



# MANUAL DE INSTALACIÓN

DESARROLLO DE SISTEMA WEB Y APLICACIÓN  
MÓVIL PARA CONSULTAR LA UBICACIÓN DE  
LAS CLÍNICAS VETERINARIAS EN LA CIUDAD  
DE QUITO

Carlos Lucero – Omar Espín  
ESFOT

# ÍNDICE DE CONTENIDOS

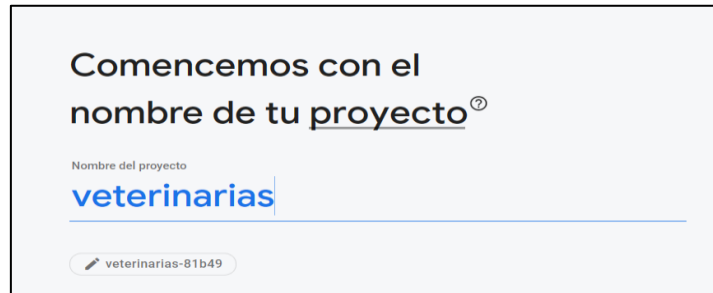
1. Despliegue del Sistema Web en Firebase. ....	2
1.1. Creación de un proyecto nuevo en Firebase .....	2
1.2. Conexión de Firebase con Angular .....	2
1.3. Configuración de las librerías en Angular.....	2
1.4. Configuración del Hosting .....	3
1.5. Instalación de las Herramientas de Firebase.....	3
1.6. Inicializar el proyecto para configurar el Hosting.....	3
1.7. Visualización del Sistema Web .....	5
2. Generar APK de la Aplicación Móvil. ....	5
2.1. Construcción del proyecto.....	5
2.2. Integración del framework Capacitor con Android .....	5
2.3. Abrir el proyecto con la ayuda de Android Studio .....	6
2.4. Generar una APK firmada.....	6
2.5. Despliegue de la Aplicación Móvil en la <i>Play Store</i> .....	8
3. Credenciales de acceso. ....	8
3.1. Sistema Web .....	8
3.2. Aplicación Móvil .....	8
4. Repositorio del Código Fuente .....	8

## 1. Despliegue del Sistema Web en Firebase.

A continuación, se describe los pasos a seguir para la puesta a producción del sistema web en el hosting de *Firebase*.

### 1.1. Creación de un proyecto nuevo en Firebase

Ingresar a *Firebase* y crear un nuevo proyecto como se muestra en la **Fig. 1**



**Fig. 1** Creación de un nuevo proyecto en Firebase

### 1.2. Conexión de Firebase con Angular

En la **Fig. 2** se muestra la configuración de las variables de entorno en el proyecto de Angular utilizando las credenciales proporcionadas por *Firebase*, dichos cambios se lo realiza en el archivo ***enviroment.prod.ts***.

```
export const environment = {  
  firebase: {  
    projectId: 'veterinarias-1c7dc',  
    appId: '1:607408468635:web:4149e8db36a58df5bc4c3e',  
    storageBucket: 'veterinarias-1c7dc.appspot.com',  
    apiKey: 'AIzaSyCmSH4gBpyqKBT-j_FXy6_lj6Hypz1mirE',  
    authDomain: 'veterinarias-1c7dc.firebaseio.com',  
    messagingSenderId: '607408468635',  
  },  
  production: true  
};
```

**Fig. 2** Configuración de variables de entorno en Angular

### 1.3. Configuración de las librerías en Angular

Se procede a realizar la conexión entre *Firebase* y Angular con la importación de los módulos necesarios en el archivo ***app.module.ts*** como se muestra en la **Fig.**

**3.**

```
imports: [  
  BrowserModule,  
  AppRoutingModule,  
  AngularFireAuthModule,  
  AngularFireModule.initializeApp(environment.firebase),  
  AngularFireStoreModule,  
  AngularFireStorageModule,  
  LoginModule,  
  DefaultModule,  
  AppMaterialModule,  
  BrowserAnimationsModule,  
  ReactiveFormsModule
```

**Fig. 3** Módulos Importados

## 1.4. Configuración del Hosting

En la Fig. 4 se muestra el ingreso a la plataforma de *Firebase* para realizar el proceso de despliegue del sistema web.

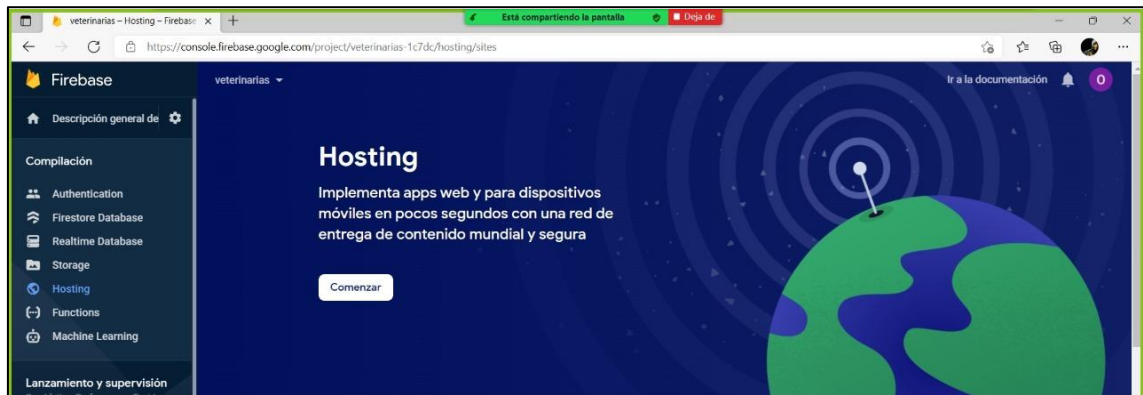


Fig. 4 Configuración del Hosting

## 1.5. Instalación de las Herramientas de Firebase

En el proyecto de Angular se abre una terminal y se ejecuta el siguiente comando, como se muestra en la Fig. 5.

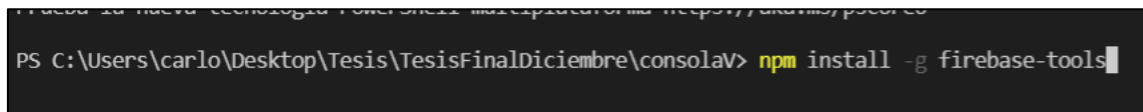


Fig. 5 Instalación de firebase-tools.

Al terminar la instalación de *firebase-tools*, se procede a realizar el enlace de la cuenta con la cuenta de Google como se muestra en la Fig. 6.



Fig. 6 Inicio de sesión de Google con firebase login.

## 1.6. Inicializar el proyecto para configurar el Hosting

Para inicializar el proyecto se debe ejecutar el comando **firebase init** con el que se va a realizar las configuraciones para el despliegue del sistema web. A continuación, se digita la letra "Y", después se selecciona la opción *hosting*: *configure files for Firebase Hosting and set up GitHub Action deploys* como se muestra en la Fig. 7.

```

##### 0000 ##### 00000000 ##### 0000 #####
## 00 00 00 00 00 00 00 00 00 00 00 00
##### 00 00000000 ##### 00000000 ##### 00000000 #####
## 00 00 00 00 00 00 00 00 00 00 00 00
## 0000 00 00 00000000 ##### 00 00 00000000 #####

You're about to initialize a Firebase project in this directory:

C:\Users\carlo\Desktop\Tesis\TesisFinalDiciembre\consolaV

Before we get started, keep in mind:

* You are initializing within an existing Firebase project directory

? Are you ready to proceed? Yes
? Which Firebase features do you want to set up for this directory? Press Space to select features, then Enter to confirm your choices.
  ( ) Realtime Database: Configure a security rules file for Realtime Database and (optionally) provision default instance
  ( ) Firestore: Configure security rules and indexes files for Firestore
  ( ) Functions: Configure a Cloud Functions directory and its files
> (*) Hosting: Configure files for Firebase Hosting and (optionally) set up GitHub Action deploys
  ( ) Hosting: Set up GitHub Action deploys
  ( ) Storage: Configure a security rules file for Cloud Storage
  ( ) Emulators: Set up local emulators for Firebase products
(Move up and down to reveal more choices)

```

Fig. 7 Configuración de Hosting.

Usar un directorio público que se encuentra por defecto, en las siguientes opciones poner la letra “N” para que no realice más configuraciones, como se muestra en la Fig. 8.

```

? What do you want to use as your public directory? public
? Configure as a single-page app (rewrite all urls to /index.html)? No
? Set up automatic builds and deploys with GitHub? No
+ Wrote public/404.html
+ Wrote public/index.html

i Writing configuration info to firebase.json...
i Writing project information to .firebaserc...

+ Firebase initialization complete!
PS C:\Users\carlo\Desktop\Tesis\TesisFinalDiciembre\consolaV>

```

Fig. 8 Directorio Público

Con el comando **ng build** se procederá a generar las carpetas necesarias para el despliegue del sistema web como se muestra en la Fig. 9.

```

+ Firebase initialization complete!
PS C:\Users\carlo\Desktop\Tesis\TesisFinalDiciembre\consolaV> ng build
✓ Browser application bundle generation complete.

Initial Chunk Files | Names | Size
main.2427709cf61a84014a94.js | main | 1.38 MB
styles.7fe5d3b80dad7086b067.css | styles | 71.60 kB
polyfills.91b950cb41fc48c02171.js | polyfills | 36.20 kB
runtime.60464eddec9dbde1afc4.js | runtime | 1.46 kB
| Initial Total | 1.48 MB

Build at: 2021-12-13T18:42:46.260Z - Hash: 9d84cfa02a22fadd21f2 - Time: 48563ms
Warning: initial exceeded maximum budget. Budget 500.00 kB was not met by 1019.53 kB with a total of 1.48 MB.

```

Fig. 9 Comando ng build.

A continuación, para realizar el despliegue correcto del sistema web se debe ingresar el comando **firebase deploy --only hosting**, como se muestra en la Fig. 10.

```

PS C:\Users\carlo\Desktop\Tesis\TesisFinalDiciembre\consolaV> firebase deploy --only hosting

=== Deploying to 'veterinarias-1c7dc'...

i deploying hosting
i hosting[veterinarias-1c7dc]: beginning deploy...
i hosting[veterinarias-1c7dc]: found 2 files in public
+ hosting[veterinarias-1c7dc]: file upload complete
i hosting[veterinarias-1c7dc]: finalizing version...
+ hosting[veterinarias-1c7dc]: version finalized
i hosting[veterinarias-1c7dc]: releasing new version...
+ hosting[veterinarias-1c7dc]: release complete

+ Deploy complete!

```

Fig. 10 Despliegue del sistema web

## 1.7. Visualización del Sistema Web

Al terminar con el proceso de despliegue se procede a acceder a la URL generada automáticamente en este caso es: <https://veterinarias-1c7dc.web.app> por medio de un navegador para comprobar el correcto funcionamiento del sistema web, como se muestra en la Fig. 11.

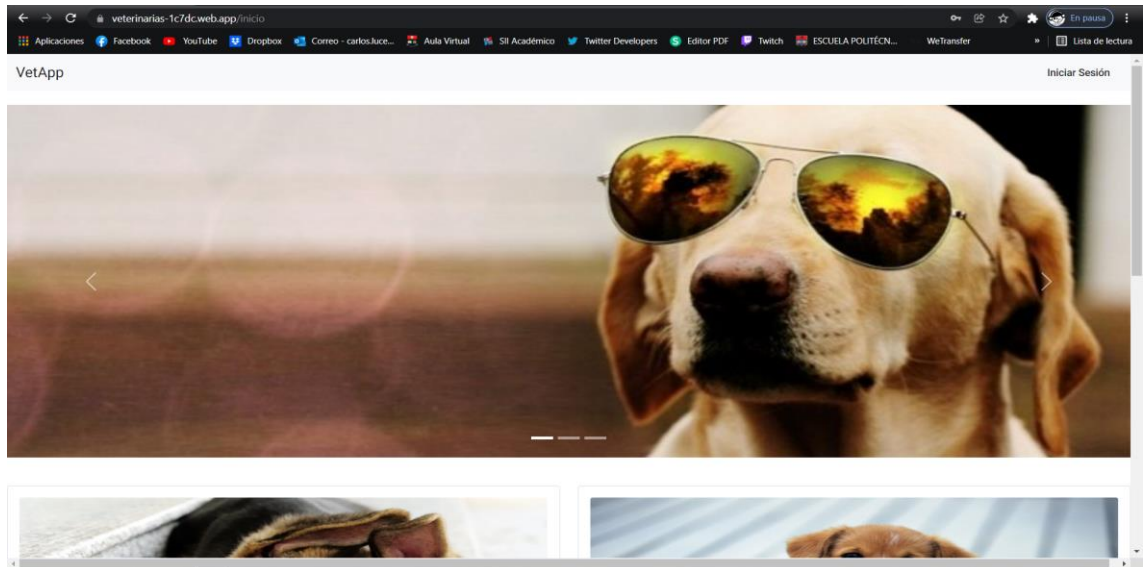


Fig. 11 Sistema Web Desplegado

## 2. Generar APK de la Aplicación Móvil.

### 2.1. Construcción del proyecto

Ejecutar el comando **ng build** para generar los complementos y librerías que necesitamos para la apk, como se muestra en la Fig. 12.

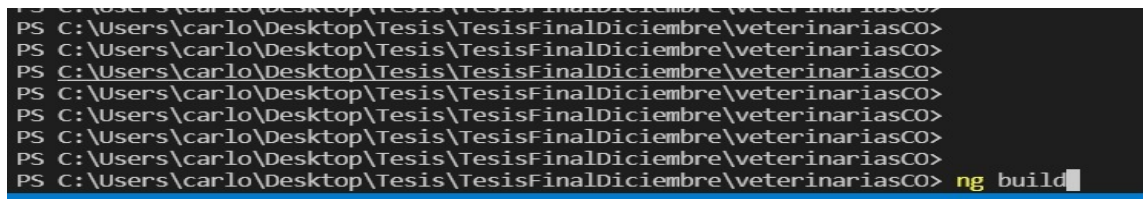


Fig. 12 Comando ng build

### 2.2. Integración del framework Capacitor con Android

En la Fig. 13 se muestra el comando que se debe utilizar para agregar Android al proyecto con la ayuda de capacitor para instalar los módulos necesarios para convertir al proyecto a Android.



Fig. 13 Integrar capacitor con Android

### 2.3. Abrir el proyecto con la ayuda de Android Studio

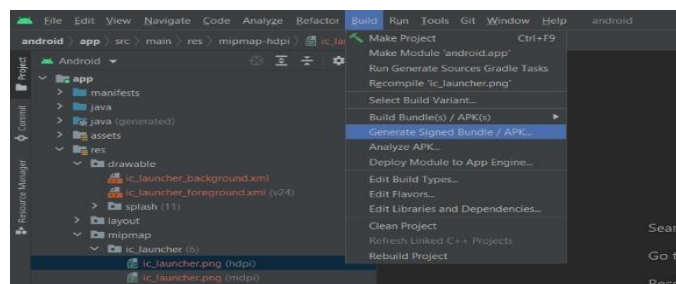
Una vez generada la carpeta de Android en el proyecto se procede a ejecutar el comando que se muestra en la **Fig. 14**, para que *Android Studio* abra el proyecto y genere las librerías necesarias.

```
PS C:\Users\carlo\Desktop\Tesis\TesisFinalDiciembre\veterinariasCO> npx cap open android
```

**Fig. 14** Abrir el proyecto en Android Studio.

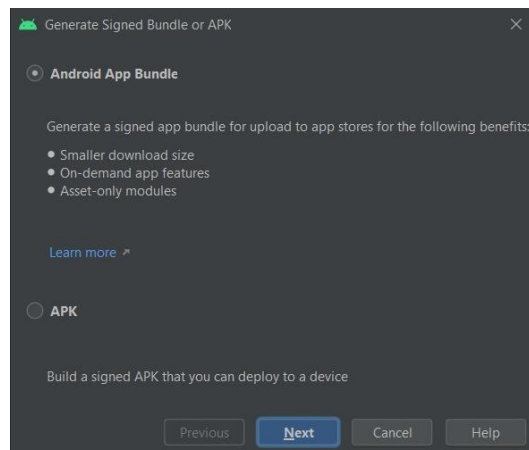
### 2.4. Generar una APK firmada

Una vez que el proyecto se ha abierto correctamente en *Android Studio* se procede a generar una APK Firmada como se muestra en la **Fig. 15**.



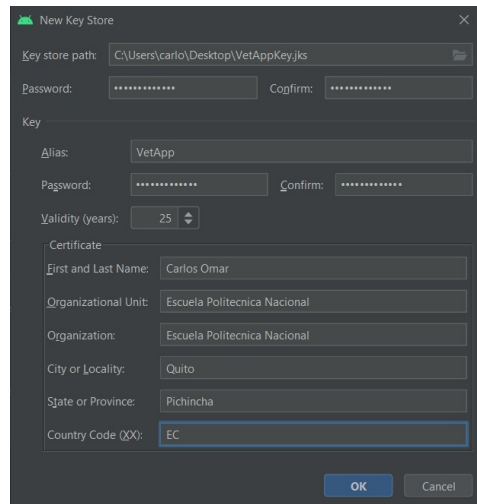
**Fig. 15** Paquete de Instalación Firmado

Realizado el paso anterior, en la ventana emergente se debe seleccionar la opción *Android App BundleK*, como se muestra en la **Fig. 16**.



**Fig. 16** Formato para el paquete de instalación

A continuación, se despliega un formulario el cual se lo debe llenar con la información del propietario de la APK para así poder generar el paquete de instalación, como se muestra en la **Fig. 17**.



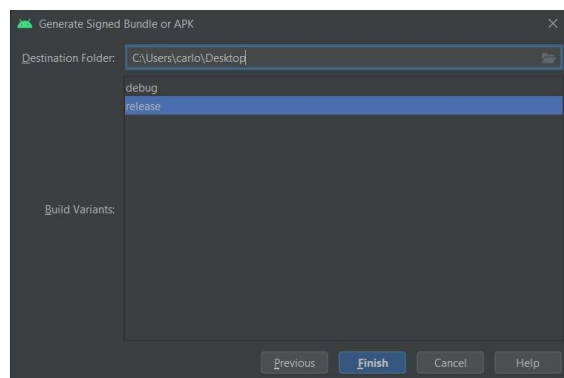
The 'New Key Store' dialog box is shown with the following fields:

- Key store path: C:\Users\carlo\Desktop\VetAppKey.jks
- Password: [masked]
- Confirm: [masked]
- Key Alias: VetApp
- Key Password: [masked]
- Key Confirm: [masked]
- Validity (years): 25
- Certificate Information:
  - First and Last Name: Carlos Omar
  - Organizational Unit: Escuela Politecnica Nacional
  - Organization: Escuela Politecnica Nacional
  - City or Locality: Quito
  - State or Province: Pichincha
  - Country Code (XX): EC

Buttons: OK, Cancel

**Fig. 17** Formulario para crear el Archivo de Firma

Por último, se selecciona la variante de despliegue que va a generar un paquete de instalación, seleccionar la variante **release** como se muestra en la **Fig. 18**.



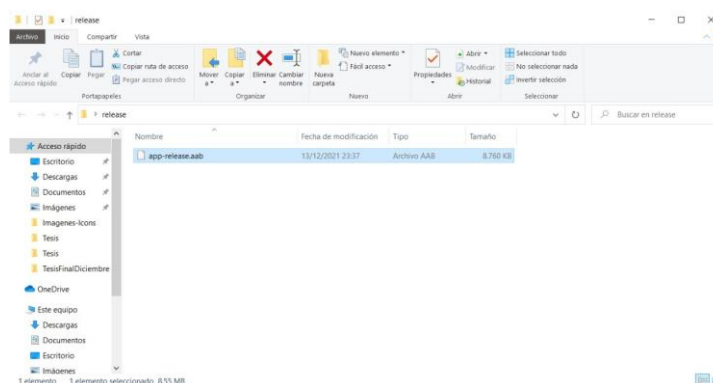
The 'Generate Signed Bundle or APK' dialog box is shown with the following fields:

- Destination Folder: C:\Users\carlo\Desktop
- Build Variants:
  - debug
  - release (selected)

Buttons: Previous, Finish, Cancel, Help

**Fig. 18** Seleccionar el despliegue que ha de tener la aplicación.

Una vez finalizado el proceso se va a crear un archivo **.abb** el cual es la aplicación que contiene la firma y la que se va a poder subir a la **Play Store** el archivo generado se muestra en la **Fig. 19**.

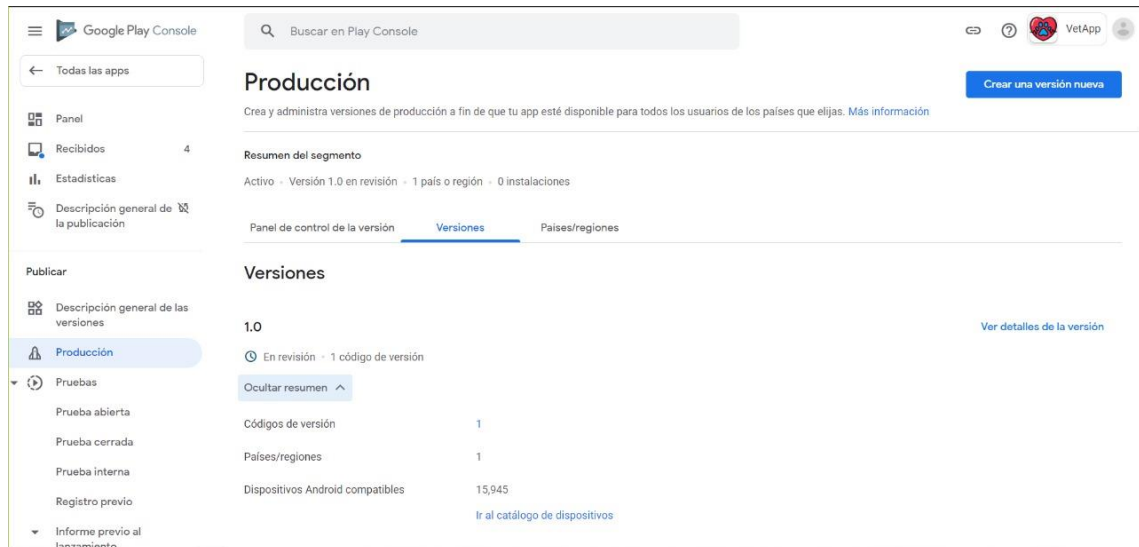


**Fig. 19** Paquete de instalación formato.abb



## 2.5. Despliegue de la Aplicación Móvil en la *Play Store*

En la **Fig. 20** se muestra el despliegue de la aplicación móvil en la *Play Store*, la cual está en un estado de revisión para su puesta en producción.



**Fig. 20** Aplicación Móvil desplegada en la *Play Store*.

## 3. Credenciales de acceso.

### 3.1. Sistema Web

Usuario: victoromarmh96@gmail.com

Contraseña: 123456789

### 3.2. Aplicación Móvil

No se requiere.

## 4. Repositorio del Código Fuente

- **Sistema Web**  
<https://github.com/CarlosIsma/TesisVetApp2021SW>
- **Aplicación Móvil**  
<https://github.com/CarlosIsma/TesisVetApp2021AM>