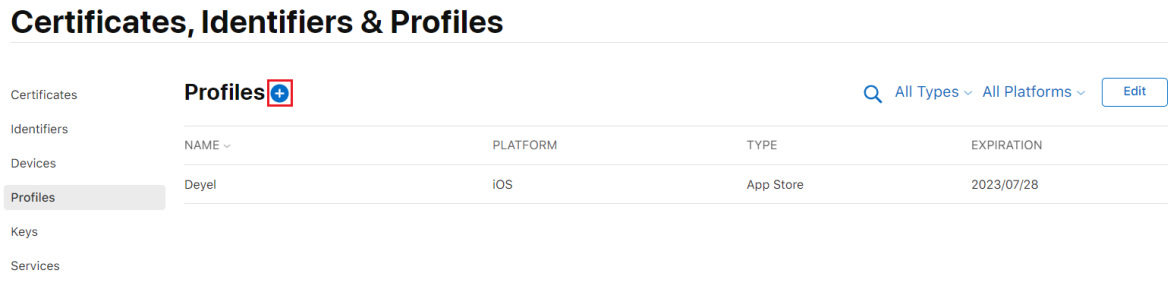
8. The certificate is created and ready to be downloaded as a file with the name "distribution.cer".



Creation of the Provisioning Profile

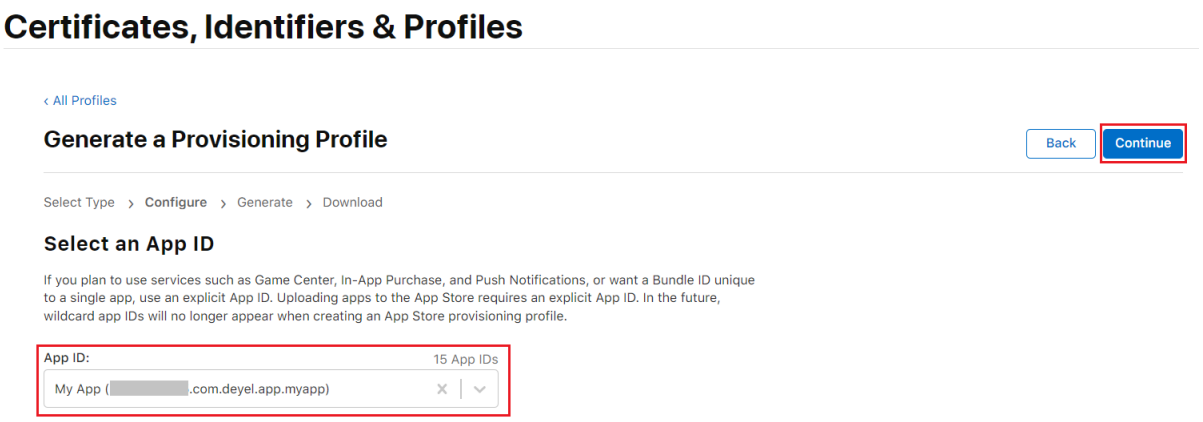
To generate a provisioning profile for application distribution in the [Apple developer panel](), the following steps should be taken:

1. Click on “Create a new profile”.

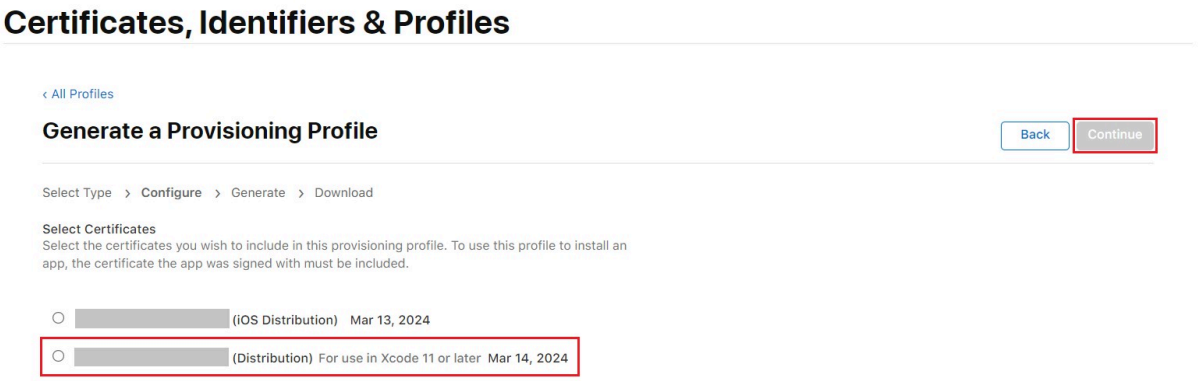


2. Select “Apple Store” to be able to upload to the application store or select “Ad Hoc” to distribute via QR code.

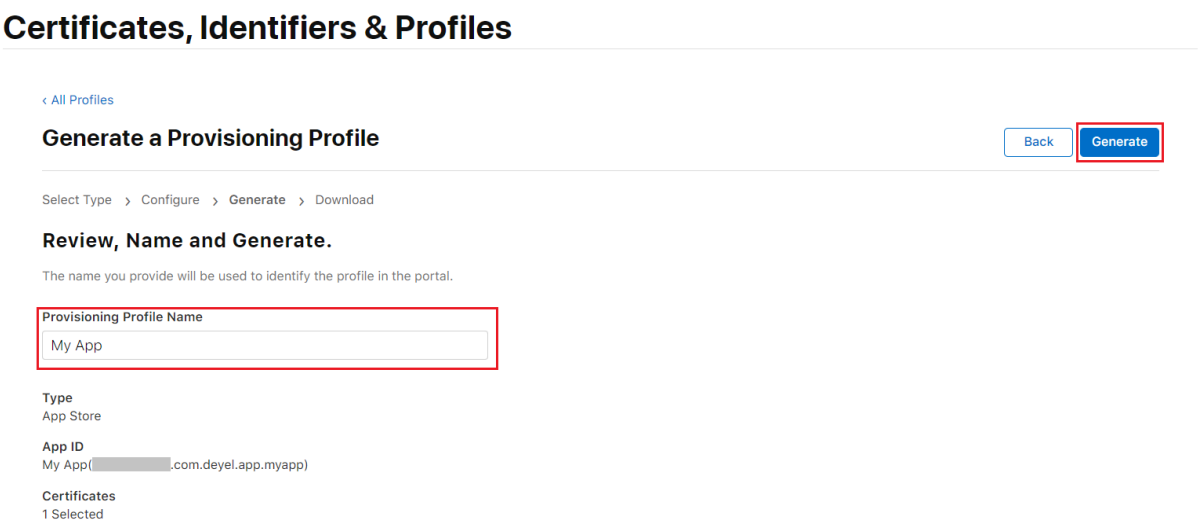
3. Select the application to associate with this profile using the corresponding “App ID”.



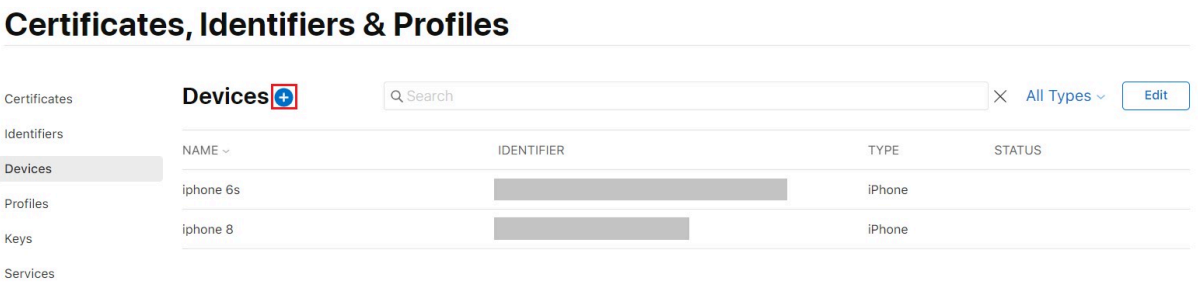
4. Select the previously created certificate to associate it with the profile.

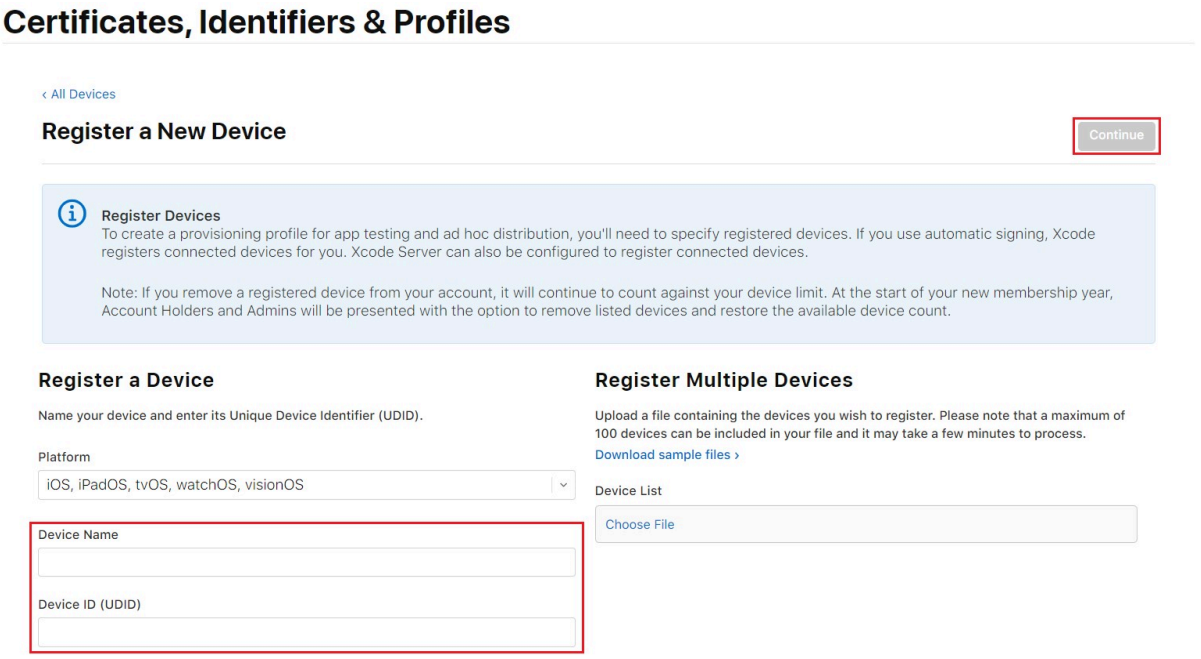


5. Define a descriptive name for the profile and before generating the profile, validate that the field values correspond to those previously entered.

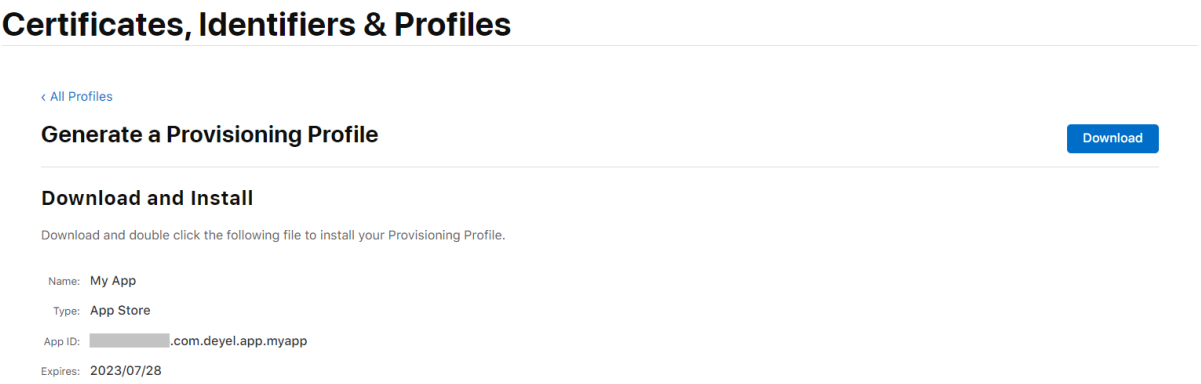


6. If distribution is via QR code, it is necessary to upload the devices that will be able to install and use the application. To do this, it is necessary to obtain the UUID (Universal Unique Identifier) of each device and register them in the “Devices” section, in the Apple developer panel. One way to get it is by scanning the QR code on the website [UUID].





7. Finally, the provisioning profile is generated and it is possible to download it as a file with the “.mobileprovision” extension.



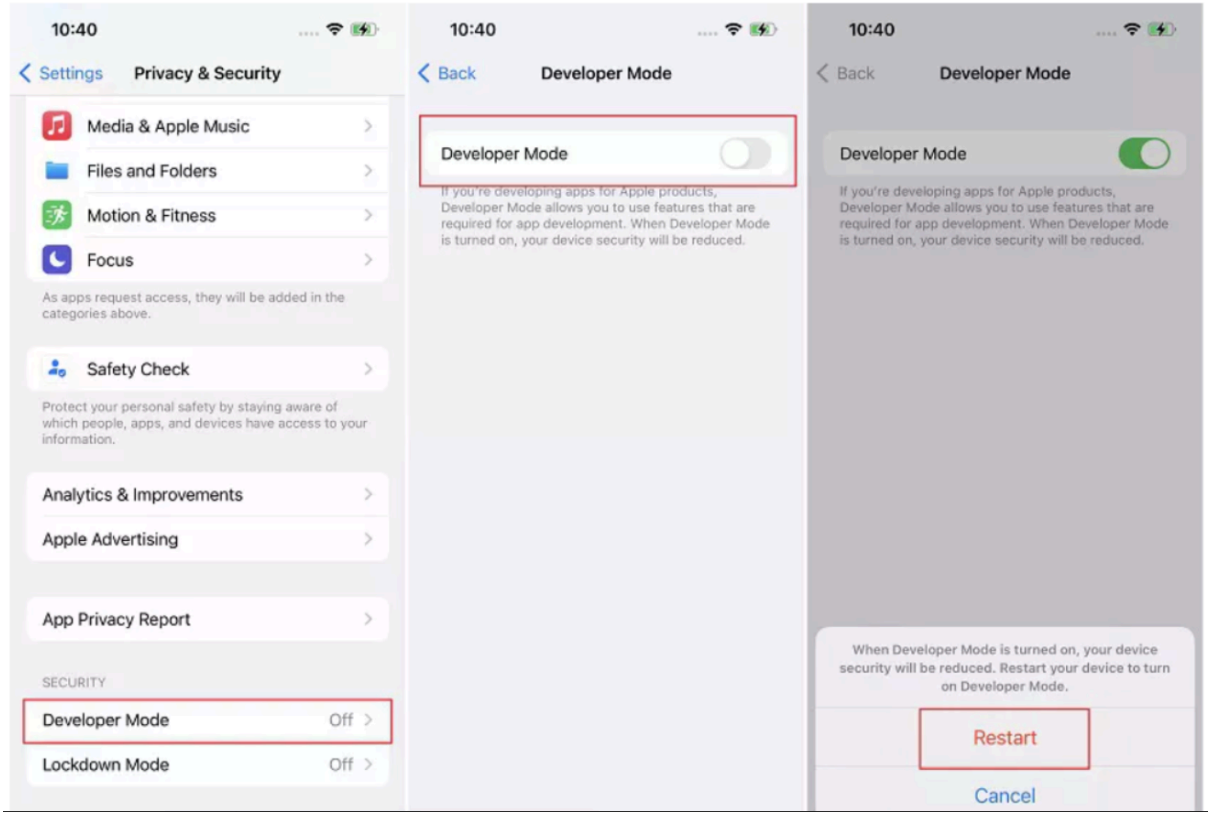
Mobile Application Distribution in Deyel

Each distribution option has different characteristics.

Distribution Via QR Code

For this option it is necessary to upload a distribution certificate and an “Ad Hoc” provisioning profile, as previously described in step 2 of the “Creation of the Provisioning Profile” section.

In order to install applications with this type of distribution, the device must have the “Developer Mode” enabled. To enable it, the following steps must be followed on such device:



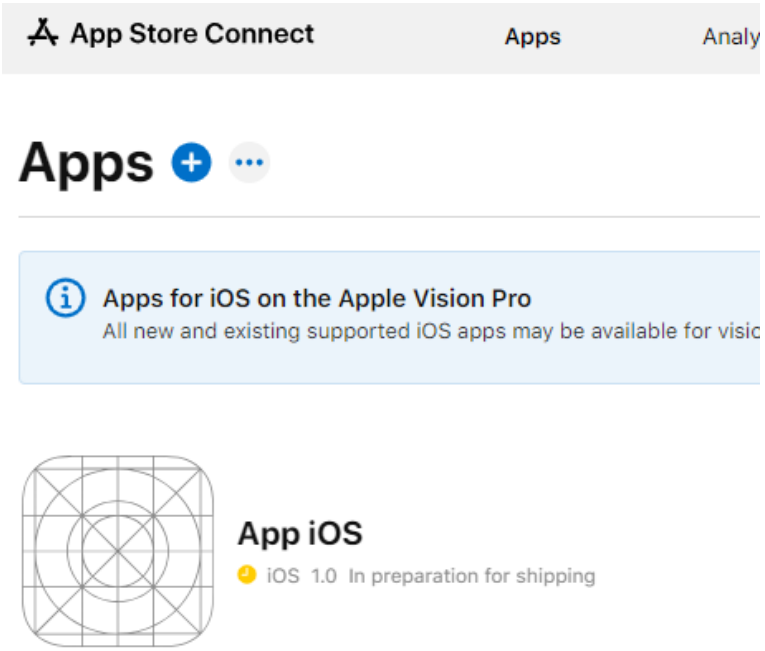
Distribution Via Store

Once the application building in the Deyel environment is made. The tasks described below can be performed for a complete upload of the application to the store.

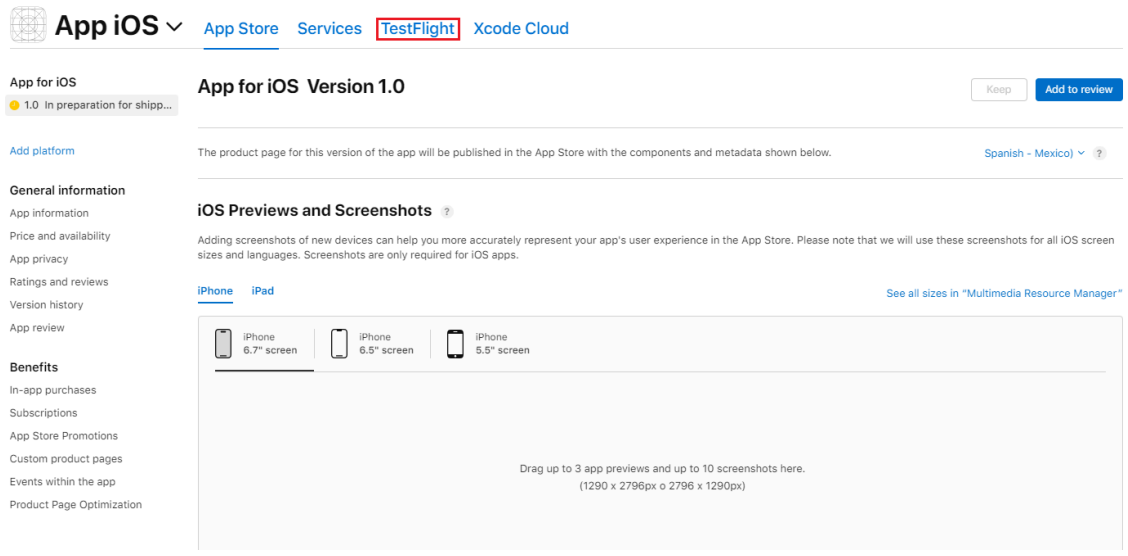
Tests with TestFlight

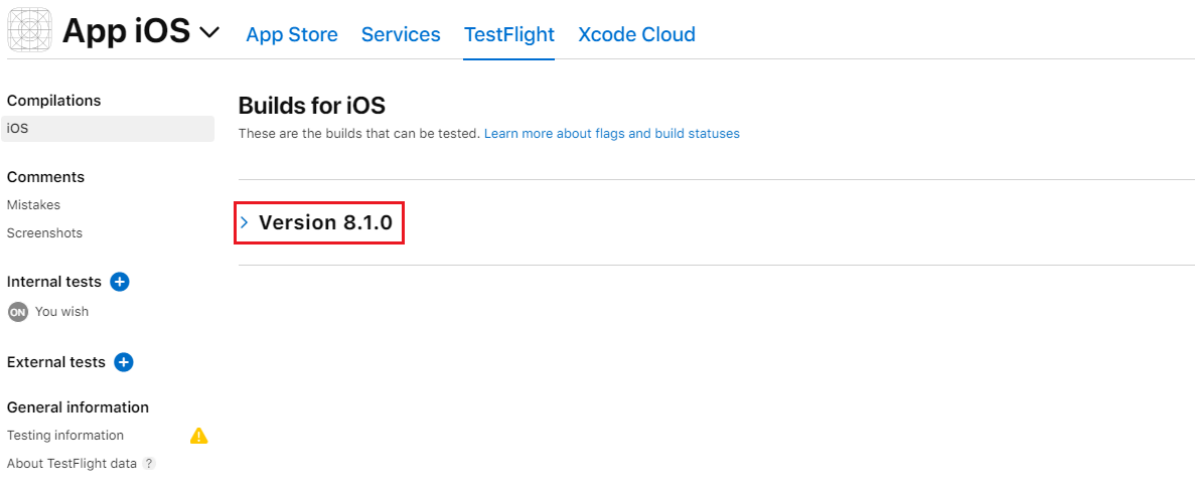
The application can be enabled in TestFlight, which is a service that allows to install and test apps before uploading them to the App Store, with the aim of conducting “beta” tests with a defined group of testers. To do this, it is necessary to perform the following steps:

1. Login to https://appstoreconnect.apple.com/ with the username and password of the developer account and in the application panel, select the application created using its descriptive name, for example “iOS App”.

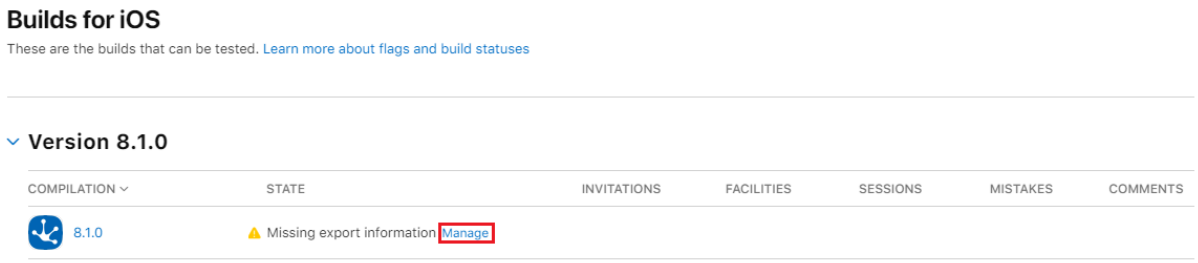


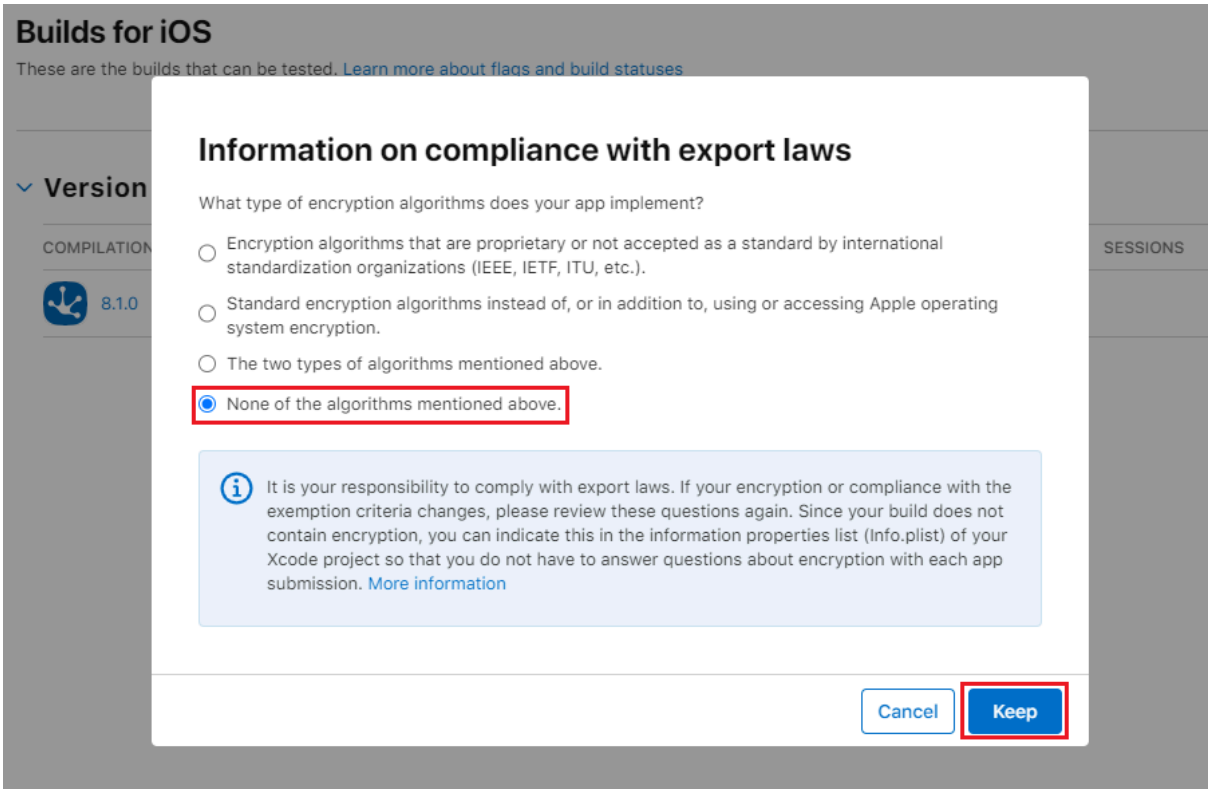
2. Within the application, go to the “TestFlight” tab to search for the uploaded application.



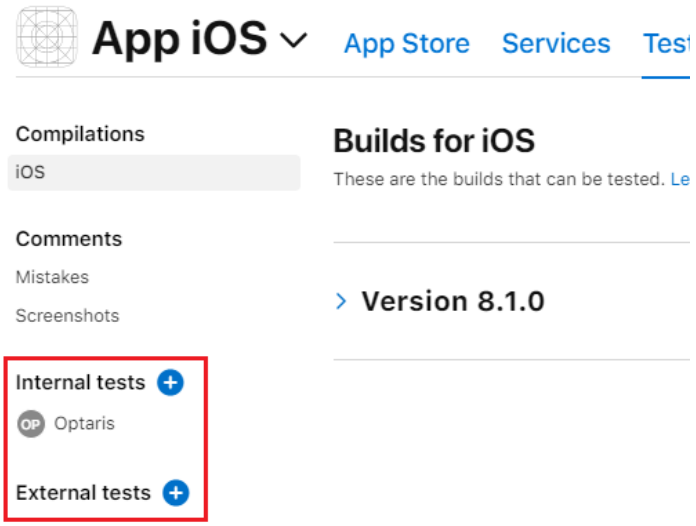


3. To enable the application for review by testers, it is necessary to define a security validation on compliance with laws and regulations related to the export of the application.





4. Once the application is enabled for review, it is necessary to define tester users.
    a. Login to https://appstoreconnect.apple.com/access/users and define the users who should test the application.
    There are two types of testers:
        i. Internal: users with an Apple developer account (free or paid).
        ii. External: users who do not have an Apple developer account and want to test the application. They can be uploaded individually or in bulk (using a file with an extension ".csv"). To use this type of testers, it is necessary for the application to undergo review by the Apple team. This review may take 1 day.
    b. Within the "TestFlight" section, define the testers according to their type.



c. Testers must install the TestFlight application from the device.
d. Testers receive an invitation to access the application made available by the developer on App Store Connect. It is possible to invite testing via email or a public link. The trial lasts 90 days and then is disabled.

For detailed information consult Beta testing with TestFlight.

Complete the Technical Data Sheet of the Application and Submit it for Review

If the tests are satisfactory, the technical data sheet of the mobile application can be completed and sent for review to the Apple team from the “App Store” tab within App Store Connect If the review is correct, the application will automatically be available in the App Store. However, the review may lead to interaction with the Apple team, guided by the interface, to resolve issues that may arise.

For detailed information, refer to the App Store Connect Guide.