BACK-END

AULA 1

Nesta disciplina, iremos mergulhar na parte invisível dos sistemas: responsável por processar dados, validar regras, armazenar informações e entregar resultado ao usuário.

Objetivos:

* Compreender o papel do back-end no desenvolvimento de sistemas.
* Aprender os fundamentos das linguagens e tecnologias utilizadas.
* Criar APIs e sistemas funcionais que interaja, com banco de dados.
* Desenvolver habilidades práticas para atuar como programador back-end.

**O que é Back-End?**

**Conceito**

* Parte do sistema que roda no servidor.
* Gerencia toda a lógica do sistema, como regras de negócio, banco dados, etc.
* Envia e recebe dados do front-end via APIs HTTP.

**Responsabilidades**

* Processar requisições.
* Validar dados.
* Acessar e manipular banco de dados.
* Garantir segurança e integridade da aplicação.

Gerenciamento e processamento dos dados/lógica do sistema.

**Analogia**

* O front-end é o garçom, a decoração e o cardápio.
* O back-end é a cozinha, os chefs, os ingredientes, a contabilidade. Tudo depende deles para que o restaurante funcione, mas **VOCÊ** não os **VÊ.**

**Front-End VS Back-End**

Front:

* Visível; Utiliza HTML, CSS, JavaScript; Responsável pela interface, design, usabilidade; É encontrado no navegador.

Back:

* Invisível; utiliza PHP, Python, Java...; Responsável pela lógica, dados regras do sistema; Encontrado no servidor.

Resumo:

O Back-End é tudo que funciona por TRÁS do sistema, trabalhando no processamento de todo o funcionamento e lógica dele. Ele cuida da segurança, das regras de negócio, da lógica, da performance e da comunicação com o banco de dados.

Linguagens:

Algumas das principais linguagens no Back-End são:

* PHP – Uma linguagem simples, fácil de aprender, muito utilizada na web.
* Node.js – Executa o JavaScript no servidor.
* Python – Linguagem clara e objetiva, muito utilizada em áreas de dados.
* Java – Uma linguagem muito complexa, utilizada em GRANDES sistemas, principalmente empresariais.
* C# (.NET) – Linguagem usada em sistemas corporativos e no Windows.

Utiliza protocolos HTTP/HTTPS, APIs e JSON.

**Ferramentas Essenciais**

* VS Code – Um editor de código gratuito, com interface simples e moderna, e conta com suporte para várias linguagens de programação. Conta também com suporte para Git.
* Terminal / CMD – Executa comandos de instalação, compilação...
* Navegador – Usado para testar sistemas, ver resultados, inspecionar erros e respostas
* Servidor Local – Permite simular o ambiente de produção na sua máquina.

**Node.js**

**Plataforma** que permite a utilização do JavaScript no servidor, assim, podendo utilizar o JavaScript para construir aplicações e sistemas.

Possui algumas peculiaridades, como ser baseado no motor V8 (Chrome), ser leve, escalável, assíncrono, e ter pacotes disponíveis via NPM (Gerenciador de pacotes).

Em 1995, JavaScript começou a ser utilizado na Web, de 1999 a 2009, eram utilizados muitos servidores PHP, Java e ASP. Em 2009, foi criado o Node.js, permitindo a utilização do JavaScript no Back-End. A partir de 2010, o JavaScript se popularizou no ambiente Back-End.

Ferramentas essenciais para a utilização do Node.js:

* VS Code
* Node.js
* NPM
* Terminal
* Navegador