

**INSTITUTO POLITÉCNICO NACIONAL**

**ESCUELA SUPERIOR DE CÓMPUTO**

**ALUMNO:**

**ALDAVERA GALLAGA IVÁN**

**APPLICATION DEVELOPMENT FOR MOBILE DEVICES**

**PROFESOR:**

**ALEJANDRO SIGFRIDO CIFUENTES ÁLVAREZ**

**TÍTULO DEL REPORTE**

**PLANTILLAS**

**FECHA DE REALIZACIÓN:**

**25 – JUNIO – 2020**

**VERSIÓN DEL REPORTE:**

**V1**

**ÍNDICE**

OBJETIVO………………………………………………………………..1

CONCEPTOS…………………………………………………………….1

DESARROLLO…………………………………………………………...1

LISTADO DE SOFTWARE……………………………………….2

LISTADO DE HARDWARE………………………………………2

PRUEBAS…………………………………………………………………2

CONCLUSIONES………………………………………………………..4

BIBLIOGRAFÍA…………………………………………………………...4

**OBJETIVO**

Conocer los distintos tipos de plantillas o layouts que maneja Android.

**CONCEPTOS**

**Layout:** sirve para hacer referencia al esquema que será utilizado y cómo están distribuidos los elementos y formas dentro de un diseño. Es un vocablo del idioma inglés, y no existe o no forma parte del texto del diccionario de la Real Academia Española, sin embargo, se traduce como "disposición, plan o diseño".

**RelativeLayout** es un grupo de vistas que muestra vistas secundarias en posiciones relativas. La posición de cada vista puede especificarse como relativa a elementos equivalentes (como a la izquierda o por debajo de otra vista) o en posiciones relativas al área RelativeLayout superior (como alineada a la parte inferior, izquierda o central).

**LinearLayout** es un grupo de vista que alinea todos los campos secundarios en una única dirección, de manera vertical u horizontal.

**FrameLayout** está diseñado para bloquear un área en la pantalla o para mostrar un solo elemento. En general, FrameLayout debe usarse para mantener una vista secundaria única, porque puede ser difícil organizar las vistas secundarias de una manera que sea escalable a diferentes tamaños de pantalla sin que los elementos secundarios se superpongan entre sí

**TableLayout** es un objeto ViewGroup que muestra elementos View secundarios en filas y columnas. TableLayout posiciona sus elementos secundarios en filas y columnas. Los contenedores de TableLayout no muestran líneas de bordes para las filas, columnas o celdas. La tabla tendrá la misma cantidad de columnas que la fila con la mayor cantidad de celdas. Una tabla puede dejar celdas vacías. Las celdas pueden abarcar varias columnas, al igual que en HTML.

**GridLayout** inicialmente parece como si es una forma de crear tablas, muy parecido a TableLayout (android.widget.TableLayout). Sin embargo, es mucho más flexible que el control TableLayout. Por ejemplo,las celdas pueden abarcar filas, a diferencia de TableLayout. Es flexible, sin embargo, viene el hecho que en verdad ayuda a alinear objetos a través de las líneas verticales de la cuadrícula creada mientras construimos una vista con GridLayout.

**Scrollview** es un grupo de vista que permite desplazar la jerarquía de vista ubicada dentro de él. La vista de desplazamiento puede tener solo un hijo directo colocado dentro de ella.

**DESARROLLO**

Para el desarrollo de esta práctica se utilizó el código proporcionado por el profesor para los siete distintos tipos de layouts, los cuales funcionaron sin ningún problema.

**LISTADO DE SOFTWARE:**

* Java Development Kit 8: Necesario para el desarrollo de aplicaciones Android.
* Android Studio 4.0: IDE para el desarrollo de aplicaciones Android
* Windows 10: Sistema operativo sobre el que corren los anteriores programas.
* Android 10. Sistema Operativo sobre el que corre la aplicación

**LISTADO DE HARDWARE:**

* Laptop HP con GB RAM DDR4 y procesador Intel Core i3
* Dispositivo Móvil Motorola One con 4GB de RAM y procesador SoC Qualcomm Snapdragon 625
* Cable tipo C para la transferencia de datos.

**PRUEBAS**

|  |  |  |
| --- | --- | --- |
|  |  |  |
| Ejemplo 1. RelativeLayout | Ejemplo 2.  LinearLayout | Ejercicio 3.  LinearLayout dividido |

|  |  |  |
| --- | --- | --- |
|  |  |  |
| Ejemplo 4.  FrameLayout | Ejemplo 5.  TableLayout | Ejemplo 6.  GridLayout |

|  |
| --- |
|  |
| Ejemplo 7. Scrollview. |

**CONCLUSIONES**

Los layouts nos permiten acomodar las cosas de manera que estas se vean mejor en la pantalla, se pueden utilizar varios layouts al mismo tiempo y algunos tienen ciertas especificaciones a utilizar.

**BIBLIOGRAFÍA**

* <https://www.significados.com/layout/>