v. 4, n. 15, 2024

Journal of
Engineering
Research

PROPUESTA DE MÉTODO PARA CREAR UNA APLICACIÓN MÓVIL CON TABS EN ANDROID STUDIO

Leopoldo Rangel Madrigal
TecNM CRODE Celaya
Celaya, Guanajuato México

Resumen: En la actualidad, en la mayoría de los entornos de desarrollo de aplicaciones móviles nos proporcionan diversas herramientas que nos permiten organizar de distinta forma la información que se le presenta al usuario, una de ellas muy utilizada es mediante Tabs, sin embargo en diversas ocasiones el entorno nos genera todo el código para crear dichos Tabs, lo que provoca que se genere mucho código innecesario, lo que conlleva a que cuando se requiere hacer modificaciones muy específicas no es tan fácil de realizarlas y el manteamiento de la aplicación se llega a dificultar. El presente trabajo propone la creación, mediante Android Studio de Tabs manejando ViewPager2, FrameLayout, la clase FragmentStateAdapter, también se podrá controlar los componentes que se incrusten en uno de los fragments mediante programación, todo esto sin utilizar plantillas o generadores de código es decir hacerlo totalmente desde cero, para tener todo el control y manipulación de dichos componentes.
Palabras-clave: Android, Tabs, desarrollo, aplicación, Fragment y ViewPager2

INTRODUCCIÓN

El desarrollo de aplicaciones móviles va en aumento, por lo que en la actualidad los distintos Entornos de Desarrollo Integrado (IDE) nos proporcionan una gran gama de herramientas brindándonos diversas posibilidades para crear de distintas formas nuestras interfaces gráficas, solo que cuando se trata de utilizar componentes un poco complejos como son los Tabs "vistas deslizantes que permiten navegar entre pantallas del mismo nivel, como pestañas, con un gesto horizontal del dedo o deslizamiento. Este patrón de navegación también se conoce como paginación horizontal." (Android Developers Febrero 2024) usualmente el IDE nos proporciona utilizar un generador de código para su creación lo que provoca que nos genera código que en la mayoría de los casos no se ocupa (incensario), y al momento de hacer cambios específicos o dar mantenimiento a la aplicación el haberlo creado de esta manera nos complica el poder llevar acabo esta tarea.

En ocasiones el mejor camino cuando se van a utilizar Tabs es crearlos desde cero es decir sin utilizar generador de código del IDE, ya que de esta forma tenemos el control de todo el aspecto gráfico y de lógica, facilitado la tarea del mantenimiento de la aplicación.

En el presente trabajo propone un método basado en el trabajo de Rishabh Kumar para crear en Android Studio una aplicación móvil que maneje tres Tabs utilizando ViewPager2, FrameLayout y la clase FragmentStateAdapter también se podrá manipular los componentes que se incrusten en uno de los fragments mediante programación, sin utilizar el template que nos proporciona IDE para así tener un mejor control. Cabe señalar que "ViewPager2 reemplaza a androidx.viewpager.widget.ViewPager, abordando la mayoría de los puntos débiles de su predecesor, incluida la compatibilidad con el diseño de derecha a izquierda, la orientación vertical, las colecciones de fragmentos modificables, etc." (Android Developers Mayo 2024)

METODOLOGÍA

ENTORNO DE DESARROLLO

El método propuesto fue desarrollado en Android Studio Giraffe 2022.3.1 con la API 34, utilizando lenguaje de programación JAVA y fue probado con un emulador con API 34

DESCRIPCIÓN DEL MÉTODO

De forma general el método consiste en crear tres Fragment Blanck que nos servirá como los tres tabs (Contactos, llamadas y mensajes) que tendrá nuestra aplicación móvil, en el activity_main.xml tendremos

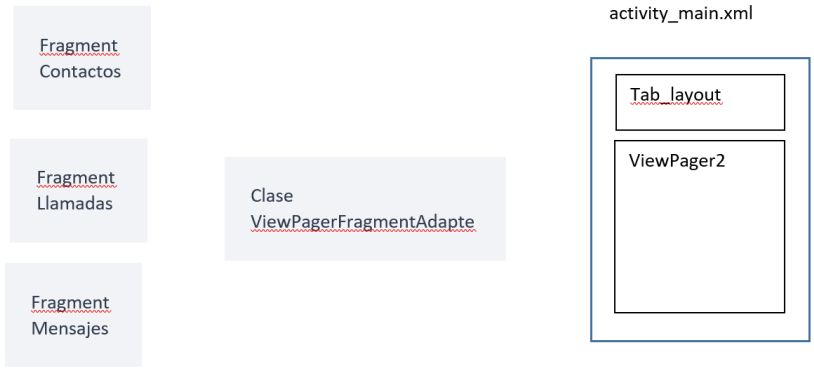

Imagen 1

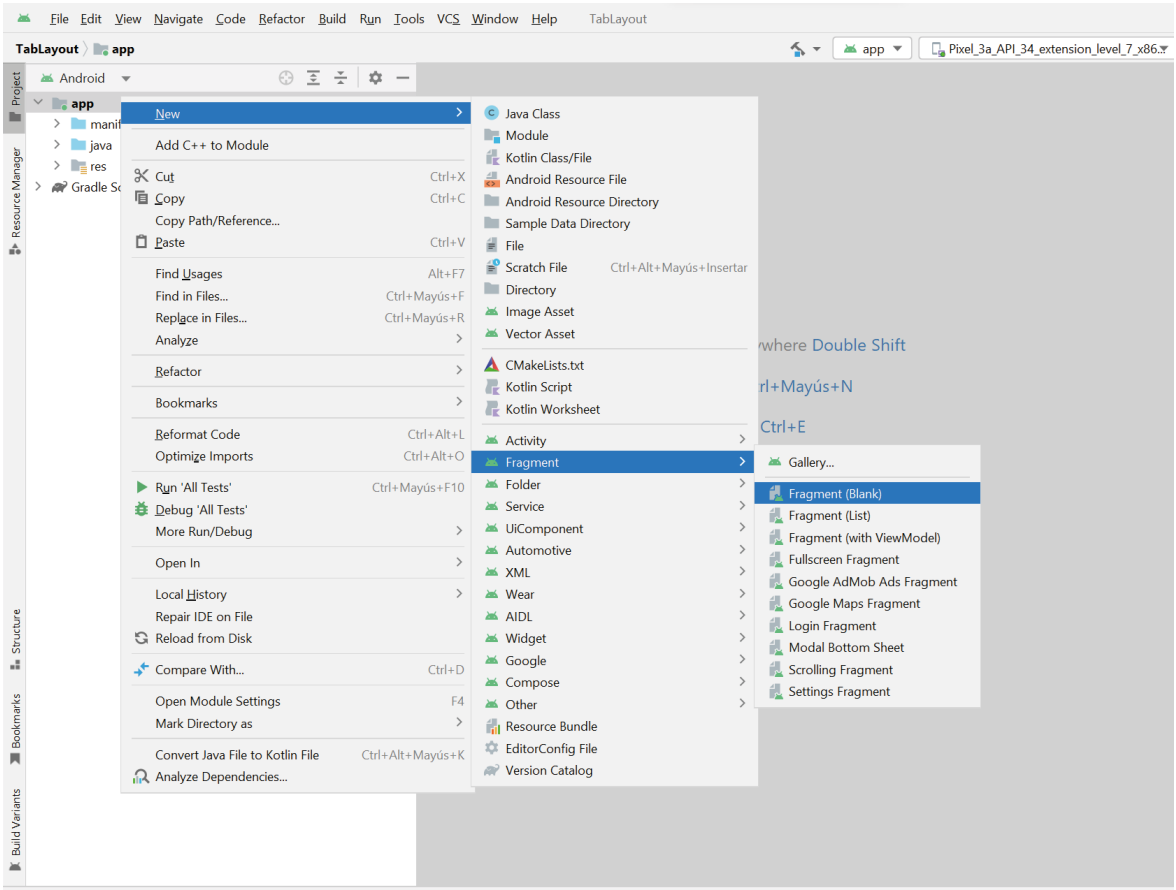

Imagen 2

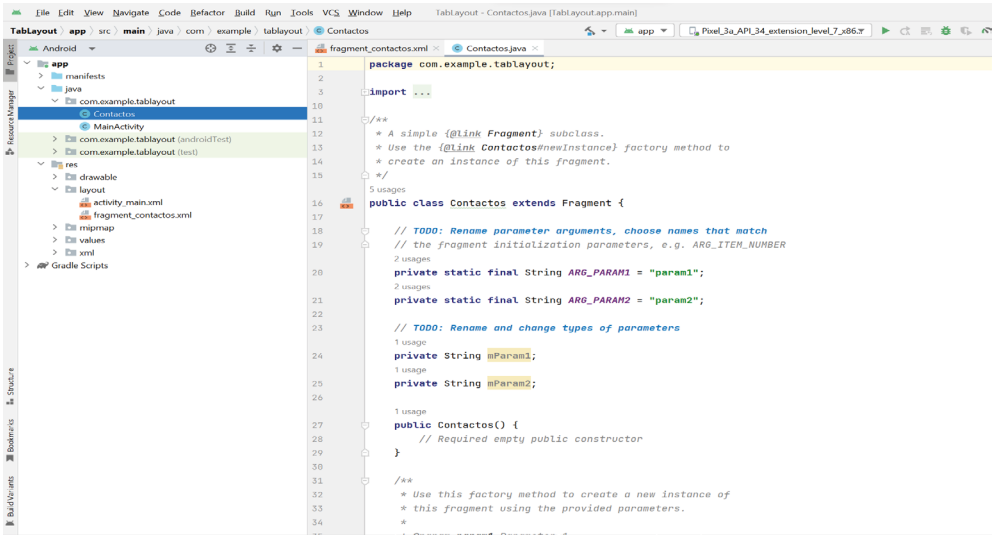

Imagen 3

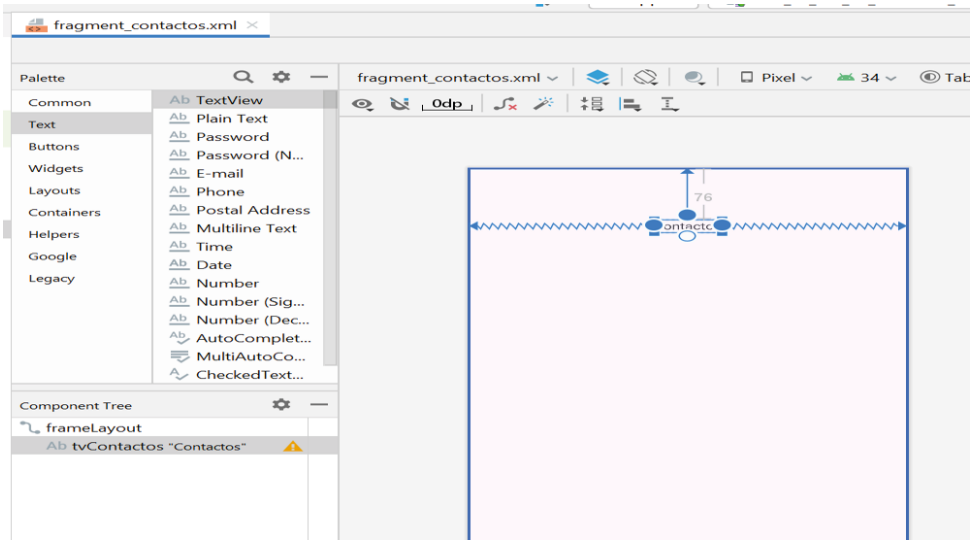

Imagen 4

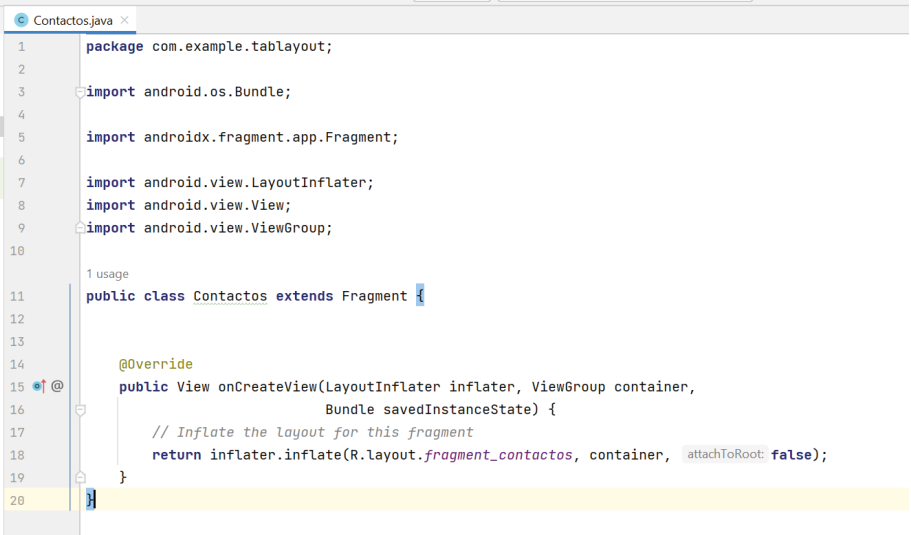

Imagen 5

tab_layout que muestra el nombre de cada pestaña y view_pager2 en el que cargaremos el fragment que seleccione el usuario, y por ultimo a través de una clase que llamaremos ViewPagerFragmentAdapter nos servirá para controlar que fragment cargaremos en el ViewPager2

Por ultimo en el fragment Contactos tendremos un formulario el cual podrá llenar el usuario y un botón "Limpiar", el cual, al tocar borrar la información ingresada, con esta simple funcionalidad se mostrar la forma de controlar los componentes que estén incrustados en un fragment de un Tab en particular.

CREACIÓN DE LOS 3 FRAGMENTA

Partiremos de un nuevo proyecto con una Empty Activity, después crear los Fragment para nuestras pestañas, como va a contener 3 pestañas, será necesario crear tres Fragment, el primer Fragment Black con el nombre de Contactos

Se generarán dos archivos **fragment_contactos.xml** y **Contactos.java** como se muestra en la imagen 3

Vamos a modificar la parte grafica de nuestro fragment creado, convertir el **FrameLayout** en un **ConstraintLayout,** y colocar un nuevo **TextView** en el **ConstraintLayout** quedado de la siguiente forma imagen 4

Modificar la parte lógica del Fragment, abrir el archivo **Contactos.java,** después de borra todo el código incensario nos deberá quedar como en la imagen 5 el archivo Contactos.java

Repetir los pasos, pero ahora con el nombre de mensajes y llamadas respectivamente. De tal forma que tendremos tres tabs con el mismo diseño. El proyecto tendrá la siguiente estructura imagen 6

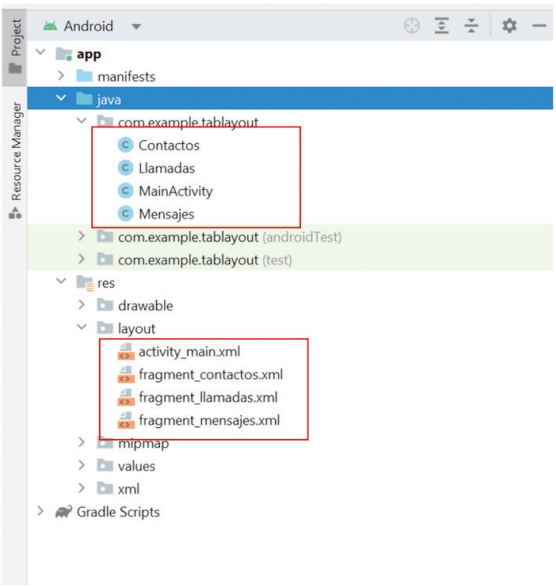

Imagen 6

CREACIÓN DE LA CLASE VIEWPAGERFRAGMENTADAPTER

Crear una nueva clase, que se llamará **ViewPagerFragmentAdapter**

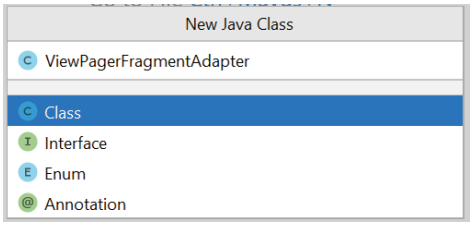

Imagen 7

Extender la clase **FragmentStateAdapter** como se muestra en la imagen 8

Implementar los métodos **createFragment**, **getItem y** el constructor de la clase**,** quedado de la siguiente forma imagen 9

Modificaremos el método **createFragment** para que dependiendo de la posición que el usuario seleccione, creara el **fragment** correspondiente, y el método **getItemCount** para que regrese el número de tabs que estamos utilizado para este caso regresamos el valor de 3

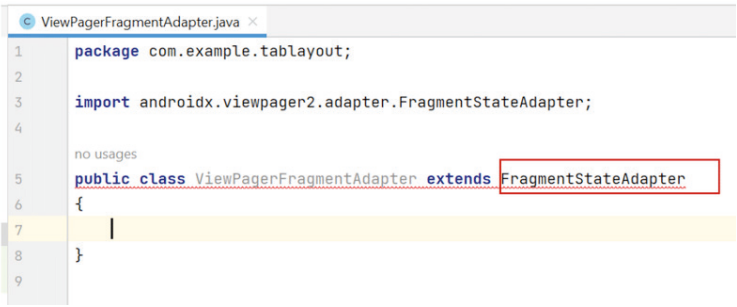

Imagen 8

Imagen 9

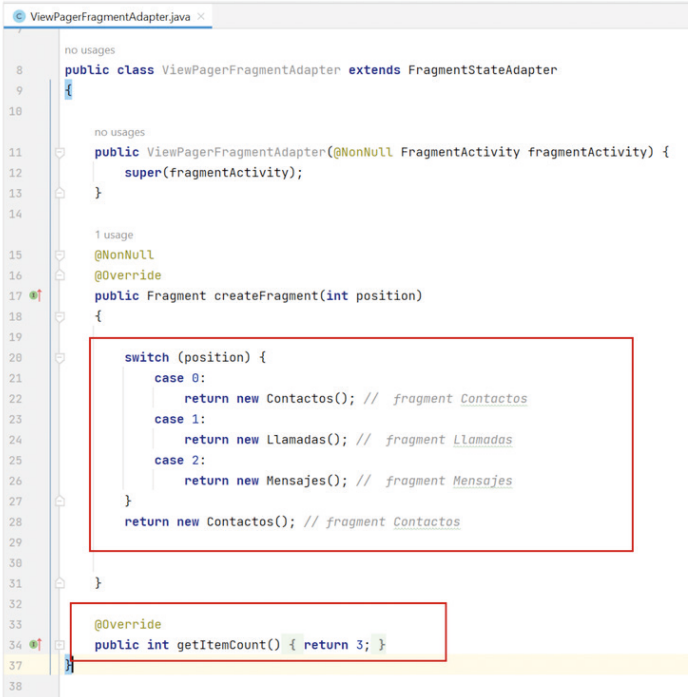

Imagen 10

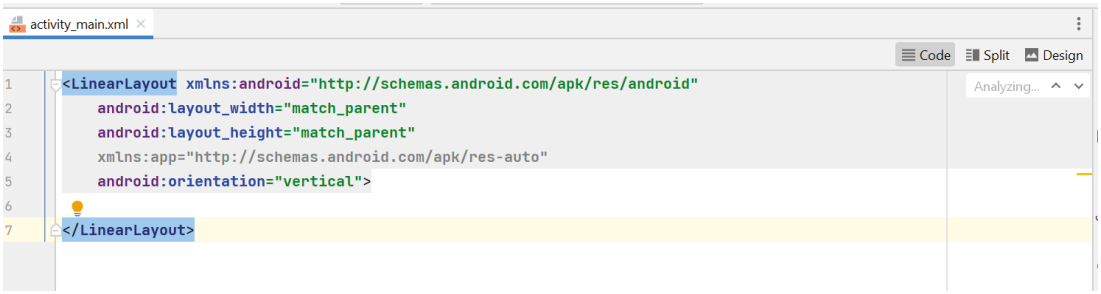

Imagen 11

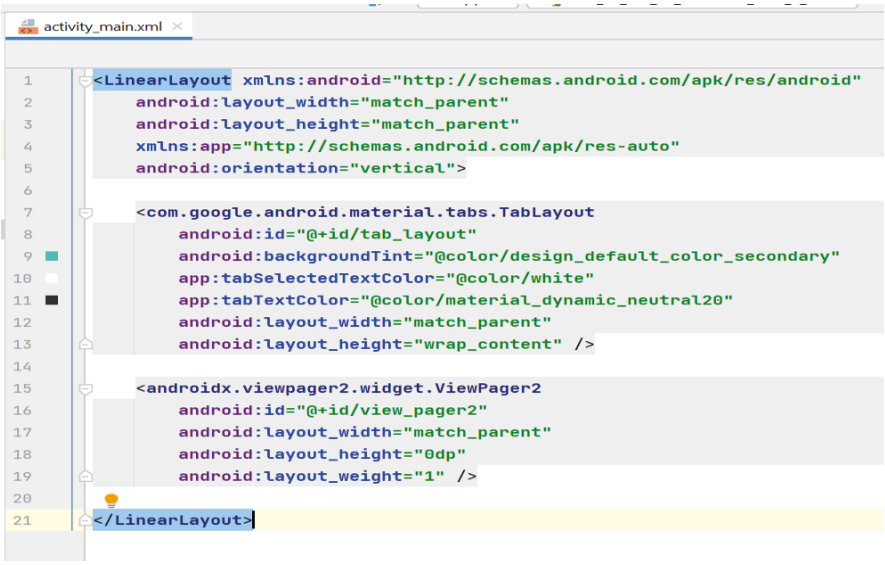

Imagen 12

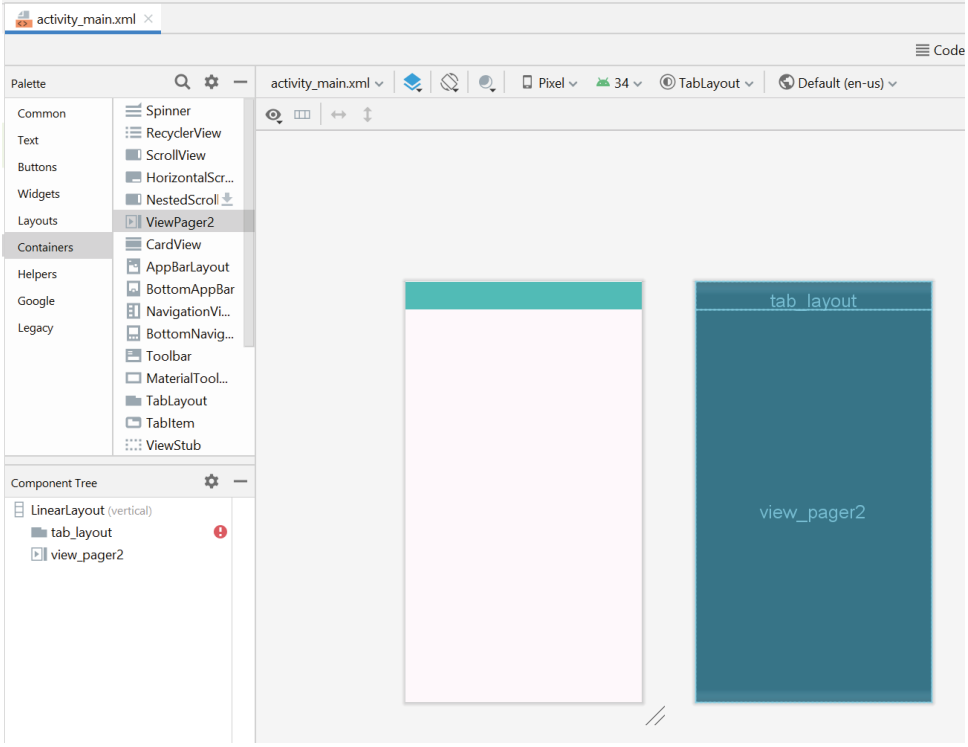

Imagen 13

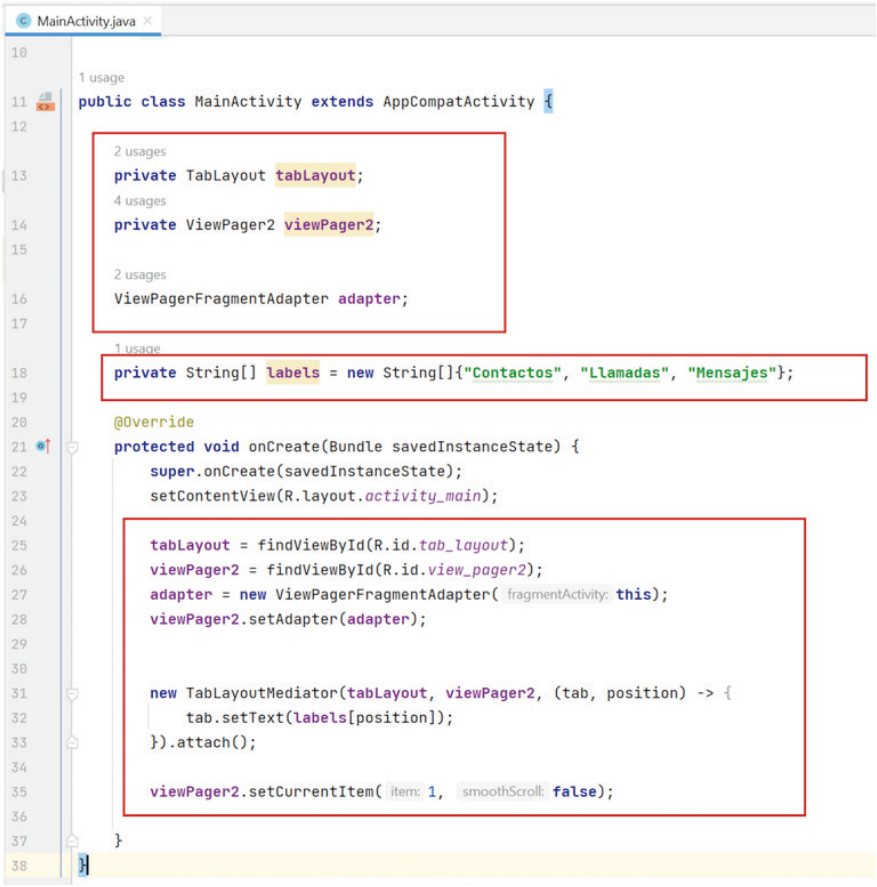

Imagen 14

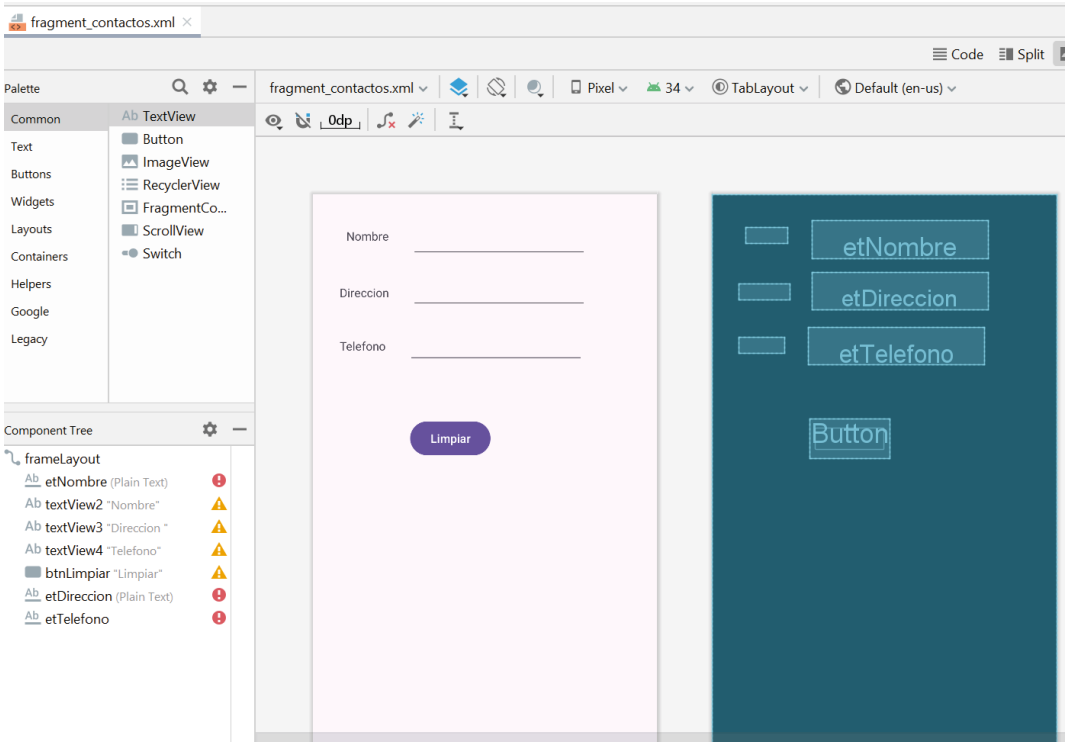

Imagen 15

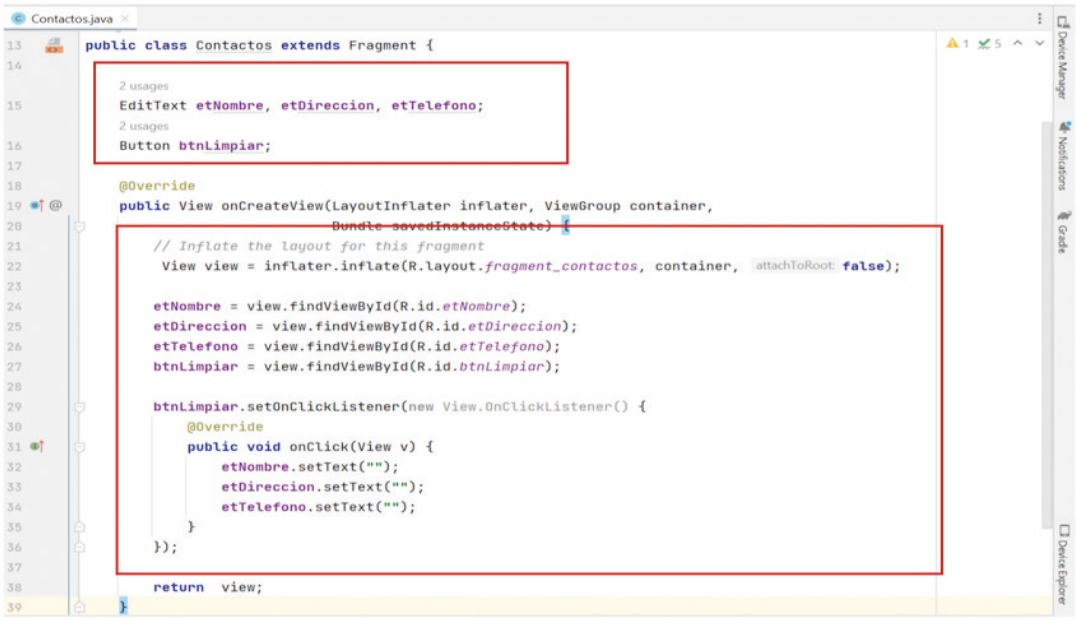

Imagen 16

CREACIÓN DE LA INTERFAZ GRAFICA

Modificar la parte grafica de nuestro **activity_main.xml** y crear un **LinearLayout** vertical desde código como se muestra en la imagen 11

Dentro del **LinearLayout** crear y configurar un **TabLayout** y un **ViewPager2** como se muestra imagen 12, el componente **ViewPager2** será sobre el cual se creará la interfaz gráfica del fragment que llamemos y el **TabLayout** nos mostrará las etiquetas de cada uno de los tabs

Quedado de la siguiente forma en la parte grafica imagen 13

PROGRAMACIÓN DE LA MAINACTIVITY.JAVA

Abrir el archivo MainActivity.java, declaramos la variable que vamos a ocupar, enlazamos las variables y a través de la clase **TabLayoutMeditor** vamos a asignar las etiquetas que tendrá cada uno de los Tabs

El código final quedara de la siguiente forma.

CONTROLAR COMPONENTES DENTRO DE LOS FRAGMENTS

Modificar el archivo contactos.java de la siguiente forma:

Modificar parte grafica del fragment Contactos quedado de la siguiente forma

Declaramos las variables para poder hacer el enlace de sus respectivos ids, es necesario crear un objeto de la clase View y a partir de ese objeto enlazar las variables, lo último que faltaría seria utilizar el método setOnClickListener, para programar la lógica de que al dar click sobre el botón que dice limpiar, se borre la información que se tenga en las cajas de texto, quedando el código final de la siguiente forma imagen 16

RESULTADOS

Probamos la aplicación y deberá mostrarnos los tres Tabs Contactos, llamadas y mensajes, cada vez que seleccionemos un tab nos creará y mostrará el Fragment correspondiente con su interfaz gráfica, de tal forma que nos mostrará un formulario en Contactos y en las pestañas de mensajes y llamadas solo mostrar el nombre de la pestaña correspondiente.

Cabe señalar que también se puede cambiar de tabs simplemente arrastrando la pantalla de izquierda a derecha o viceversa, ya que es una de las funcionalidades que nos permite el **ViewPager2**

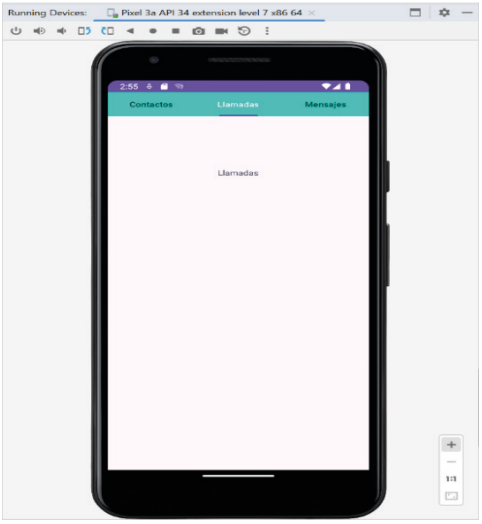

Imagen 17

Para probar la funcionalidad de la pestaña Contactos llenamos los campos con cierta información y damos click sobre limpiar y dicha información se deberá borrar.

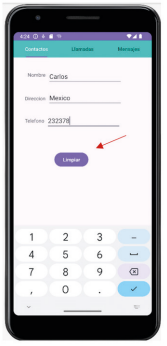

Imagen 18

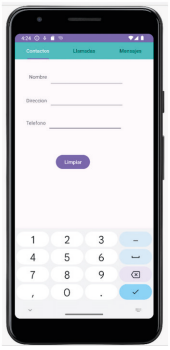

Imagen 19

CONCLUSIÓN

El presente trabajo proporciono un método para poder crear una aplicación móvil con manejo de Tabs creados desde cero sin utilizar el templetes de Android Studio, lo que nos permitió tener un mejor entendimiento del código y más limpio, por lo cual nos facilitara hacer cambios que se requieran en la aplicación, por otra parte se explicó la forma de poder controlar los componentes que se incrustan en los tabs, con un ejemplo de funcionalidad muy simple, pero que se puede escalar a cosas más complejas siendo la base lo que se explicó en este trabajo, el lector puede tomar este método como una base para poder utilizarlo en diversas aplicaciones con distintos propósitos utilizando los tabs.

REFERENCIAS

• Rishabh Kumar (Junio 2022), Creating Whatsapp Like Tabs With New ViewPager2 Android, https://www.codewithrish.com/creating-whatsapp-like-tabs-with-new-viewpager2-android

• Android Developers (Febrero 2024) https://developer.android.com/guide/navigation/navigation-swipe-view?hl=es-419

• Android Developers (Mayo 2024) https://developer.android.com/reference/kotlin/androidx/viewpager2/widget/ViewPager2