
PROJETO ATHUS

Padrões de Commits(Commit Pattern)

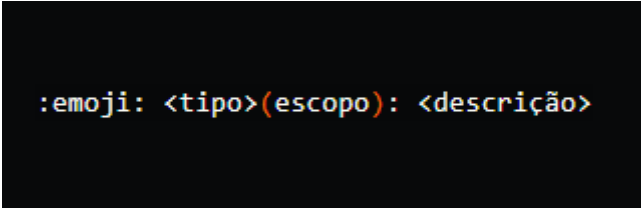
O que é o Conventional Commits ?

O Conventional Commits é uma convenção simples de mensagens de commit, que segue um conjunto de regras e que ajuda os projetos a terem um histórico de commit explícito e bem estruturado.

O **Conventional Commits** deixara a linha do tempo de um projeto mais estruturada e fácil de entender. Quando documentados utilizando o **Conventional Commits** nos mostram quem alterou, quando, em qual contexto e qual tipo de alteração foi feita.

Como utilizar ?

A mensagem do commit deve ser estruturada da seguinte forma:

A diagrama de um commit estruturado, mostrando a sintaxe: :emoji: <tipo>(escopo): <descrição>. O texto está em uma caixa preta com fundo branco, onde 'emoji' e 'escopo' são destacados em cor laranja.

```
:emoji: <tipo>(escopo): <descrição>
```





















Os commits terão um emoji para dar uma identificação visual a mensagem, um tipo, um escopo para dar contexto, e uma descrição onde poderá colocar os detalhes do commit.

Tipos de Commits

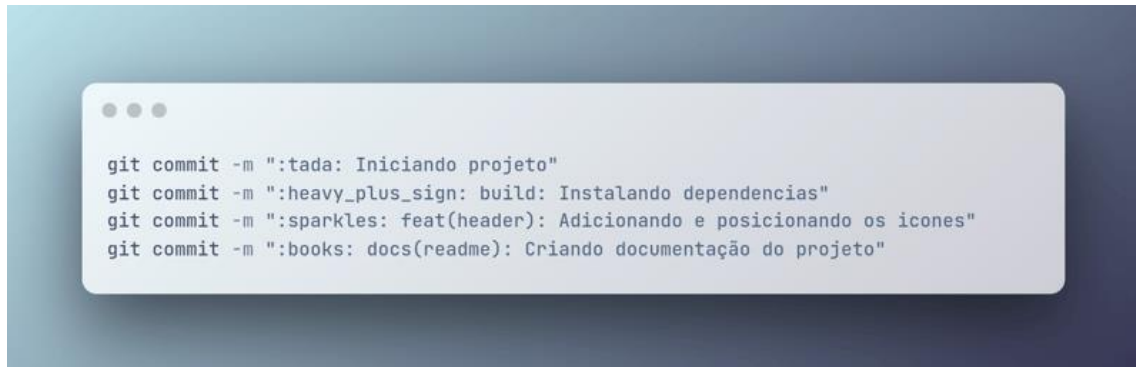
O commit possui os elementos estruturais abaixo (tipos), que informam a intenção do seu commit ao utilizador(a) de seu código.

- **fix** - Commits do tipo fix indicam que seu trecho de código commitado está solucionando um problema (bug fix), (se relaciona com o PATCH do versionamento semântico).
- **feat** - Commits do tipo feat indicam que seu trecho de código está incluindo um novo recurso (se relaciona com o MINOR do versionamento semântico).
- **docs** - Commits do tipo docs indicam que houveram mudanças na documentação, como por exemplo no Readme do seu repositório. (Não inclui alterações em código).
- **style** - Commits do tipo style indicam que houveram alterações referentes a formatações de código, semicolons, trailing spaces, lint... (Não inclui alterações em código).
- **refactor** - Commits do tipo refactor referem-se a mudanças devido a refatorações que não alterem sua funcionalidade, como por exemplo, uma alteração no formato como é processada determinada parte da tela, mas que manteve a mesma funcionalidade, ou melhorias de performance devido a um code review.
- **build** - Commits do tipo build são utilizados quando são realizadas modificações em arquivos de build e dependências.
- **test** - Commits do tipo test são utilizados quando são realizadas alterações em testes, seja criando, alterando ou excluindo testes unitários. (Não inclui alterações em código)
- **chore** - Commits do tipo chore indicam atualizações de tarefas de build, configurações de administrador, pacotes... como por exemplo adicionar um pacote no gitignore. (Não inclui alterações em código)

Emojis

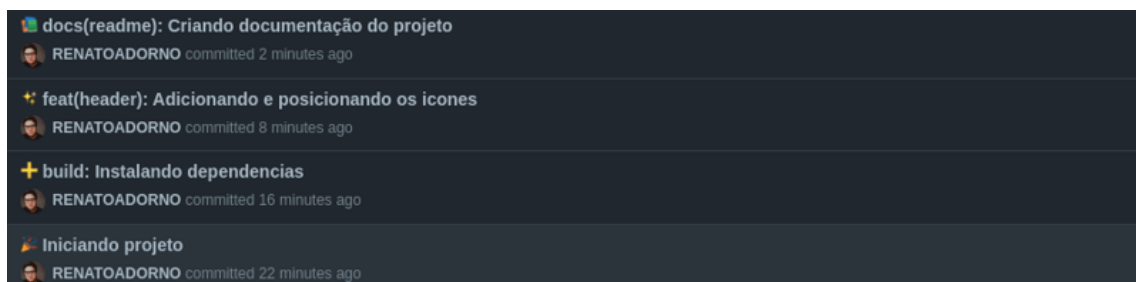
Tipo de commit	Emojis
Commit inicial	 :tada:
Tag de versão	 :bookmark:
Novo recurso	 :sparkles:
Lista de ideias (tasks)	 :soon:
Bugfix	 :bug:
Documentação	 :books:
Testes	 :test_tube:
Adicionando um teste	 :white_check_mark:
Teste de aprovação	 :heavy_check_mark:
Acessibilidade	 :wheelchair:
Texto	 :pencil:
Package.json em JS	 :package:
Em progresso	 :construction:
Arquivos de configuração	 :wrench:
Removendo uma dependência	 :heavy_minus_sign:
Adicionando uma dependência	 :heavy_plus_sign:
Revertendo mudanças	 :boom:
Alterações de revisão de código	 :ok_hand:
Refatoração	 :recycle:
Mover/Renomear	 :truck:

Exemplos de utilização:



```
git commit -m ":tada: Iniciando projeto"
git commit -m ":heavy_plus_sign: build: Instalando dependencias"
git commit -m ":sparkles: feat(header): Adicionando e posicionando os icones"
git commit -m ":books: docs(readme): Criando documentação do projeto"
```

Seguindo esses 4 commits acima teremos seguinte histórico:



Considerações Finais

Esta documentação estabelece os padrões a serem seguidos de acordo com a convenção de commits. Pedimos que todos sigam esses padrões para garantir um ambiente mais organizado e compreensível para todos os membros da equipe.

Se preferir, você pode acessar mais detalhes diretamente no site abaixo:

<https://dev.to/renatoadorno/padroes-de-commits-commit-patterns-41co>