






# SISB020 - Desenvolvimento de Software para Dispositivos Móveis

## Aula 4

Prof. Thiago Cavalcante

# Desenvolvimento do App

-  **Onde** o app vai ser publicado
-  **Quem** vai fazer o app
-  Quais são as **partes** de um app
-  Quanto **tempo** demora para criar um app
-  Qual **tecnologia** será usada para a criação do app

# Quanto tempo demora para criar um app

- Baixa complexidade
  - Poucas telas e UI simples
  - Não requer back-end nem integração com APIs
  - Ex.: calculadoras, relógios, listas de tarefas
- Média complexidade
  - Requer back-end, banco de dados, APIs, autenticação de usuário (login)
  - Design de UI customizado
  - Ex.: bancos, catálogos de produtos, gerenciamento de projetos
- Alta complexidade
  - Tendem a ser os mais bem sucedidos
  - Além do back-end, tem processamento de pagamentos, vários idiomas, integração com terceiros, animações, processamento em tempo real
  - Ex.: redes sociais, jogos, e-commerce

# O que pode atrasar o desenvolvimento

- Mudanças de escopo
- Programadores inexperientes
- Inovação com tecnologias emergentes
  - Inteligência artificial
  - Aprendizagem de máquina
  - Realidade virtual
  - Realidade aumentada

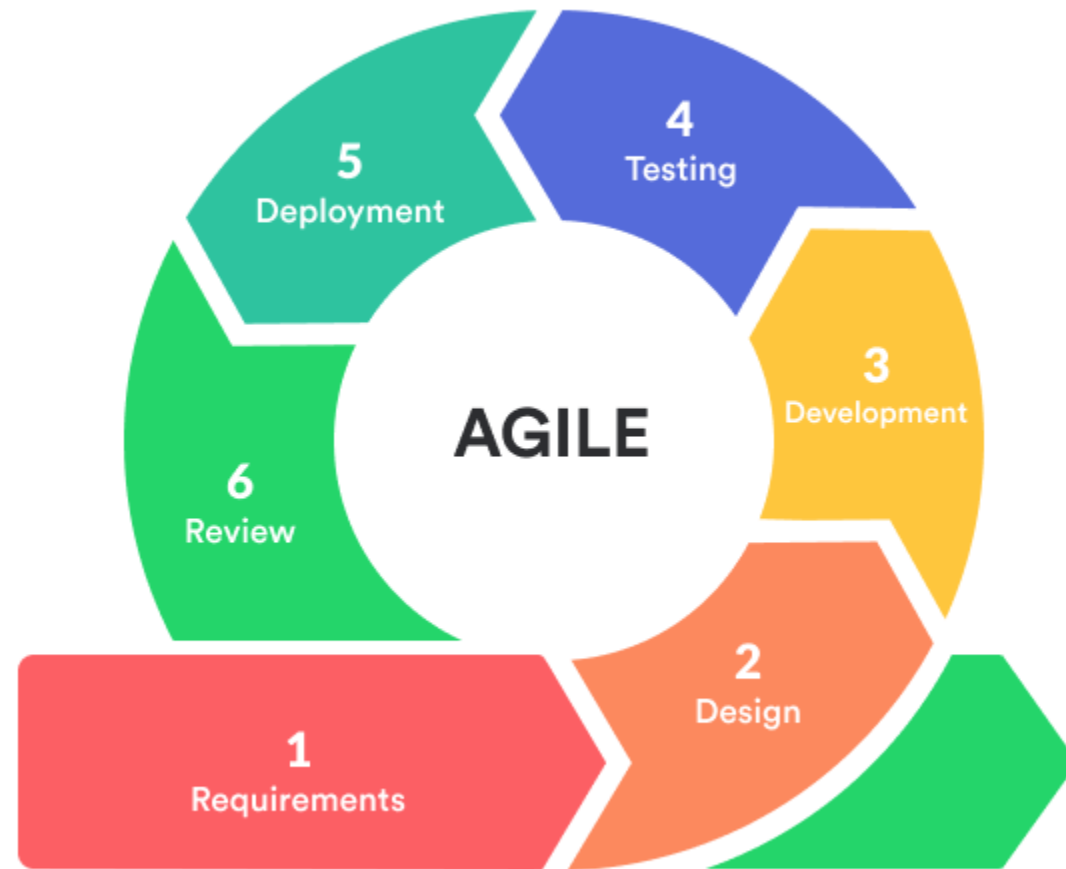
# O que pode acelerar o desenvolvimento

- Criar um *roadmap*
- Programadores experientes
- Criar um MVP — *metodologias ágeis*

# Vantagens das metodologias ágeis

- Reações imediatas às mudanças e desenvolvimentos do mercado
- Entrega rápida
- Visão geral clara do progresso do projeto
- Termos comuns: MVP, *release*, *roadmap*, *backlog*
- Exemplos: Scrum, Kanban, Lean

“ Fazer o mínimo possível, testar o produto, coletar feedback, analisar os dados e voltar ao trabalho. ”

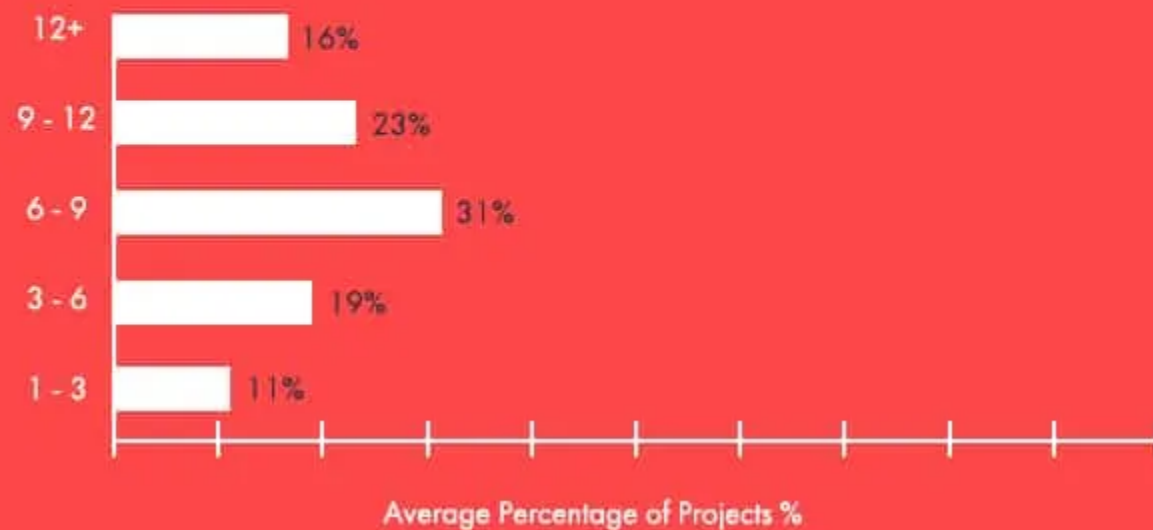


## HOW LONG DOES IT TAKE TO MAKE AN APP?

INVONTO  
Connecting People Through Digital

### Average Time Estimate Of App Development Projects

Months






## HOW LONG DOES IT TAKE TO MAKE AN APP?

INVONTO  
Connecting People Through Digital

### Average App Development Timeline



# Qual tecnologia será usada

- App web
- App web progressivo (PWA — *Progressive Web App*)
- App híbrido
- App nativo multiplataforma 
- App nativo

# App Development Approaches

Mobile Web App



Hybrid App

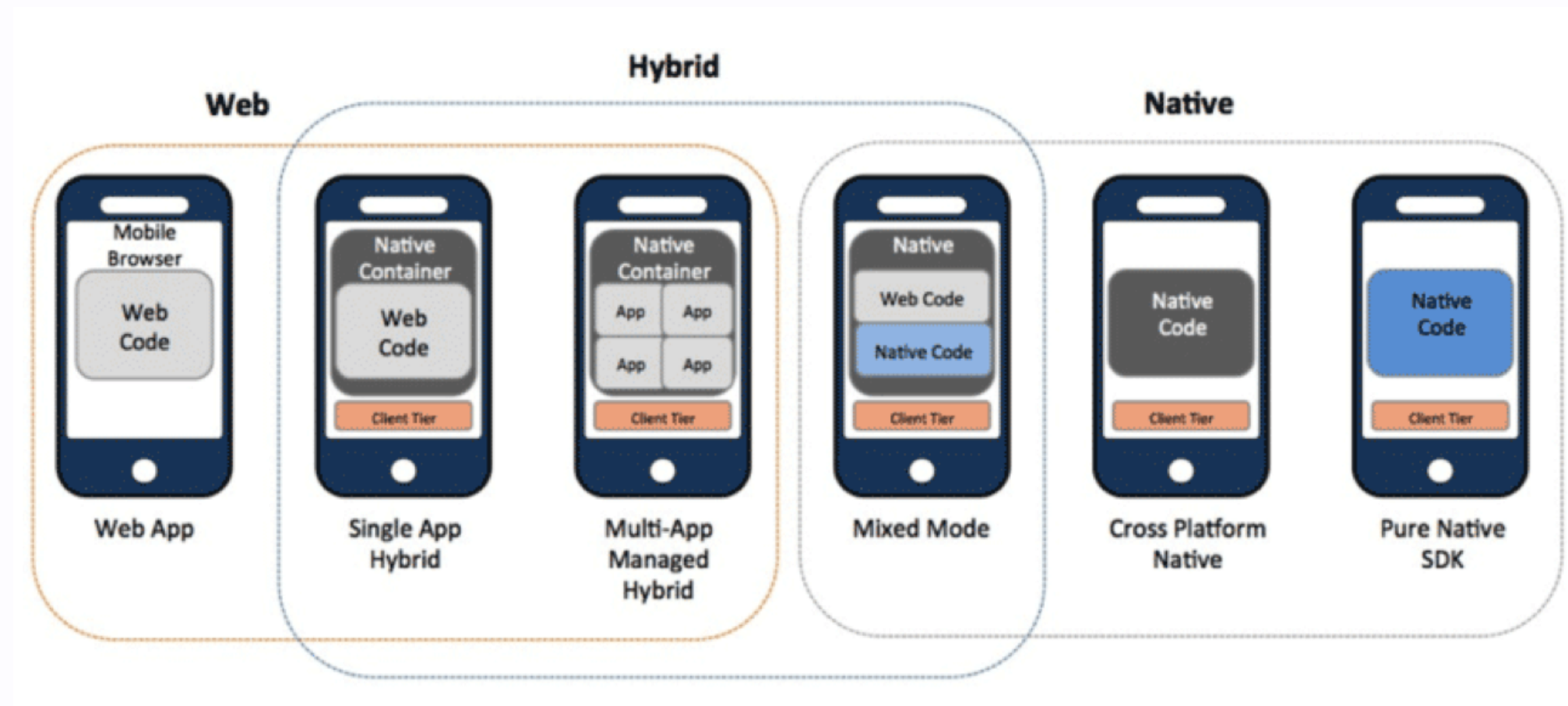


Mixed Hybrid App



Native App





# App Web

## Vantagens

- Única base de código
- Independente da loja
- Alta flexibilidade para desenvolvimento
- Custo baixo, desenvolvimento rápido
- Funciona em qualquer tela

## Desvantagens

- Sem notificações
- Não funcionam offline
- Interface não segue regras de design padrão
- Não interagem com 100% do hardware
- Performance baixa
- Não são encontrados na loja

# Exemplos



# App web progressivo

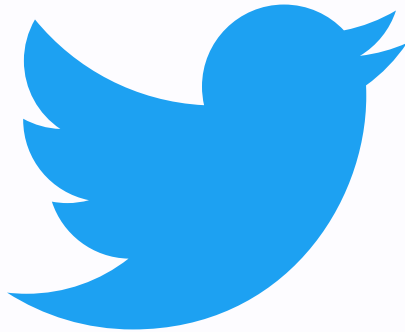
## Vantagens

- Única base de código
- Usuário pode criar um atalho na tela do celular
- Uso de APIs modernas e plugins para fornecer mais funcionalidades

## Desvantagens

- Não é compatível com todos os browsers
- Interface não segue regras de design padrão
- Não interagem com 100% do hardware
- Não são encontrados na loja

# Exemplos





# App híbrido

## Vantagens

- Única base de código
- Custo baixo, desenvolvimento rápido
- Não precisam de um browser
- Podem acessar funcionalidades do aparelho

## Desvantagens

- Performance baixa (animações, monitoramento)
- UI/UX de qualidade inferior
- Interface não segue regras de design padrão

# Exemplos



# App nativo multiplataforma

## Vantagens

- Única base de código
- O código é compilado para cada plataforma
- Performance média/alta
- Custo mais baixo, escalável
- UI/UX impecáveis e consistentes

## Desvantagens

- Requer experiência
- *Deployment* mais lento
- Novas funcionalidades para Android/iOS não podem ser usadas imediatamente

# Exemplos



# App nativo

## Vantagens

- Performance alta
- Total acesso às funcionalidades do aparelho
- Notificações
- Experiência do usuário de acordo com o padrão

## Desvantagens

- Tudo em dobro (app, código, equipe, bugs)
- Alto custo de desenvolvimento e manutenção
- Requer experiência
- *Deployment* mais lento

# Exemplos



# Testes

- Experiência do usuário
- Funcionais
- Performance
- Segurança
- Dispositivo e Plataforma

# Testes de Experiência do Usuário

- Implementação bate com o que foi feito na fase de design
- Visual consistente
  - Fontes
  - Estilos
  - Cores
  - Espaçamentos



# Testes Funcionais

- É impossível prever o comportamento do usuário
- Quanto mais usuários testarem o app, melhor (diferentes resultados)
- Testes de sistema (geral) e testes unitários (específicos)

“ Volte aos documentos de planejamento e design e verifique cada funcionalidade do app; não assuma que algo funciona na versão final apenas porque funcionou durante a fase de desenvolvimento. ”

# Testes de Performance

- Velocidade de resposta às requisições dos usuários
- Velocidade de carregamento das telas
- Drenagem de bateria
- "Vazamento" de memória
- Eficiência de uso da internet
- Tamanho do app
- Funcionamento do app caso haja um pico de usuários

# Testes de Segurança

- Qualquer vulnerabilidade pode levar a um hackeamento
- Rastreamento das sessões de login tanto no app quanto no backend
- Término automático da sessão após algum tempo
- Armazenamento de credenciais em serviços confiáveis
- Verificação de formulários para entrada de dados

# Testes de Dispositivo e Plataforma

- Lançamentos de novos dispositivos e novas atualizações de sistema ocorrem periodicamente
- Várias empresas customizam o Android
- Aplicativos móveis precisam ser testados em vários dispositivos (ou simuladores)