



Desenvolvimento de Sistemas

Frontend, Backend e API

O que é Front-end?



- O Front-end é a parte visível de uma aplicação, ou seja, tudo que o usuário interage diretamente: botões, menus, formulários, cores, animações, etc. É a interface gráfica.
- **Principais características:**
 - Executa no navegador do usuário.
 - Foco em experiência do usuário (UX) e design (UI).
 - Trabalha com HTML, CSS e JavaScript como base.

Linguagens e Tecnologias:



-
- HTML → Estrutura da página.
- CSS → Estilo e layout.
- JavaScript → Interatividade.
- Frameworks/Bibliotecas: React.js, Angular, Vue.js, Bootstrap (para estilização)
-
- **Exemplo prático:**
- Quando você acessa um site e vê um botão "Enviar", isso é Front-end. Se ao clicar aparece uma animação ou muda de cor, isso também é Front-end.

O que é Back-end?



- O Back-end é a parte lógica e funcional da aplicação, que roda no servidor. Ele processa dados, faz cálculos, acessa banco de dados e envia respostas para o Front-end.
-
- **Principais características:**
- Executa no servidor.
- Foco em lógica, segurança e integração.
- Trabalha com bancos de dados e APIs.

Linguagens e Tecnologias:

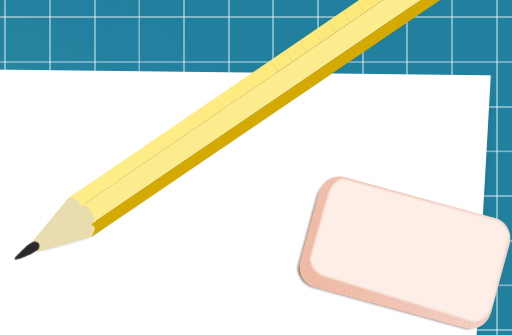
- JavaScript (Node.js) → Muito usado em APIs.
- Python (Django, Flask) → Ideal para aplicações web.
- Java (Spring Boot) → Corporativo.
- PHP (Laravel) → Sites dinâmicos.
- Ruby (Rails) → Aplicações rápidas.
- Banco de Dados:
- MySQL, PostgreSQL (relacionais)
- MongoDB (NoSQL)
-
- **Exemplo prático:**
- Quando você clica em "Enviar" e seus dados são salvos no banco, isso é Back-end. Ele pega os dados, valida, armazena e retorna uma resposta.





Comunicação Frontend e Backend?

- Eles se comunicam via API (Application Programming Interface).
- **O que é uma API?**
- É um conjunto de regras que permite que sistemas diferentes se comuniquem. Geralmente, usamos APIs REST ou GraphQL.
-



- **Funcionamento:**

- O Front-end faz uma requisição HTTP (GET, POST, PUT, DELETE) para a API.
- O Back-end processa e retorna uma resposta em JSON.
- O Front-end exibe os dados para o usuário.
-

- **Exemplo simples de fluxo:**

- Usuário preenche um formulário no site (Front-end).
- O site envia os dados para `https://api.meusite.com/usuarios` (API).
- O servidor (Back-end) salva no banco e retorna `{ "status": "sucesso" }`.
- O Front-end mostra uma mensagem: "Cadastro realizado com sucesso".

Front - end



Back - end



Front - end



Back - end

