

Cross-Platform

Ionic + Cordova/Capacitor

Elaborado por:

- Duarte Serrão (A83630)
- Joana Alves (PG50457)

Temas a apresentar

1. Visão Geral

- a. Diferenças entre Desenvolvimento Nativo e *Cross-platform*
- b. Vantagens e desvantagens de desenvolvimento *cross-platform*

2. Ionic + Cordova/Capacitor

- a. Contextualização
- b. Diferença entre Ionic e Cordova/Capacitor (use cases)
- c. Conhecimento Prévio
- d. Arquitetura
- e. Vantagens/Desvantagens
- f. Componentes básicos
- g. Aplicações que usam Ionic

3. Considerações Finais



01

Visão Geral

Desenvolvimento Nativo

- Escrita de código separado por plataforma;
- Usa ferramentas e tecnologias específicas;
- Oferece uma experiência mais personalizada;
- Oferece alto desempenho;
- Desenvolvimento mais caro e demorado.



Desenvolvimento Cross-Platform

- Única base de código para duas ou mais plataformas:
 - Diferentes dispositivos;
 - Diferentes sistemas operativos;
 - Diferentes *browsers*.
- Só é preciso de uma *framework* cross-platform para desenvolver a aplicação;



Desenvolvimento Cross-Platform

Vantagens

- Redução do Tempo e Custo
- Alcance
- Facilidade de Manutenção
- Consistência de Design
- Acesso a Recursos Nativos

Desvantagens

- Desempenho
- Limitações de Recursos
- Dependências de Terceiros
- Limitações de Personalização
- Adaptação



02

Ionic +
Cordova/Capacitor

Contextualização



ionic



Use Cases

IONIC

- Navegação
- Formulários
- Listas
- Animações
- Botões

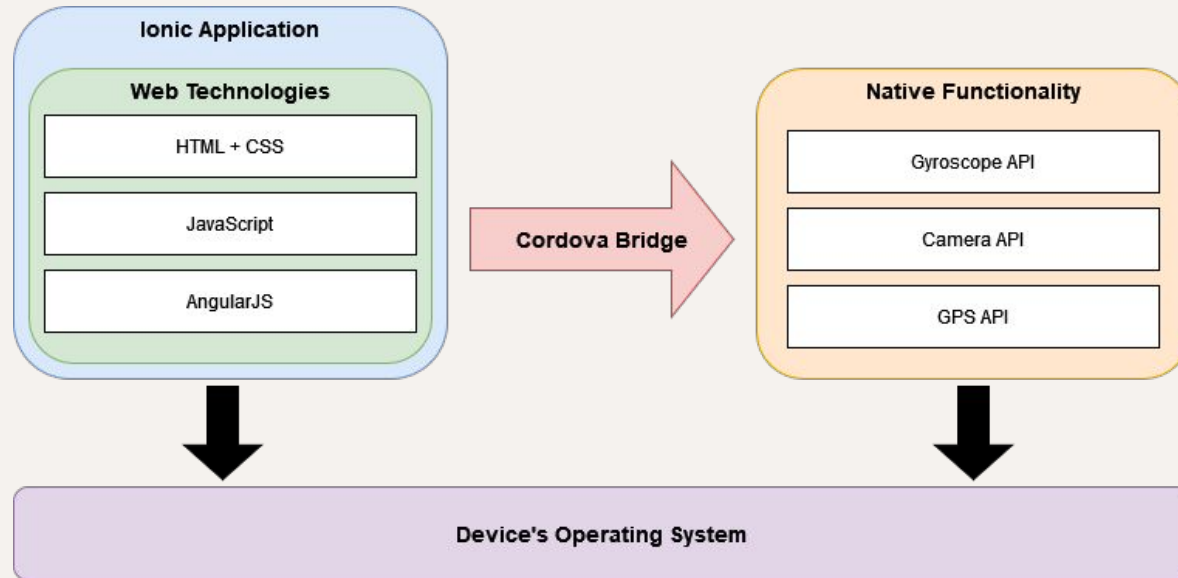
CORDOVA/CAPACITOR

- Câmera
- Geolocalização
- Giroscópio
- Sistema de Ficheiros
- Notificações *push*

Conhecimento Prévio

- Tecnologias **Web**:
 - HTML
 - CSS
 - JavaScript
- Framework **Angular**
- **Comunidade** Ativa
- **Documentação** Abrangente

Arquitetura



Ionic + Cordova/Capacitor

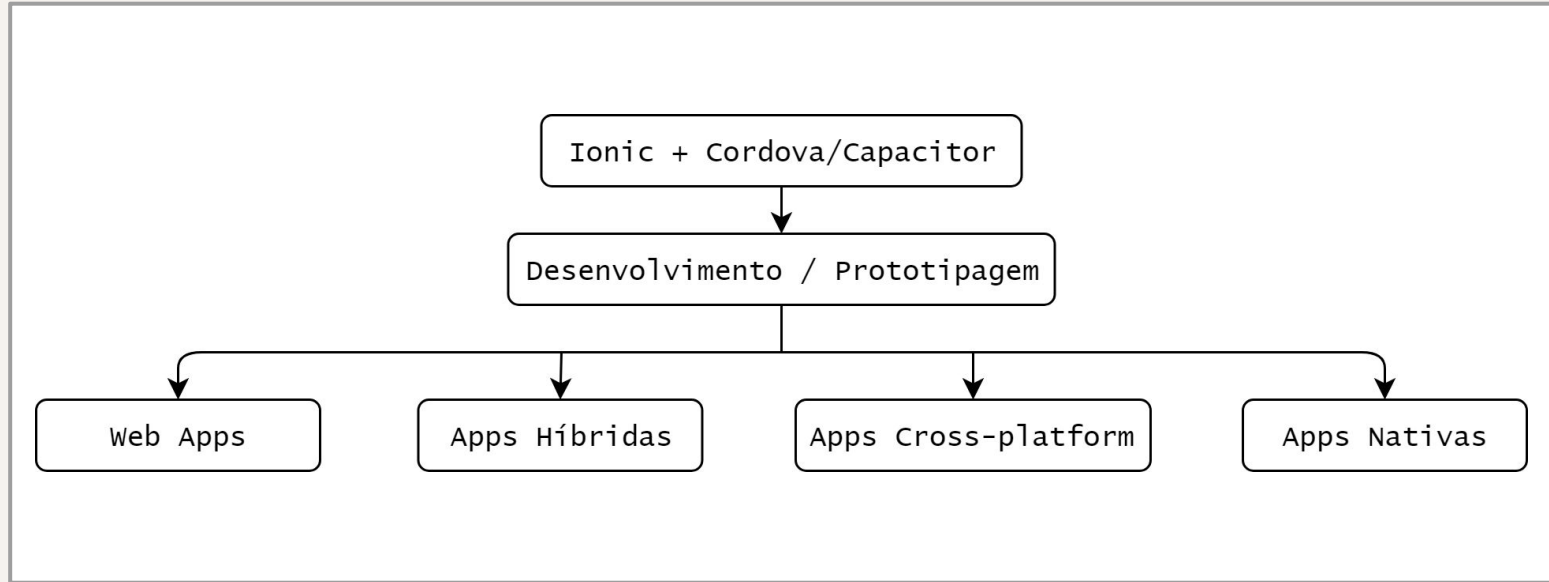
Vantagens

- Curva de Aprendizagem
- Biblioteca de Componentes
- Integração
- *Plugins*

Limitações

- Desempenho
- Personalização
- Dependência de *Plugins*

Aplicabilidade



Componentes Básicos

Componente	Descrição
Content	Contém outros componentes para interação e navegação.
Navigation	Para navegação na aplicação
Item	Container de UI genérico que pode ser usado em listas
Input	Componente para inserção de dados
Button	Botão básico que manipula eventos ao pressionar nele

Aplicações que usam Ionic

The logo for MarketWatch, featuring the word "Market" in black and "Watch" in green, with a green upward-pointing arrow integrated into the letter 't' of "Watch".

MarketWatch

Ajuda utilizadores a descobrir notícias, análises e dados do mercado de ações mais recentes.

The logo for Sanvello, consisting of a green rectangle with a white stylized 'S' shape cutout on the right side, and the word "SANVELLO" in black capital letters centered within the green area.

SANVELLO

Ajuda utilizadores a gerir o stress, a ansiedade e a depressão.

The logo for SworKit, featuring the word "SWORKIT" in orange capital letters, with a small orange circle above the 'O' and a small orange dot above the 'R'.

App de fitness que oferece um plano personalizado para cada utilizador.



03

Considerações Finais

- Deve-se escolher o melhor método de desenvolvimento, dependendo dos requisitos;
- Ionic serve para Interface de Utilizador;
- Cordova ou Capacitor servem para aceder a funcionalidades nativas;
- É possível fazer uma aplicação completa só com Ionic, mas não é possível só com Cordova/Capacitor;