



# Manual de instalación y configuración

**Alexander Cabezas Medina**

06/2018

© Alexander Cabezas Medina

Reservados todos los derechos. Está prohibido la reproducción total o parcial de esta obra por cualquier medio o procedimiento, comprendidos la impresión, la reprografía, el microfilme, el tratamiento informático o cualquier otro sistema, así como la distribución de ejemplares mediante alquiler y préstamo, sin la autorización escrita del autor o de los límites que autorice la Ley de Propiedad Intelectual.



# Índice

1. Introducción.....	1
1.1 Contexto y justificación del Trabajo.....	2
2. Configuración de ambiente de desarrollo para las aplicaciones móviles. ....	2
3. Instalación y configuración de mongoDB .....	4
4. Instalación y configuración de nodeJS.....	5

## **Lista de figuras**

**No se encuentran elementos de tabla de ilustraciones.**

# 1. Introducción

En este manual se explica el proceso necesario para realizar la instalación y configuración del ambiente de desarrollo de todas las aplicaciones relacionadas con E-Health.

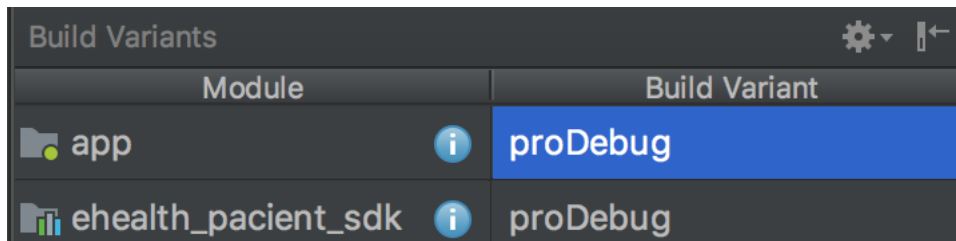
## 2. Configuración de ambiente de desarrollo para las aplicaciones móviles.

En este apartado se explica el proceso que se debe llevar a cabo para poder ejecutar las aplicaciones móviles del proyecto E-Health. El código fuente de ambas aplicaciones será aportado dentro un archivo comprimido .zip, el cual contendrá los proyectos con el código fuente, configuraciones y recursos necesarios para importar dichos proyectos a AndroidStudio y su posterior ejecución en dispositivos móviles o emuladores.

### 2.1 Aplicación destinada a pacientes.

Para poder importar el proyecto de la aplicación móvil destinada a los pacientes se debe descomprimir el archivo .zip suministrado, luego se debe abrir el proyecto desde AndroidStudio. El IDE comenzará el proceso de descarga de las dependencias, una vez terminado será posible ejecutar la aplicación en un dispositivo móvil o emulador.

El proyecto fue desarrollado con dos product flavors, uno llamado “mock” y otro “pro”, el product flavor mock era solo para motivos de desarrollo y no debe ser utilizado para pruebas. Para realizar las pruebas pertinentes a este trabajo final de master se deben realizar en el product flavor pro, en particular con el build variant que se muestra en la siguiente imagen.



### 2.2 Aplicación destinada a médicos.

Para poder importar el proyecto de la aplicación móvil destinada a los médicos se debe descomprimir el archivo .zip suministrado, luego se debe abrir el proyecto desde AndroidStudio. El IDE comenzará el proceso de descarga de las dependencias, una vez terminado será posible ejecutar la aplicación en un dispositivo móvil o emulador.

El proyecto fue desarrollado con dos product flavors, uno llamado “mock” y otro “pro”, el product flavor mock era solo para motivos de desarrollo y no debe ser utilizado para pruebas. Para realizar las pruebas pertinentes a este trabajo final de master se deben realizar en el product flavor pro, en particular con el build variant que se muestra en la siguiente imagen.

Module	Build Variant
app	proDebug
ehealth_doctor_sdk	proDebug

### 2.3 configurar las aplicaciones para utilizar un host local.

Los proyectos mencionados anteriormente ya vienen configurados para utilizar como host un servidor público en el cual se encuentra desplegado el backend, esto incluye la base de datos y la aplicación en nodeJS, esto es con el fin de simplificar las pruebas y no tener que instalar de forma local lo necesario para ejecutar el backend.

Sin embargo, si se quiere utilizar otro host será necesario modificar el código fuente de las aplicaciones, el host se encuentra definido en la clase `NetworkManagerImpl.class` dentro de cada uno de los módulos tipo librería en cada uno de los proyectos.

Dentro de la clase `NetworkManagerImpl.class` se debe identificar donde se crea la instancia de `Retrofit`, que es la librería que se utiliza para realizar las peticiones de datos al servidor y en el método `baseUrl`, se debe colocar el nuevo host.



### 3. Instalación y configuración de mongoDB

La aplicación del backend utiliza como DBMS mongoDB, sin embargo, es importante destacar que su instalación y configuración no es necesaria, ya que se tiene un servidor ya desplegado en una IP publica a la cual se tiene acceso desde cualquier dispositivo con una conexión a internet. De todas formas, se menciona el proceso que se debe realizar si se quiere instalar de forma local en un equipo de trabajo.

Principalmente se deben seguir las instrucciones de descarga e instalación que se indican en la página oficial de mongoDB, <https://docs.mongodb.com/manual/>.

Luego de instalar mongoDB se debe, desde una terminal ejecutándose en mongoDB realizar las siguientes acciones.

Crear y utilizar la base de datos use tfmehealth.

Crear las colecciones necesarias.

```
db.createCollection('appointments')
db.createCollection('doctors')
db.createCollection('patients')
db.createCollection('specialties')
db.createCollection('waits')
```

Ademas se debe realizar una inserción por defecto a una de las colecciones.

```
db.specialties.insert({name:'general'})
```

## 4. Instalación y configuración de nodeJS

La aplicación del backend fue desarrollada en nodeJS, sin embargo, no es necesaria su instalación y configuración ya que se provee un servidor con la aplicación desplegada la cual se puede utilizar desde cualquier dispositivo móvil con una conexión a internet. De igual forma se indica cómo se debe instalar y configurar nodeJS para ser utilizado desde un equipo local de desarrollo.

Para descargar e instalar nodeJS, se deben seguir las instrucciones que se ofrecen en la página oficial de nodeJS, <https://nodejs.org/es/>

Luego se debe descomprimir el proyecto, el usuario se debe dirigir hasta el directorio donde se encuentra el archivo app.js que es TFM\_EHealth y desde una terminal en esa dirección debe iniciar la aplicación con el comando node app.