**Git**

Funciona **no seu computador** para **rastrear as mudanças** que você faz nos prompts.

**GitHub**

Funciona **online** para **compartilhar o repositório da Biblioteca de Prompts** com o time.

**Repositório (Repo)**

Um repositório é onde todos os prompts serão armazenados, organizados em pastas. Ele mantém o histórico de mudanças e permite que várias pessoas trabalhem de forma coletiva.

**Exemplo:**

O repositório **Biblioteca de Prompts** contém todos os prompts organizados por área.

**Branch**

Uma branch é uma “cópia” do repositório onde você pode fazer alterações sem afetar a branch principal (main). Depois de finalizar as alterações, você pode mesclar a branch de volta na principal.

**Exemplo:**

Criar a branch marketing/email-nutricao-lancamento para trabalhar em um novo prompt para criação de emails.

**Commit**

Um commit é feito para "criar” ou "salvar” mudanças feitas em um arquivo. Ele inclui uma mensagem que explica o que foi criado ou alterado.

**Exemplo:**

feat: Adicionando prompt de nutrição por email

**Pull Request (PR)**

Um Pull Request é uma solicitação para que suas alterações sejam revisadas e aprovadas antes de serem integradas na branch principal (main).

**Exemplo:**

Abrir um PR para que alguém do time revise o novo prompt antes de integrá-lo ao repositório.

**Merge (não se preocupe com isso)**

O merge é o processo de unir as alterações feitas em uma branch secundária (por exemplo, marketing/nutricao-v1) na branch principal (main).

**Clone**

O clone é uma cópia do repositório GitHub que você baixa para o seu computador. Isso permite que você trabalhe localmente nos arquivos.

**Push**

O push é o comando que você usa para enviar as mudanças feitas no seu ambiente local para o repositório no GitHub.

**Comando:**

git push origin nome-da-branch

**Pull**

O pull é o comando que você usa para **atualizar seu repositório local** com as mudanças mais recentes feitas no GitHub.

**Comando:**

git pull

### **🧩 Boas Práticas de Versionamento**

**Histórico de Versões**

Mantenha um histórico claro das alterações feitas em cada arquivo, usando o padrão:

• **v1:** Primeira versão do prompt.

• **v1.1, v1.1.1 e etc :** Pequenos ajustes ou melhorias.

• **v2:** Mudanças grandes na estrutura do prompt.

**Exemplo:**

# **Histórico de Versões**

* v1 (01-01-2025): Criação do prompt.
* v1.1 (05-01-2025): Ajuste no tom de comunicação.
* v2 (05-02-2025): Inclusão de uma chamada para ferramenta.

**Mensagens de Commit**

Use mensagens claras e descritivas nos commits.

**🔄 Comandos Básicos do Git**

| **Comando** | **Descrição** |
| --- | --- |
| git clone | Clona o repositório para o seu computador. |
| git add . | Adiciona todas as mudanças para commit. |
| git commit -m "msg" | Cria um commit com a mensagem especificada. |
| git push | Envia as mudanças para o GitHub. |
| git pull | Puxa as mudanças mais recentes do GitHub. |