**Informe del Proyecto: APP CITAS MÉDICAS**

Este informe se basa en la presentación 'Avance del Proyecto', y detalla los puntos más importantes del desarrollo de la aplicación móvil para la gestión de citas médicas, así como conceptos clave relacionados con las tecnologías utilizadas.

**Proyecto**

**APP CITAS MÉDICAS**

Vista de ventana de ingreso y registro

**ANDROID**

Android es un sistema operativo móvil de código abierto basado en el kernel de Linux. Fue diseñado principalmente para dispositivos móviles como smartphones y tablets.   
Su arquitectura se compone de varias capas, que incluyen el núcleo Linux, las bibliotecas nativas (como SQLite, WebKit y OpenGL), el entorno de ejecución Android Runtime (ART), el framework de aplicaciones y la capa de aplicaciones del usuario.  
El desarrollo en Android se realiza mediante Android Studio, que permite crear interfaces con XML y programar la lógica en Java o Kotlin.

**ANDROID STUDIO**

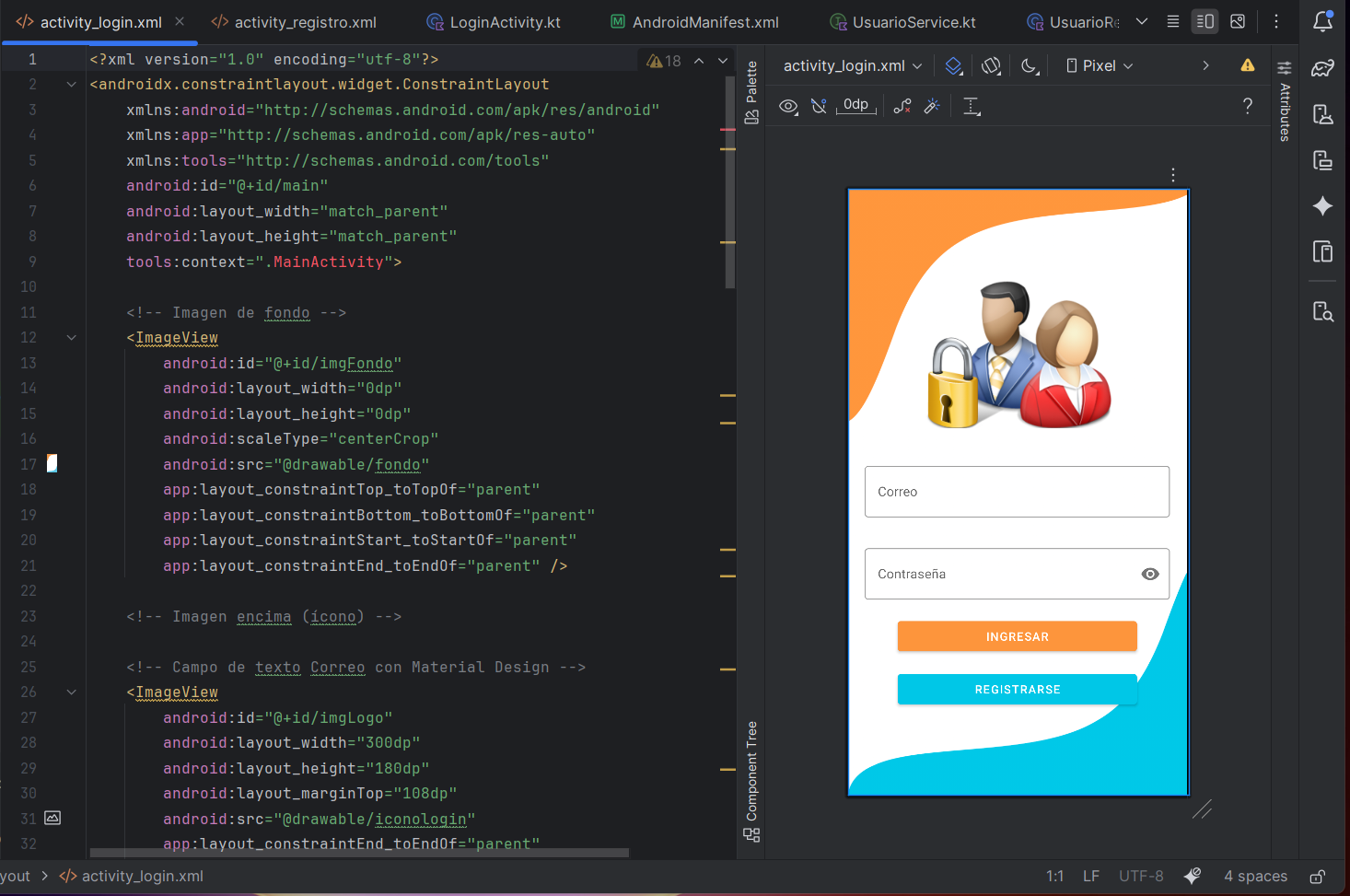
* Android Studio es el entorno de desarrollo integrado (IDE) oficial para el desarrollo de aplicaciones Android.
* Proporciona herramientas avanzadas para la codificación, depuración, pruebas y empaquetado de aplicaciones.
* Entre sus componentes principales están el SDK (Software Development Kit), el AVD (Android Virtual Device) para la emulación de dispositivos y el JDK (Java Development Kit), necesario para compilar el código fuente.

**KOTLIN**

* Kotlin es un lenguaje moderno, conciso y seguro desarrollado por JetBrains. Desde 2017, Google lo reconoce como lenguaje oficial para Android.
* Ofrece ventajas como la interoperabilidad total con Java, la reducción de código boilerplate, la seguridad frente a null pointer exceptions y la compatibilidad multiplataforma.
* Permite compartir código entre Android, iOS y aplicaciones de escritorio, lo que lo convierte en una excelente opción para proyectos escalables.

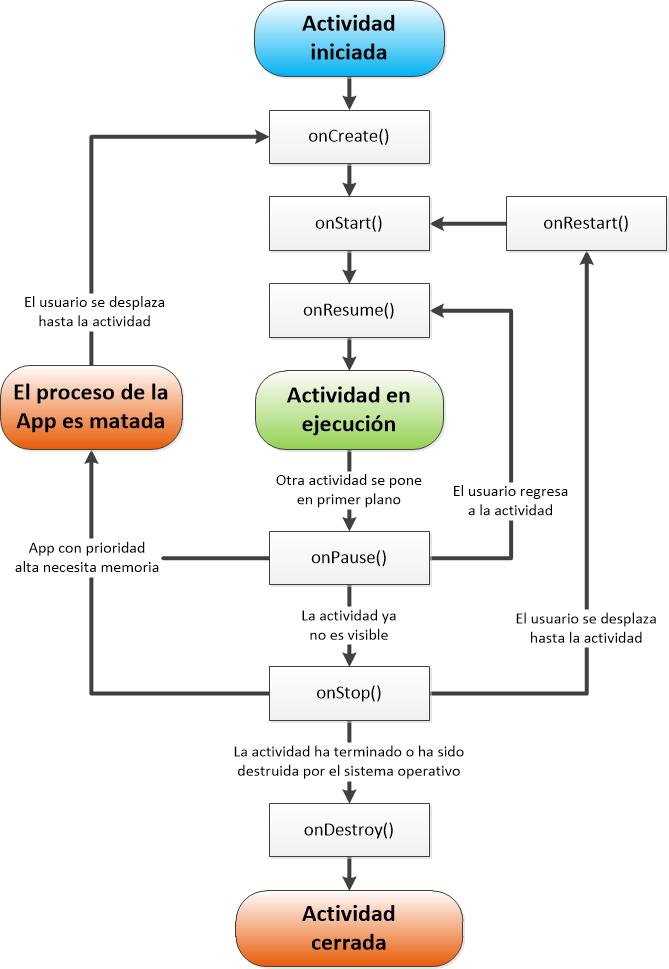
**Activity**

Es el componente principal que representa una sola pantalla o ventana donde el usuario interactúa con la aplicación. Cada pantalla que ves en tu dispositivo es una Activity, y esta gestiona la lógica y la interfaz de usuario de esa pantalla. Una Activity consta de dos partes principales: un archivo de layout (XML) que define la estructura visual y un archivo de código (Kotlin o Java) que controla su comportamiento y responde a las interacciones del usuario.

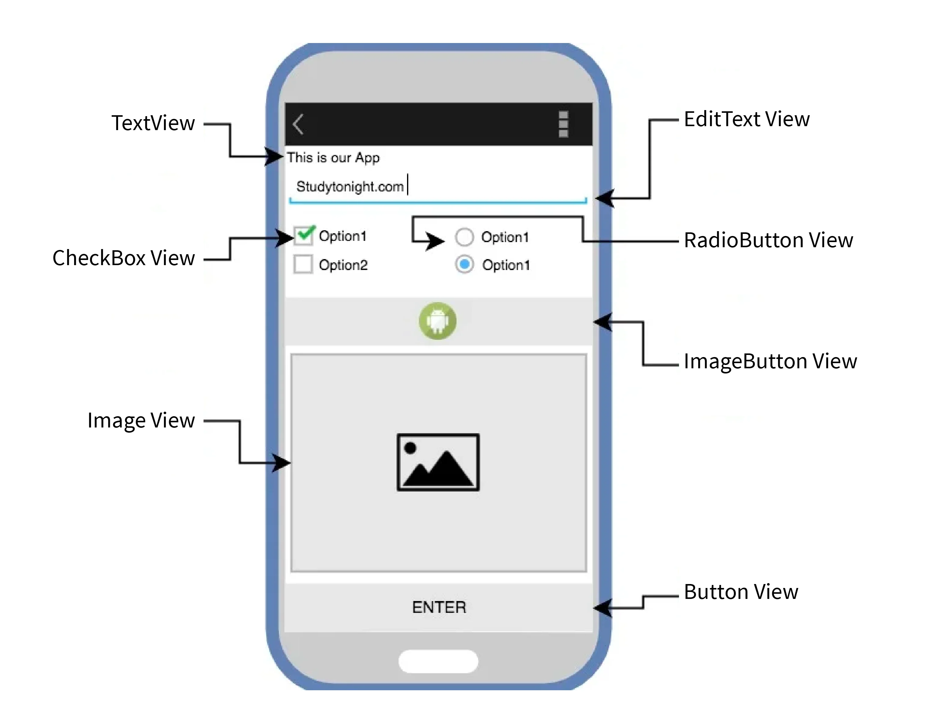


**Ciclo de vida de una Activity**

* Una Activity representa una única pantalla con una interfaz de usuario. Su ciclo de vida está controlado por el sistema Android, que llama a distintos métodos (onCreate, onStart, onResume, onPause, onStop y onDestroy) dependiendo del estado de la aplicación.
* Entender este ciclo es fundamental para gestionar recursos y mantener una experiencia fluida para el usuario.

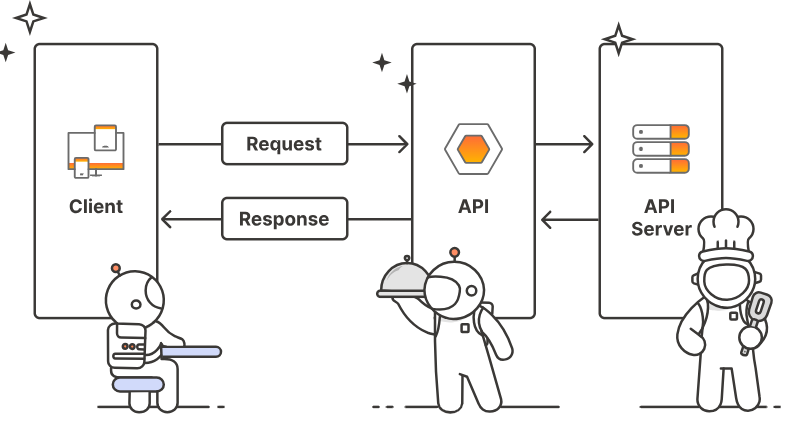
****

**View en Android**

* Las vistas o 'Views' son los componentes visuales de una interfaz en Android. Cada elemento que el usuario ve o con el que interactúa se deriva de la clase base View, de la cual se extienden componentes como TextView, Button, ImageView y otros.   
  Estas vistas se organizan en jerarquías dentro de contenedores llamados ViewGroups, como LinearLayout o ConstraintLayout.

**API: Interfaz de Programación de Aplicaciones.**

Una API es un conjunto de reglas y definiciones que permite que dos aplicaciones o sistemas se comuniquen entre sí.  
Una API puede estar basada en diferentes protocolos, como SOAP, gRPC, o HTTP.

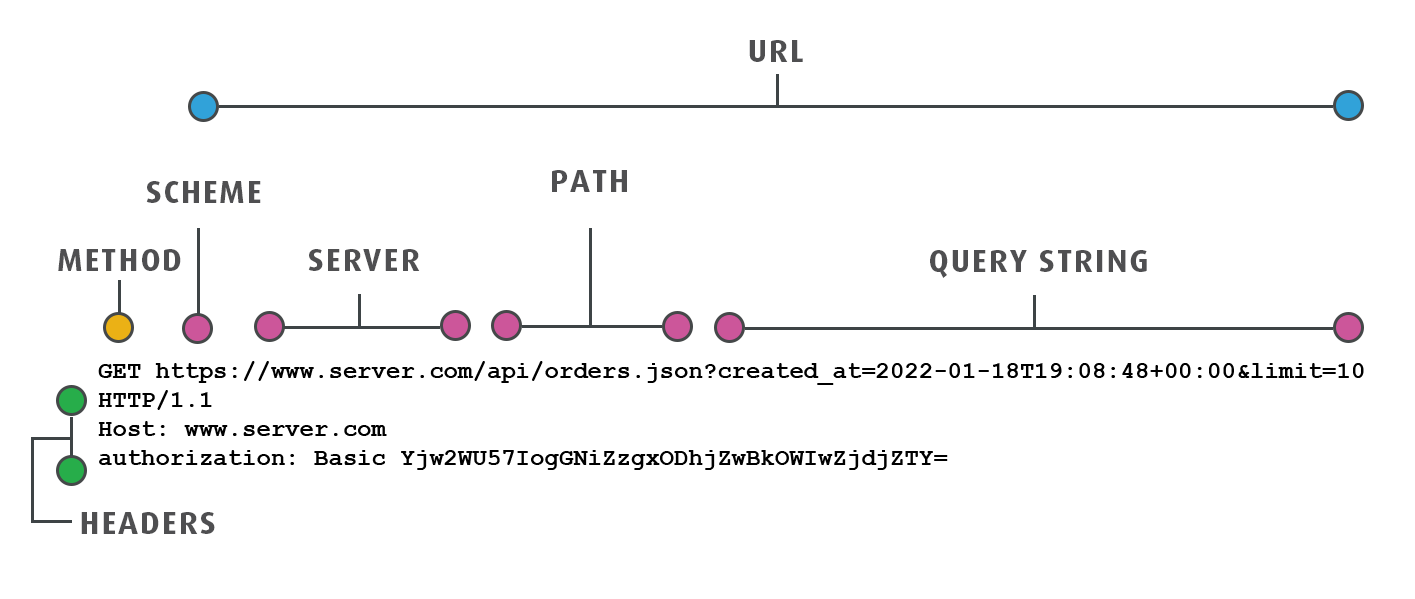


**HTTP**

* En los protocolos basados en el modelo cliente-servidor, como es el caso del HTTP, una sesión consta de tres fases:
  + El cliente establece una conexión TCP (o la conexión correspondiente si la capa de transporte corresponde a otro protocolo).
  + El cliente manda su petición, y espera por la respuesta.
  + El servidor procesa la petición, y responde con un código de estado y los datos correspondientes.
  + Protocolo de Transferencia de Hipertexto. Es un protocolo diseñado para transferir información entre computadoras.

  
Cada interacción se basa en un modelo de petición (Request) y respuesta (Response).   
Los códigos de estado HTTP indican el resultado de cada solicitud, siendo 200 para éxito, 404 para recurso no encontrado y 500 para error interno del servidor.

**Request/Petición**

****

Los códigos de estado indican el éxito o el fracaso de una solicitud. Son un número de 3 dígitos donde el primer dígito significa la clase o categoría de respuesta.



**Conclusiones**

* El proyecto 'APP CITAS MÉDICAS' representa un avance significativo en el desarrollo de soluciones móviles para la gestión de servicios de salud.
* Su implementación en Android, utilizando Kotlin y Android Studio, garantiza compatibilidad, rendimiento y facilidad de mantenimiento.
* Además, el uso de APIs y el modelo cliente-servidor permiten la comunicación eficiente entre la aplicación y el backend, asegurando una experiencia confiable para los usuarios.