1. **Completa la siguiente tabla (1,5 puntos):**

|  |  |
| --- | --- |
| **PLATAFORMA DE DESARROLLO** | **LENGUAJE DE PROGRAMACIÓN** |
| Android Studio | Java, Kotlin |
| Flutter | Dart |
| Xamarin | C# .NET |
| Xcode | Swift, Objective-C |
| IntelliJ | Java, kotlin, dart, JavaScript, TypeScript, HTTML, CSS … |

1. **Explica con tus palabras las seis devoluciones de llamada del ciclo de vida de la clase *activity* de una app. (2 puntos)**

**El ciclo de vida de la calse activity son las diferentes etapas por las que pasa la actividad.**

**Durante el ciclo de vida existen seis devoluciones de llamadas o callbacks principales,**

**esto permite a los desarrolladores realizar cambios en cada etapa. Las devoluciones de**

**llamada del ciclo de vida de la clase activity son → onCreate(), onStart(), onResume(),**

**onPause(), onStop(), onDestroy()**

**onCreate() → Este se invoca cuando la actividad esta siendo creada. En esta devolucion es donde se realiza la inicializacion basica, como por ejemplo la creacion de la interfaz de usuario.**

**OnStart() → Este sera invocado cuando la actividad se hace visible para el usuario, pero sin que este en primer plano, es decir antes de que la actividad se haga interactiva.**

**OnResume() → Este indica que la actividad esta a punto de interactuar con el usuario, es decir la actividad pasara a primer plano, estando lista para recibir información del usuario. En esta devolución se inician las animaciones y registra los receptores de eventos.**

**OnPause() → La actividad aun sera visible, pero no en primer plano. Esta devolución de llamadas para detener animaciones, liberar recursos. La actividad sigue existiendo pero no será visible.**

**OnStop() → Esta sera invocado cuando la actividad ya no sea visible para el usuario. En esta etapa, la actividad es ocultada y libera recursos innecesarios. Actividad existente pero no visible.**

**OnDestroy() → Esta sera invocada cuando la actividad esta siendo destruida. Este es el ultimo ciclo de vida y se utiliza para liberar cualquier recurso retenido. Despues de esta invocacion la actividad deja de existir**

1. **Explica con tus propias palabras qué es una API.(1,5 puntos)**

**Una API o Interfaz de Programacion de Aplicaciones, La API es como un camarero que facilita la comunicación entre diferentes partes de una cocina , permitiendo que solicites y recibas informacion sin tener que conocer los detalles internos**

1. **Busca ejemplos de APIs para aplicaciones móviles. Selecciona tres y explícalos. (5 puntos)**

**NASA API → La NASA proporciona a los desarrolladores acceder a los datos y recursos, la API que explicare sera NASA Open API.**

**NASA Open API ofrece información / datos que nos proporciona la propia nasa, permitiendo a los desarrolladores obtener la información recopilada por la nasa para sus proyectos, aplicaciones o sitios web.**

**Los recursos principales que nos ofrecen son APOD, Mars Rover Photos, EPIC, DONKI.**

**APOD → Proporciona la imagen astronomica destacada del dia. https://apod.nasa.gov/apod/astropix.html**

**Mars Rover Photos → Permite el acceso de las imagenes tomadas por el robot rovers en marte.**

**https://data.nasa.gov/Space-Science/Mars-Rover-Photos-API/929k-jizu/about\_data**

**EPIC → Imagenes de la tierra tomadas desde el espacio.**

**https://epic.gsfc.nasa.gov**

**DOKI → Proporciona datos de eventos relacionados con el clima espacial.https://kauai.ccmc.gsfc.nasa.gov/DONKI/**

**Para acceder a toda esta información tendremos que registrarnos en el portal de desarrolladores de la NASA, para que así nos den una clave API permitiendonos acceder a la NASA OPEN API**

**En conclusion con esta herramienta los desarrolladores, lo podran utilizar para aplicaciones educativas, noticias espaciales, exploracion espacial … , tendran datos para enriquecer su proyecto.**

**YouTube API → YouTube al pertenecer a Google, esta API ES PARTE DE A PLATAFORMA DE Google API, permitiendo a los desarrolladores interactuar con diferentes servicios de YouTube, como por ejemplo la reproducción de videos, listas de reproducción …**

**Existen diferentes APIs, tenemos los principales ;**

**YouTube Data API → nos proporciona acceso a recursos como videos, lista de reproducción y canales.**

**YouTube Analytics API → Este nos permite acceder a los datos sobre el rendimiento de los videos y canales.**

**YouTube Player API → Este nos facilita el reproductor de video de YouTube en aplicaciones o sitios web**

**Estas herramientas son utilizados para muchas funciones, yo comentare unos pocos. La reproduccion de videos ya comentada anteriormente, gestion de listas de reproduccion, acceso a metadatos , comentarios o suscripciones...**

**Para utilizar la API de YouTube el desarrollador deberá de crear un proyecto en la consola de desarrolladores de google y habilitar la API de YouTube, a continuación solicitaremos la clave para autenticar y autorizar el acceso a los recursos de YouTube**

**Spotify API → Esta API nos permite recopilar los datos disponibles de Spotify, o hacer funciones como crear listas…**

**Existen varios recursos, aquí veras los principales;**

**Spotyfi Web API → Ofrece el acceso a casi todas las funciones de la aplicación de Spotify, como busqueda de canciones o artistas, detalles de artistas, lista de reproduccion…**

**Spotify Player SDK → Este permite la reproducción de música de la aplicación, es decir, permite que en tu aplicación un reproductor.**

**Spotify for Developers → Incluye herramientas y recursos adicionales para los desarrolladores, documentacion, SDKs …**

**Existe una aplicación externa de spotify, llamada stats.fm que recopila toda la información de tu cuenta y te muestra las estadísticas, como una recopilación de tus datos, por ejemplo, un ranking de tus artistas o canciones mas escuchados y mucho mas.**

**Como accedemos a la API, nos tendremos que registrar en el portal de desarrolladores de Spotify para obtener credenciales de API, incluyendo un ID de cliente y un secreto de cliente.**

**Para la autentificación se utiliza Oauth y las solicitudes para la API deben de incluir el token de acceso correspondiente.**