**Roteiro**

**Parte 1**

Olá Pessoal! Sejam bem vindos! Hoje vamos falar sobre um tema muito importante para o desenvolvimento de sistemas: os frameworks.

Mas antes disso, comente se você já utilizou algum framework? Qual foi a sua experiência?

**Parte 2**

Para começar, o que é um framework? Um framework, ou uma estrutura de trabalho, é um conjunto de bibliotecas, que abordam funcionalidades, e estruturas, para o desenvolvimento de aplicações, com a finalidade de fornecer soluções e facilidade para um problema em comum entre os desenvolvedores. Onde pode incluir funções como:

- Bibliotecas e módulos para tarefas específicas;

- Padrões de design recomendados para estruturar o código;

- Ferramentas de desenvolvimento;

- Standards de codificação.

**Parte 3**

Agora que já entendemos o que é um framework, vamos conhecer alguns tipos diferentes de frameworks.

Full-Stack: Primeiro, temos os frameworks de desenvolvimento full-stack, que abrangem tanto o front-end quanto o back-end. Um exemplo famoso é o Django, que usa Python para lidar com toda a lógica da aplicação.

Microframework: Depois, temos os Microframework são Frameworks que possuem uma estrutura inicial muito mais simples quando comparado a um Framework convencional. É inútil sem uma base, mas facilita a fornecedor recursos específicos para um framework. Um exemplo é o Flask, também em Python.

API frameworks: Por último, temos os frameworks de API são estruturas que combinam APIs e frameworks que ajudam na criação de interfaces para comunicação entre sistemas. O FastAPI, por exemplo, é uma excelente opção para criar APIs de forma rápida e eficiente.

**Parte 4**

Além dos tipos de frameworks, precisamos falar sobre a arquitetura deles. A mais comum é o MVC - Model-View-Controller. Ele divide o código em três partes:

Model-View-Controller (MVC): O MVC é um padrão de arquitetura de software. O MVC sugere uma maneira que de dividir as responsabilidades dentro de um software web.

Model: Este componente encapsula as informações essenciais e a funcionalidade do sistema.

View: Ela controla a interface do usuário, manipulando as ações do usuário.

Controller: Gerencia a entrada do usuário e organiza as interações entre o Model e a View.

Model-View-ViewModel (MVVM): Construído para aplicações mais modernas, principalmente aquelas que assumem interfaces de usuário complexas e dinâmicas.

Model: Semelhante ao MVC, também manipula as infomações essenciais.

View: Representação visual e interação com o usuário.

ViewModel: Ele aprimora ainda mais a interação entre o View e o Model, assim como o Controller do MVC.

**Parte 5**

Agora vamos abordar um framework específico, o React. O React é um framework JavaScript criado pelo Facebook (atual Meta) que é usado para criar interfaces de usuário (UI) em aplicativos web.

A biblioteca foi criada por um time de desenvolvedores do Facebook, em 2011, com a finalidade de otimizar a atualização do feed de notícias da rede social.

**Parte 6**

O React é utilizado em diversas aplicações web, tanto para dispositivos móveis, quanto para desktops, como por exemplo o Facebook, que utiliza o React para criar interfaces de usuário, onde precisa ser atualizada constantemente, também para aplicações que precisam ser escaláveis e mantidas por equipes grandes. Além de aplicações móveis nativas para iOS e Android, permitindo que os desenvolvedores criem aplicações móveis com alta qualidade e que se comportem como aplicações nativas, sendo construídas usando tecnologias web, como JavaScript e CSS.

**Parte 7**

E aí, pessoal, gostaram do vídeo? Espero que tenham ficado claro como os frameworks são essenciais no desenvolvimento de sistemas!

Se você gostou do vídeo, não esqueça de deixar seu like, se inscrever no canal e compartilhar com seus amigos!

Até a próxima aula! Valeu!