



# **CÓDIGOS DE ALTA PERFORMANCE - WEB**

**Aula 03 – FRAMEWORKS**

**Prof. Dr. Diogo Rodrigues**

# FRAMEWORKS

Conjunto de ferramentas, bibliotecas e padrões que fornecem uma estrutura para o desenvolvimento do software.

Objetivo é simplificar o processo de desenvolvimento

# EXEMPLOS DE FRAMEWORKS



**Desenvolvimento WEB:**

Angular, React e Vue.JS



**Desenvolvimento WEB com  
Pytho:**

DJANGO e FLASK



**Desenvolvimento JAVA**

SPRING

# ANGULAR



Desenvolvido e mantido  
pela google



Usa TypeScript como  
linguagem principal

# REACT



Desenvolvido pelo  
facebook



Utiliza biblioteca



Baseado em componentes  
reutilizáveis



Usa JSX, uma extensão do  
JavaScript que permite  
escrever HTML no código

# VUE.JS



**FRAMEWORK PROGRESSIVO  
PARA CONSTRUÇÃO DE  
INTERFACES DE USUÁRIO**



**FACIL CURVA DE  
APRENDIZADO**

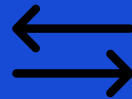
# DJANGO



Desenvolvido em Python



FrameWork de alto nível para desenvolvimento WEB



Segue padrão Model-View-Templeate (MVT)



Fornecer um ORM ( Object-relational Mapping) integrado para interagir com banco de dados

# FLASK



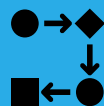
FrameWork de desenvolvimento  
WEB com Python



Mais minimalista do que o DJANGO  
oferecendo mais liberdade ao  
desenvolvedor WEB



Considerado um microframework



Otimo para pequenos projetos.





# NODE.JS

**Ambiente de execução  
JavaScript**

**Permite realização de  
operações locais sem a  
necessidade dos navegadores**

# NPM (NPX)

O **NPM Create** é um dos comandos disponibilizados pelo NPM (Node Package Manager, ou Gerenciador de pacotes NODE)

Vem instalado junto com o NODE, e é capaz de instalar ou executar comandos em linha de comando.

# PACOTES

Projeto escrito em NODE  
que foi disponibilizado do  
uma pessoa ou equipe na  
comunidade DEV.

São disponibilizados no  
site do NPM

<https://www.npmjs.com>





# **INICIALIZANDO PROJETOS COM FRAMEWORKS**

```
npx create-react-app login-app  
cd login-app
```

# REACT

1-Criando aplicação com REACT

```
cd ..
```

```
npm start
```

# REACT

2- executado APP

```
✓ Project name: ... <your-project-name>
✓ Add TypeScript? ... No / Yes
✓ Add JSX Support? ... No / Yes
✓ Add Vue Router for Single Page Application development? ... No / Yes
✓ Add Pinia for state management? ... No / Yes
✓ Add Vitest for Unit testing? ... No / Yes
✓ Add an End-to-End Testing Solution? ... No / Cypress / Playwright
✓ Add ESLint for code quality? ... No / Yes
✓ Add Prettier for code formatting? ... No / Yes

Scaffolding project in ./<your-project-name>...
Done.
```

```
$ npm create vue@latest
```

# VUE.JS

## 1- Criando projeto VUE.JS

```
$ cd <your-project-name>  
$ npm install  
$ npm run dev
```

# VUE.JS

## 2-Executando app VUE



```
npm init @angular myApp
```

# ANGULAR

1- Criando projeto Angular.

```
npm start
```

# ANGULAR

## 2- iniciando App Angular