



Intensivão de **JAVASCRIPT**

T

Apostila Completa

AULA 4

Guia passo a passo: Projeto Chatbot com IA



Parte 1

Instalando as ferramentas



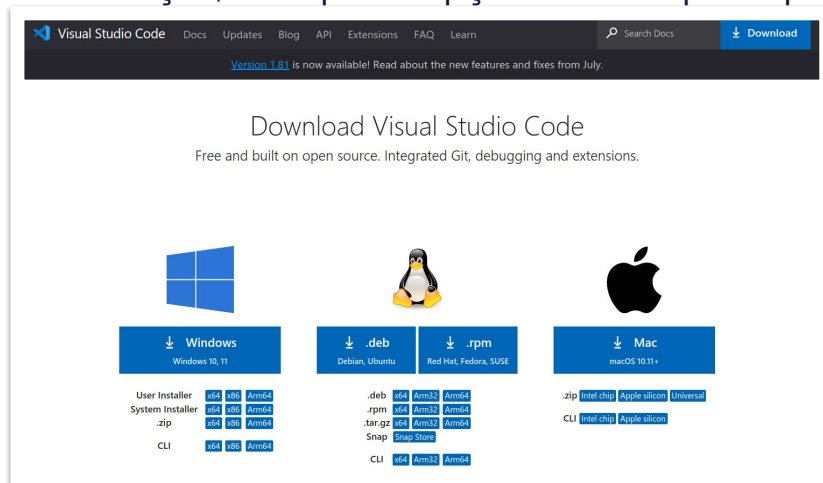


Instalando as ferramentas – VS CODE

Caso você ainda não tenha o Visual Studio Code instalado, basta seguir os procedimentos abaixo. A instalação do VS Code é totalmente gratuita, e você pode usá-lo em seu computador sem precisar pagar nada. O link para fazer o download do programa é mostrado abaixo:

<https://code.visualstudio.com/>

- 1 - Baixe o arquivo de instalação correspondente ao seu sistema operacional (Windows, MacOS ou Linux).
- 2 - Execute o arquivo de instalação e siga as instruções na tela. Em geral, é só continuar clicando em "Próximo".
- 3 - No Windows, durante a instalação, marque a opção "Add to path" para adicionar o VS Code às suas variáveis de ambiente.





Instalando as ferramentas – Node.js

O Node.js é um ambiente de execução de código JavaScript do lado do servidor. Ele permite que você execute código JavaScript fora do navegador, o que significa que você pode criar aplicativos de servidor, scripts de linha de comando e muito mais usando JavaScript. Para instalar o Node.js, você pode seguir os seguintes passos:



- Acesse o site oficial do Node.js em <https://nodejs.org>.
- Na página inicial, você verá duas versões para download: LTS (Long Term Support) e Current. A versão LTS é recomendada para a maioria dos usuários, pois é mais estável e possui suporte a longo prazo. Selecione a versão LTS ou a versão mais recente, se preferir.
- Após selecionar a versão desejada, você será redirecionado para a página de download. Escolha o instalador adequado para o seu sistema operacional (Windows, macOS ou Linux) e clique no link para iniciar o download.
- Após o download ser concluído, execute o instalador e siga as instruções na tela para concluir a instalação.
- Após a instalação ser concluída, você pode verificar se o Node.js foi instalado corretamente abrindo o terminal ou prompt de comando e digitando o comando `node -v`. Se a versão do Node.js for exibida, significa que a instalação foi bem-sucedida.

Parte 2

Apresentação do Projeto Chatbot com IA!

Apresentação do Projeto Chatbot com IA

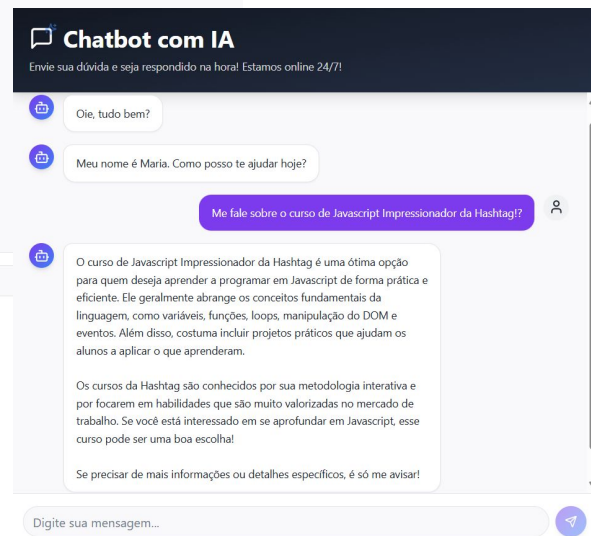
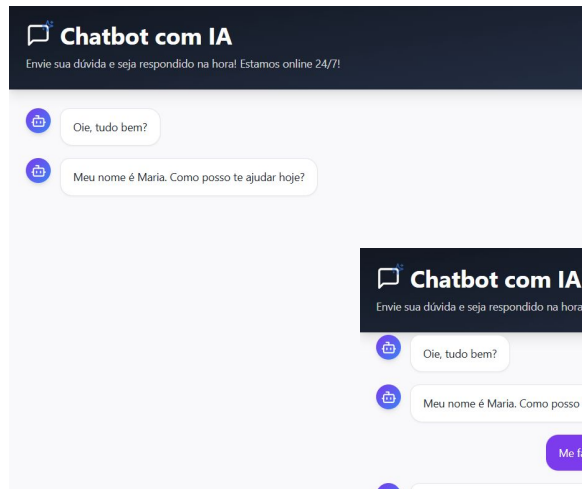


Bem-vindo ao **Chat Moderno com IA**, um projeto desenvolvido para demonstrar como a **inteligência artificial** pode transformar a forma como interagimos digitalmente, tornando a comunicação mais fluida, acessível e envolvente.

Nesta aplicação, você poderá **conversar em tempo real com uma IA**, experimentando uma interface moderna construída com **JavaScript (TypeScript)** e desenvolvida na plataforma **Lovable**.

O projeto combina tecnologias atuais e eficientes — como **React**, **TailwindCSS** e **Supabase** — para criar uma experiência interativa completa, que une **design moderno**, **comunicação instantânea** e **respostas inteligentes**.

Nosso objetivo é apresentar uma plataforma de conversa que vai além de um simples chat: um ambiente inovador, intuitivo e impulsionado por IA. 🚀



Parte 3

Front-end x Back-end



Projetos Frontend e Projetos Backend em JavaScript são duas áreas distintas do desenvolvimento web que trabalham juntas para criar aplicativos web dinâmicos e interativos.

Projetos Frontend:

O Frontend refere-se à parte da aplicação web com a qual os usuários interagem diretamente. Isso inclui o layout, design, e interatividade que o usuário vê e com o qual interage em um navegador da web.

- **Tecnologias Frontend em JavaScript:** No frontend, JavaScript é a linguagem principal para tornar as páginas web interativas. Junto com HTML (HyperText Markup Language) e CSS (Cascading Style Sheets), o JavaScript é fundamental para criar uma experiência de usuário envolvente e responsiva.
- **Frameworks e Bibliotecas:** Existem muitos frameworks e bibliotecas JavaScript populares para o frontend, como React.js, Angular.js e Vue.js. Essas ferramentas ajudam os desenvolvedores a criar interfaces de usuário complexas e dinâmicas de forma mais eficiente.
- **Responsividade e Design:** No frontend, os desenvolvedores se concentram na responsividade do design, garantindo que o aplicativo funcione bem em diferentes dispositivos e tamanhos de tela.



Projetos Backend:

O Backend é a parte invisível de um aplicativo web que lida com a lógica de negócios, processamento de dados e interações com o servidor. Ele suporta e gerencia o funcionamento do frontend.

- **Tecnologias Backend em JavaScript:** Node.js é a tecnologia principal para desenvolvimento backend em JavaScript. Ele permite que os desenvolvedores usem JavaScript no servidor, o que simplifica a comunicação entre frontend e backend, já que a mesma linguagem pode ser usada em ambos os lados.
- **Frameworks e Bibliotecas:** Existem diversos frameworks e bibliotecas para o desenvolvimento backend em JavaScript, como Express.js, Nest.js, e Koa.js. Essas ferramentas fornecem estruturas e funcionalidades para criar servidores web robustos e escaláveis.
- **Gerenciamento de Dados:** No backend, os desenvolvedores lidam com operações de banco de dados, autenticação de usuários, manipulação de arquivos e outras tarefas relacionadas ao armazenamento e recuperação de dados.



Projetos FullStack:

Um projeto fullstack de JavaScript é aquele que engloba tanto o desenvolvimento do frontend quanto do backend utilizando JavaScript como linguagem principal. Nesse tipo de projeto, JavaScript é utilizado em todas as camadas da aplicação web, desde a interface do usuário até a lógica de servidor e o banco de dados, se aplicável.

Componentes de um Projeto Fullstack em JavaScript:

Frontend:

- Responsável pela interface do usuário (UI) que os usuários interagem diretamente em um navegador web. Utiliza tecnologias como HTML, CSS e JavaScript para criar uma experiência interativa e responsiva. Frameworks como React.js, Angular.js, Vue.js são comuns para construção de interfaces dinâmicas.

Backend:

- Lida com a lógica de negócios, processamento de dados e interações com o servidor. JavaScript pode ser usado no servidor com o auxílio de plataformas como Node.js, permitindo que os desenvolvedores usem uma linguagem comum em todo o projeto. Frameworks como Express.js, Nest.js, e Koa.js são comumente usados para criar APIs RESTful e gerenciar rotas no servidor.

Parte 4

Tecnologias e Ferramentas utilizadas



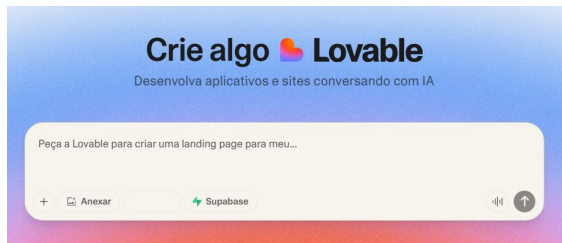
♥ JavaScript (TypeScript)

O **JavaScript** é a linguagem responsável por adicionar interatividade e lógica às páginas web. No nosso projeto, usaremos **TypeScript**, que é uma versão aprimorada do JavaScript — ele adiciona **tipagem estática**, ajudando a detectar erros e tornando o código mais organizado e fácil de manter.

♥ Lovable

O **Lovable** é uma plataforma que permite criar aplicações web completas de forma visual, sem precisar configurar manualmente o ambiente de desenvolvimento. Ele combina design e código em um único espaço, facilitando o processo de construção de projetos modernos com **React**, **TailwindCSS** e **TypeScript**.

Para começar, basta acessar o site <https://lovable.dev>, criar uma conta e iniciar um novo projeto. A interface é intuitiva e permite adicionar componentes, conectar APIs e visualizar o resultado em tempo real.





♥ Supabase

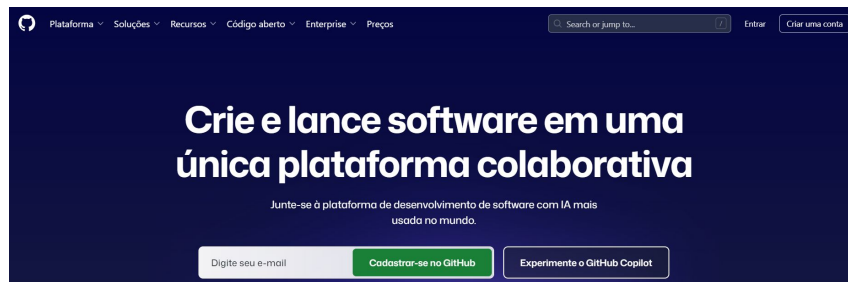
O **Supabase** é uma plataforma que fornece serviços prontos para autenticação, banco de dados e armazenamento de arquivos — funcionando como uma alternativa moderna ao Firebase.

No nosso projeto, ele será usado para **armazenar informações e mensagens do chat**.

♥ GitHub

O **GitHub** é uma plataforma usada para **armazenar, versionar e compartilhar códigos**. Ele permite trabalhar em equipe e acompanhar o histórico de alterações do projeto.

Para utilizá-lo, você pode criar uma conta em <https://github.com>, instalar o **Git** no seu computador





Tecnologias e Ferramentas utilizadas

🧩 O que é o Git?

Git é um **sistema de controle de versão**. Ele permite que você acompanhe mudanças no seu código ao longo do tempo, trabalhe em equipe sem sobrescrever o trabalho dos outros e volte para versões anteriores se algo der errado.

É uma ferramenta **fundamental para desenvolvedores**, especialmente quando usada junto com plataformas como **GitHub** (que armazena seus projetos online e facilita a colaboração).

💻 Como instalar o Git no computador 👉 Para Windows:

1. Vá até o site oficial: <https://git-scm.com/>
2. Clique em **Download for Windows**.
3. O instalador será baixado (.exe). Abra-o.
4. Vá clicando em **Next** nas etapas, mantendo as configurações padrão (isso já configura o Git Bash e o Git na linha de comando).
5. Clique em **Install** e depois em **Finish**.

Após isso, você pode abrir o **Git Bash** ou usar o Git direto no **Prompt de Comando (CMD)** ou **terminal do VS Code**.



Tecnologias e Ferramentas utilizadas

Prompt de IA

Um **prompt de IA** é o comando ou instrução que você envia para um modelo de inteligência artificial, como o ChatGPT. Ele orienta a IA sobre **o que fazer, como responder e qual estilo seguir**.

Durante o desenvolvimento do projeto, você poderá usar prompts para gerar ideias de código, resolver erros ou até criar textos automáticos para o chat.

A chave está em ser claro e específico — quanto mais contexto o prompt tiver, mais precisa será a resposta da IA.





Tecnologias e Ferramentas utilizadas

O Papel do Desenvolvedor na Era da IA

A inteligência artificial é uma ferramenta poderosa para acelerar processos e gerar soluções iniciais, mas **não substitui o raciocínio lógico, a experiência prática e o olhar crítico do desenvolvedor.**

Especialmente em projetos com JavaScript e frameworks como React, é essencial compreender o funcionamento do código, identificar falhas e aplicar correções com precisão.

A IA pode sugerir caminhos, mas é o desenvolvedor quem garante que o projeto funcione de verdade. Essa parceria entre humano e máquina é o que torna o processo criativo e técnico realmente eficiente.



Parte 5

Construindo com IA: Etapas de um Projeto Real



Construindo com IA: Etapas de um Projeto Real

✂ Etapa 1 — Criando Conta no Lovable

Para começar a utilizar o Lovable e gerar projetos com inteligência artificial, é necessário criar uma conta gratuita na plataforma. O processo é simples e rápido, e permite acesso ao painel onde você poderá interagir com a IA, enviar prompts e receber sugestões de layout e código.

✓ Passo a passo:

1. **Acesse o site oficial:**

Vá para lovable.dev usando seu navegador preferido.

2. **Crie sua conta gratuita:**

Clique em “Sign Up” ou “Criar conta” e preencha os dados solicitados. O cadastro é gratuito e não exige cartão de crédito.

3. **Abra o painel de criação de projeto:**

Após o login, você será direcionado ao painel principal. É aqui que você poderá iniciar novos projetos, conversar com a IA e acompanhar os resultados gerados.

Com a conta criada e o painel aberto, você estará pronto para começar a etapa de criação. A IA do Lovable permite que você envie um prompt descrevendo o que deseja — como o estilo visual, funcionalidades ou estrutura do projeto — e ela gera automaticamente o layout e o código correspondente.



Construindo com IA: Etapas de um Projeto Real

Etapa 2 — Criando o Prompt e Enviando a Imagem

Nesta etapa, o objetivo é orientar a inteligência artificial para gerar um layout e código de chatbot com base nas instruções fornecidas. Isso é feito por meio de um prompt textual e uma imagem de referência que complementa o pedido.

No exemplo, utilizamos o seguinte prompt:

“Quero que você me ajude a criar um chatbot moderno. Como você constrói em React, gostaria que fizesse os códigos em JavaScript e não em TypeScript. Se não for possível, pelo menos escreva TypeScript como se fosse JavaScript.

Esse chatbot deve estar integrado com a API do ChatGPT. Pra isso, me solicite a chave de API que te mando por aqui e você pluga no código. Enquanto você não tiver uma, use uma ‘fake’.

A base do visual do chatbot está na imagem que te mandei. Se conseguir deixar ele mais visual, moderno e chamativo, pode implementar as alterações. Não precisa ser exatamente igual ao que mandei.”

Esse prompt é direto, detalhado e fornece à IA todas as informações necessárias para gerar um projeto funcional e visualmente atrativo.



Construindo com IA: Etapas de um Projeto Real

Imagem de Referência

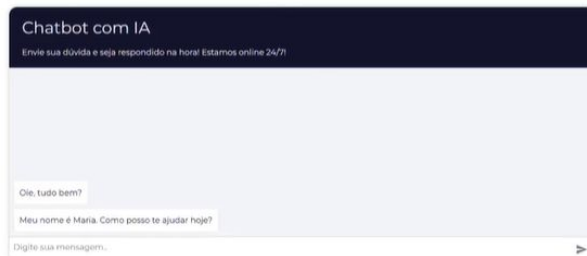
Junto ao prompt, foi anexada uma imagem contendo instruções visuais sobre o estilo desejado para o chatbot. Essa imagem ajuda a IA a compreender melhor o tipo de layout esperado, servindo como guia para a estética e estrutura do projeto.

Resultado Esperado

Com o prompt e a imagem enviados, a IA é capaz de gerar automaticamente:

- A estrutura básica do chatbot
- O layout visual estilizado
- O código em JavaScript (ou TypeScript adaptado)
- Sugestões de integração com a API do ChatGPT

Essa abordagem acelera o processo de prototipagem e garante que o resultado esteja alinhado com os objetivos visuais e funcionais do projeto.





Construindo com IA: Etapas de um Projeto Real

⚙️ Etapa 3 — Entendendo o Resultado e Corrigindo os Erros e Entendendo Variáveis de Ambiente

Após o envio do prompt e da imagem de referência, a inteligência artificial gera automaticamente um projeto com base nas instruções fornecidas. O resultado é visualmente atrativo e tecnicamente estruturado, mas nem sempre funcional em sua totalidade.

💻 Resultado Gerado pela IA

O projeto entregue pela IA inclui:

- Estrutura em **React**
- Código em **TypeScript**
- Estilização com **Tailwind CSS**

O layout gerado é moderno e bem organizado, porém, ao executar o projeto, foi identificado um erro de funcionamento. Isso revela uma limitação comum em soluções geradas por IA: a ausência de ajustes finos e configurações específicas que exigem intervenção humana.



Construindo com IA: Etapas de um Projeto Real

Diagnóstico do Problema

Ao investigar o erro, foi constatado que o projeto tentava acessar **variáveis de ambiente** que não estavam definidas corretamente. Esse tipo de falha é comum quando a IA presume configurações que não foram explicitadas ou que dependem do ambiente local de desenvolvimento.

Explicação da Solução Aplicada

Para corrigir o problema, foram realizadas as seguintes ações:

- Criação manual das variáveis de ambiente necessárias
- Ajuste das chamadas no código para garantir compatibilidade
- Testes de execução para validar o funcionamento completo do projeto

Essas etapas demonstram a importância do conhecimento técnico e da lógica de programação na validação e correção de projetos gerados por IA. Vamos entender as etapas realizadas a partir do próximo slide.



Construindo com IA: Etapas de um Projeto Real

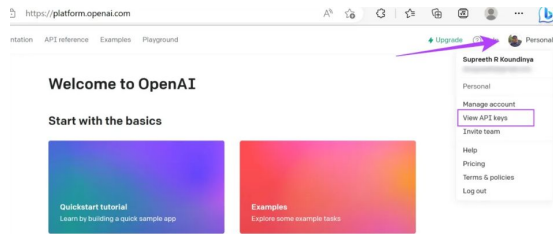
Etapa 4 — Integração com OpenAI

Nesta etapa, o objetivo é conectar o projeto gerado pela IA à API da OpenAI, permitindo que o chatbot utilize inteligência artificial para responder aos usuários com base nos prompts enviados.

Gerando a Chave de API

Para realizar essa integração, é necessário obter uma chave de API diretamente na plataforma da OpenAI. O processo é simples:

1. **Criar uma conta** na [OpenAI Platform](#)
2. **Gerar uma API Key** no painel de usuário
 - Clique na foto do seu perfil no canto superior direito e selecione 'Exibir chaves de API'.
 - Clique no botão 'Criar nova chave secreta'. Isso gerará uma chave de API.
 - Você terá a opção de nomear a chave de API e, uma vez visualizada para você, clique no botão copiar.
 - Depois de copiar a chave, você pode colá-la na ferramenta que requer o uso da chave para acessar o GPT ou qualquer outro produto da OpenAI.



API keys

Your secret API keys are listed below. Please note that we do not display your secret API keys again after you generate them.

Do not share your API key with others, or expose it in the browser or other client-side code. In order to protect the security of your account, OpenAI may also automatically rotate any API key that we've found has leaked publicly.

| NAME | KEY | CREATED | LAST USED |
|------------|------------|-------------|-------------|
| Secret key | [REDACTED] | 29 Mar 2023 | 29 Mar 2023 |

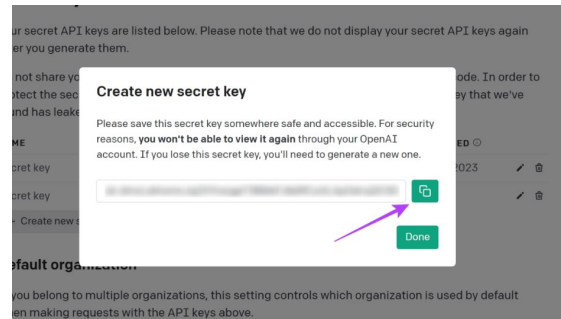
+ Create new secret key

Default organization

If you belong to multiple organizations, this setting controls which organization is used by default when making requests with the API keys above.

Personal

Note: You can also specify which organization to use for each API request. See [Authentication](#) to learn more.



3. Inserir a chave no Lovable como uma **variável de ambiente**, garantindo segurança e organização

Essa chave é essencial para que o chatbot consiga se comunicar com os modelos da OpenAI e gerar respostas inteligentes.



Construindo com IA: Etapas de um Projeto Real

🧩 Variável de ambiente no Projeto

Para manter o projeto organizado e seguir boas práticas, foi criada uma pasta chamada `config`, contendo o arquivo `api.js` com o seguinte conteúdo:

```
// ⚠️ AVISO DE SEGURANÇA:  
// Armazenar chaves de API no front-end expõe sua chave publicamente.  
// Qualquer pessoa pode inspecionar o código e usar sua chave da OpenAI.  
// Isso pode resultar em cobranças inesperadas na sua conta.  
//  
// Para produção, use sempre um backend para proteger suas chaves.  
  
export const OPENAI_API_KEY = import.meta.env.VITE_OPENAI_API_KEY;
```

Esse código acessa a chave de forma segura, utilizando o sistema de variáveis de ambiente do Vite. A chave não aparece diretamente no código-fonte, o que reduz o risco de exposição — embora, em ambientes de produção, o ideal seja sempre utilizar um backend para proteger dados sensíveis.



Construindo com IA: Etapas de um Projeto Real

A Relação entre IA e o Desenvolvedor JavaScript

A IA pode gerar estruturas e acelerar o desenvolvimento, mas **a responsabilidade pela segurança, organização e funcionamento do projeto continua sendo do desenvolvedor**. Entender como configurar variáveis de ambiente, proteger chaves e estruturar o código são habilidades fundamentais para garantir que a aplicação seja segura e escalável.

A integração com a OpenAI é um ótimo exemplo de como o desenvolvedor precisa atuar como ponte entre automação e lógica real — ajustando, validando e protegendo cada etapa do processo.

Etapa 5 — Ajustando o Backend e Simplificando o Projeto

Após a geração automática do projeto pela IA, foi identificado que a estrutura incluía uma camada de backend utilizando **Supabase** e **Lovable Cloud**. Embora funcional, essa abordagem pode ser complexa para projetos iniciais ou protótipos rápidos. Por isso, optou-se por simplificar a arquitetura.

| Ajuste | Descrição |
|--------------------------|--|
| Remoção do Supabase | Eliminação da camada de backend para reduzir complexidade |
| Manutenção da variável | A chave da OpenAI continua protegida via <code>VITE_OPENAI_API_KEY</code> |
| Requisições no front-end | Comunicação direta com a API, sem intermediários |



Construindo com IA: Etapas de um Projeto Real

Decisão Técnica

A estrutura de backend foi removida e substituída por uma abordagem mais direta:

- A **chave da API da OpenAI** foi mantida como **variável de ambiente**, garantindo segurança e organização
- As **requisições à API** passaram a ser feitas **diretamente pelo front-end**, utilizando JavaScript/TypeScript com suporte do Vite

Essa simplificação torna o projeto mais leve, fácil de entender e ideal para fins educacionais ou testes rápidos.

IA como Assistente, Desenvolvedor como Decisor

Essa etapa reforça um ponto essencial: **a IA é uma ferramenta de apoio, não uma autoridade final**. Ela pode sugerir estruturas, gerar código e acelerar processos, mas **cabe ao desenvolvedor avaliar, adaptar e decidir** o que faz sentido para o projeto.

Ao simplificar o backend, o desenvolvedor assume o controle da arquitetura, tornando o projeto mais acessível e funcional para o contexto desejado. Essa autonomia é o que transforma a IA em uma parceira criativa — e não em uma chefe.



Construindo com IA: Etapas de um Projeto Real

⚡ Etapa 7 — Integração com GitHub

Após validar o funcionamento do projeto gerado pela IA, é hora de garantir que ele esteja versionado e acessível para futuras edições. A integração com o GitHub permite controle de versão, colaboração em equipe e edição local com ferramentas como o VS Code.

🔗 Fluxo de Integração

A plataforma Lovable oferece suporte direto para sincronização com o GitHub, facilitando o processo de publicação e edição do projeto. O fluxo básico envolve:

- **Conectar a conta do GitHub** dentro do painel do Lovable
- **Criar um novo repositório** para armazenar o projeto gerado
- **Clonar o repositório no VS Code**, permitindo edição local com mais controle
- **Trabalhar localmente**, utilizando comandos como **push**, **pull**, **commit** e **branch** para gerenciar o código

Essa etapa transforma o projeto em algo profissional, versionado e pronto para evoluir com segurança e organização.



Construindo com IA: Etapas de um Projeto Real

Benefícios da Integração

- Permite **salvar o histórico de alterações** do projeto
- Facilita a **colaboração com outros desenvolvedores**
- Garante que o projeto esteja **acessível de qualquer lugar**
- Abre espaço para **melhor organização e escalabilidade**
-

Git + VS Code: Ferramentas do Desenvolvedor Moderno

A integração com GitHub e o uso do VS Code são práticas essenciais para qualquer desenvolvedor que deseja manter controle sobre seu código, aplicar boas práticas de versionamento e trabalhar com eficiência. Mesmo que o projeto tenha sido gerado por IA, é o desenvolvedor quem garante sua evolução e estabilidade.



Construindo com IA: Etapas de um Projeto Real

Reflexão Final — Etapa 9

Ao longo do processo de criação do chatbot com apoio da inteligência artificial, ficou evidente que o uso da IA exige conhecimento técnico, senso crítico e capacidade de adaptação. Mesmo com a geração automática de código e layout, foi necessário compreender variáveis de ambiente, lidar com erros de execução e realizar integrações com APIs externas.

Lição Principal

A inteligência artificial é uma **ferramenta poderosa**, mas **não substitui o papel do programador**. Ela potencializa a produtividade, acelera tarefas e oferece sugestões, mas **é o desenvolvedor quem interpreta, corrige e transforma o projeto em algo funcional e seguro**.

Saber **o que pedir, como interpretar o resultado e corrigir o que vier errado** são habilidades fundamentais para tirar o máximo proveito da IA no desenvolvimento de software.

**A IA é uma ferramenta, não uma muleta.
Use-a para criar, aprender e evoluir.**