**SERVICIO NACIONAL DE APRENDIZAJE – SENA**

Logotipo

Descripción generada automáticamente

**CENTRO DE COMERCIO REGIONAL ANTIOQUIA**

**TECNOLOGÍA EN ANÁLISIS Y DESARROLLO DE SOFTWARE - 2675805**

**Evidencia de producto: GA5-220501095-AA1-EV08 - Maquetación de la Interfaz gráfica en XML -Android**

**DANIEL FELIPE ARIAS CORREDOR**

**2023**

*Introducción*

Este documento tiene como propósito presentar los resultados de la consulta realizada acerca del lenguaje XML, así como elementos para el desarrollo de aplicaciones en Android. Primero se realizará una definición de XML, seguido de los lenguajes de programación mas usados para Android. Tercero, se tratarán los layouts más usados para Android. Cuarto, se hablará de diseño responsivo en aplicaciones web y por último se mencionarán los entornos de desarrollo o IDEs mas usados en desarrollo de interfaces de usurario Android.

*¿Qué es XML?*

XML o Extensive Markup Language es un lenguaje de marcado similar a HTML, pero a diferencia de este que es usado para dar estructura a una interfaz web y tiene etiquetas predefinidas, XML no tiene etiquetas predefinidas y su uso está enfocado en servir como estándar para compartir información entre diferentes sistemas. XML requiere una correcta escritura para salvaguardar la integridad de la información y para que sea entendible entre sistemas, por lo que se espera que las etiquetas estén correctamente cerradas y se respete su jerarquía. La figura 1 presenta un ejemplo de código XML.

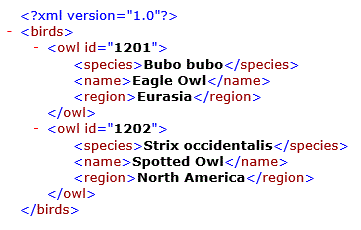


Figura . Ejemplo de código xml. Recuperado de: https://help.seagullscientific.com/2022/es/Content/Resources/Images/XML\_sample\_file.GIF

El uso más común de XML es la transferencia de datos, normalmente dos sistemas o aplicaciones compartirán información en formato XML o JSON, siempre que sea por mutuo acuerdo. Adicional, permite dar estructura a la información que es presentada en interfaces de usuario y a poder ser filtrada según etiquetas XML.

Este lenguaje puede ser escrito en editores, navegadores web o mediante librerías de otros lenguajes de programación que permitan transformar información a código XML y viceversa.

*Lenguajes para el desarrollo de aplicaciones Android*

Se debe tener en cuenta que Android al ser un sistema operativo no cuenta con un lenguaje propio de programación para sus aplicaciones, sin embargo, existen lenguajes y tecnología populares y ampliamente extendidos para desarrollo de aplicaciones en Android. Entre los lenguajes de programación mas populares para el desarrollo de aplicaciones Android se encuentran:

* Kotlin: Es un lenguaje de programación diseñado por JetBrains, considerado el oficial para el desarrollo de aplicaciones Android. Permite crear código sencillo, pero eficiente. Google recomienda usar Kotlin para desarrollar proyectos en Android
* Java: Lenguaje de programación multipropósito de alto nivel, mediante el cual se ha desarrollado gran parte de Android. Es una de las opciones mas populares ya que ha sido el lenguaje de desarrollo de aplicaciones Android por defecto.
* JavaScript: Lenguaje bastante usado para crear aplicaciones híbridas, especialmente junto con frameworks como ReactNative o vue js. Usado universalmente en la creación de interfaces de usuario para añadir dinamismo e interactividad.
* C#: Opción de lenguaje común para aplicaciones Android híbridas

Cabe resaltar que estos lenguajes suelen ser usados junto con frameworks y entornos de desarrollo que permiten una construcción de aplicaciones Android de manera sencilla y estructurada.

*Layouts para Android*

Para Android un Layout es un contenedor donde se pueden colocar elementos de diseño de una interfaz de aplicación tales como imágenes, formularios, widgets, etc. El layout establece el orden visual de estos elementos. Los layout pueden ser creados o cargados a través de archivos XML. Normalmente se trabajan en IDEs o editores como Android Studio. Existen diferentes tipos de layout para ordenar los elementos de una interfaz, los más usados son:

* Linear layout: Apila todos los elementos debajo del anterior o también en horizontal

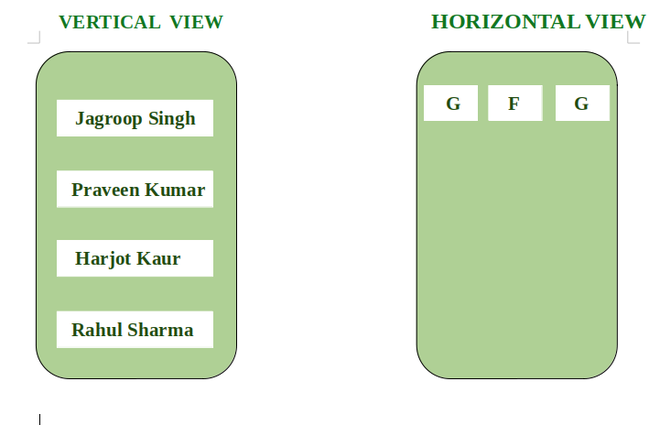


Figura . Linear layout. Recuperado de: https://media.geeksforgeeks.org/wp-content/uploads/20210127084730/LLayout-660x425.png

* Relative layout: Posiciona un elemento de forma relativa a otro, es decir a la derecha de…, diagonal a…, siempre teniendo un punto de referencia.

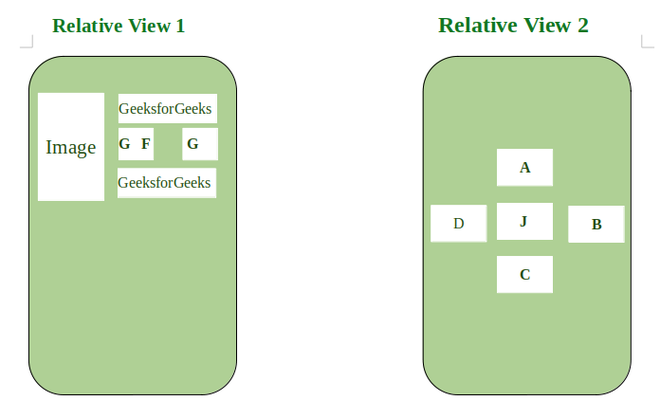


Figura . Relative layout. Recuperado de: https://media.geeksforgeeks.org/wp-content/uploads/20210127085726/RLayoutView-660x419.png

* Table layout: Consiste en una especie de tabla o rejilla donde se pueden poner elementos entre filas y columnas. El resultado final consiste en un conjunto ordenado de elementos como si se tratara de una tabla.

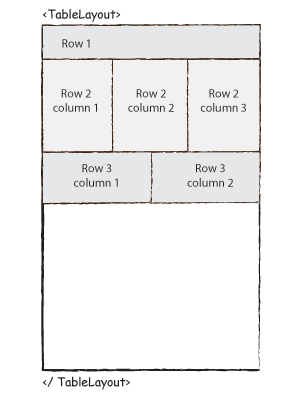


Figura . Table layout. Recuperado de: https://www.tutorialspoint.com/android/images/table.jpg

* Absolute layout: Posiciona los elementos de manera absoluta en el espacio destinado a la interfaz, de modo que la posición de un objeto viene dada por coordenadas en pixeles.

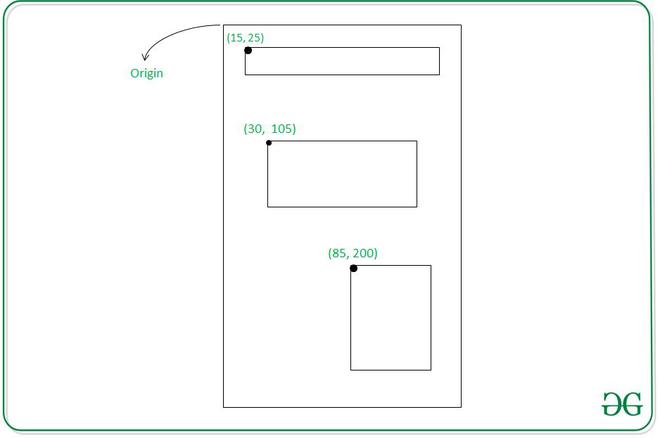


Figura . Absolute layout. Recuperado de: https://media.geeksforgeeks.org/wp-content/uploads/20201124234331/absolutelayout-660x438.JPG

*Diseño responsivo*

Actualmente se espera que una página web pueda visualizarse correctamente en cualquier dispositivo. El termino diseño responsivo se refiere a unas interfaces web que se adaptan gráficamente en tamaño y distribución de sus elementos, a diferentes tipos de dispositivos donde se puedan visualizar. Por tanto, un sitio web responsivo puede visualizarse de manera clara y cómodo tanto desde un monitor de escritorio, como desde una Tablet o un teléfono. Como desventaja tiene que su planificación y desarrollo requieren de tiempo y un presupuesto elevado si se crea desde cero. En caso de aplicaciones antiguas, adaptar un diseño responsivo puede resultar complejo.

Es de resaltar que un sitio web responsivo no tiene una página web diferente para cada tipo de pantalla, sino que una misma interfaz se adapta a cada una de ellas. La figura 6 presenta un ejemplo de interfaz responsiva, donde una misma página puede adaptarse al tamaño de pantalla de diferentes dispositivos y ser clara y cómoda de ver tanto en pantalla de escritorio, como computador portátil, Tablet y smartphone.

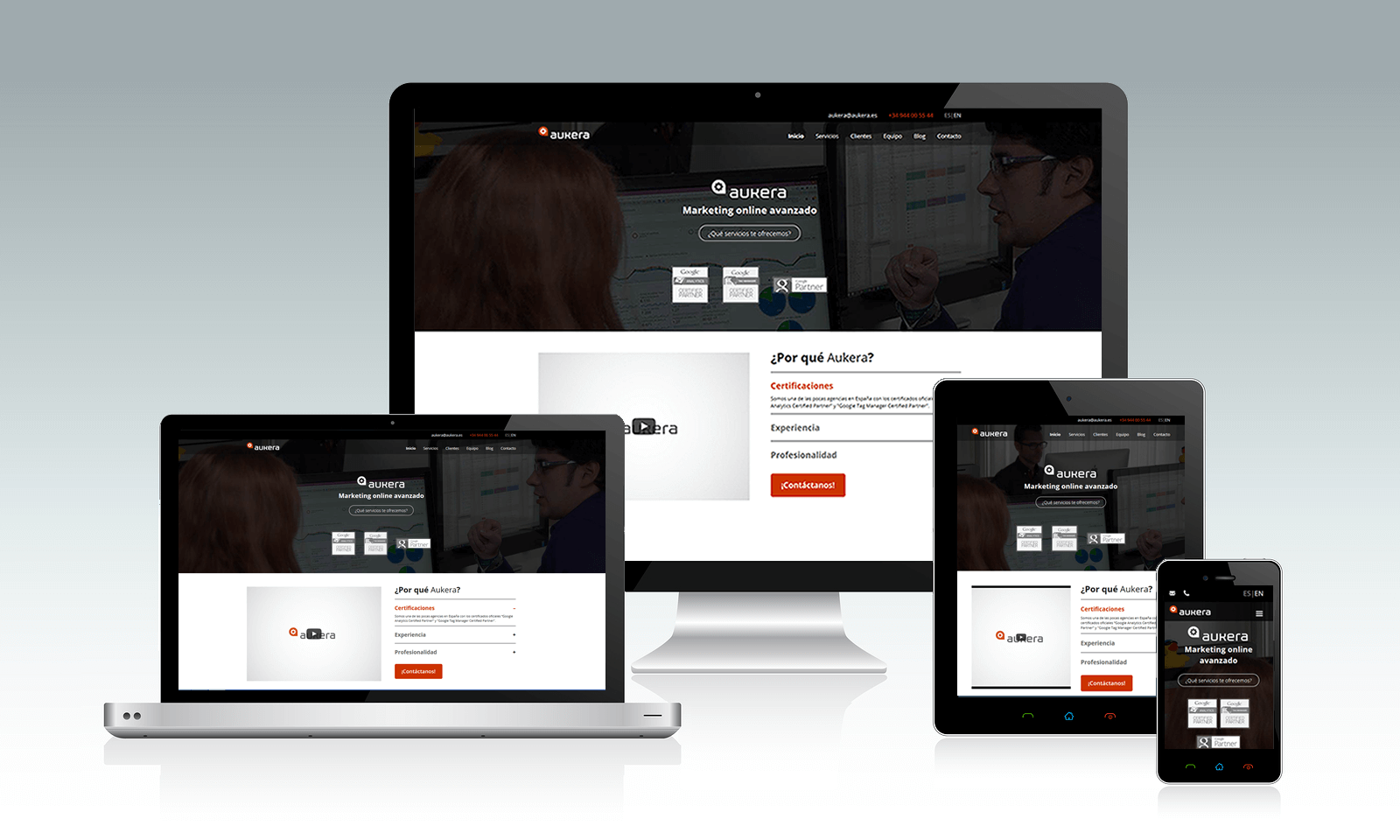


Figura . Ejemplo de diseño web responsivo. Recuperado de: https://aukera.es/blog/imagenes/imagen11.png

*IDEs para el diseño de interfaces de usuario Android*

En este apartado se tratarán los IDEs o entornos de desarrollo mas comunes a la hora de crear layouts de interfaces para una aplicación Android. Entre estos entornos resaltan:

* Android Studio: Entorno de desarrollo mas famoso para trabajar aplicaciones Android. Su característica mas notable son sus emuladores o dispositivos virtuales, que permite probar la aplicación sin necesidad de un dispositivo físico real. Presenta además un editor de diseño virtual, donde se permite mover y arrastrar elementos de layouts en lugar de tener que escribir código desde cero. Los resultados finales se pueden comprobar con diversos dispositivos, así como distintas versiones de Android. La figura 7 presenta une jmplode desarrollo de un layout en Android Studio.

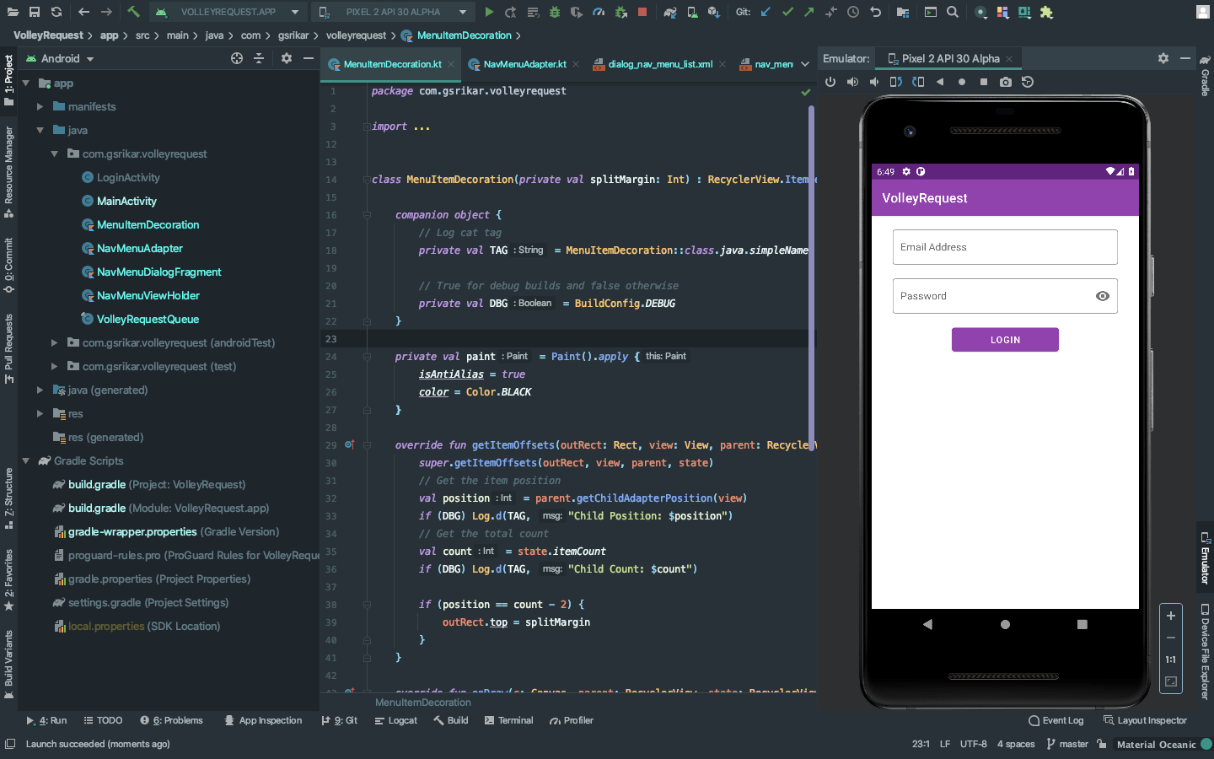


Figura . Interfaz de android studio. Recuperado de: https://1.bp.blogspot.com/O7QLJ\_FsBTg/X3CaoiMkshI/AAAAAAAAMOA/KMTZEfGZD2487FxPw5rfUT70YvCqe\_4PACLcBGAsYHQ/s1440/android-studio-emulator-tool-window.png

* IDE QT: Es un IDE multiplataforma para desarrollo móvil. Cuenta con un sofisticado editor de código para lenguajes como C, JavaScript, QML, entre otros. Contiene un apartado de diseño de interfaz de usuario, permitiendo crear interfaces con widgets predeterminados. Otro apartado útil es el emulador de dispositivos, que permite probar aplicaciones en entornos similares a dispositivos reales
* Eclipse: Entorno tradicional de programación para Java. Este IDE es compatible con Android mediante la extensión Google ADT. Compatible con varios lenguajes y con una amplia comunidad de usuarios, permiten diversas funcionalidades.
* Jetbrains Rider: IDE ideado para el desarrollo con framework Xamarin, que utiliza lenguajes como C# para crear aplicaciones Android. Permite control de versiones y pruebas unitarias.

*Referencias*

AWS. *¿Qué es XML?* Recuperado de: <https://aws.amazon.com/es/what-is/xml/#:~:text=El%20lenguaje%20de%20marcado%20extensible,datos%20y%20aplicaciones%20de%20terceros>.

IONOS. *¿Diseño web móvil, responsivo o app?* Recuperado de: <https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/pagina-web-movil-responsiva-o-app/>

Mdn web docs. *Introducción a XML.* Recuperado de: <https://developer.mozilla.org/es/docs/Web/XML/XML_introduction>

Mejor Programación. *Tipos de layout.* Recuperado de: <https://www.mejorprogramacion.com/curso_android_java/3-1-layout/>

Medewar, S. *7 mejores IDE para el desarrollo de aplicaciones móviles.* Recuperado de: <https://geekflare.com/es/best-ide-for-mobile-app-development/>