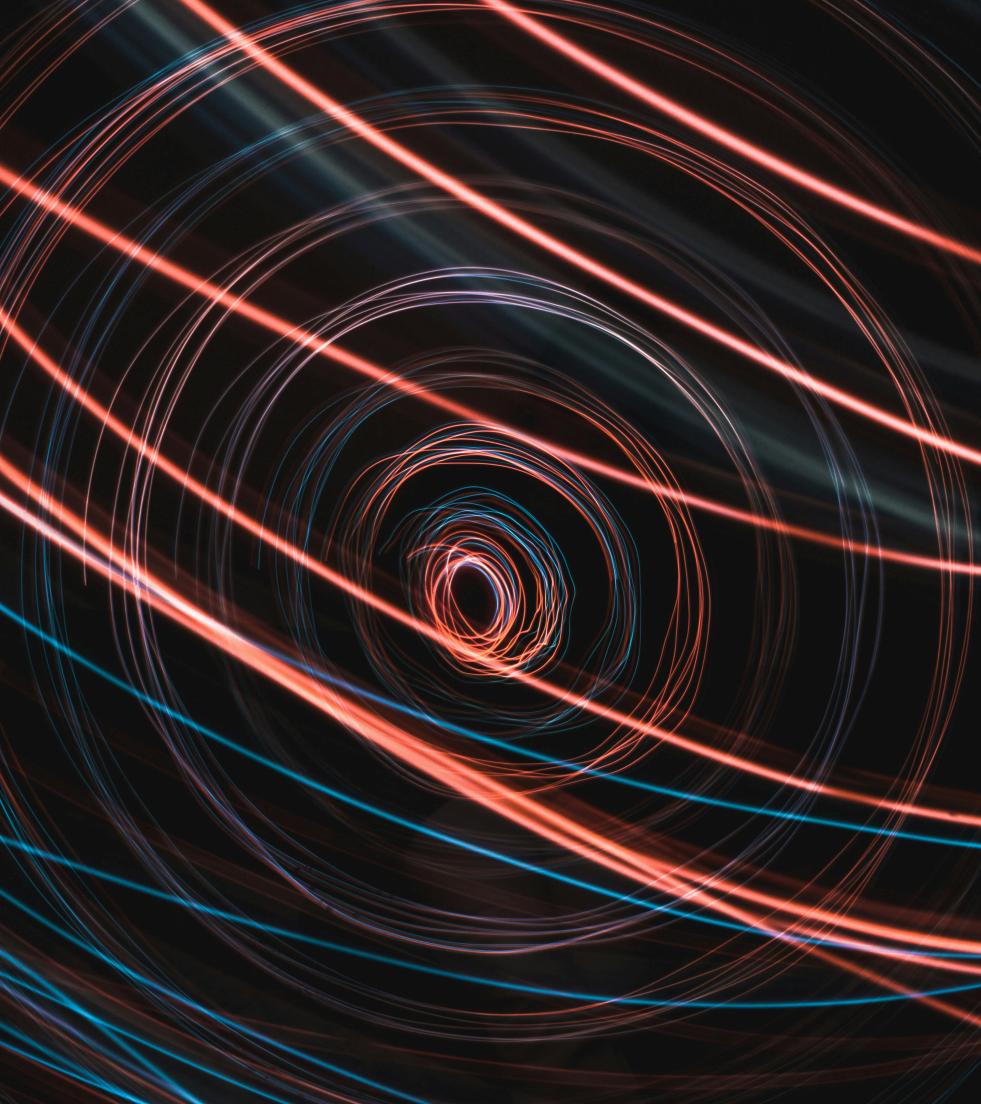


Aula Zero

Programação Web 1



Objetivos de Aprendizagem

- Definir os conceitos gerais de PW
- Apresentar a metodologia

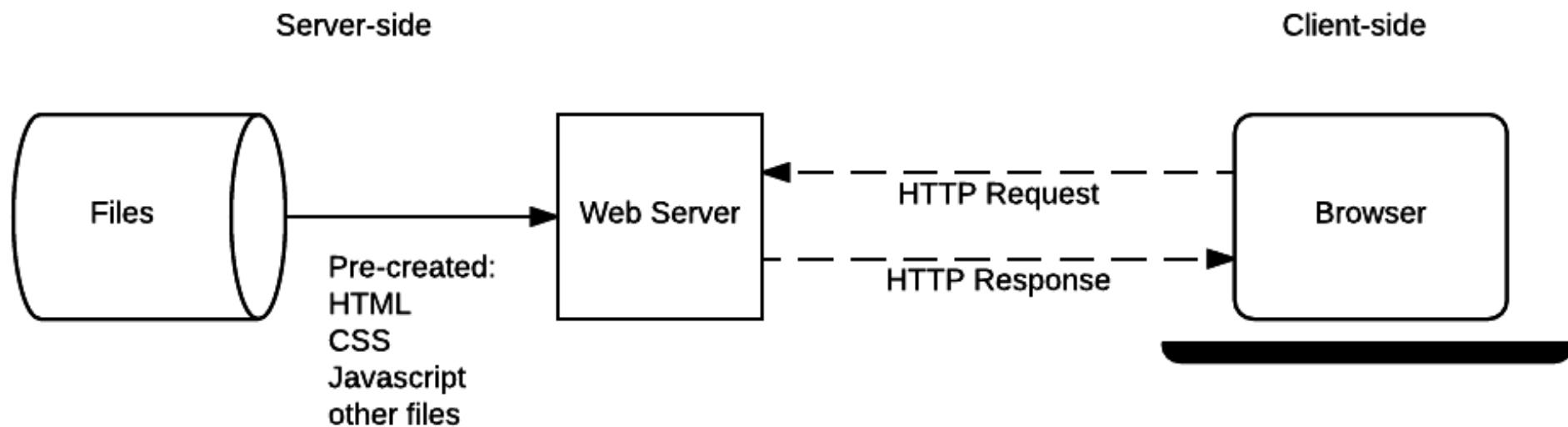
Agenda

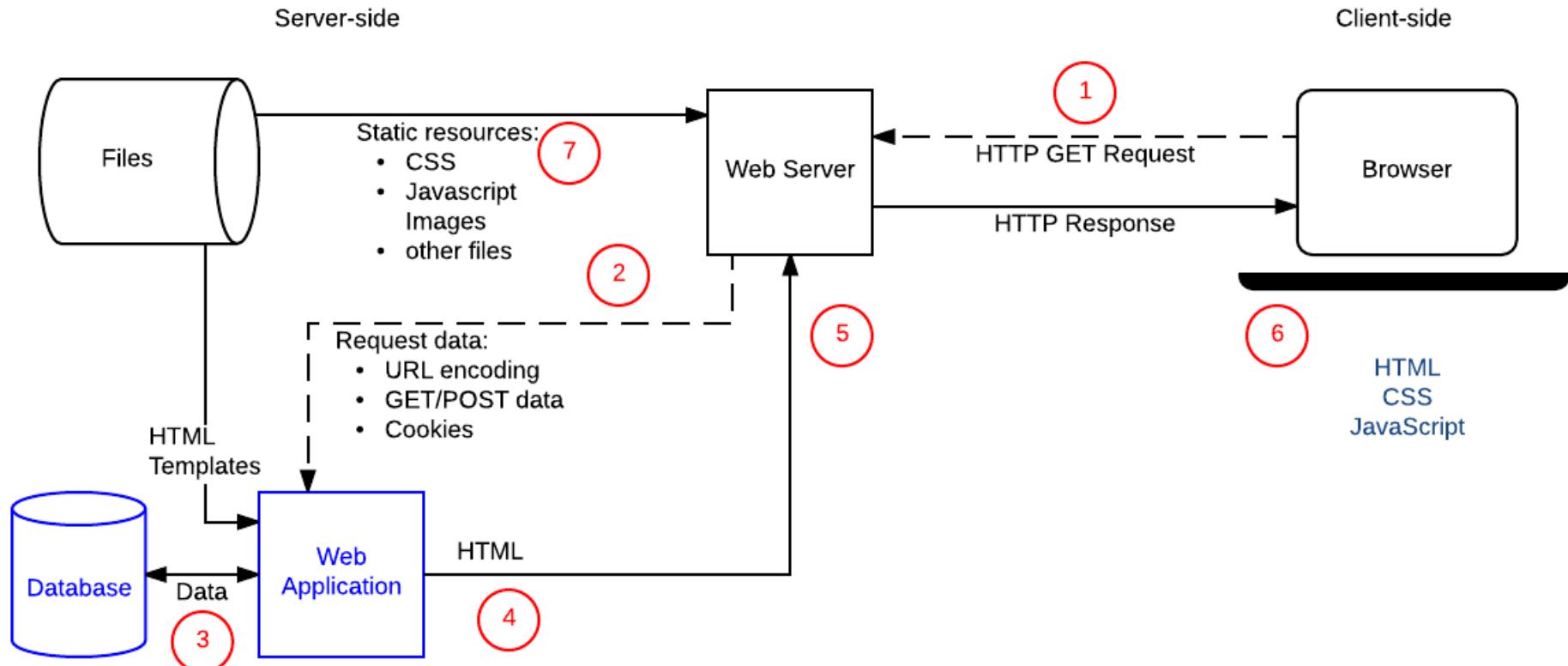
- O que é Programação Web?
- Conceitos e Tecnologias
- Metodologia

Conceitos de PW

Desenvolvimento Web (Programação Web) é o processo de criação de aplicações (sites) utilizando tecnologias implantadas em servidores remotos acessíveis através da Internet.

Sites Estáticos





Desenvolvimento Web

- **Frontend**
- **Backend**
- **Fullstack**

Desenvolvimento *Frontend*

O **Frontend** é responsável pela interface do usuário e pela experiência de interação com o sistema. É o que o usuário vê e interage diretamente no navegador.

Desenvolvimento *Frontend*

Principais responsabilidades

- *Design* da interface gráfica (UI)
- Experiência do usuário (UX)
- **Responsividade** e acessibilidade
- Interações dinâmicas (ex.: cliques, animações)

Desenvolvimento *Frontend*

Tecnologias Típicas

- **HTML5:** Estrutura o conteúdo web 
- **CSS3:** Estiliza e organiza o layout 
- **JavaScript:** Adiciona interatividade 
- **Frameworks:** React, Angular, Vue.js   

Desenvolvimento *Frontend*

Atividades Típicas

- Criação de botões, menus, formulários
- Garantir que o site funcione bem em dispositivos móveis e de mesa
- Implementar animações e transições
- Visual adequado para aplicação (Cores, fontes, *layout*, etc)

Desenvolvimento ***Backend***

O **Backend** envolve a lógica na retaguarda da aplicação, priorizando a interação com bancos de dados, autenticação e outras funcionalidades essenciais que acontecem no servidor e são transparentes ao usuário.

Desenvolvimento *Backend*

Principais responsabilidades

- Processamento de dados e lógica de negócios
- Gerenciamento de banco de dados
- Autenticação e autorização de usuários
- Integração com APIs e serviços externos

Desenvolvimento *Backend*

Tecnologias Típicas

- **Linguagens de programação:** Python, Java, Ruby, PHP, Node.js
 -     
- **Bancos de dados:** MySQL, PostgreSQL, MongoDB
 -   
- **APIs:** RESTful, GraphQL
- **Servidores:** Apache, Nginx

Desenvolvimento *Backend*

Atividades típicas

- Armazenamento e recuperação de dados de um banco de dados
- Autenticação de usuários
- Criação de APIs para comunicação com o *frontend*

Desenvolvimento *Fullstack*

Aplicado por alguns desenvolvedores que reúnem as habilidades tanto de FE quanto de BE.

Desenvolvimento *Fullstack*

Principais responsabilidades

- Combinar habilidades de *frontend* e *backend*
- Entender todo o ciclo de desenvolvimento de uma aplicação
- Integrar APIs, banco de dados e lógica de interface

Desenvolvimento *Fullstack*

Tecnologias Típicas

- **Linguagens e frameworks Fullstack:** Node.js, Ruby on Rails, Django
 -   
- **Conhecimento de tecnologias de frontend e backend**
- **Ferramentas DevOps:** Docker, Kubernetes, CI/CD  

Desenvolvimento *Fullstack*

Exemplos de atividades

- Desenvolver uma aplicação completa, desde a interface até a lógica no servidor.
- Implementar segurança na autenticação e autorização.
- Integração de banco de dados com a interface do usuário.

Característica	Frontend	Backend	Fullstack
Objetivo	Interface e interação do usuário	Lógica de negócios	Ambos, frontend e backend
Tecnologias	HTML, CSS, JavaScript, Frameworks	Linguagens de servidor, bancos	Combinação de frontend e backend
Atividades	Design responsivo, UX, UI	Criação de APIs, autenticação	Desenvolvimento completo
Responsabilidade	Estética e usabilidade	Funcionalidade e desempenho	Equilíbrio entre usabilidade e funcionalidade
Desafios	Acessibilidade, compatibilidade	Escalabilidade, segurança	Manter o equilíbrio entre os dois lados

Que caminho seguir?

- **Frontend:** Se você é criativo, gosta de *design* e da interação com o usuário.
- **Backend:** Se você prefere lidar com lógica, servidores e a manipulação de dados.
- **Fullstack:** Se você quer ter uma visão global do desenvolvimento e gosta de trabalhar com todos os aspectos de uma aplicação.

Tecnologias e Ferramentas

- HTML5 
- CSS3 
- Node.js (Express) 
- APIs REST
- Postman 
- MongoDB 
- Git 
- Visual Studio Code  (ou IDE de preferência)

Por que?



2025 Developer Survey

This annual Developer Survey provides a crucial snapshot into the needs of the global developer community, focusing on the tools and technologies they use or want to learn more about.

Programming, scripting, and markup languages

- *Most popular technologies*

- *Admired and Desired*

Databases

- *Most popular technologies*

- *Admired and Desired*

Web frameworks

- *Most popular technologies*

- *Admired and Desired*

Integrated development environment

- *Most popular technologies*

- *Admired and Desired*

PW1 2025.2

Metodologia PW1 2025.2

- A cada aula
 - Abordagem de um conceito específico
 - Demonstrações de código exemplificando o conceito
 - Até 50 minutos de exposição
 - Atividades didáticas
- Atividades avaliativas em algumas aulas (N1)
- Trabalho final (N2)
 - Individual
 - Tema livre
 - Criação de aplicação com funcionalidades

Perguntas

Referências

- *Web development*

- Arquitetura Cliente/Servidor

- Stack Overflow 2025 Survey

José Roberto Bezerra

jroberto@ifce.edu.br

<https://github.com/jroberto76>

Powered by  Sliddev

Foto by José Campos from Unsplash

