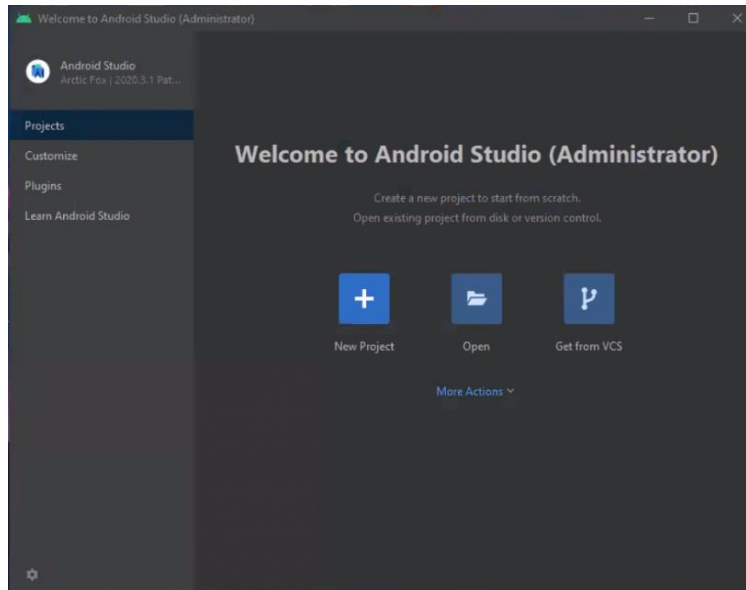


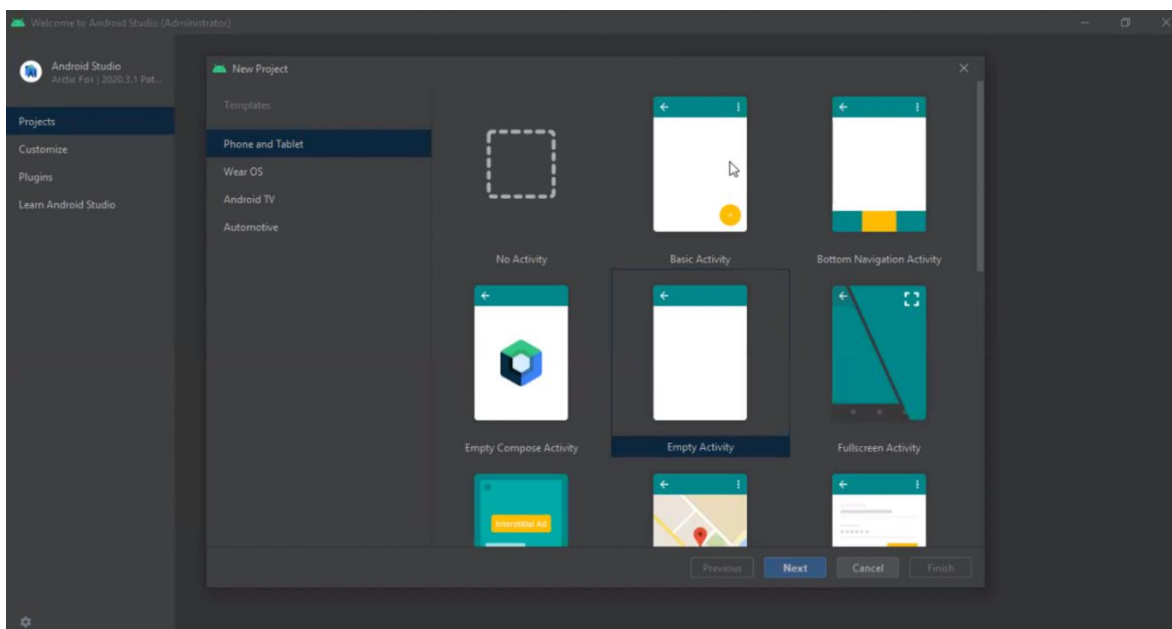
Android Studio

Reporte de actividades, 21 y 22 de Enero

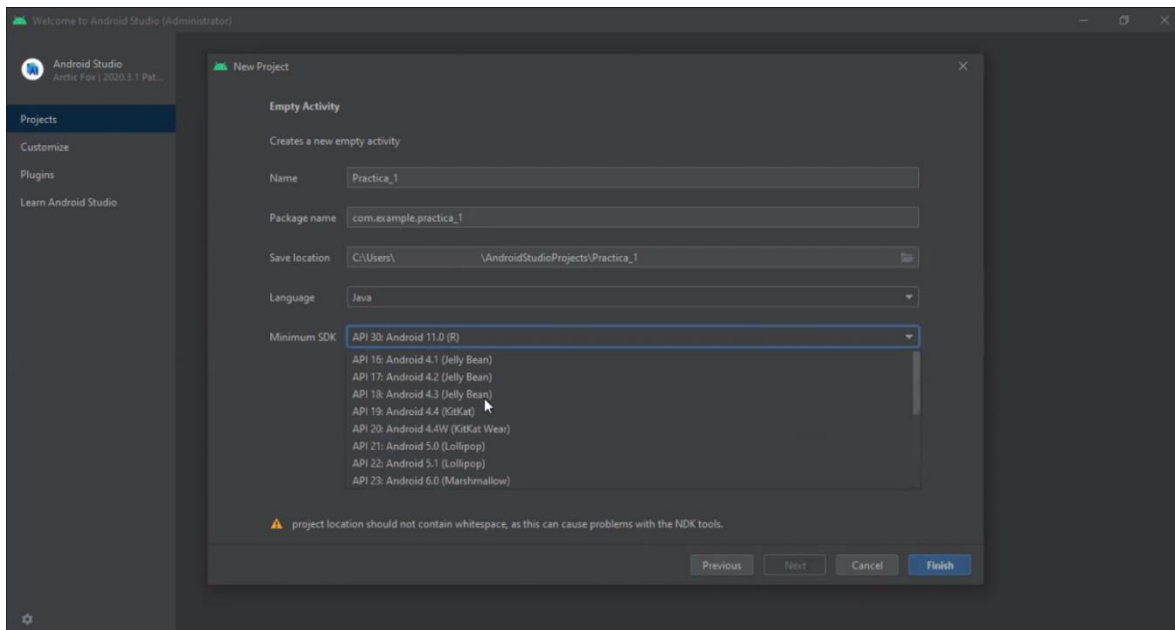
Para comenzar con el primer proyecto del curso utilizando Android Studio primero se realizó la instalación, una vez terminada abrimos el programa, nos aparecerá la siguiente pantalla donde elegiremos la opción “New Project”.



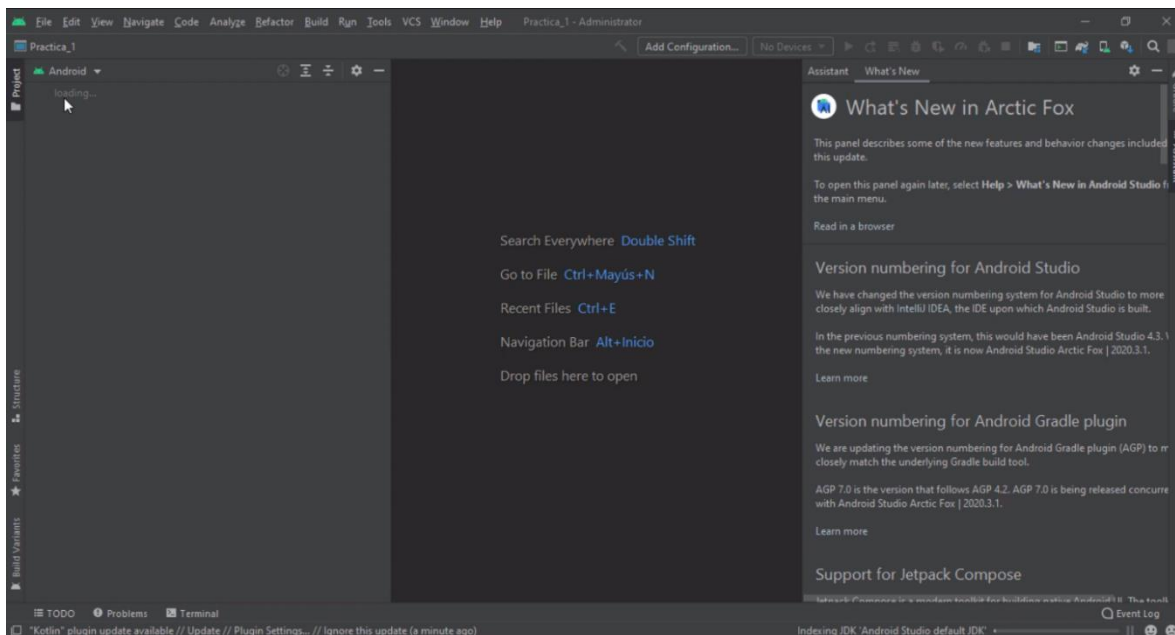
Nos aparecerá la siguiente pantalla donde tendremos que elegir la plantilla que queremos realizar, para este proyecto elegiremos la opción “Empty Activity”



En la siguiente pantalla tendremos que realizar la configuración de nuestro proyecto, pondremos el nombre del proyecto, la carpeta en la que estará almacenado, el lenguaje que utilizaremos será Java y el Kit de desarrollo será el que tenga la mayor compatibilidad con la versión de Android, en este caso será la API 21: Android 5.0 (Lollipop)

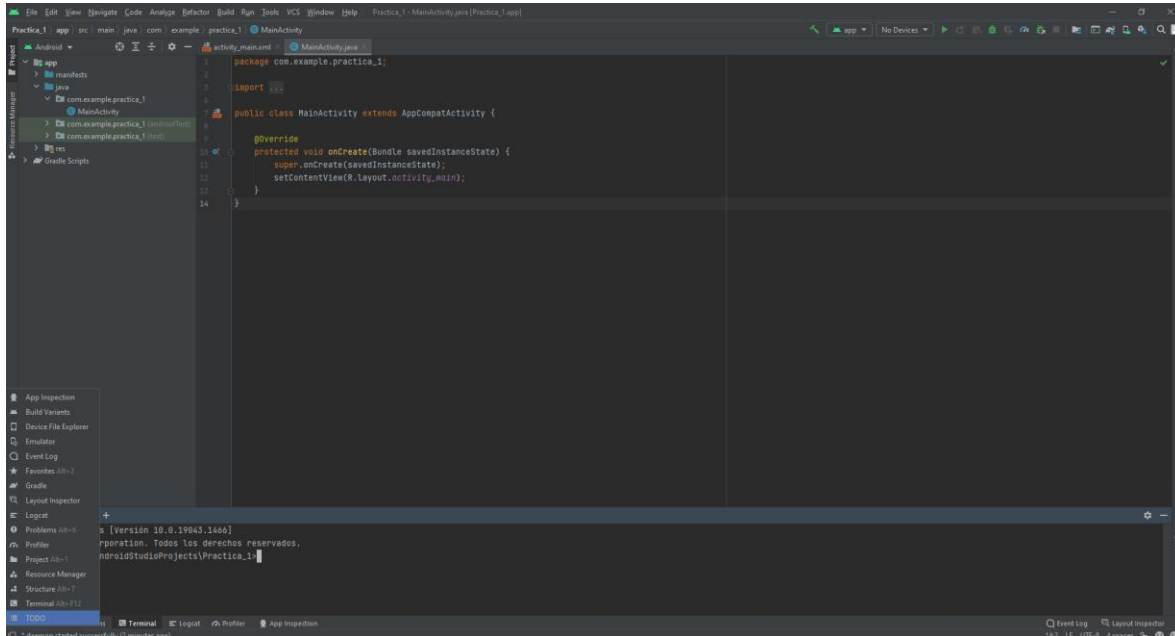


El software ya está listo para comenzar con nuestro primer proyecto

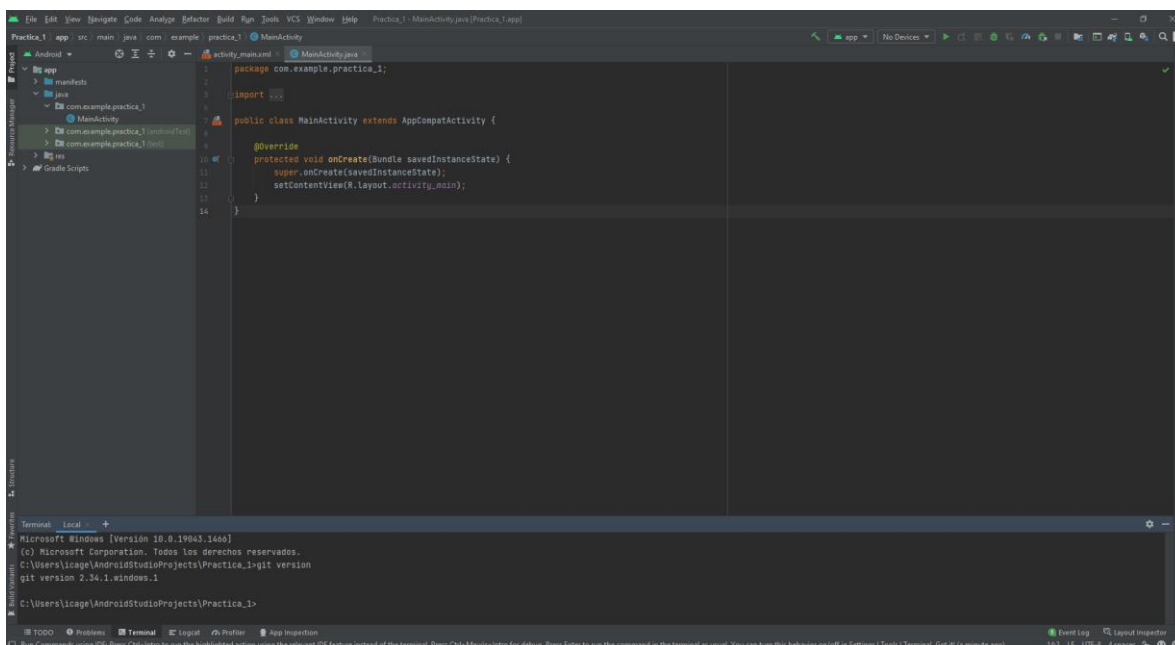


Uso de comandos y guardar el subir el proyecto a GitHub

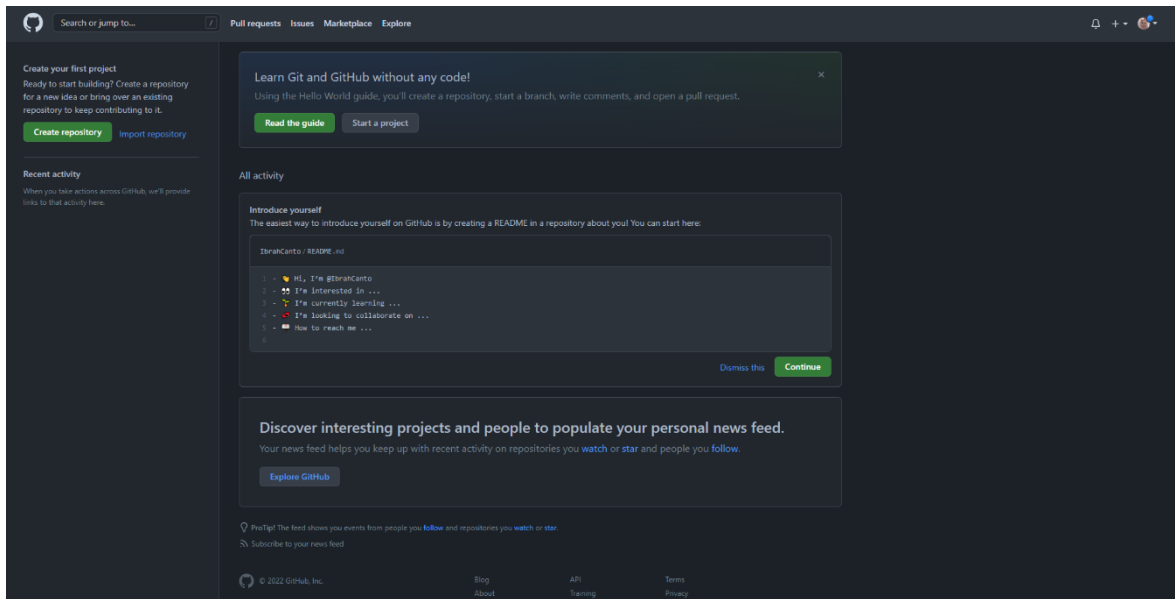
Para empezar a utilizar los comandos en Android Studio debemos abrir la terminal, esto lo hacemos presionando Alt+F12 o seleccionando en la parte inferior izquierda para que nos aparezca un menú donde podremos abrir la terminal.



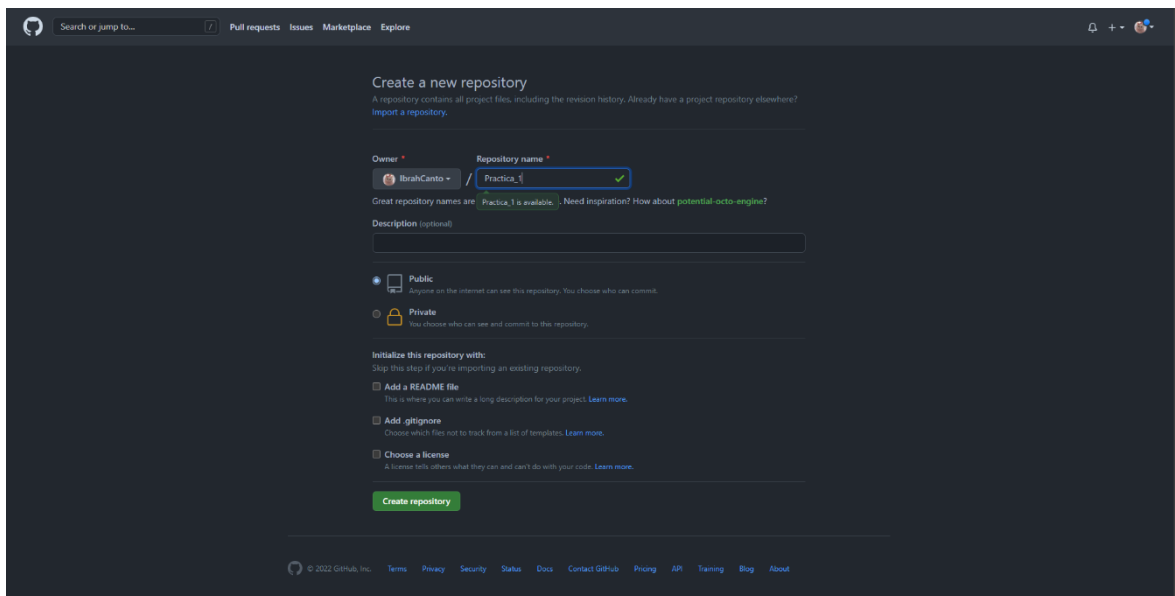
Para comenzar a utilizar el software de control de versiones y subir nuestro proyecto a GitHub debemos instalar el programa Git, una vez terminado realizamos una comprobación desde la terminal de Android Studio escribiendo 'git version'



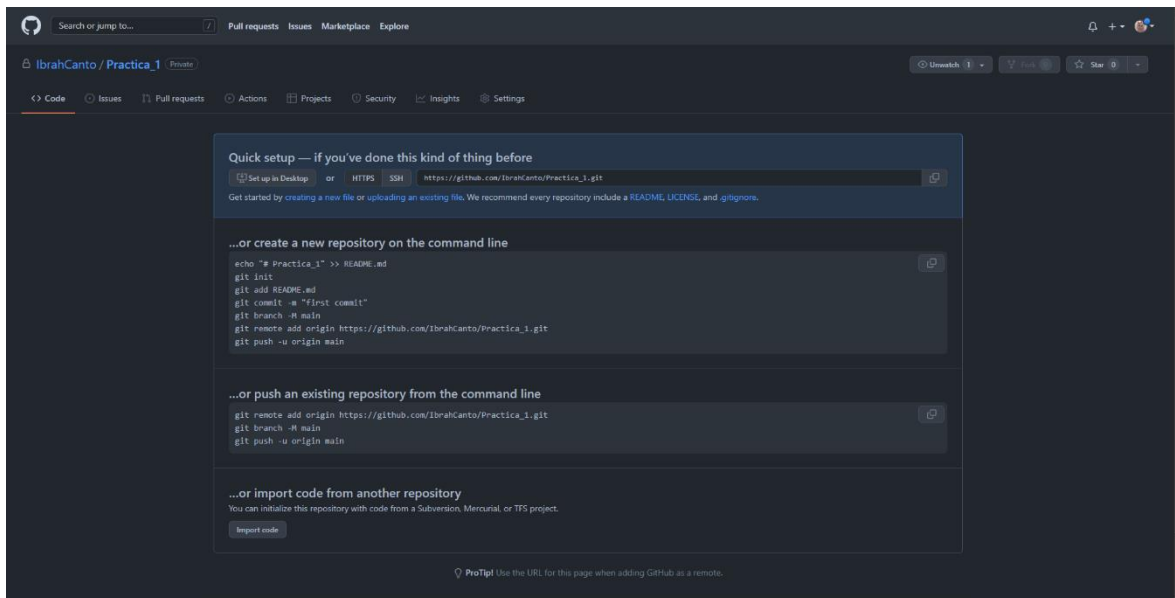
Una vez comprobado que Git este funcionando ya podremos subir nuestro proyecto como un repositorio en GitHub, para esto debemos tener una cuenta en github.com e iremos donde dice ‘Create repository’



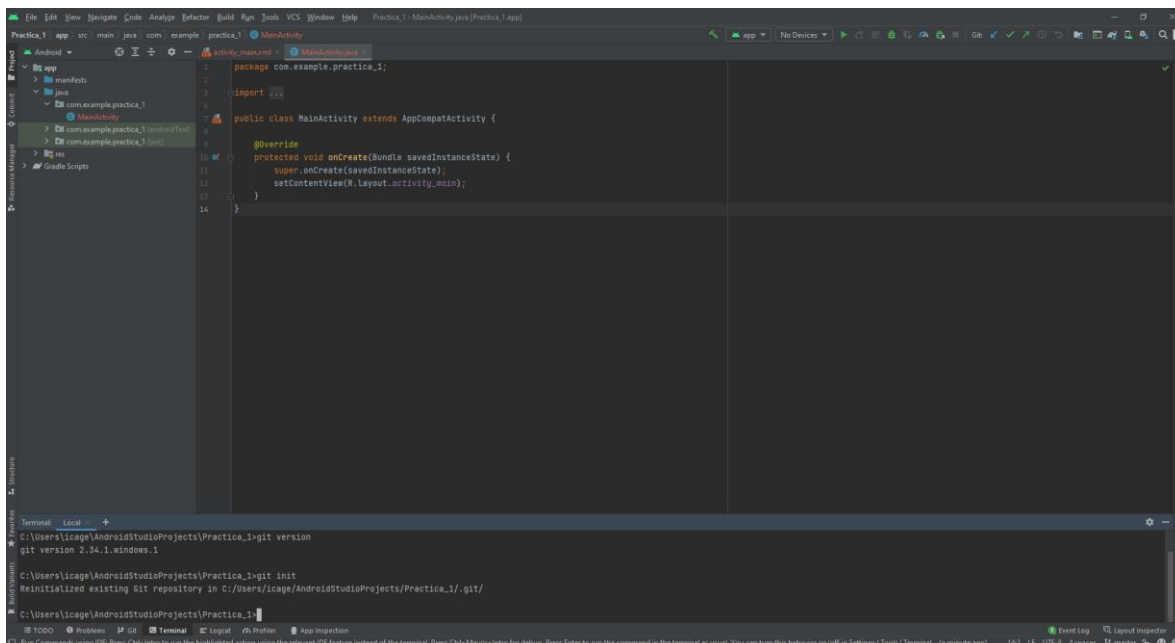
Pondremos el nombre del repositorio en donde estará almacenado nuestro proyecto y una descripción, elegiremos Private para tener un repositorio privado y presionamos donde dice ‘Create repository’



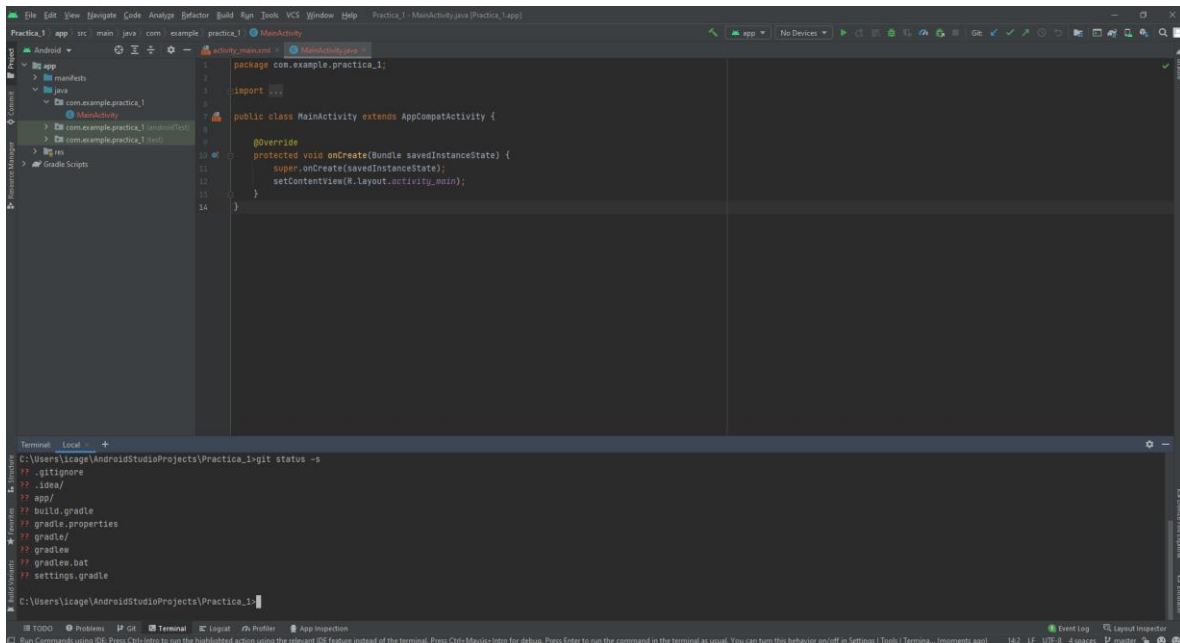
Nos aparecerá la siguiente pantalla y continuaremos desde la terminal de Android Studio.



Desde la terminal de Android Studio escribiremos 'git init' para comenzar



Para hacer la comprobación de los datos que se subirán a nuestro repositorio en GitHub escribiremos en la terminal 'git status -s'



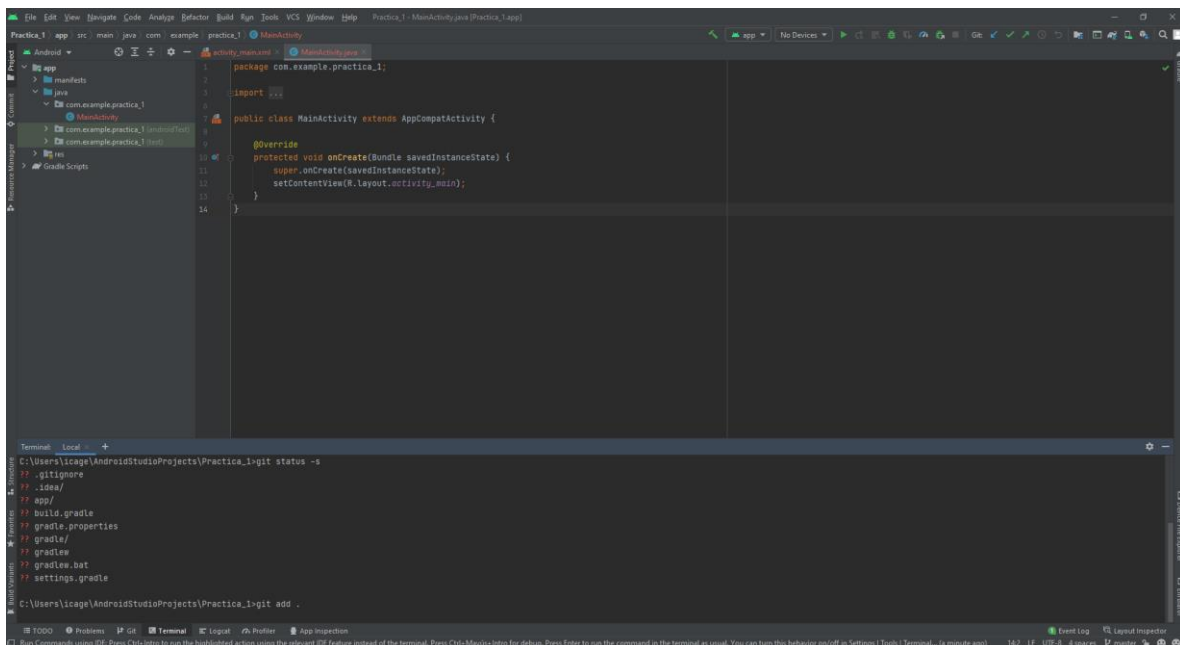
The screenshot shows the Android Studio IDE. The main editor displays the `MainActivity.java` file with the following code:

```
1 package com.example.practica_1;
2
3 import androidx.appcompat.app.AppCompatActivity;
4
5 public class MainActivity extends AppCompatActivity {
6
7     @Override
8     protected void onCreate(Bundle savedInstanceState) {
9         super.onCreate(savedInstanceState);
10        setContentView(R.layout.activity_main);
11    }
12
13 }
14
```

The terminal window at the bottom shows the command `git status -s` being executed, resulting in the following output:

```
C:\Users\lucag\AndroidStudioProjects\Practica_1>git status -s
?? .gitignore
?? .idea/
?? app/
?? build.gradle
?? gradle.properties
?? gradle/
?? gradlew
?? gradlew.bat
?? settings.gradle
```

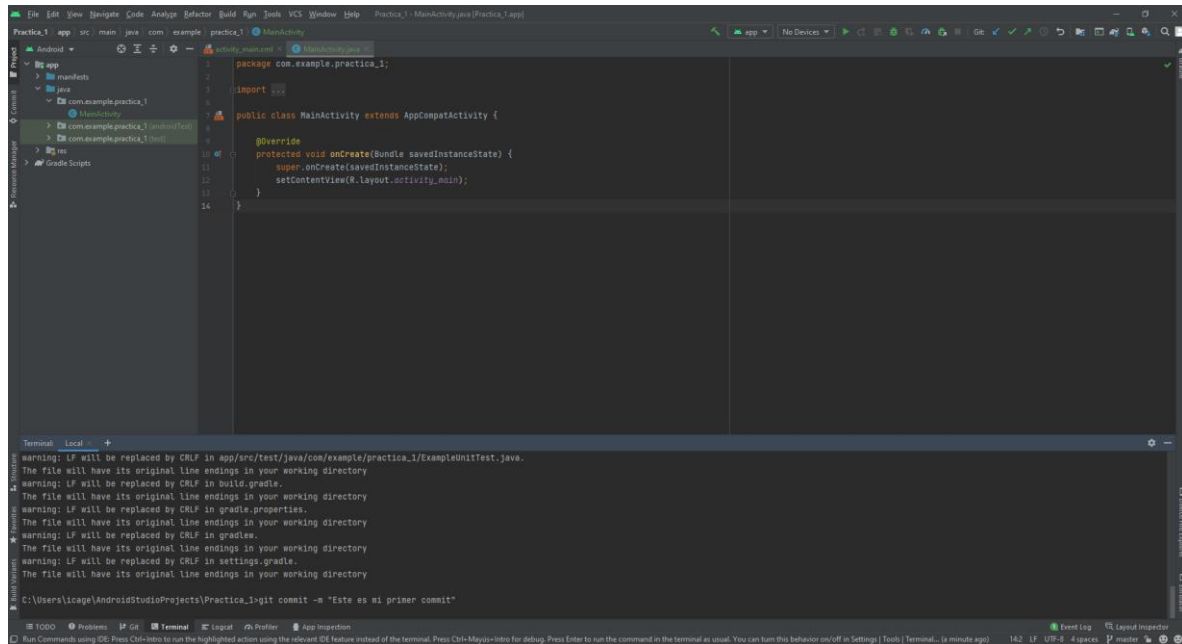
Una vez hecha la comprobación escribiremos 'git add .' esto seleccionara todos los datos de nuestro proyecto que se subirán al repositorio en GitHub



The screenshot shows the Android Studio IDE with the same `MainActivity.java` file. The terminal window now shows the command `git add .` being executed, resulting in the following output:

```
C:\Users\lucag\AndroidStudioProjects\Practica_1>git add .
?? .gitignore
?? .idea/
?? app/
?? build.gradle
?? gradle.properties
?? gradle/
?? gradlew
?? gradlew.bat
?? settings.gradle
```

Una vez seleccionados todos los datos que se subirán a nuestro repositorio en GitHub vamos a realizar nuestro primer commit, esto funciona como una confirmación de los datos que están listos para ser subidos y se puede acompañar con un mensaje, la forma en la que debemos escribirlo en la terminal de Android Studio es ‘git commit -m “Este es mi primer commit”’, dentro de las comillas escribiremos el mensaje que acompañara nuestro commit.



```
package com.example.practica_1;

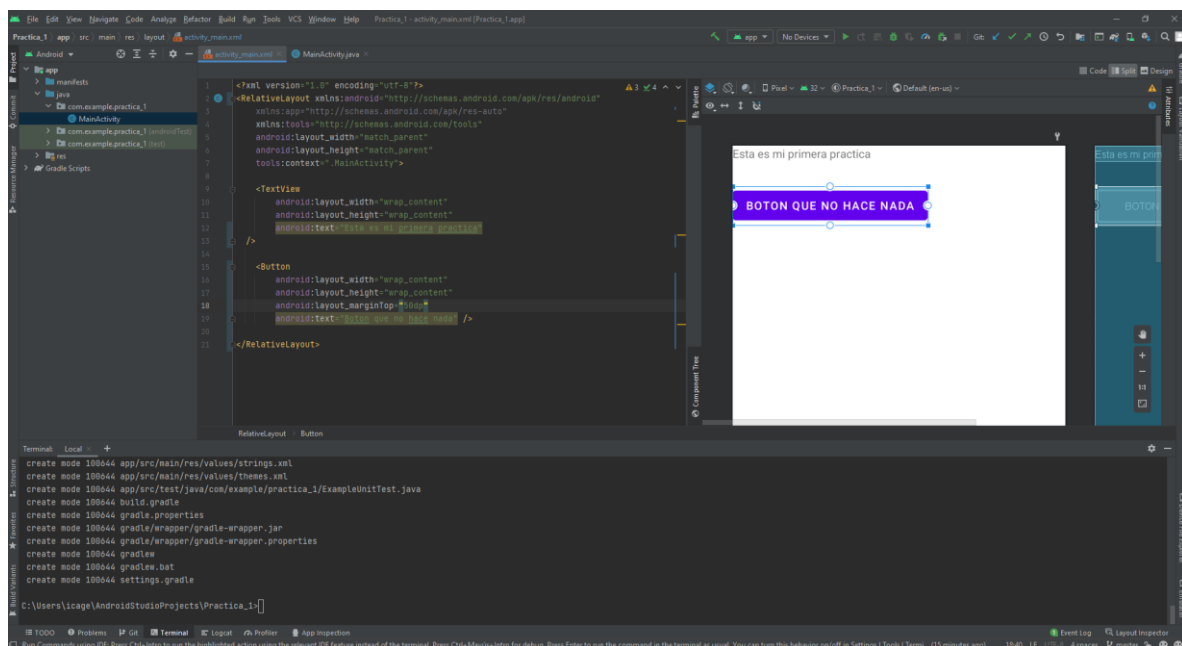
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```

```
Terminal: Local +
warning: LF will be replaced by CRLF in app/src/test/java/com/example/practica_1/ExampleUnitTest.java.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in build.gradle.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in gradle.properties.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in gradlew.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in settings.gradle.
The file will have its original line endings in your working directory
C:\Users\lucage\AndroidStudioProjects\Practica_1>git commit -m "Este es mi primer commit"
```

Para poder hacer la comprobación del control de versiones debemos realizar un cambio en nuestro proyecto.



```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

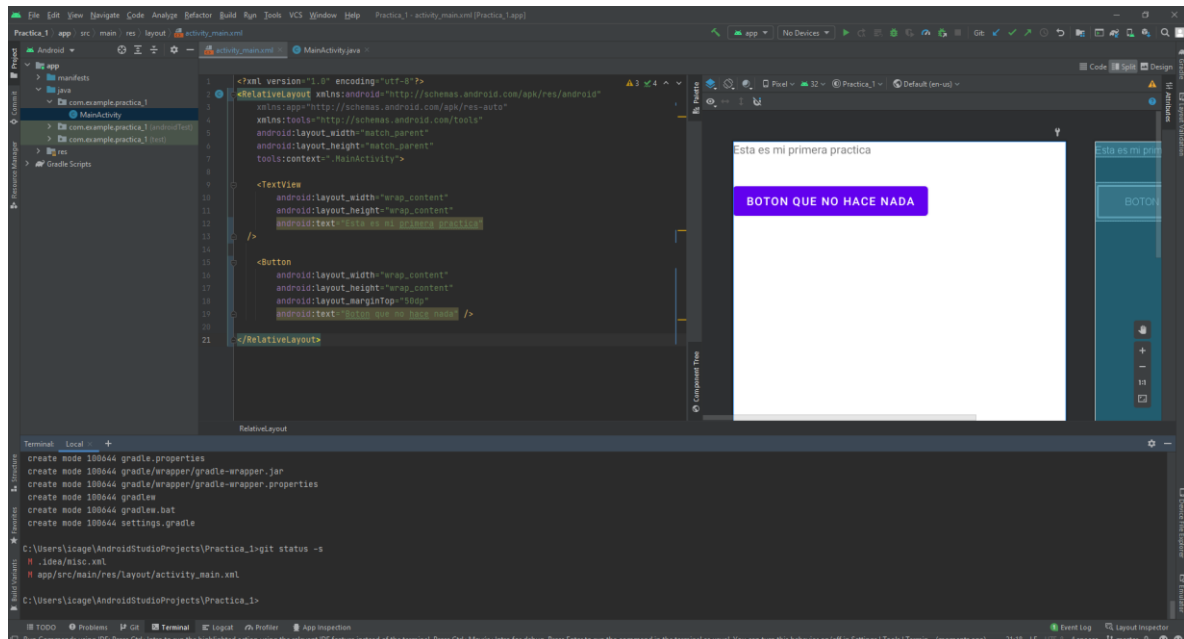
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Esta es mi primera practica"
    />

    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="100px"
        android:text="BOTON QUE NO HACE NADA"
    />

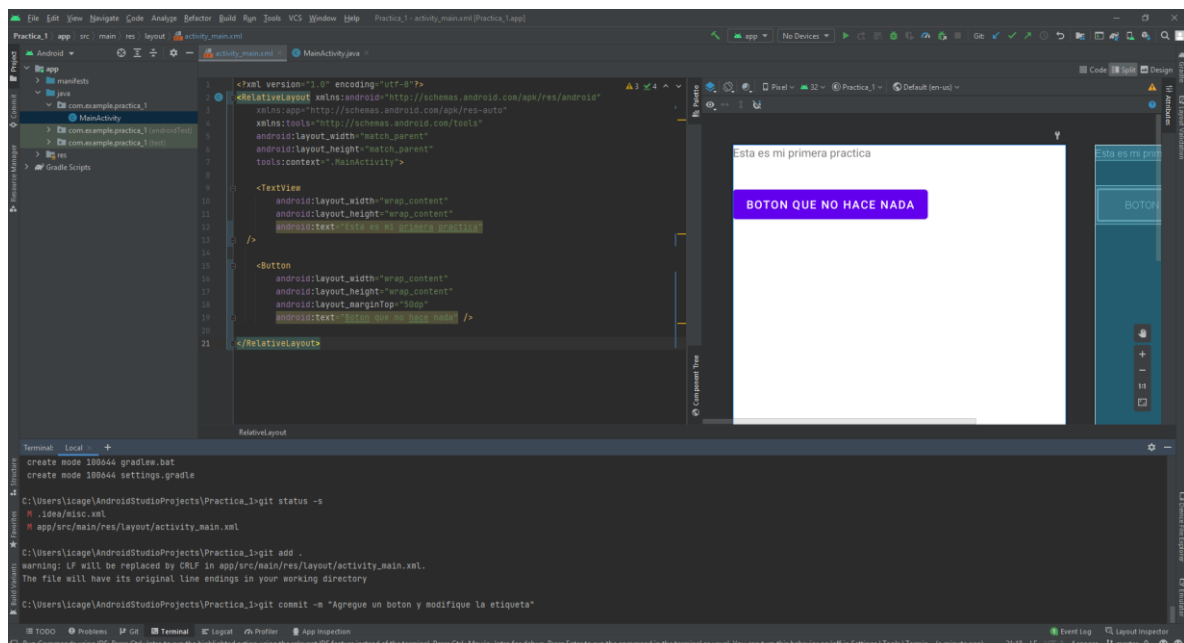
</RelativeLayout>
```

```
Terminal: Local +
create mode 100644 app/src/main/res/values/strings.xml
create mode 100644 app/src/main/res/values/themes.xml
create mode 100644 app/src/test/java/com/example/practica_1/ExampleUnitTest.java
create mode 100644 build.gradle
create mode 100644 gradle.properties
create mode 100644 gradle/wrapper/gradle-wrapper.jar
create mode 100644 gradle/wrapper/gradle-wrapper.properties
create mode 100644 gradlew
create mode 100644 gradlew.bat
create mode 100644 settings.gradle
C:\Users\lucage\AndroidStudioProjects\Practica_1>
```

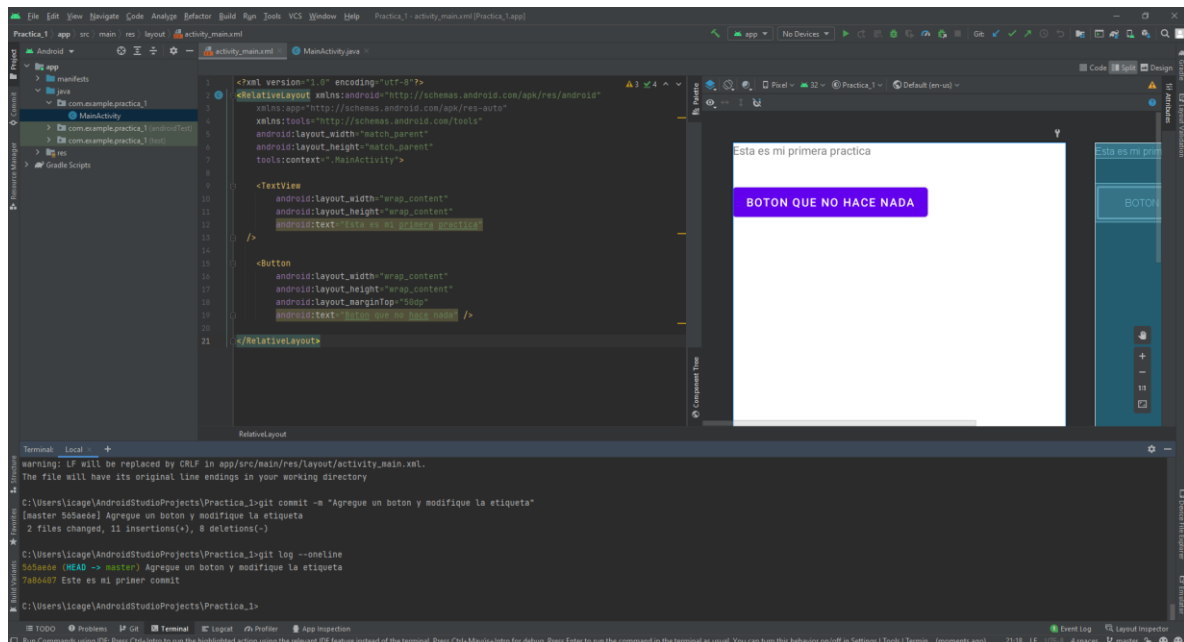
Una vez hecho el cambio podemos comprobar los datos que fueron modificados escribiendo en la terminal 'git status -s'



Este cambio lo vamos a acompañar con otro commit, este se llamó 'Agregue un botón y modifique la etiqueta'

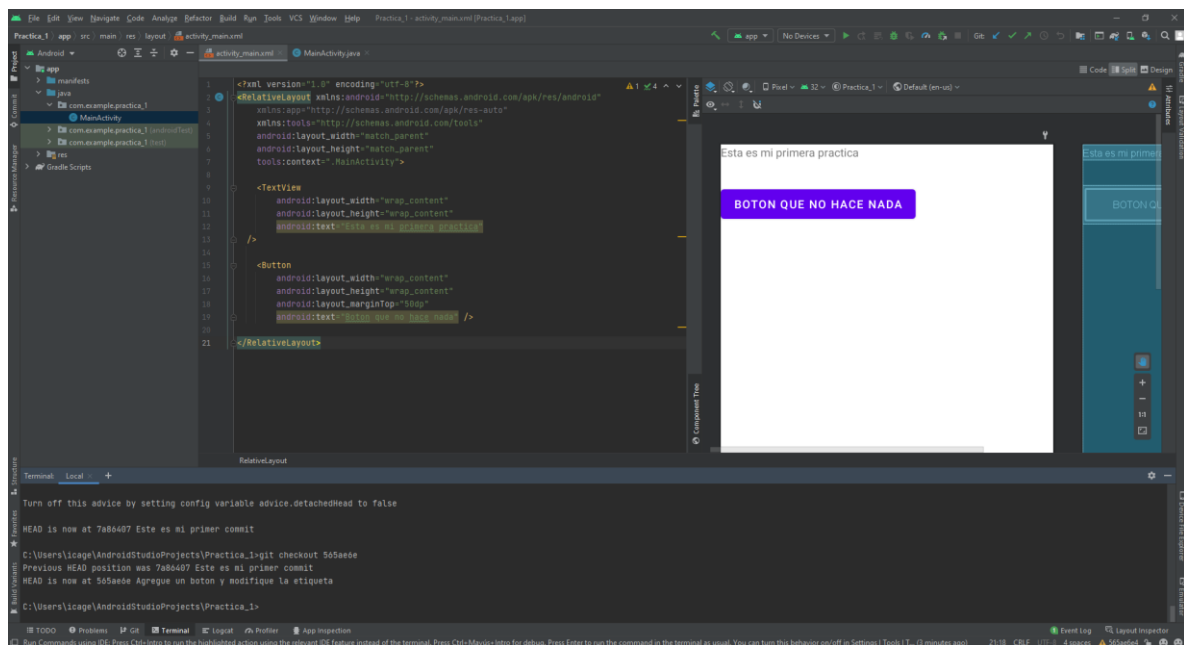


Para poder ver el historial de todos los commits realizados en el proyecto debemos escribir en la terminal 'git log --oneline' esto nos mostrara nuestros commits acompañados de una clave que lo identifica



The screenshot shows the Android Studio IDE. The main editor displays the 'activity_main.xml' file, which contains an Android layout with a text view and a button. The text view has the text 'Esta es mi primera practica' and the button has the text 'BOTON QUE NO HACE NADA'. The terminal at the bottom shows the output of the 'git log --oneline' command, which lists two commits: '565e0e Agregue un boton y modifique la etiqueta' and '7a86407 Este es mi primer commit'. The terminal also shows the command 'git checkout 565e0e' being executed.

Para elegir el commit al que deseemos regresar debemos escribir en la terminal 'git checkout' acompañado de la clave del commit al que deseemos regresar, ejemplo git checkout 7a86407' que representa el commit 'Este es mi primer commit'



The screenshot shows the Android Studio IDE. The main editor displays the 'activity_main.xml' file, which contains an Android layout with a text view and a button. The text view has the text 'Esta es mi primera practica' and the button has the text 'BOTON QUE NO HACE NADA'. The terminal at the bottom shows the output of the 'git checkout 565e0e' command, which indicates that the HEAD is now at 565e0e and that the commit 'Agregue un boton y modifique la etiqueta' has been checked out. The terminal also shows the command 'git log --oneline' being executed, which lists the same two commits as in the previous screenshot.