



# Tecnologías para Front-end

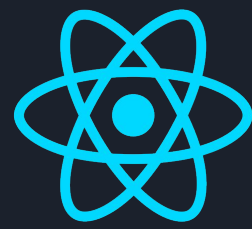
González, Cristian  
Olivera, Lucas  
Pardal, Brenda  
Pardal, Juliana



Flutter



NativeScript



React Native



PhoneGap



Ionic



Xamarin



# Ionic



Ionic Framework es un toolkit open source para crear aplicaciones de escritorio y móviles usando tecnologías web (HTML, CSS, y JavaScript). Está centrado en frontend y en las interacciones del usuario con las aplicaciones. (controls, interactions, gestures, animations)

## VENTAJAS:

- + Usa HTML y CSS como tal.
- + Modifica aspectos propios de Android o iOS de forma transparente para el programador.
- + Tiene documentación extensiva.
- + Permite aprender el desarrollo móvil sin dejar de aprender web.



# Ionic



## DESVENTAJAS:

- Al ser basado en componentes Web requiere el uso de plug-ins para acceder a funciones nativas.
- Los plug-ins hacen que el debugging se vuelva más difícil.
- Solo tiene soporte para Angular (aunque se pretende incorporar React y Vue).
- Testing basado en Angular.



# Xamarin



# Microsoft

Xamarin es un framework desarrollado por Microsoft como una extensión de su plataforma .NET para el desarrollo de apps móviles.

## VENTAJAS:

- + C# como lenguaje predeterminado.
- + Acceso a plataforma .NET (Azure y todos sus servicios para incorporar fácilmente).
- + Permite desarrollo para Android, iOS, tvOS, watchOS, macOS y Windows.
- + Código compilado a nativo.
- + Velocidad próxima a la nativa.
- + Desarrollo de UI común ágilmente por medio de Xamarin.Forms.
- + Permite acceder a funcionalidad como ARKit, CoreML, Fingerprint, Bluetooth, etc.
- + Fácil binding de librerías nativas.



# Xamarin



# Microsoft

## DESVENTAJAS:

- Overhead significativo para las apps.
- Para una GUI nativa, requiere código diferente para cada plataforma.
- Delay entre actualización de la plataforma destino a Xamarin.
- No posee Hot Reload.
- Dificultad al exportar código producido para trabajar con otra tecnología.
- Es pago para uso en medianas a grandes empresas.



# FLUTTER



Flutter es el nuevo framework mobile de Google para crear interfaces nativas de alta calidad en iOS y Android en un tiempo récord.

## VENTAJAS:

- + Funciona con código existente.
- + Utilizado por desarrolladores y organizaciones de todo el mundo.
- + Es gratuito y Open Source.
- + Utiliza Hot Reload.
- + Rendimiento e integración de funciones nativas.
- + Su lenguaje Dart es fácil de aprender.
- + Multiplataforma.
- + Desarrollo rápido.





# FLUTTER



Flutter no es una buena elección cuando:

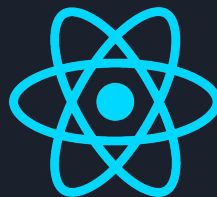
- La aplicación a desarrollar necesita soportar 3D Touch.
- La aplicación a desarrollar requiere múltiples interacciones con un sistema operativo, o requiere bibliotecas nativas raras y poco conocidas.
- Se necesita una UI minimalista, pero depende de un uso significativo del hardware del teléfono.
- Se quiere crear una Instant App.

Además, hay que tener en cuenta que Flutter es un framework reciente y, por lo tanto, menos estable que su competidor React Native.





# React Native



React Native es un framework mobile de código abierto que utiliza JavaScript y fue desarrollado por Facebook en 2015.

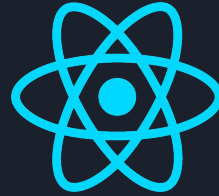
## VENTAJAS:

- + Desarrollo multiplataforma, cubre tanto iOS como Android utilizando una única base de código.
- + La aplicación híbrida se procesa de forma nativa debido a los componentes reutilizables.
- + Menor uso de memoria.
- + Excelente tiempo de respuesta
- + Ahorro de tiempo para desarrolladores.
- + Código abierto y rendimiento potente.
- + Hot Reload = fast coding.
- + Estabilidad en el mercado.
- + Comunidad activa





# React Native



## DESVENTAJAS:

- Los componentes en React Native son bastante básicos, por lo que si necesita algo adicional, se requiere un esfuerzo considerable al diseñar.
- React Native es sólo una envoltura sobre los métodos nativos, por lo que requiere un puente para traducir esas llamadas a una API nativa.
- UI frágil debido a que las actualizaciones pueden romper la interfaz de usuario.
- Aplicaciones más grandes que las nativas.