

3. Ambiente de Desenvolvimento (6h)

Objetivos da Aula:

- Configurar o ambiente de desenvolvimento para Flutter e Dart.
 - Configurar o Emulador
 - Criar uma aplicação Flutter básica interativa.
-

Instalação do Flutter e Dart

O que é Flutter?

O **Flutter** é um **framework de desenvolvimento de interfaces** criado pelo Google, utilizado para construir **aplicações nativas** para **Android, iOS, Web, Desktop e Embedded** a partir de um único código-fonte. Ele se destaca por sua alta performance, facilidade de desenvolvimento e rica personalização de interfaces.

Principais características do Flutter

- ✓ **Cross-platform** – Permite criar aplicativos para múltiplas plataformas com um único código-fonte.
 - ✓ **Hot Reload** – Atualiza as alterações no código **instantaneamente**, sem precisar recompilar toda a aplicação.
 - ✓ **Alto desempenho** – Utiliza o próprio **motor gráfico Skia**, garantindo animações suaves e renderização eficiente.
 - ✓ **UI rica e personalizável** – Oferece widgets modernos para criar interfaces elegantes e interativas.
 - ✓ **Código nativo** – Compila diretamente para código nativo, sem a necessidade de JavaScript bridges.
-

O que é Dart?

O **Dart** é a **linguagem de programação** utilizada no Flutter. Criado pelo Google, é uma linguagem moderna, eficiente e otimizada para construção de interfaces gráficas rápidas e fluidas.

Principais características do Dart

- ✓ **Sintaxe limpa e fácil de aprender** – Similar ao Java, JavaScript e C#.
 - ✓ **Compilação Just-in-Time (JIT) e Ahead-of-Time (AOT)** – Permite desenvolvimento rápido com Hot Reload e alta performance na versão final.
 - ✓ **Orientado a objetos** – Baseado em classes e herança, facilitando a organização do código.
 - ✓ **Garbage Collector eficiente** – Gerencia memória automaticamente para evitar vazamentos.
-

Importância do Flutter e Dart

4.1 Para Empresas e Desenvolvedores

- **Menos tempo e custo** – Desenvolva para múltiplas plataformas sem precisar duplicar código.
- **Facilidade de manutenção** – Código mais limpo e organizado.
- **Comunidade ativa** – Grande suporte da Google e desenvolvedores ao redor do mundo.

4.2 Para o Mercado de Tecnologia

- **Adoção crescente** – Empresas como Google, Alibaba e Nubank utilizam Flutter.
 - **Expansão para Web e Desktop** – Além do mobile, o Flutter também é usado para aplicações Web e Windows/Mac/Linux.
-

Instalação do Flutter no Windows via CLI

1.1 Criar estrutura de diretórios

```
mkdir C:\src
```

1.2 Baixar e Instalar o Flutter SDK

```
# Baixar o Flutter SDK
```

```
Invoke-WebRequest -Uri  
"https://storage.googleapis.com/flutter_infra_release/releases/sta  
ble/windows/flutter_windows_3.29.0-stable.zip" -OutFile  
"C:\src\flutter\flutter.zip"
```

Extrair os arquivos

```
Expand-Archive -Path "C:\src\flutter\flutter.zip" -DestinationPath  
"C:\src\flutter"
```

Extrair Arquivos Via 7zip(Mais Rápido)

usar o aplicativo 7zip e extrair arquivos

Adicionar Flutter ao PATH pelo CMD ou Manualmente

```
setx /M PATH "%PATH%;C:\src\flutter\bin"
```

Verificar instalação do Flutter

```
flutter --version
```

Instalar o Android SDK via CLI

2.1 Baixar o Android SDK (Command Line Tools)

Baixar o Android Command Line Tools

```
Invoke-WebRequest -Uri  
"https://dl.google.com/android/repository/commandlinetools-win-110  
76708_latest.zip" -OutFile "C:\src\android\cmdline-tools.zip"
```

Extrair arquivos

```
Expand-Archive -Path "C:\src\android\cmdline-tools.zip"  
-DestinationPath "C:\src\android"
```

Renomear pasta necessária

```
Rename-Item -Path "C:\src\android\cmdline-tools" -NewName "latest"
```

2.2 Configurar Variáveis de Ambiente

```
[System.Environment]::SetEnvironmentVariable("PATH",  
"$env:PATH;C:\src\android\comand-line\latest\bin;C:\src\android\platform-tools", [System.EnvironmentVariableTarget]::Machine)
```

```
# Atualizar a sessão do terminal  
$env:PATH=[System.Environment]::GetEnvironmentVariable("PATH",  
[System.EnvironmentVariableTarget]::Machine)
```

Testar SDK

```
sdkmanager --list
```

3. Instalar Componentes do Android SDK

```
sdkmanager --install "platform-tools" "platforms;android-34"  
"build-tools;34.0.0" "emulator"
```

Verificar instalação

```
adb --version  
emulator --version
```

4. Criar um Emulador Android - A

4.1 Baixar a Imagem do Sistema

```
sdkmanager --install "system-images;android-34;google_apis;x86_64"
```

4.2 Criar um Emulador

```
avdmanager create avd -n MeuEmulador -k  
"system-images;android-34;google_apis;x86_64" --device "pixel_4"
```

4.3 Iniciar o Emulador

```
emulator -avd MeuEmulador -no-snapshot
```

O **Android Emulator** será iniciado.

4. Criar um Emulador Android - B

4.1 Instalar o BlueStack

Baixe a Versão 5.0 do BlueStack

4.2 Configuração do BlueStack

Realizar as Configurações do BlueStack

5. Testar o Flutter com o Emulador

```
flutter doctor
```

Se tudo estiver correto, o Flutter mostrará que o ambiente está pronto.

Criar e rodar um projeto Flutter

```
flutter create meu_app  
cd meu_app
```

```
flutter run
```

Isso iniciará o aplicativo no emulador.

Prática Guiada - Olá, Mundo

```
// Importa o pacote Flutter, que contém os widgets e ferramentas
necessárias para criar o aplicativo.
import 'package:flutter/material.dart';

// A função principal do programa, que é o ponto de entrada da
aplicação.
void main() {
  runApp(MyApp()); // Executa o aplicativo, iniciando o widget raiz
  "MyApp".
}

// Define o widget principal da aplicação.
class MyApp extends StatelessWidget {
  // Construtor do widget MyApp.
  @override
  Widget build(BuildContext context) {
    // Retorna o MaterialApp, que é a base para aplicativos com estilo
    Material Design.
    return MaterialApp(
      home: Scaffold( // Scaffold fornece uma estrutura básica para
uma tela, com barra de app, corpo e outros elementos.
        appBar: AppBar( // Define a barra superior (AppBar) com o
título do aplicativo.
          title: Text('Hello World App'), // Título exibido na AppBar.
        ),
        body: Center( // Centraliza os widgets filhos na tela.
          child: Text(
            'Olá, Mundo!', // Exibe o texto "Olá, Mundo!" no centro da
tela.
          ),
        ),
      ),
    );
  }
}
```

```
    ),
    ),
    );
}
```