

Práctica React Native

JOSÉ JIMÉNEZ DIO

28 de mayo de 2023



Introducción

Esta practica se enfoca en el desarrollo de una aplicación móvil desarrollada con el framework de react native cli, demostrando los conocimientos que se tiene con este.

Objetivo

Desarrollar una aplicación móvil y consumir una API noticias

Herramientas de desarrollo

Sistema operativo: Linux

Sistema de desarrollo destino: Android

Dependencias a instalar:

Node.js, Framework de react native, JDK de desarrollo java SE versión 11, Android studio, GitHub

Para la instalación y configuración del entorno de desarrollo se puede guiar de la documentación de react native, ya sea para sistema operativo Windows, macOS o Linux: [Configuración del entorno de desarrollo · React Nativo \(reactnative.dev\)](https://reactnative.dev/docs/environment-setup)

Descripción de actividades con planeación

Diseño de pantallas en Figma

Creación del repositorio en GitHub

Creación del proyecto

Vinculación del proyecto creado con el repositorio creado

Implementación del diseño en la aplicación móvil

Creación del icono y el splash de la aplicación

Generación del apk de la app.

Clonar repositorio

Desarrollo de actividades

Diseño de pantallas en figma

Para la creación de los mockups se utilizó figma, creando las pantallas con la paleta de colores elegido. A continuación, se muestra como quedaron los diseños de cada pantalla.



Imagen 1. Pantalla de Home

Información

Titulo de la noticia

Autor

Fecha de publicación

Descripcion de la noticia

Imagen 2. Pantalla de Detalle

Creación del repositorio en GitHub

Para crear el repositorio, abres GitHub, le das clic en 'File', que se encuentra en la parte superior derecha, eliges la opción de 'New repository...', y te abre una ventana como se muestra a continuación, en la cual se nombra el repositorio, se describe el repositorio brevemente, eliges la ubicación donde estará el repositorio, marcamos que inicie con README, en cuanto al Git ignore se elige Android, siendo un repositorio para desarrollo móvil, no elegimos licencia y por ultimo le damos en el botón de 'Create repository', y listo, nos creara el repositorio.

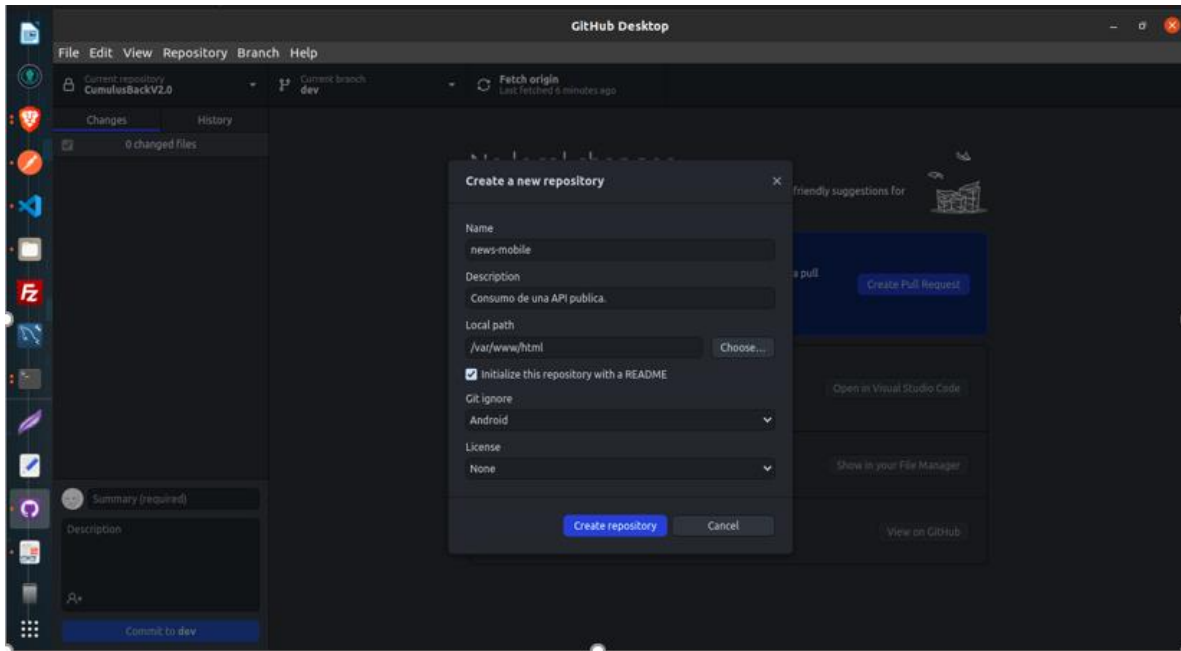


Imagen 3. Creación del repositorio en GitHub

Creación del proyecto

Teniendo en cuenta la instalación de las dependencias antes mencionadas, podremos iniciar con la creación del proyecto, para ello abrimos la terminal, en la ubicación que se desee tener el proyecto, y después instalar react native globalmente con el comando: 'npm uninstall -g react-native-cli @react-native-community/cli', después de esto, se crea el proyecto con el comando: 'react-native init newsMobile', como se muestra en la imagen 4, 'newsMobile' es el nombre del proyecto, esto puede cambiar, según el desarrollador quiera nombrar el proyecto.

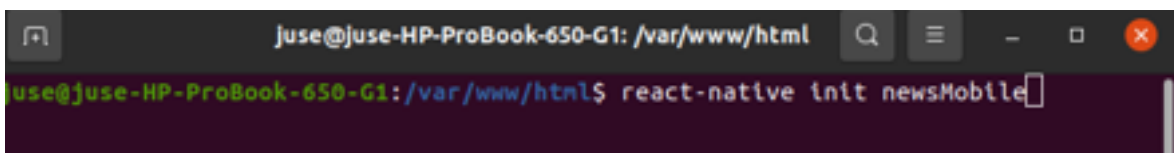


Imagen 4. Creación del proyecto



Imagen 5. Proyecto creado

Cuando se termine de crear el proyecto, se entra al proyecto que se creó con el nombre que se le asignó, con el comando: 'cd newsMobile'(Imagen 6), tomando en cuenta que 'newsMobile' es el nombre del proyecto.

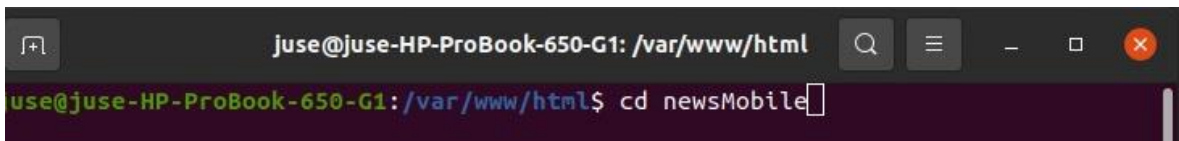


Imagen 6. Ingresar al proyecto para poder ejecutar

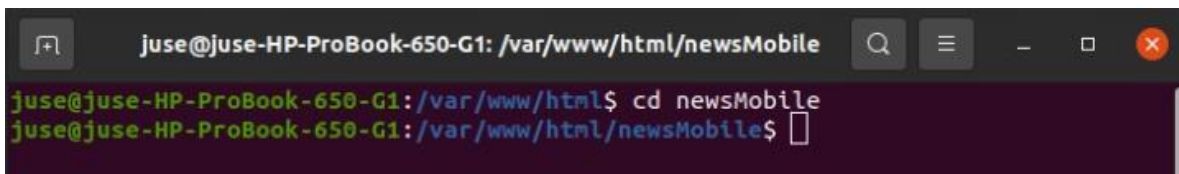


Imagen 7. Se ingreso a la carpeta del proyecto

Para poder seguir con esto debemos hacer una configuración a nuestro teléfono, que es el de activar el modo desarrollador y el modo depuración por USB:

Primero ir a la configuración del teléfono, y buscar la opción "Acerca del teléfono" o "Información del teléfono". Puede estar ubicada en la sección "Ajustes" o "Sistema", dependiendo de la versión de Android y la capa de personalización del fabricante.

Dentro de "Acerca del teléfono", buscar la opción "Número de compilación" o "Número de versión". Toca repetidamente (generalmente alrededor de 7 veces) en esta opción hasta que aparezca un mensaje indicando que el modo desarrollador se ha habilitado.

Volver a la configuración principal y buscar la nueva opción llamada "Opciones de desarrollador" o "Desarrollador" que ahora debería estar disponible.

Ingresa a "Opciones de desarrollador" y busca la configuración "Depuración USB" o "Depuración de Android". Activar esta opción. Es posible que se te solicite confirmar la acción.

Listo, la depuración por USB esta activa, ahora procedemos a conectar tu teléfono Android la computadora mediante un cable USB, para poder ejecutar en tiempo real nuestra aplicación de react native.

Terminada de hacer la configuración del dispositivo móvil, se sigue con el siguiente paso. Una vez dentro del proyecto, se conecta el teléfono celular, o se ejecute el emulador de Android, en este caso voy a usar el teléfono físico para poder hacer la ejecución de la aplicación, conectamos el teléfono a la computadora y en el teléfono aparece una notificación como se muestra en la **Imagen 8**, lo cual debemos elegir el de transferencia de archivo, también nos aparece una notificación que el modo depuración esta activa como se muestra en la **Imagen 9** (El teléfono tiene que estar en modo de depuración), y cuando la computadora reconozca el teléfono, se ejecuta el siguiente comando para iniciar la instalación de la app en el dispositivo conectado: `'npx react-native run-android'`(**Imagen 10**), una vez terminada de ejecutar este comando, se ejecuta este otro comando para iniciar el servidor: `'npx react-native start'`(**Imagen 13**), esta inicia el servidor, esperamos que inicie y presionamos la letra 'r' (**Imagen 14**) para recargar la aplicación, así como se muestra en las imágenes, y listo, ya podemos ver nuestra aplicación ejecutándose.

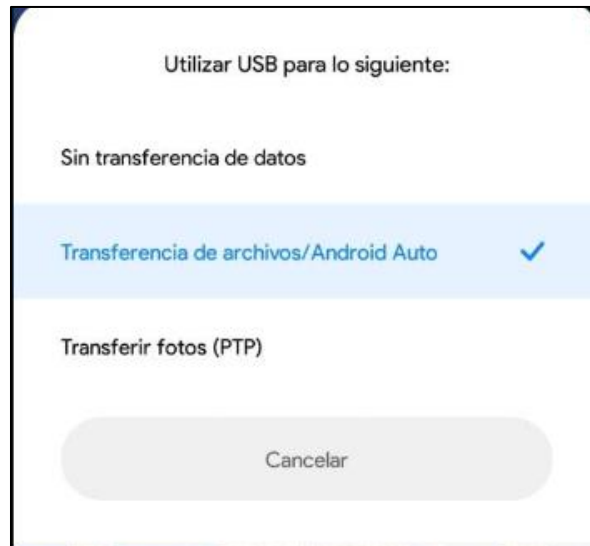


Imagen 8. Conexión por USB, con opción de transferencia de archivos



Imagen 9. Notificación de Modo depuración

```
juse@juse-HP-ProBook-650-G1: /var/www/html/newsMobile$ npx react-native run-android
Info Starting JS server...
Info Installing the app...

> Task :app:installDebug
Installing APK 'app-debug.apk' on 'M2006C3LG - 11' for :app:debug
Installed on 1 device.

BUILD SUCCESSFUL in 20s
38 actionable tasks: 2 executed, 36 up-to-date
Info Connecting to the development server...
Info Starting the app on "NV994LPB0N8965C6"...
Starting: Intent { cmp=com.newsmobile/.MainActivity }
juse@juse-HP-ProBook-650-G1: /var/www/html/newsMobile$
```

Imagen 10. Ejecución e instalación de la app en el dispositivo

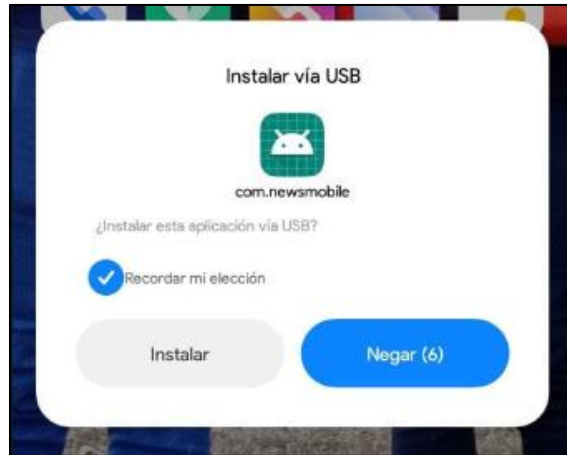


Imagen 11. Permitir en el dispositivo móvil instalar la app




Imagen 12. App instalada en el dispositivo móvil, falta iniciar el servidor

```
juse@juse-HP-ProBook-650-G1: /var/www/html/newsMobile$ npx react-native run-android
info Starting JS server...
info Installing the app...

> Task :app:installDebug
Installing APK 'app-debug.apk' on 'M2006C3LG - 11' for :app:debug
Installed on 1 device.

BUILD SUCCESSFUL in 20s
38 actionable tasks: 2 executed, 36 up-to-date
info Connecting to the development server...
info Starting the app on "NV994LPB0N8965C6"...
Starting: Intent { cmp=com.newsmobile/.MainActivity }
juse@juse-HP-ProBook-650-G1: /var/www/html/newsMobile$ npx react-native start



Welcome to Metro v0.73.9
Fast - Scalable - Integrated

r - reload the app
d - open developer menu
i - run on iOS
a - run on Android

□
```

Imagen 13. Iniciar el servidor para recargar la app

```
juse@juse-HP-ProBook-650-G1:/var/www/html/newsMobile$ npx react-native start

  Welcome to Metro v0.73.9
  Fast - Scalable - Integrated

r - reload the app
d - open developer menu
i - run on iOS
a - run on Android

info Reloading app...
BUNDLE ./index.js

LOG Running "newsMobile" with {"rootTag":11}
```

Imagen 14. Con la letra 'r', reiniciar el servido

Una vez ejecutada la aplicación en el dispositivo móvil sin modificaciones, se ve como se muestra a continuación en la **Imagen 15**.

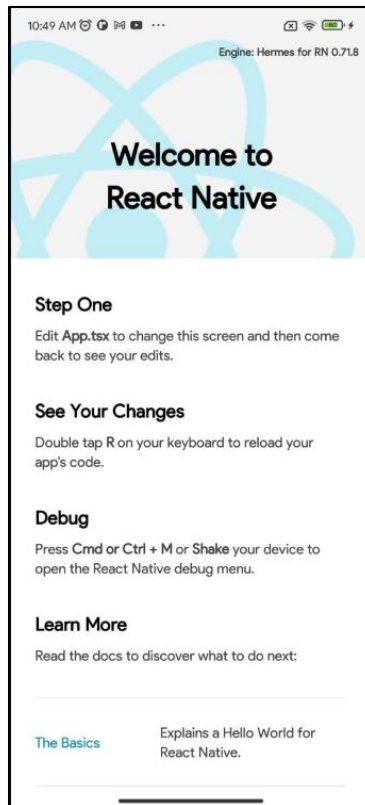


Imagen 15. App móvil instalada y en ejecución

Vinculación del proyecto creado con el repositorio creado.

Teniendo el repositorio creado y el proyecto, vinculamos el repositorio con el proyecto, moviendo los archivos que están en el proyecto a la carpeta del repositorio creado, dentro del proyecto creado existe la carpeta de 'node_modules' que no copiaremos al repositorio, porque esta carpeta se crea al ejecutar 'npm install' dentro del repositorio, instalando las librerías que se tiene en el package.json, cada modificación que se haga se sube al repositorio.

Implementación del diseño en la aplicación móvil

En esta parte del proyecto se inicia con la implementación del diseño que se desarrolló en figma, cada diseño terminado y modificado se suben al repositorio de GitHub.

Se analiza las librerías que se van a ocupar, como lo son los de navegación y sus componentes, estos componentes se instalan con los comandos: 'npm install

@react-navigation/native @react-navigation/native-stack' y 'npm install react-native-screens react-native-safe-area-context' asi como se muestra en las siguientes imágenes.



Imagen 16. Instalación de la librería de navegación

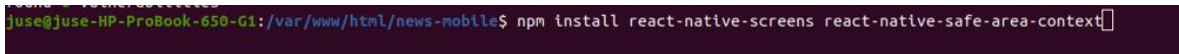


Imagen 17 Instalación del componente de la librería de navegación

Creación del icono y el splash de la aplicación

Para la creación del icono y del splash se utilizó canva, tomado en cuenta la paleta de colores elegido, así como también se tuvo que adaptar el icono a diferentes tamaños y sustituir el icono credo por los que traía por default, para que al momento de instalarlos en dispositivos diferentes mantenga el icono que se a diseñado para ello. Las medidas que se tuvieron que adaptar el icono son las siguientes:

- mdpi (densidad de pantalla baja): 48x48 píxeles
- hdpi (densidad de pantalla media): 72x72 píxeles
- xhdpi (densidad de pantalla alta): 96x96 píxeles
- xxhdpi (densidad de pantalla extra alta): 144x144 píxeles
- xxxhdpi (densidad de pantalla extra extra alta): 192x192 píxeles

Colocar cada archivo de imagen(icono) en las carpetas correspondientes dentro de android/app/src/main/res/

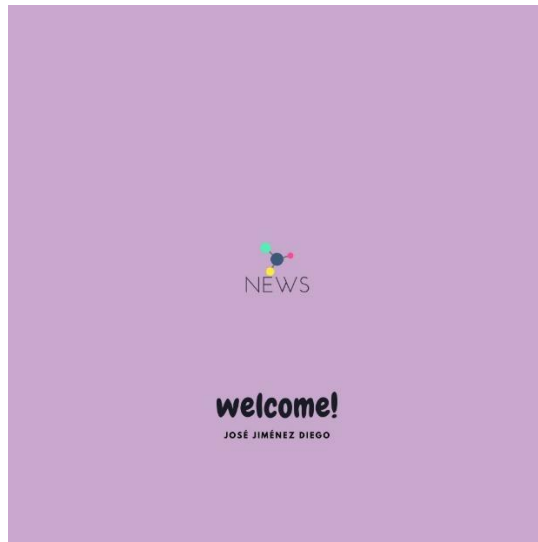


Imagen 18. Splash de la aplicación



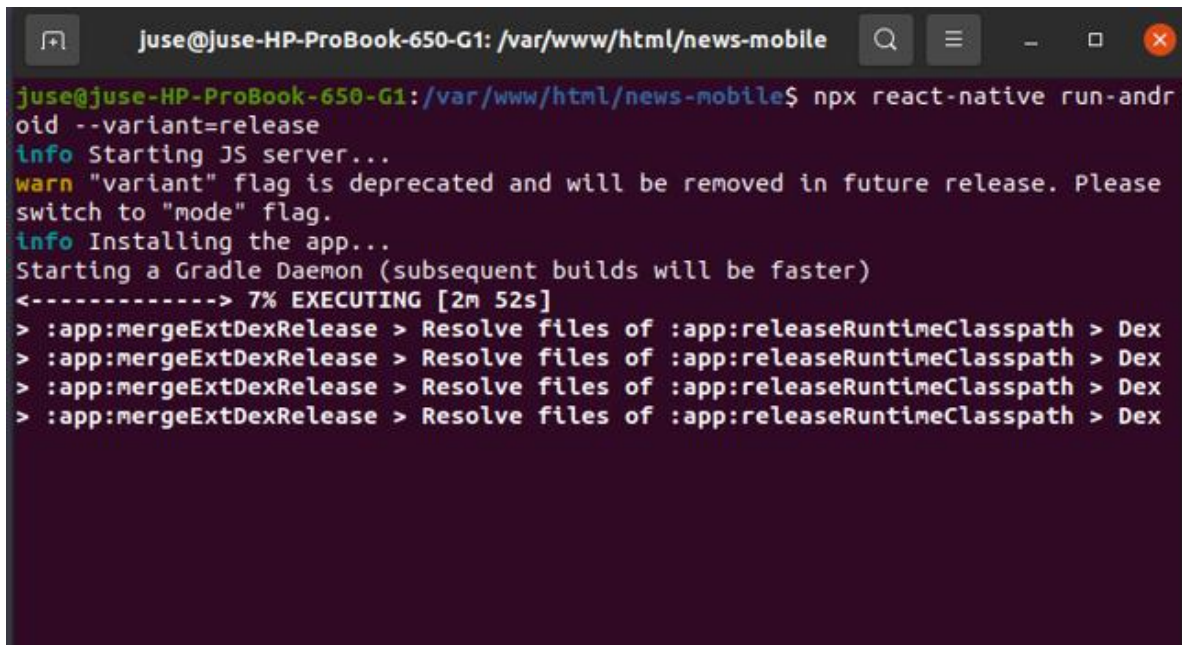
Imagen 19 Icono de la aplicación

Generación del apk de la app.

Para generar el apk es con el siguiente comando: `npx react-native run-android --variant=release`

Este comando compilará y generará el apk, y una vez finalizado el proceso, el apk debe estar ubicado en la ubicación: **android/app/build/outputs/apk/release/app-release.apk**

El archivo apk generado se llama app-release.apk.



```
juse@juse-HP-ProBook-650-G1: /var/www/html/news-mobile$ npx react-native run-android --variant=release
info Starting JS server...
warn "variant" flag is deprecated and will be removed in future release. Please switch to "mode" flag.
info Installing the app...
Starting a Gradle Daemon (subsequent builds will be faster)
<-----> 7% EXECUTING [2m 52s]
> :app:mergeExtDexRelease > Resolve files of :app:releaseRuntimeClasspath > Dex
> :app:mergeExtDexRelease > Resolve files of :app:releaseRuntimeClasspath > Dex
> :app:mergeExtDexRelease > Resolve files of :app:releaseRuntimeClasspath > Dex
> :app:mergeExtDexRelease > Resolve files of :app:releaseRuntimeClasspath > Dex
```

Imagen 20 Generar el apk

Clonar repositorio

Para clonar el proyecto se debe de agregar esta liga en la terminal con uso de GitHub CLI: **gh repo clone DiegoJose99/news-mobile**

o usando SSH con la siguiente liga: **git clone git@github.com:DiegoJose99/news-mobile.git**

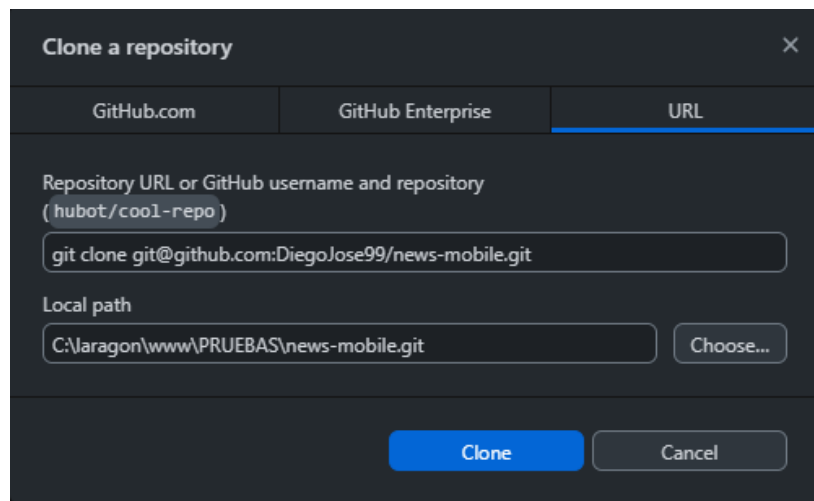


Imagen 21 Clonar repositorio con SSH

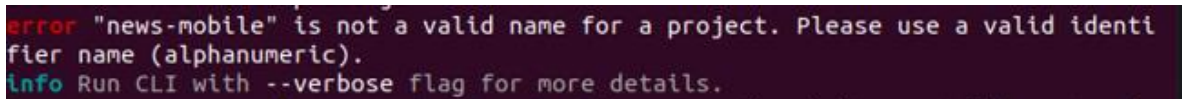
Complicaciones

Estas son algunas de las complicaciones durante la creación del proyecto, la primera dificultad a la que me enfrente fue al crear el proyecto, porque react native no permite crear el nombre de proyectos con `_`, `-`, `.` y entre otros, y yo estaba creando el proyecto con nombre `news-mobile`, (**Imagen 18 y 19**) no me permitió, y opte por crear el nombre del proyecto como `'newsMobile'`, (**Imagen 20**) así como se muestra en esta imagen, y así si me permitió crear el proyecto.



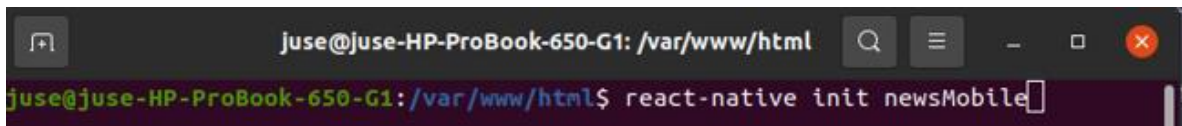
```
juse@juse-HP-ProBook-650-G1: /var/www/html$ react-native init news-mobile
```

Imagen 22 Intento de crear el proyecto incorrecto



```
error: "news-mobile" is not a valid name for a project. Please use a valid identifier name (alphanumeric).  
info Run CLI with --verbose flag for more details.
```

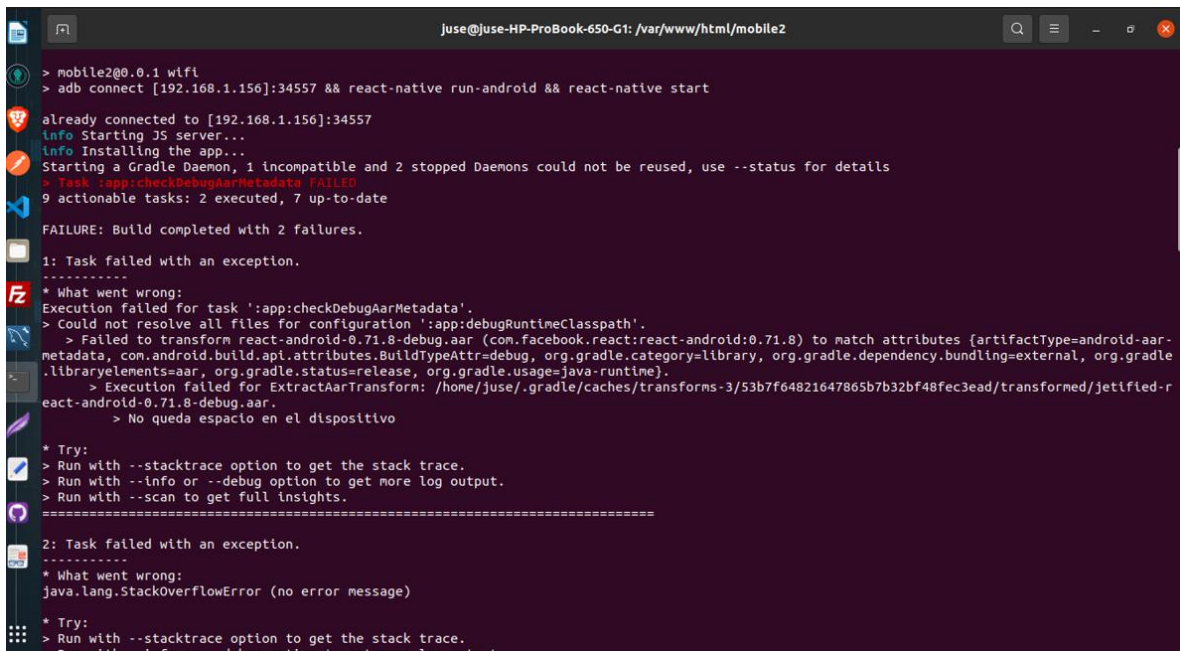
Imagen 23 Error al crear el proyecto por el nombre



```
juse@juse-HP-ProBook-650-G1: /var/www/html$ react-native init newsMobile
```

Imagen 24 Creación del proyecto con nombre correcto

La segunda complicación con la que me encontré fue el de no podía descargar las dependencias del proyecto por falta de espacio en mi computadora, y me arrojaba el error que se muestra en la imagen, lo cual se soluciono eliminando algunas cosas para poder tener mas espacio dentro de la computadora.

A terminal window on a Linux system showing the output of a React Native build command. The user has connected to a device via adb and run the command 'react-native run-android'. The build process starts, but fails at the 'app:checkDebugAarMetadata' task. The error message indicates that the build failed due to a 'StackOverflowError' (no error message) and a 'No queda espacio en el dispositivo' (No space left on device) error. The terminal also shows the 'Task failed with an exception' message and the 'FAILURE: Build completed with 2 failures.' message. The user is prompted to try running with '--stacktrace' or '--info' options to get more details.

```
Juse@Juse-HP-ProBook-650-G1: /var/www/html/mobile2
> mobile2@0.0.1 wifi
> adb connect [192.168.1.156]:34557 && react-native run-android && react-native start

already connected to [192.168.1.156]:34557
info Starting JS server...
info Installing the app...
Starting a Gradle Daemon, 1 incompatible and 2 stopped Daemons could not be reused, use --status for details
> Task :app:checkDebugAarMetadata FAILED
9 actionable tasks: 2 executed, 7 up-to-date

FAILURE: Build completed with 2 failures.

1: Task failed with an exception.
-----
* What went wrong:
Execution failed for task ':app:checkDebugAarMetadata'.
> Could not resolve all files for configuration ':app:debugRuntimeClasspath'.
  > Failed to transform react-android-0.71.8-debug.aar (com.facebook.react:react-android:0.71.8) to match attributes {artifactType=android-aar-metadata, com.android.build.api.attributes.BuildTypeAttr=debug, org.gradle.category=library, org.gradle.dependency.bundling=external, org.gradle.libraryElements=aar, org.gradle.status=release, org.gradle.usage=java-runtime).
    > Execution failed for ExtractAarTransform: /home/juse/.gradle/caches/transforms-3/53b7f64821647865b7b32bf48fec3ead/transformed/jetified-react-android-0.71.8-debug.aar.
      > No queda espacio en el dispositivo

* Try:
> Run with --stacktrace option to get the stack trace.
> Run with --info or --debug option to get more log output.
> Run with --scan to get full insights.
=====
2: Task failed with an exception.
-----
* What went wrong:
java.lang.StackOverflowError (no error message)

* Try:
> Run with --stacktrace option to get the stack trace.
> Run with --info or --debug option to get more log output.
```

Imagen 25 Error al descargar dependencias del proyecto por falta de espacio

Otra complicación que tuve en el transcurso del desarrollo de esta aplicación es el límite permitido de eventos por 'inotify' por usuarios, y para poder ajustar el valor se requieren permisos de superusuario, el error que me mando fue le siguiente que se muestra en la **Imagen 23**, y para resolver se ejecuta el comando: 'sudo sysctl fs.inotify.max_user_watches=524288', para poder restaurar los valores de evento nuevamente.

```

w/html/mobile2/node_modules/@jest'
  at FSWatcher.<computed> (node:internal/fs/watchers:244:19)
  at Object.watch (node:fs:2251:34)
  at NodeWatcher._watchdir (/var/www/html/mobile2/node_modules/metro-file-map
/src/watchers/NodeWatcher.js:145:24)
  at /var/www/html/mobile2/node_modules/metro-file-map/src/watchers/NodeWatch
er.js:47:14
  at Walker.<anonymous> (/var/www/html/mobile2/node_modules/metro-file-map/sr
c/watchers/common.js:135:31)
  at Walker.emit (node:events:526:28)
  at /var/www/html/mobile2/node_modules/walker/lib/walker.js:69:16
  at FSReqCallback.oncomplete (node:fs:188:23) {
  errno: -28,
  syscall: 'watch',
  code: 'ENOSPC',
  path: '/var/www/html/mobile2/node_modules/@jest',
  filename: '/var/www/html/mobile2/node_modules/@jest'
}

```

Imagen 26 Error de inotify, por superar número de eventos por usuario

```

juse@juse-HP-ProBook-650-G1:/var/www/html/news-mobile$ sudo sysctl fs.inotify.ma
x_user_watches=524288
[sudo] contraseña para juse:
Lo siento, pruebe otra vez.
[sudo] contraseña para juse:
fs.inotify.max_user_watches = 524288
juse@juse-HP-ProBook-650-G1:/var/www/html/news-mobile$

```

Imagen 27 Solución al error de inotify

Otra de las complicaciones fue que la API tiene una Api_key, que solo es valido por 24hrs, generado el apk y como el Api_key caduco no se mostraban las noticias, por lo que se volvió a generar otra Api_key y se actualizo la API que se encuentra en la app, y nuevamente se generó el apk.

Resultados

Estos son los resultados obtenidos del desarrollo de la aplicación movil, primero tenemos el Splash:



Imagen 28 Splash de la app móvil

Seguido tenemos la pantalla de inicio o home(**Imagen 29**), donde se muestran las lista de las noticias, al dar clic en uno de ellos te muestra el detalle de las noticias(**Imagen 29**), así como se muestra a continuación:

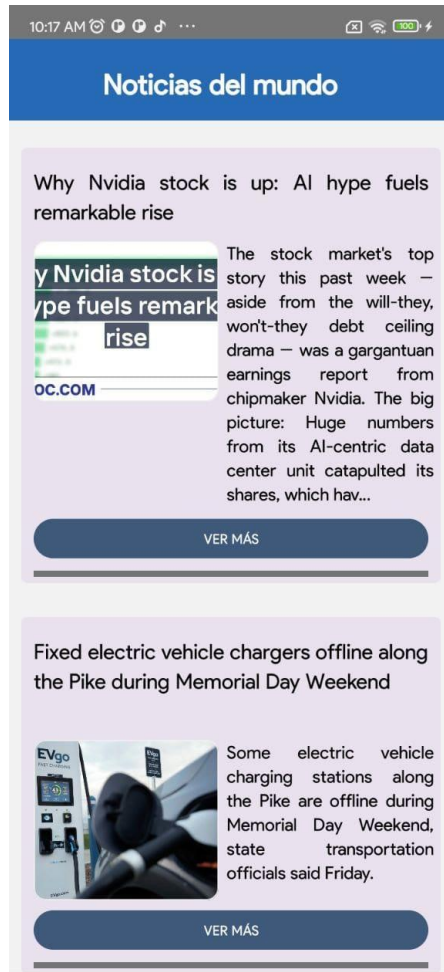


Imagen 29 Home de la aplicación móvil

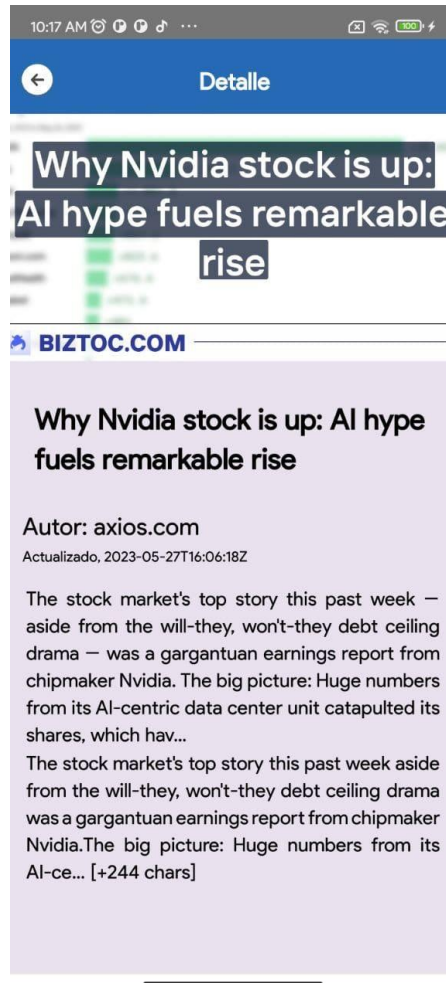


Imagen 30 Detalles de la noticia de la app móvil

Cuando la noticia no tiene una imagen, agregue una imagen que te mande por default como se ve en la **Imagen 31**, y al dar clic, como no tiene una imagen te muestra la imagen por default como se observa en la **imagen 32**.

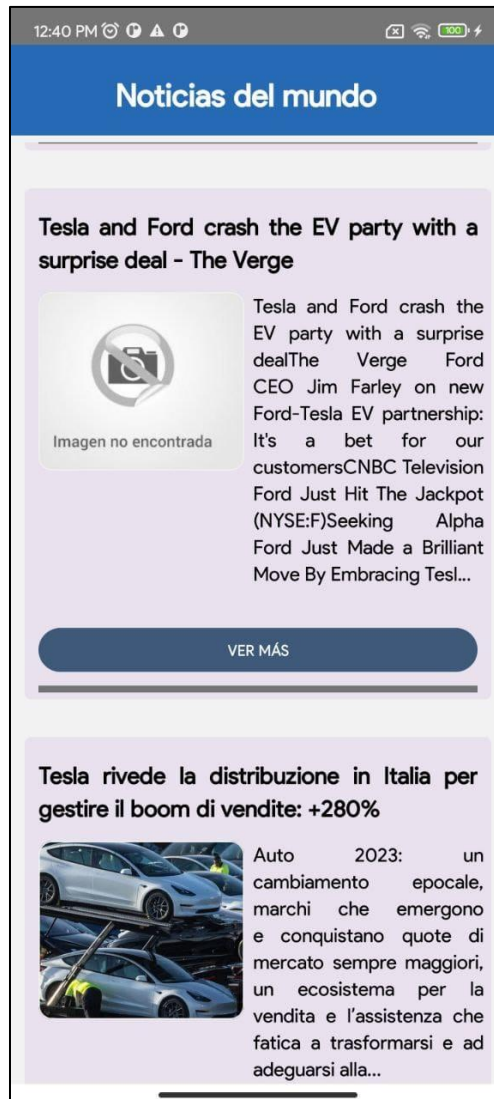


Imagen 31 Noticia sin Imagen

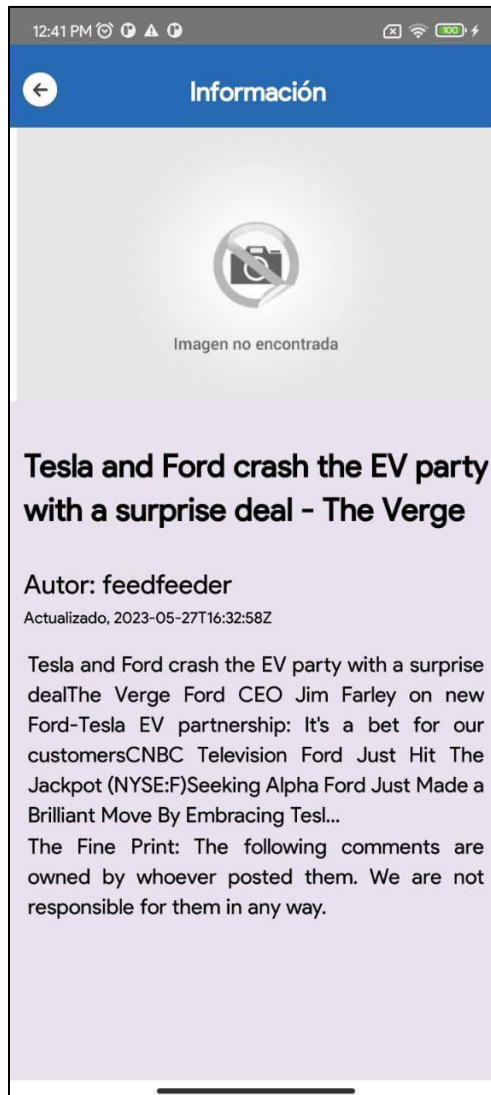


Imagen 32 Detalles de noticia sin imagen

En el repositorio de GitHub se ve de la siguiente manera con los commits realizados:

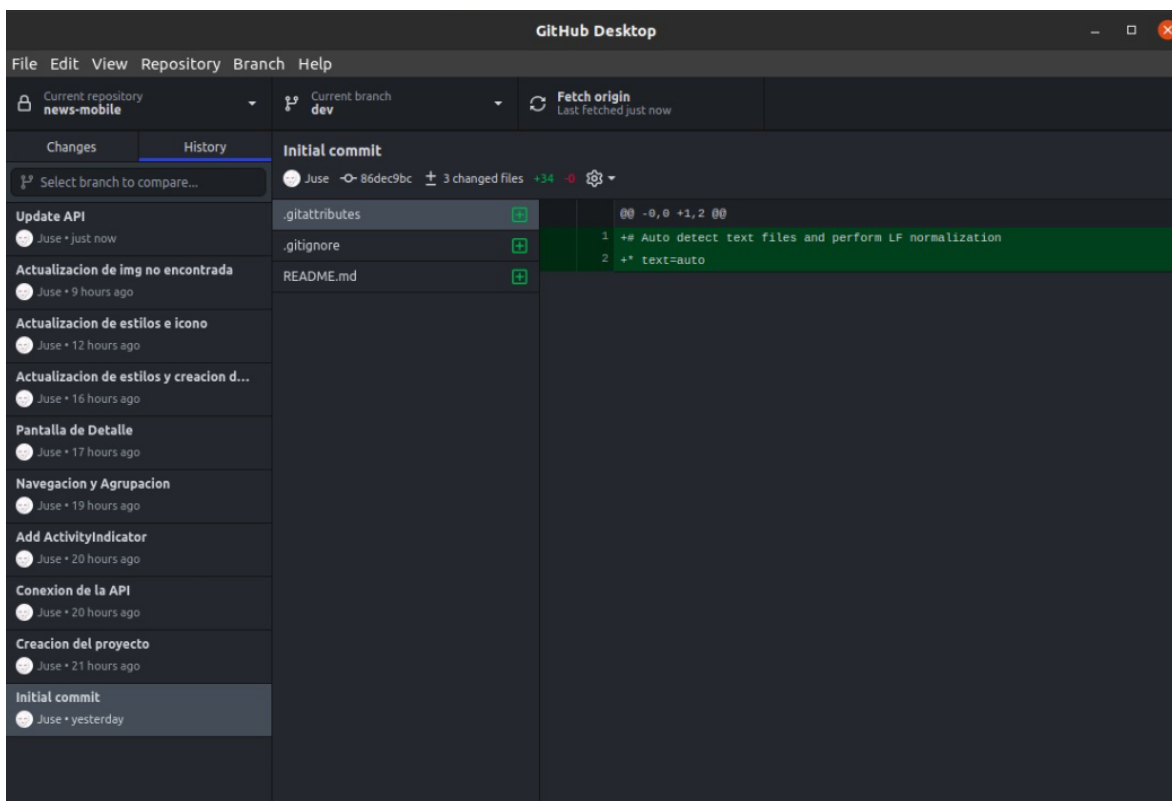


Imagen 33 Repositorio de GitHub commits