

Plano de Aprendizado

Introdução

Html e CSS

Javascript

Frontend

Backend

Tecnologias Citadas

Introdução

Esse documento será usado para registrar o caminho que eu pretendo seguir no mundo da programação. Basicamente, são os passos que pretendo dar aos poucos até virar um desenvolvedor Fullstack.

Meu objetivo é adquirir habilidades sólidas tanto no front-end quanto no back-end, permitindo que eu desenvolva aplicações web completas, desde a interface do usuário até a lógica do servidor e a gestão de banco de dados. Inicialmente, vou me concentrar em fortalecer meu conhecimento em HTML, CSS e JavaScript, fundamentais para a criação de interfaces responsivas e dinâmicas. Após concluir essa base, pretendo dominar React, uma biblioteca popular para o desenvolvimento de interfaces interativas, seguida por Node.js e Express para construir o back-end das aplicações.

Com o tempo, irei explorar bancos de dados, como MongoDB e PostgreSQL (plataforma essa que já tenho um certo conhecimento e certificação), além de ferramentas de versionamento como Git e técnicas de DevOps para facilitar o ciclo de desenvolvimento e implantação.

Esse plano não só me guiará passo a passo na minha jornada, mas também será um registro da minha evolução como programador, documentando cada conquista e os desafios superados no caminho para me tornar um desenvolvedor Fullstack completo.

Html e Css

O HTML será o ponto de partida na minha jornada para dominar o desenvolvimento web. Através dele, vou aprender a estruturar o conteúdo de sites de maneira organizada e semântica, garantindo que as páginas sejam compreensíveis tanto para os usuários quanto para os mecanismos de busca. Ao estudar HTML, vou me familiarizar com a marcação de textos, links, imagens, formulários, e aprender como utilizar tags semânticas corretamente para melhorar a acessibilidade e o SEO.

O CSS será o responsável por transformar as páginas HTML em sites visualmente atraentes e responsivos. Com o CSS, vou aprender a controlar cores, fontes, espaçamentos e layouts, criando designs que se adaptem a diferentes dispositivos e tamanhos de tela. Além disso, vou explorar ferramentas modernas como Flexbox e CSS Grid para organizar os elementos da página de forma eficiente e flexível. Estudarei também animações e transições para adicionar interatividade e movimento ao design.

Javascript

O JavaScript será o próximo passo essencial na minha jornada, permitindo que eu adicione interatividade e dinamismo às páginas web. Com JavaScript, vou aprender a manipular o DOM (Document Object Model) para modificar elementos HTML em tempo real, responder a eventos de usuários e criar funcionalidades mais complexas, como validação de formulários, animações e interações dinâmicas.

Plano de Aprendizado

Introdução

Html e CSS

Javascript

Frontend

Backend

Tecnologias Citadas

Além disso, estudarei conceitos fundamentais, como tipos de dados, estruturas de controle, funções, e a manipulação de arrays e objetos, garantindo uma compreensão sólida da linguagem. Dominar o JavaScript nativo será crucial para desenvolver aplicações web mais ricas e funcionais, além de abrir portas para a criação de lógicas de programação tanto no front-end quanto no back-end no futuro.

Frontend

No desenvolvimento front-end, uma das ferramentas mais importantes que irei dominar é o React, uma biblioteca JavaScript poderosa e amplamente utilizada para a criação de interfaces de usuário interativas e dinâmicas. O React se destaca pela sua abordagem baseada em componentes reutilizáveis, que permite dividir a interface em pequenas partes independentes, facilitando a manutenção e expansão de aplicações complexas.

Através do Virtual DOM, React otimiza a atualização de elementos na página, garantindo alto desempenho e uma experiência de usuário mais fluida. Além disso, com o uso de hooks, como useState e useEffect, vou poder gerenciar o estado e os efeitos colaterais de forma simples e eficiente, sem a necessidade de componentes de classe.

Além do React, ferramentas como Bootstrap e Sass são fundamentais no desenvolvimento front-end, otimizando a criação de interfaces eficientes e responsivas.

Backend

No desenvolvimento back-end, o Node.js se destaca como uma ferramenta poderosa para construir servidores e APIs escaláveis utilizando JavaScript. Com o Node.js, posso criar aplicações de alta performance graças ao seu modelo assíncrono e orientado a eventos, que permite processar múltiplas requisições simultaneamente sem bloquear o servidor.

Além disso, o Express.js, um framework minimalista para Node, facilita a criação de rotas, middleware e gerenciamento de requisições HTTP, tornando o desenvolvimento de APIs RESTful mais ágil e eficiente.

Ao dominar o Node.js, estarei equipado para desenvolver aplicações fullstack, conectando o front-end e o back-end de forma eficiente e escalável.

Tecnologias Citadas

- HTML: Linguagem de marcação usada para estruturar o conteúdo da web.
- CSS: Linguagem de estilo usada para descrever a apresentação de documentos HTML.
- JavaScript: Linguagem de programação usada para adicionar interatividade às páginas web.
- React: Biblioteca JavaScript para construir interfaces de usuário interativas.
- Node.js: Ambiente de execução JavaScript no lado do servidor.

**Plano de
Aprendizado**

Introdução

Html e CSS

Javascript

Frontend

Backend

Tecnologias Citadas

Express.js: Framework minimalista para Node.js, usado para criar servidores e APIs.

MongoDB: Banco de dados NoSQL orientado a documentos.

PostgreSQL: Sistema de gerenciamento de banco de dados relacional.

Git: Sistema de controle de versão.

Bootstrap: Framework front-end para desenvolvimento de sites responsivos.

Sass: Pré-processador CSS que adiciona funcionalidades como variáveis e aninhamento.