

Manual de instalación de Entorno de Desarrollo para la aplicación Banca Empresas

Lima – Perú

2023

Tabla de contenido

Manual de instalación	3
1. Prerrequisitos:	3
2. Verificación de variables de entorno de Android	3
3. Clonar Repositorio.....	4
4. Instalar dependencias	5
5. Corrección de Librería desfasada	5
6. Configurar Emulador	6
7. Ejecución de la aplicación	6
8. Generación de instalador	9
9. Versionamiento de la aplicación	20
10. Consideraciones adicionales para el desarrollo	21

Manual de instalación

Para la instalación del entorno de desarrollo para el aplicativo App Empresa, se debe instalar los prerequisites para posteriormente clonar el repositorio, configurarlo y ejecutar el emulador o dispositivo físico.

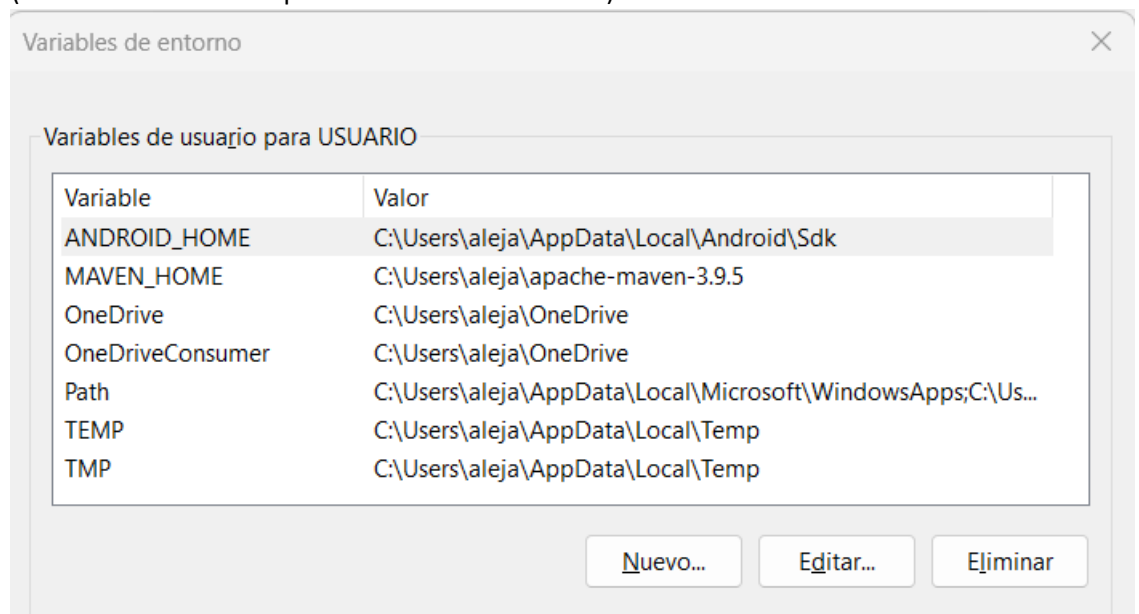
1. Prerrequisitos:

- Instalación de JDK 11
- Android SDK 13
- Instalación de Node 18.18.2
- Instalación de npm 9.8.1
- IDEs IntelliJ y AndroidStudio
- En caso de usuario de Mac se necesita tener XCode y Cocoa Pods 1.11.3 instalado

2. Verificación de variables de entorno de Android

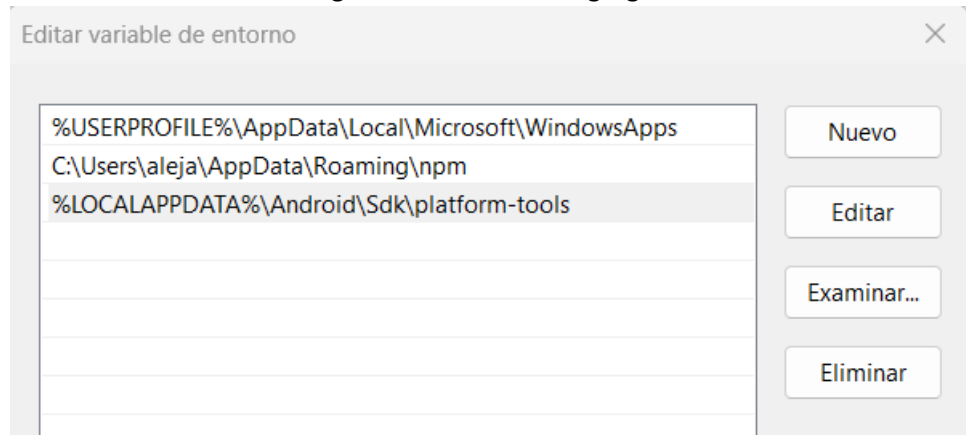
Ingresar a las variables de entorno de la computadora y verificar lo siguiente:

- En la sección Variables de entorno para USUARIO debe tener configurado la variable ANDROID_HOME. De lo contrario agregarla apuntando al sdk de Android (AndroidStudio instala por defecto el Android SDK)



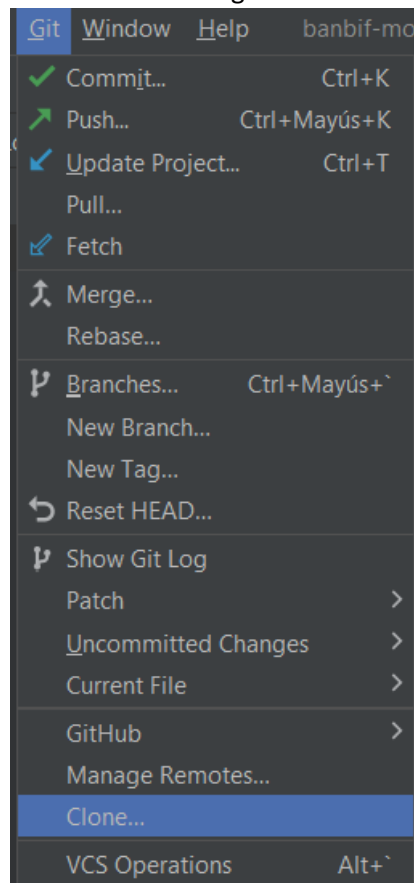
- En la sección Variables de entorno para USUARIO presionar doble click en la variable Path y verificar que se encuentre agregado el platform-tools de Android

como se muestra en la imagen. De lo contrario agregarlo

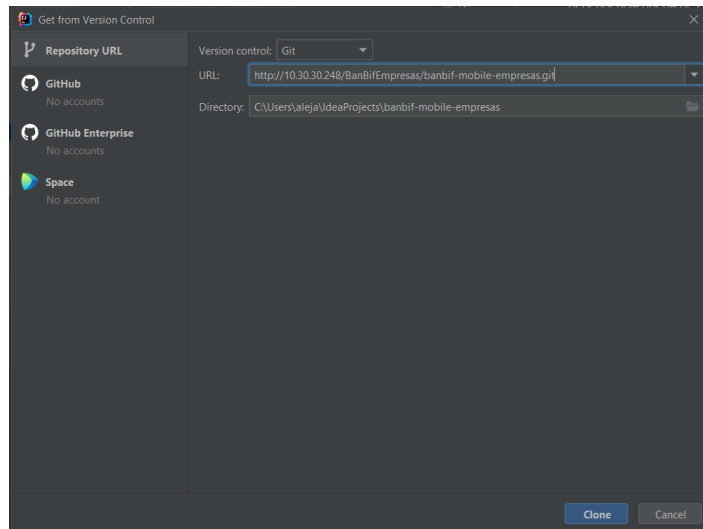


3. Clonar Repositorio

- Usando IntelliJ dirigirse al menú Git > Clone ...



- Ingresar el Url del repositorio:
 - o URL: <http://10.30.30.248/BanBifEmpresas/banbif-mobile-empresas.git>



- Cuando el IDE lo solicite ingrese las credenciales de acceso

4. Instalar dependencias

Abrir un terminal en la ruta raíz del proyecto y ejecutar el siguiente comando:

“npm install”

para instalar las dependencias del proyecto

Paso especial para usuarios Mac:

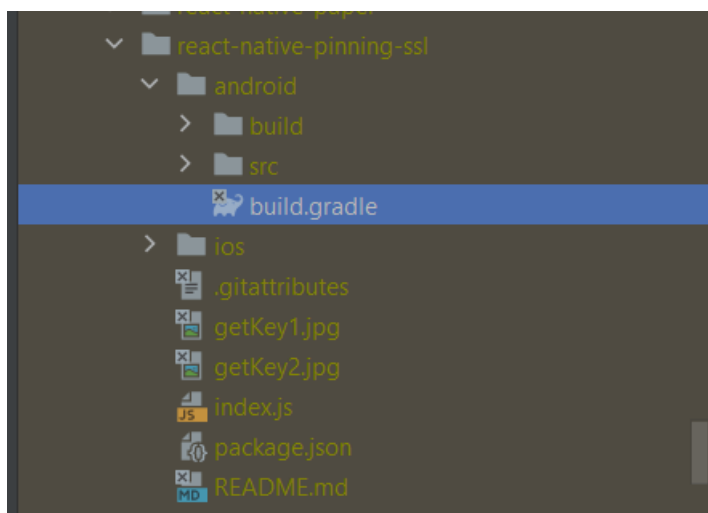
Luego de realizar el paso anterior, abrir un terminal en la carpeta ios y ejecutar el siguiente comando:

“pod install”

Para instalar las dependencias de cocoa pods para ios

5. Corrección de Librería desfasada

Abrir el archivo “/banbif-mobile-empresas/node_modules/react-native-pinning-ssl/android/build.gradle”

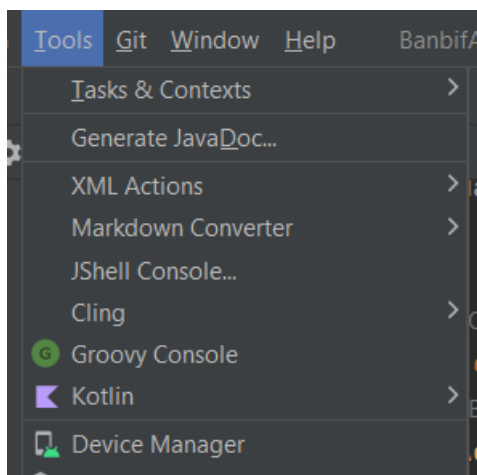


Cambiar “compile” por “implementation”. Es decir, la dependencia queda de esta forma:
“implementation 'com.facebook.react:react-native:+'”

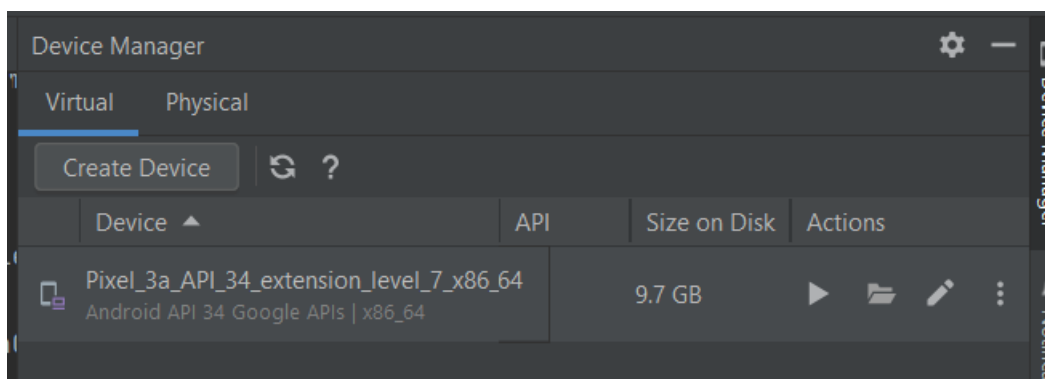
```
32  
33 dependencies {  
34     implementation 'com.facebook.react:react-native:+'  
35 }  
36
```

6. Configurar Emulador

Abrir el IDE AndroidStudio y crear un nuevo dispositivo virtual, ir al menú Tools > Device Manager



Crear un nuevo dispositivo con el Android API 33 o superior



Opcionalmente también se puede conectar un dispositivo físico para hacer las pruebas

7. Ejecución de la aplicación

Se puede ejecutar la aplicación en un simulador o un dispositivo físico:

Ejecución en un emulador:

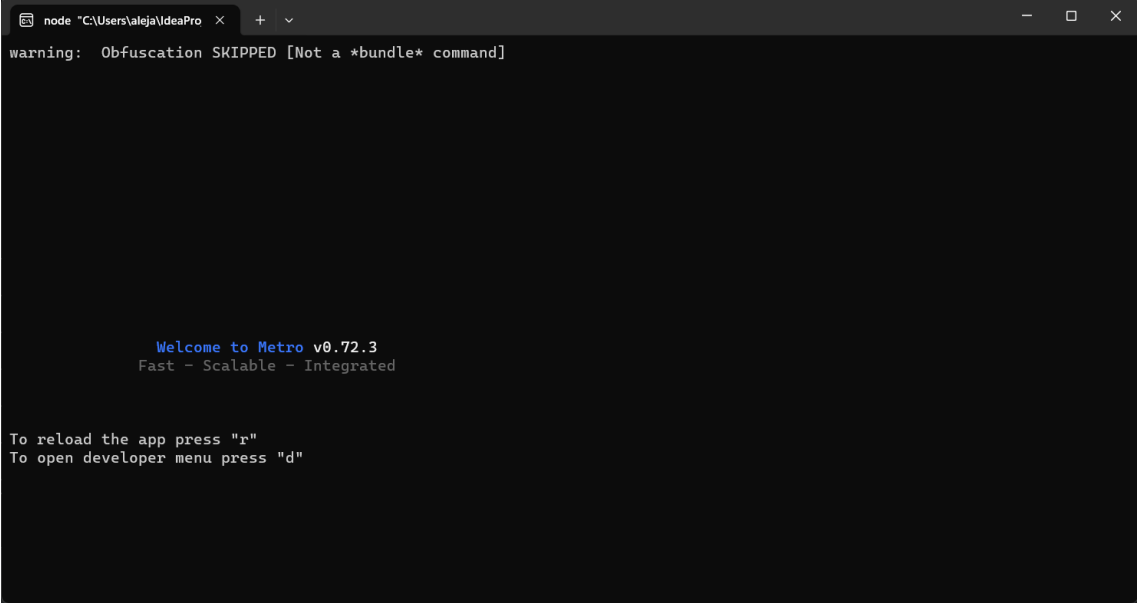
Abrir el AndroidStudio y ejecutar un simulador

Abrir un terminal en la raíz del proyecto y escribir el siguiente comando

“npx react-native run-android”

Se abra la siguiente ventana:

- Terminal que compila el bundle de la aplicación de react native

A screenshot of a terminal window with a dark background. The title bar shows 'node "C:\Users\aleja\IdeaPro' and standard window controls. The terminal output includes a warning about obfuscation, a 'Welcome to Metro v0.72.3' message with the tagline 'Fast - Scalable - Integrated', and instructions to press 'r' to reload the app and 'd' to open the developer menu.

```
node "C:\Users\aleja\IdeaPro" warning: Obfuscation SKIPPED [Not a *bundle* command]

Welcome to Metro v0.72.3
Fast - Scalable - Integrated

To reload the app press "r"
To open developer menu press "d"
```

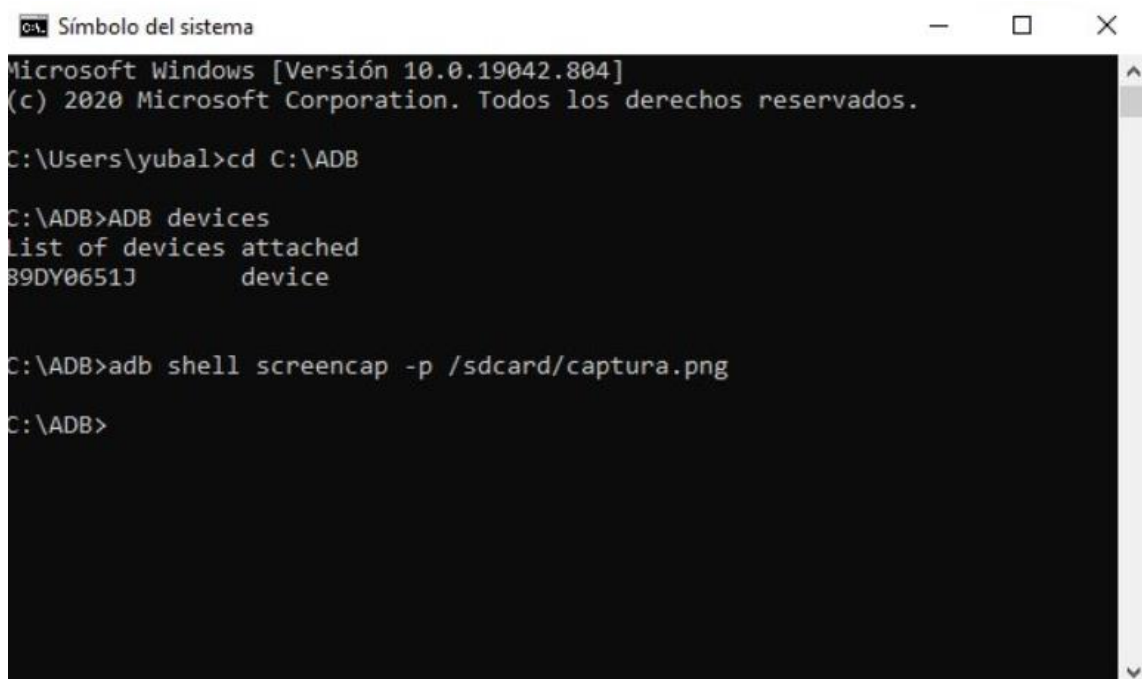
En el emulador de Android se abrirá la aplicación

Ejecución en un dispositivo físico:

Ejecutar el comando:

“adb devices”

Se listarán los dispositivos conectados



```
Símbolo del sistema
Microsoft Windows [Versión 10.0.19042.804]
(c) 2020 Microsoft Corporation. Todos los derechos reservados.

C:\Users\yubal>cd C:\ADB

C:\ADB>ADB devices
List of devices attached
89DY0651J device

C:\ADB>adb shell screencap -p /sdcard/captura.png

C:\ADB>
```

Copiar el id del device y correr el siguiente comando (ejemplo para la pantalla mostrada, reemplazar el id por el de su dispositivo):

“npx react-native run-android --deviceId=89DY0651J”

En cualquier caso la aplicación se abrirá ya sea en el emulador o el dispositivo físico:

BanBif

Empresas



Coopaci



Mherrera



Contraseña

Ingresar



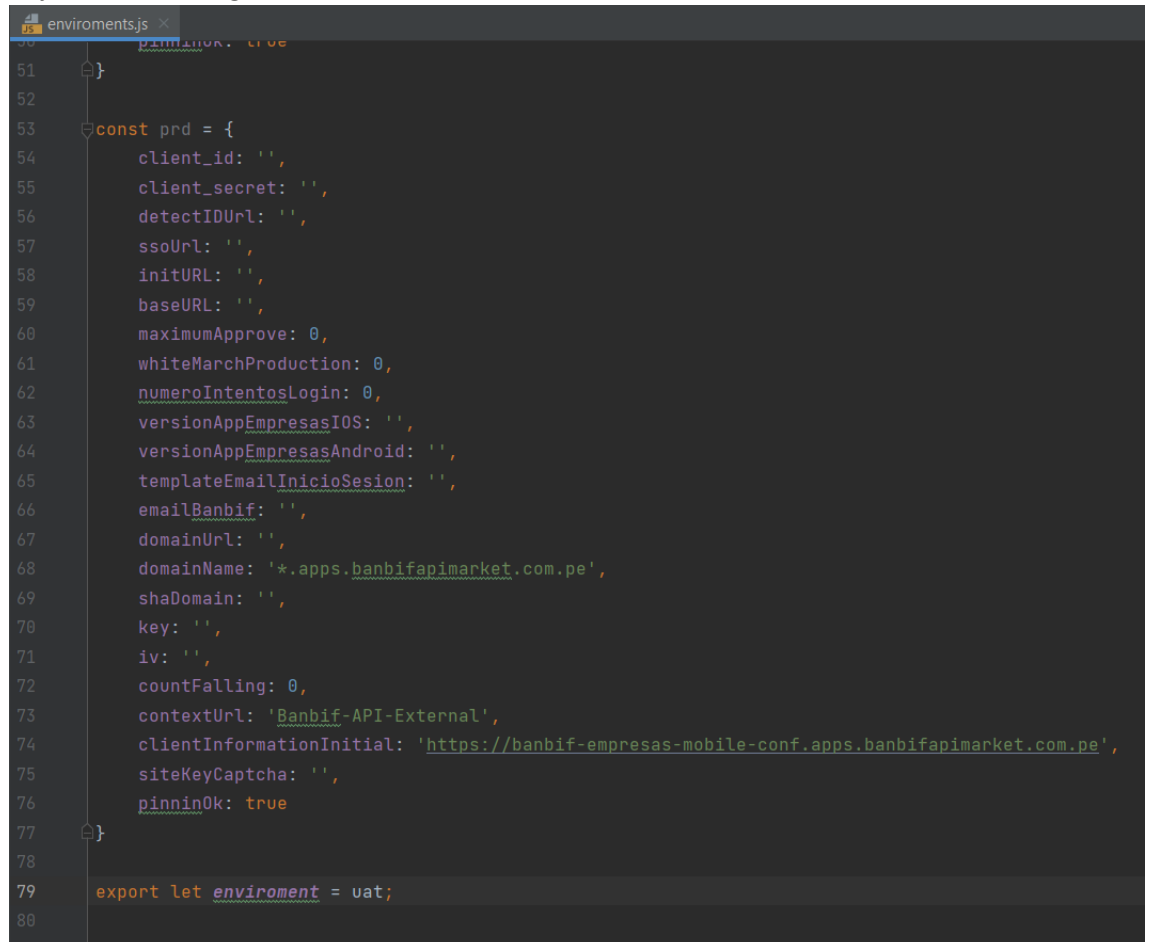
Recordarme



Token Digital

Para la generación del instalador de la aplicación, considerar los siguientes pasos comunes:

- Revisar el Branch que se quiere usar
- Hacer pull para obtener los últimos cambios
- Revisar archivo “src/utls/enviroments.js” y ver que en la línea 79 se exporte el objeto correcto según el ambiente



```
51 }
52
53 const prd = {
54   client_id: '',
55   client_secret: '',
56   detectIDUrl: '',
57   ssoUrl: '',
58   initURL: '',
59   baseUrl: '',
60   maximumApprove: 0,
61   whiteMarchProduction: 0,
62   numeroIntentosLogin: 0,
63   versionAppEmpresasIOS: '',
64   versionAppEmpresasAndroid: '',
65   templateEmailInicioSesion: '',
66   emailBanbif: '',
67   domainUrl: '',
68   domainName: '*.apps.banbifapimarket.com.pe',
69   shaDomain: '',
70   key: '',
71   iv: '',
72   countFalling: 0,
73   contextUrl: 'Banbif-API-External',
74   clientInformationInitial: 'https://banbif-empresas-mobile-conf.apps.banbifapimarket.com.pe',
75   siteKeyCaptcha: '',
76   pinninOk: true
77 }
78
79 export let environment = uat;
80
```

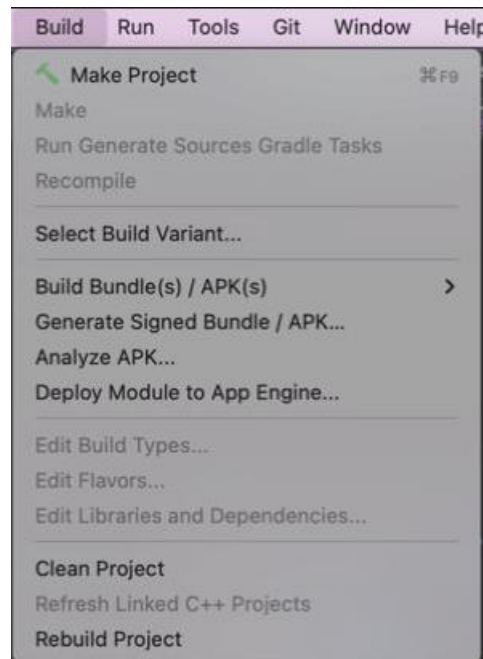
- Si es para UAT exportar “uat”
- Si es para PRODUCCION exportar “prd”
- Si es para DEV exportar “dev”

a. Android

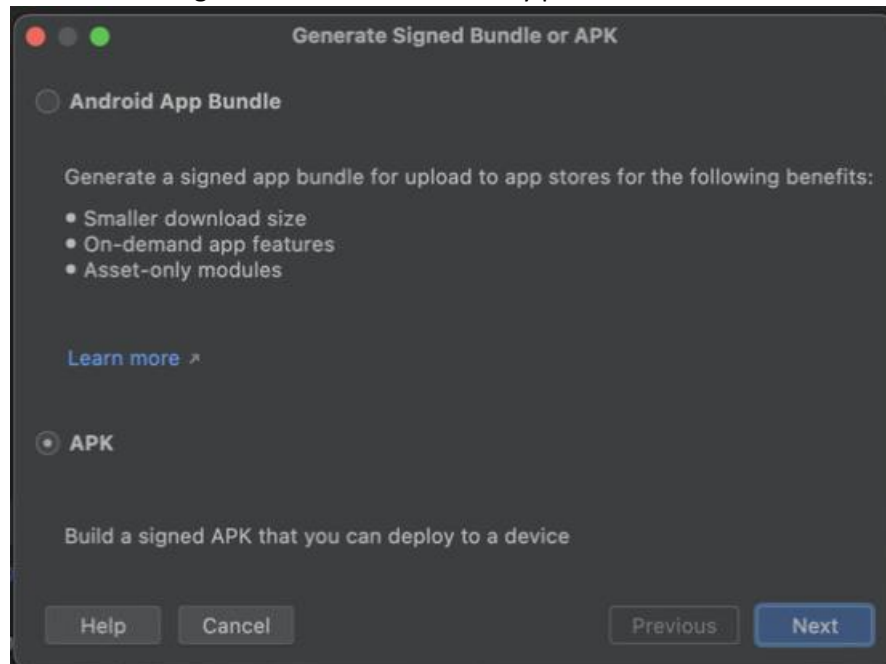
- I. Ir al archivo “Login.js” y verificar en el método “configuracion” que la variable “environment.detectIDUrl” se encuentre con el siguiente valor (Solo para UAT):

```
environment.detectIDUrl = configuration.clientDetectIdUrl
```

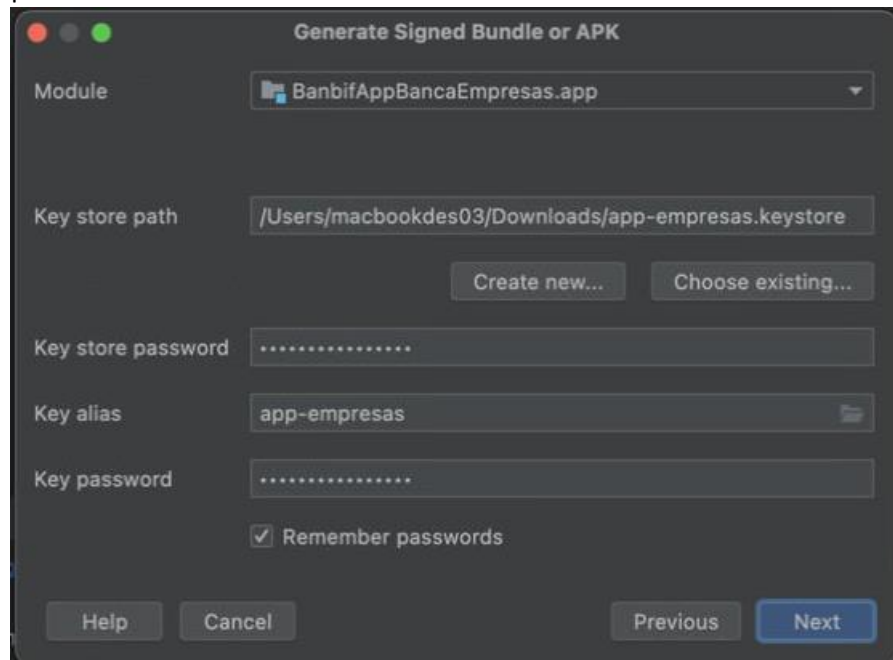
- II. Ingresar a AndroidStudio y ir al menú “Build > Generate Signed Bundle / APK ...”



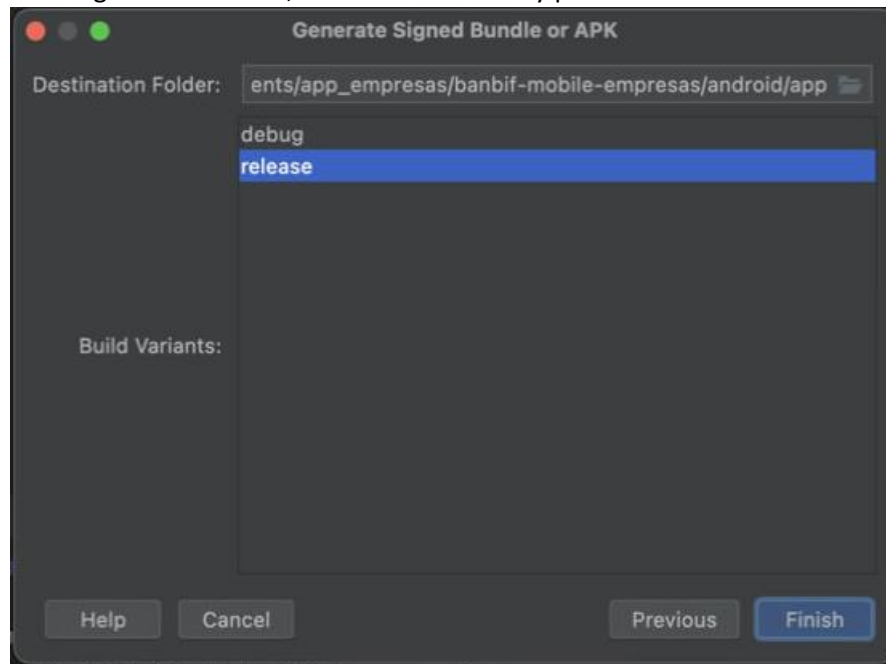
- III. En el modal de generación seleccionar APKy presionar Next



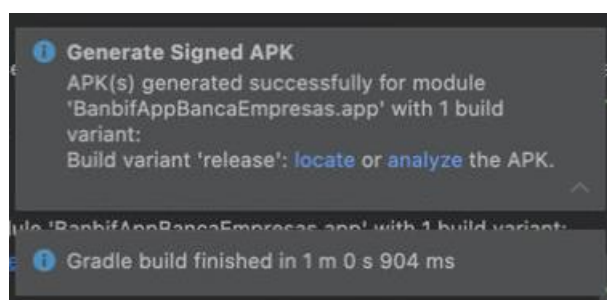
- IV. En la siguiente ventana, verificar que las credenciales estén colocadas y presionar Next



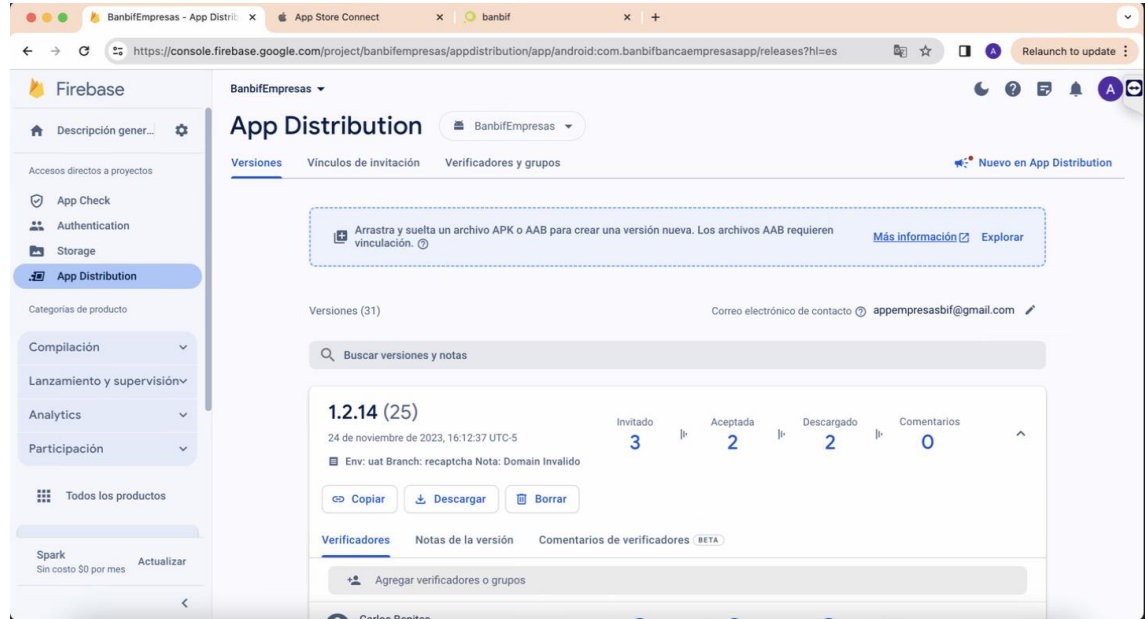
- V. En la siguiente ventana, seleccionar reléase y presionar Finish



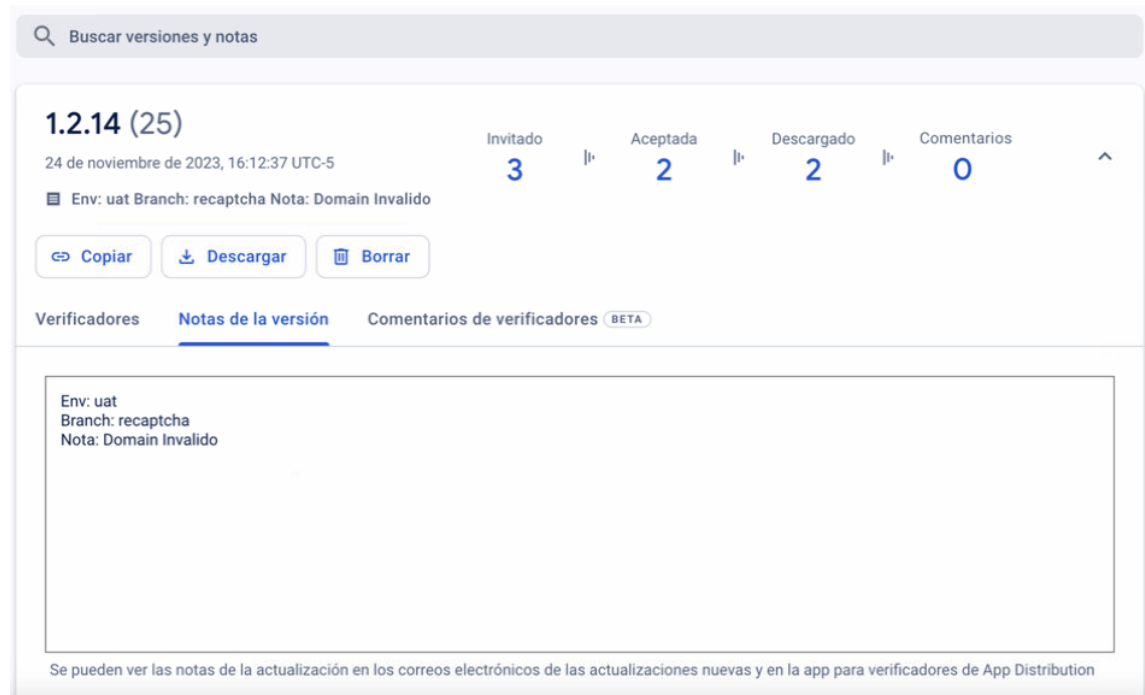
- VI. Esperar unos minutos que el proceso de build termine, se mostrara un mensaje de confirmación. Presionar el link "locate" y nos abrirá la carpeta donde se encuentra el apk.



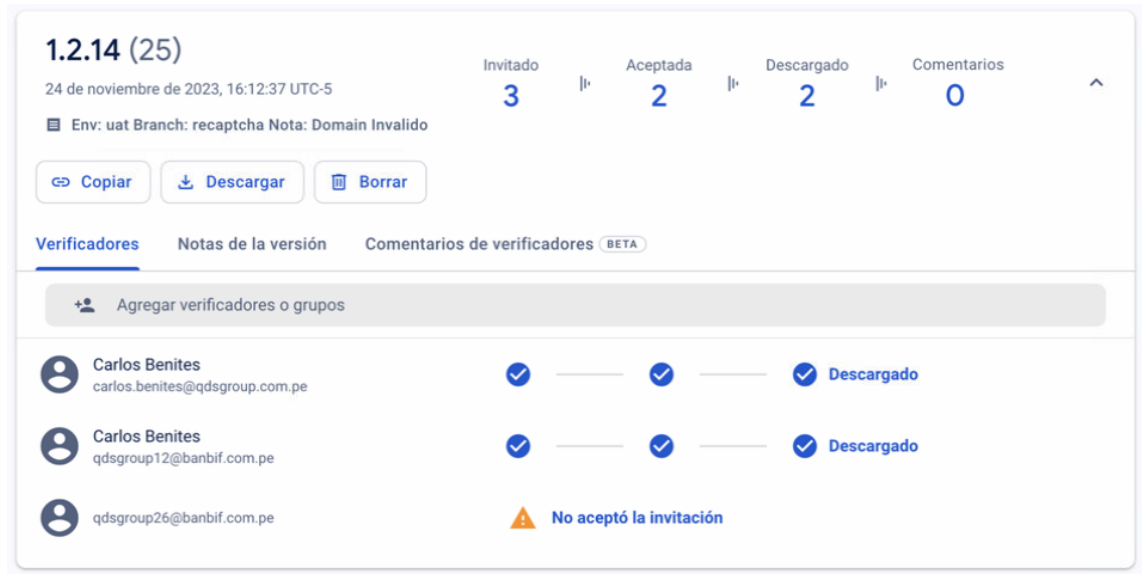
VII. Abrir Firebase App Distribution en el navegador y arrastras el apk



VIII. Luego que termine la carga del apk ingresar notas del ambiente y los cambios realizados



- IX. Por ultimo agregar los correos electrónicos de los verificadores para que se distribuya el apk por email

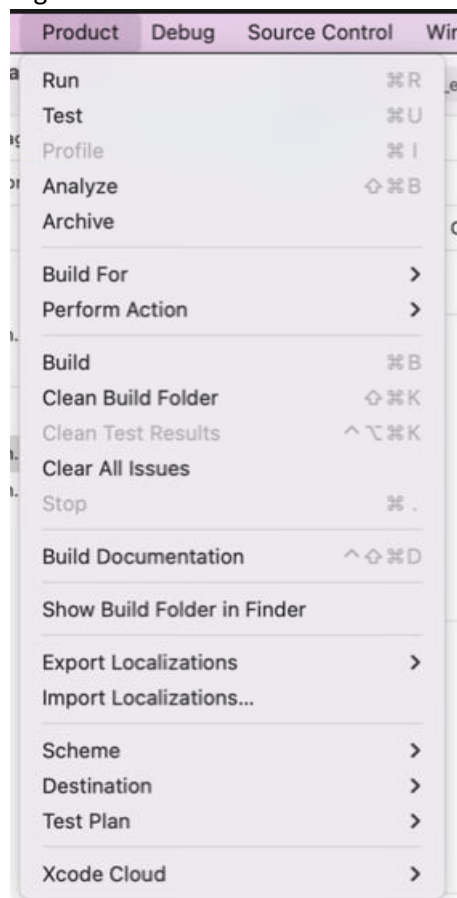


b. IOs

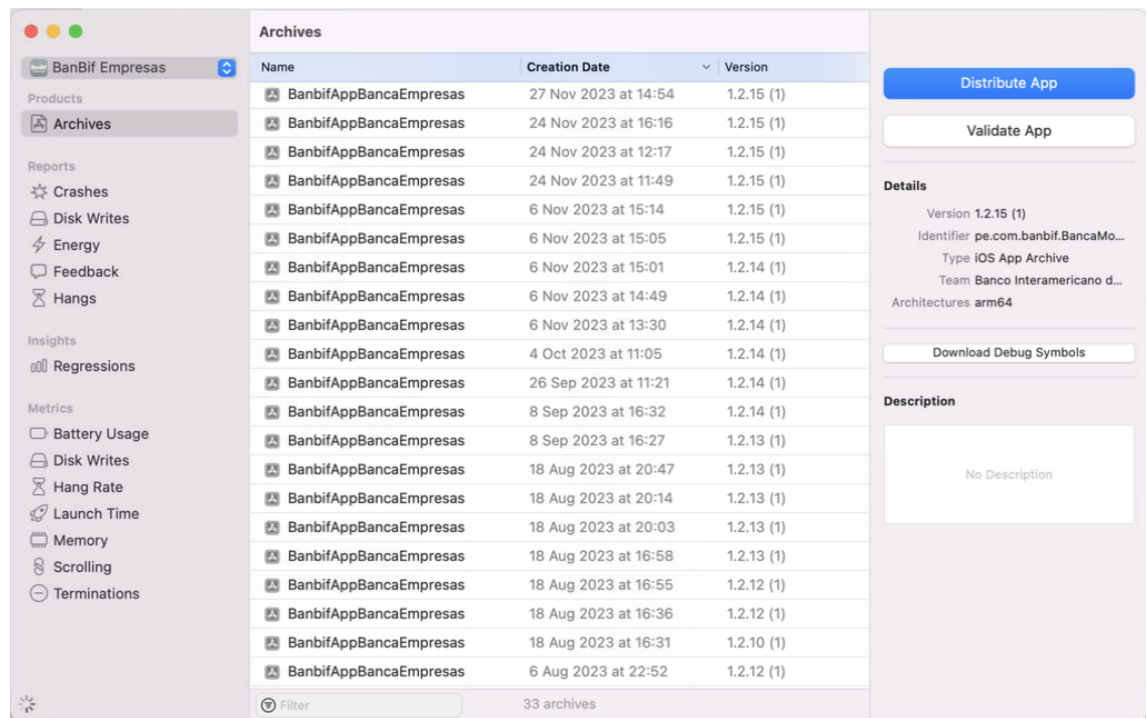
- I. Ir al archivo "Login.js" y verificar en el método "configuracion" que la variable "enviroment.detectIDUrl" se encuentre con el siguiente valor (Solo para UAT):

```
enviroment.detectIDUrl = 'http://bif3sms10:8080/detect/public/registration/mobileServices.htm?code='
```

- II. Ingresar a xCode e ir al menu "Product > Archive "

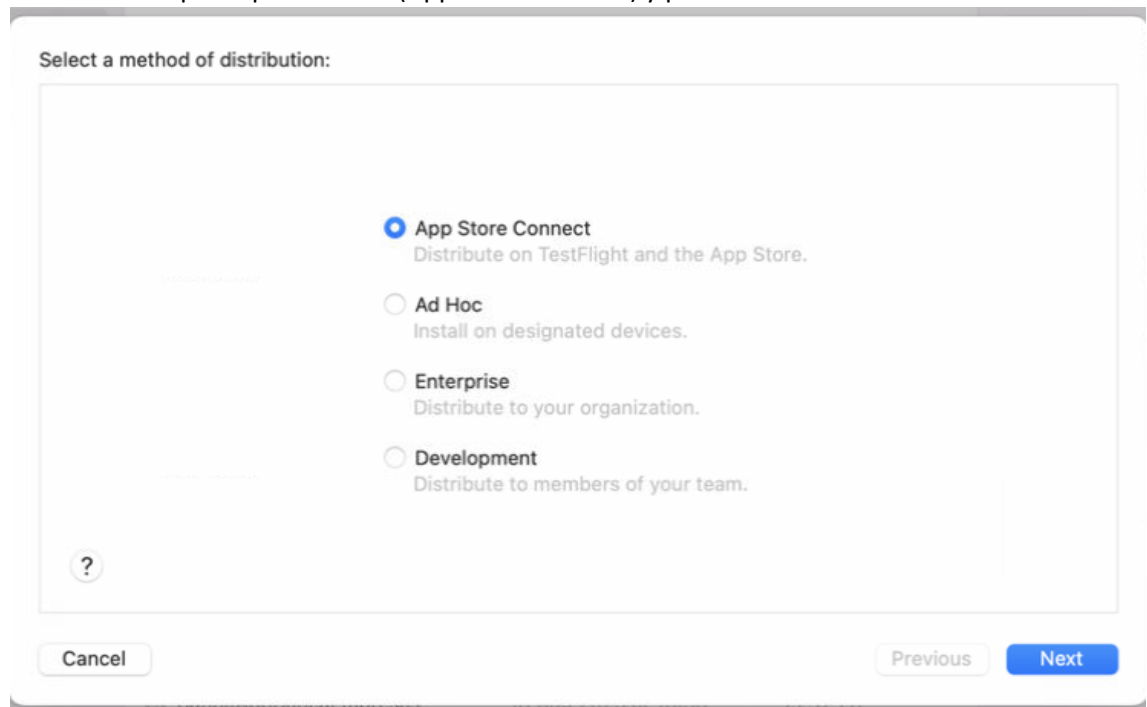


- III. Esperar que termine la compilación, posteriormente aparecer la siguiente ventana donde

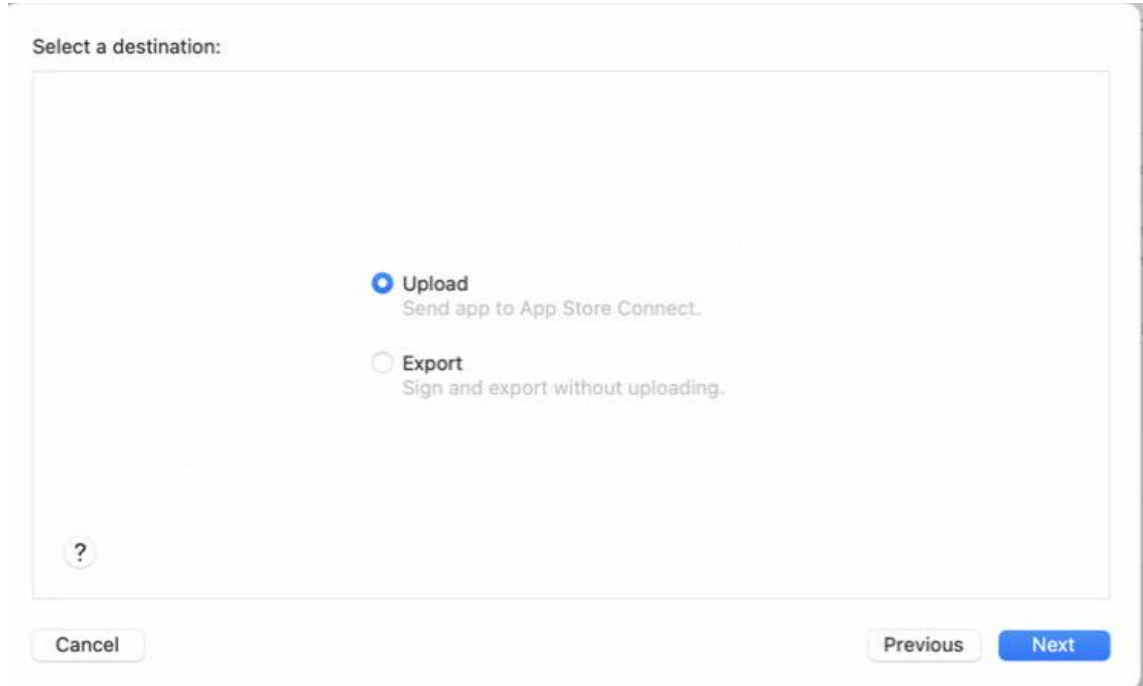


debemos presionar el botón “Distribute App”

IV. Mantener la opción por defecto (App Store Connect) y presionar Next



- V. Mantener la opción por defecto (Upload) y presionar Next



Select a destination:

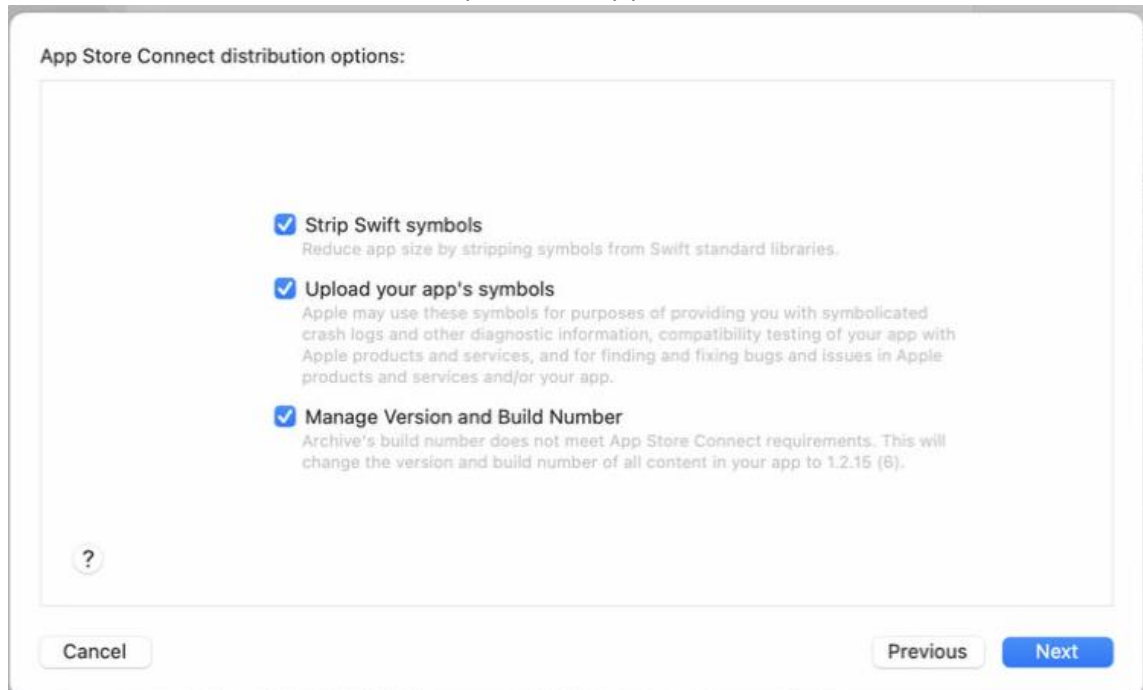
☒ **Upload**
Send app to App Store Connect.

☐ **Export**
Sign and export without uploading.

?

Cancel Previous Next

- VI. Mantener los 3 checkboxes marcados por defecto y presionar Next



App Store Connect distribution options:

☒ **Strip Swift symbols**
Reduce app size by stripping symbols from Swift standard libraries.

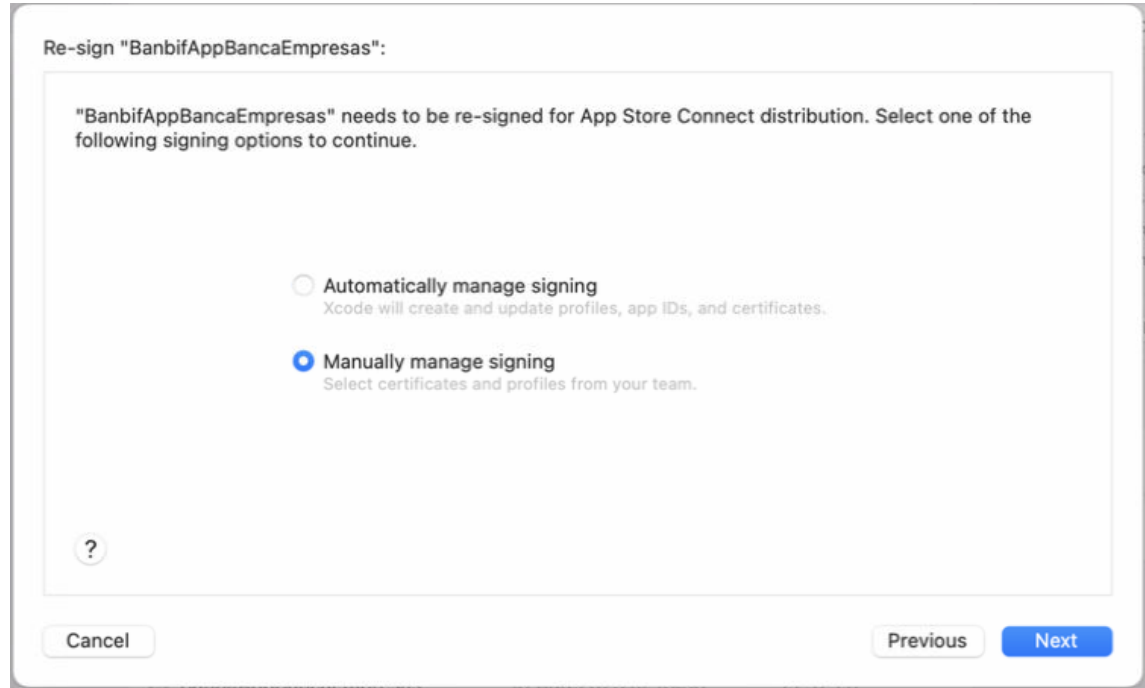
☒ **Upload your app's symbols**
Apple may use these symbols for purposes of providing you with symbolicated crash logs and other diagnostic information, compatibility testing of your app with Apple products and services, and for finding and fixing bugs and issues in Apple products and services and/or your app.

☒ **Manage Version and Build Number**
Archive's build number does not meet App Store Connect requirements. This will change the version and build number of all content in your app to 1.2.15 (6).

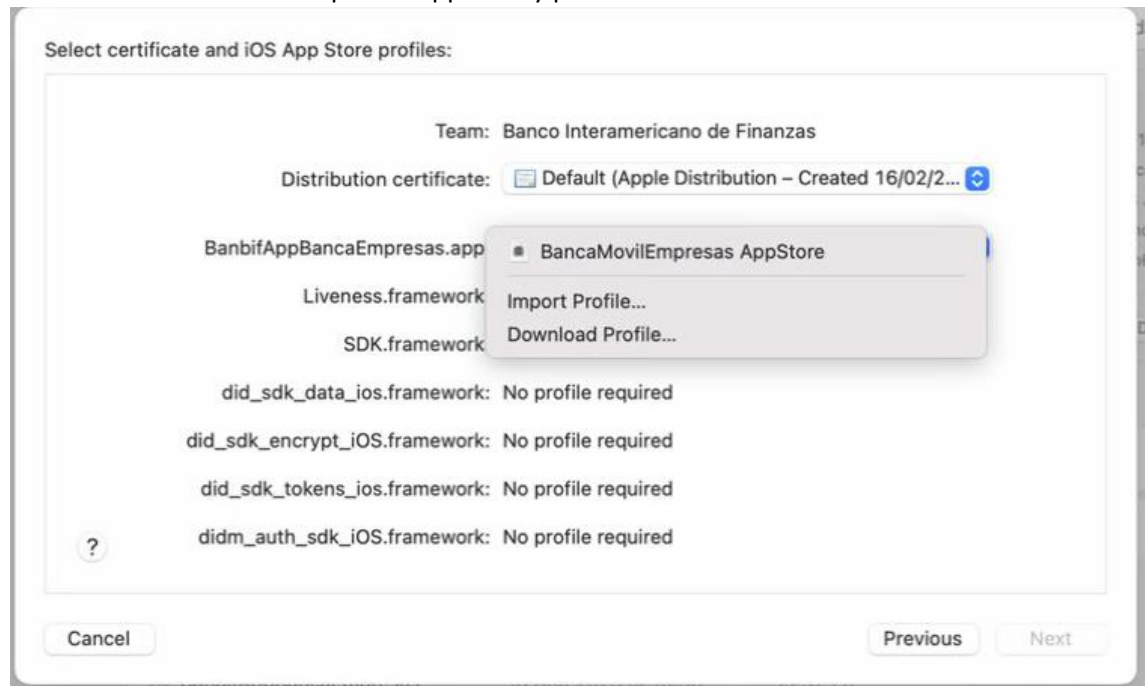
?

Cancel Previous Next

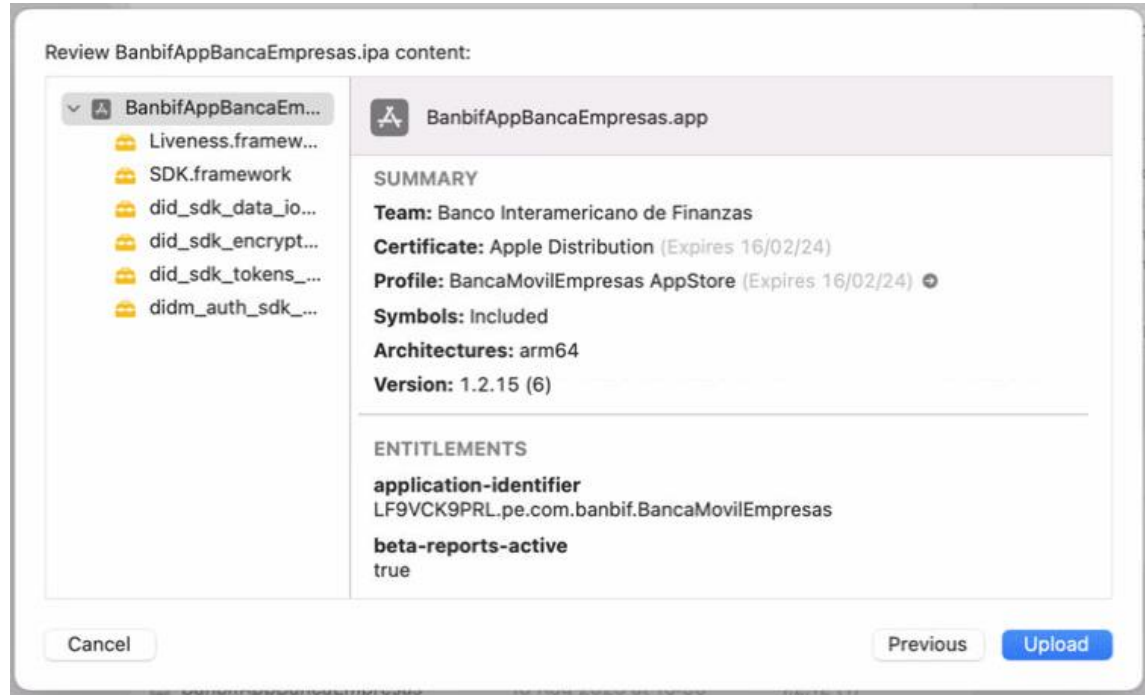
- VII. Seleccionar la segunda opcion "Manually manage signing" y presionar Next



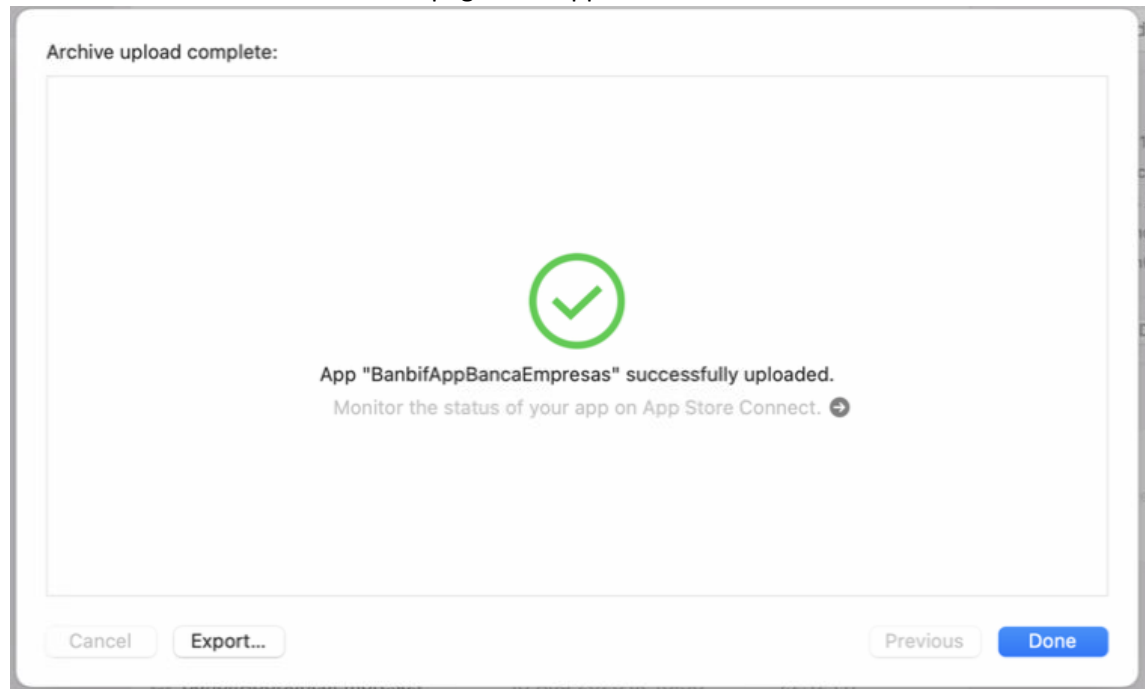
- VIII. Seleccionar BancaMovilEmpresas AppStore y presionar Next



IX. Presionar el botón Upload



X. El instalador habrá sido subido a la página de App Store Connect



XI. En la seccion Test Flight se puede ver la version subida

Compilaciones

IOS

Comentarios

Errores

Capturas de pantalla

Pruebas internas

App Store Connect Users

Pruebas externas

test

Testers App Empresas

Información general

Todos tus testers

Información para las pruebas

Acerca de los datos de TestFlight

Compilaciones para iOS

Estas son las compilaciones que pueden probarse. Más información sobre indicadores y estados de las compilaciones

✓ Versión 1.2.15

COMPILACIÓN	ESTADO	GRUPOS	INVITACIONES	INSTALACIONES	SESIONES	ERRORES	COMENTARIOS
6	(En proceso)						
5	Lista para enviar Caduca en 84 días	AC	4	3	43	-	-
4	Lista para enviar Caduca en 81 días	AC	4	2	-	-	-
3	Lista para enviar Caduca en 80 días	AC	4	2	-	-	-
2	Lista para enviar Caduca en 80 días	AC	4	2	-	6	1
1	Lista para enviar Caduca en 63 días	AC	4	1	-	-	-

> Versión 1.2.14

XII. Cuando la compilacion termine el proceso presionar Gestionar en la columna estado

✓ Versión 1.2.15

COMPILACIÓN	ESTADO	GRUPOS	INVITACIONES	INSTALACIONES	SESIONES	ERRORES	COMENTARIOS
6	Falta información de... Gestionar						
5	Lista para enviar Caduca en 84 días	AC	4	3	43	-	-
4	Lista para enviar		4	2			

XIII. Selección la ultima opcion y presionar Guardar

Información sobre el cumplimiento de las leyes de exportación

¿Qué tipo de algoritmos de encriptación implementa tu app?

☐ Algoritmos de encriptación propietarios o no aceptados como estándar por los organismos internacionales de normalización (IEEE, IETF, ITU, etc.).

☐ Algoritmos de encriptación estándar en lugar de, o además de, utilizar o acceder a la encriptación del sistema operativo de Apple.

☐ Los dos tipos de algoritmos mencionados anteriormente.

☒ Ninguno de los algoritmos mencionados anteriormente.

i

Es tu responsabilidad cumplir con las leyes de exportación. Si cambia tu encriptación o el cumplimiento de los criterios de exención, consulta de nuevo estas preguntas. Dado que tu compilación no contiene encriptación, puedes indicarlo en la lista de propiedades de información (Info.plist) de tu proyecto de Xcode para así no tener que responder a preguntas sobre la encriptación con cada envío de la app. Más información

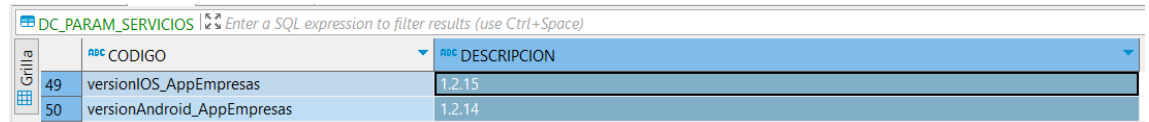
Cancelar

Guardar

9. Versionamiento de la aplicación

Para actualizar la versión de la aplicación:

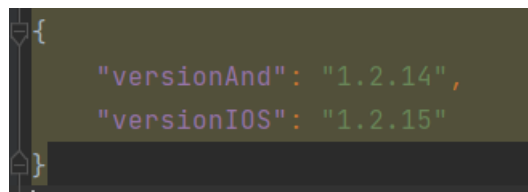
- Actualizar en la base de datos, la tabla “DC_PARAM_SERVICIOS” los registros “versionIOS_AppEmpresas” y “versionAndroid_AppEmpresas” Con la finalidad de ver el modal de actualización en la app.



DC_PARAM_SERVICIOS

	CODIGO	DESCRIPCION
49	versionIOS_AppEmpresas	1.2.15
50	versionAndroid_AppEmpresas	1.2.14

- Actualizar en el archivo “version.json”



```
{  
  "versionAnd": "1.2.14",  
  "versionIOS": "1.2.15"  
}
```

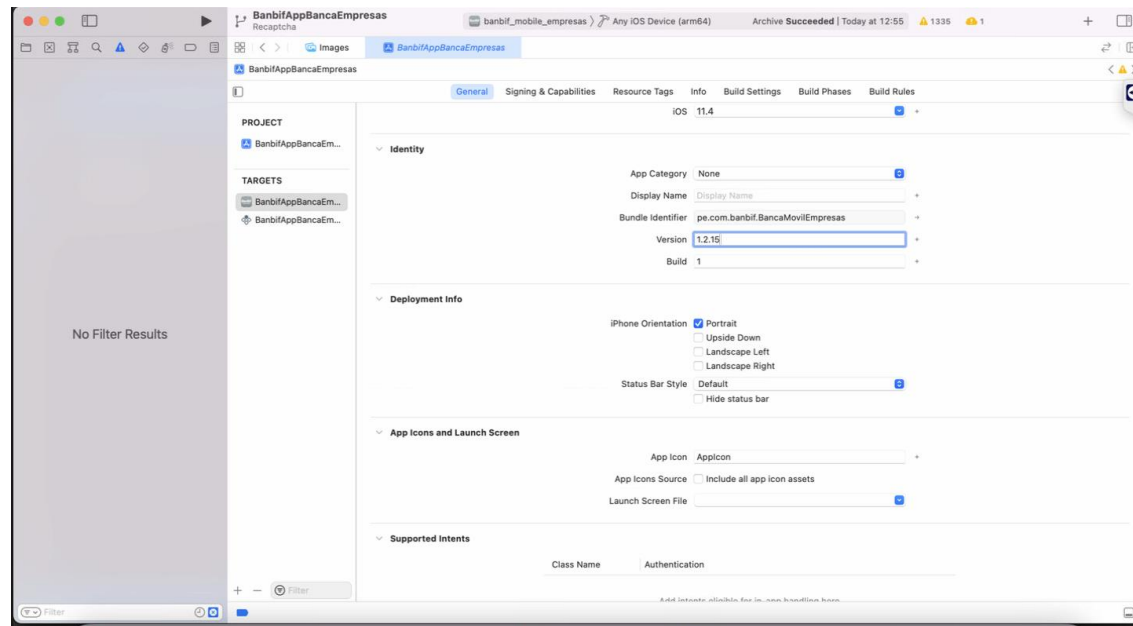
- Para Android:
 - o Dirigirse al archivo “android/app/build.gradle” y cambiar el “versionCode” y “versionName”



```
app\build.gradle  
140 defaultConfig {  
141     applicationId "com.banbifbancaempresasapp"  
142     minSdkVersion rootProject.ext.minSdkVersion  
143     targetSdkVersion rootProject.ext.targetSdkVersion  
144     versionCode 25  
145     versionName "1.2.14"  
146     buildConfigField "boolean", "IS_NEW_ARCHITECTURE_ENABLED", isNewArchitectureEnabled()  
147  
148     if (isNewArchitectureEnabled()) {  
149         // We configure the CMake build only if you decide to opt-in for the New Architecture  
150         externalNativeBuild {  
151             cmake {  
152                 cppFlags "-DVERSION_MAJOR=1 -DVERSION_MINOR=2 -DVERSION_PATCH=14"  
153             }  
154         }  
155     }  
156 }
```

- Para iOS

- Dirigirse a XCode y bajo la vista del proyecto, sección General, cambiar la versión



10. Consideraciones adicionales para el desarrollo

- En caso de agregar o modificar dependencias de node, antes de generar el instalador para iOS, abrir un terminal en la ruta ios del proyecto y ejecutar el comando “pod install”
- Carpeta donde se encuentran los correos para enrolamiento y aprobación de operaciones (**La dirección de correo del usuario tiene que tener el dominio @banbif.com.pe para recibir el email**):
 - \\10.30.30.2\mailroot\Drop
 - Solicitar las credenciales para autenticación
- Log de servicios (loguearse en RH-SSO)
 - <https://console-openshift-console.apps.ocpuat01.dombif.peru>
 - Solicitar las credenciales para autenticación
- Para probar el enrolamiento al token digital
 - En la tabla DC_USRPROFILE ubicar un registro con role 1 o 2
 - Cambiar el dni (con la finalidad de no tener el mensaje máximo de dispositivos enrolados)
 - Settear la columna “flagsofttoken” en Y
 - Settear la columna “flagtoken” en N
- Si un usuario se bloquea por número de reintentos fallidos
 - En la tabla DC_USRPROFILE ubicar un registro
 - Settear la columna “sts” en 0
 - Settear la columna “invalid” en 0
- Canal web, el ingreso se realiza con las mismas credenciales que para la app empresas
 - https://bancaporinternetuat.dombif.peru/DCIBS_BIFNET/pages/s/login.html