

Versionamento Semantico

por Matheus Anzzulin

Pontos Chave

- O que é
- Utilização

O Que é

O Versionamento Semântico, ou SemVer, é um sistema de versionamento usado no desenvolvimento de software para transmitir informações significativas sobre as mudanças em uma base de código. Segue um sistema de numeração de três partes:
major.minor.patch

Conceito principal

MAJOR.MINOR.PATCH

Major: incrementada quando alterações incompatíveis são introduzidas, como alterações interrompidas na API ou melhorias importantes de recursos.

Minor: incrementada quando novos recursos são adicionados de maneira compatível com versões anteriores, indicando melhorias ou extensões nas funcionalidades existentes.

Patch: incrementado para correções de bugs compatíveis com versões anteriores ou pequenas melhorias que não afetam a funcionalidade existente.

Utilização

O objetivo principal é permitir que os desenvolvedores entendam rapidamente o impacto das alterações feitas em uma biblioteca ou pacote de software.

exemplo: impacto que tem no projeto mudar a versão de uma lib de 4.3.4 para 5.0.0

Exemplo

Dany quer atualizar a lib de componentes que o time usa no projeto de 2.4.1 para 3.4.1 para poder utilizar um componente mais descolado que tem somente nessa versão, é seguro fazer essa mudança?

Exemplo

Se quer utilizar uma funcionalidade que tem apenas na versão 1.5.0 da lib e o projeto está utilizando a versão 1.4.3, é seguro mudar a versão?

Exemplo

Carolina mudar o payload que a api está retornando na rota Get /cards pois precisa de dados diferentes para montar o novo layout do site. Atualmente a api está na versão 1.4.1, qual deve ser a nova versão?

Labels Adicionais: Pré-lançamento e Metadata

As labels de pré-lançamento são anexados ao número da versão para indicar que o lançamento ainda não está estável ou finalizado. Eles são frequentemente usados para versões que ainda estão em desenvolvimento, teste ou refinamento antes de serem lançadas oficialmente para os usuários. Os rótulos de pré-lançamento ajudam a comunicar aos desenvolvedores e usuários que o software está em um estado de transição e pode não ser adequado para uso em produção.

Labels Adicionais: Exemplos

Alpha: Alpha releases are early versions of the software that are still undergoing significant changes and may have incomplete features or known issues. They are typically released to a limited audience for testing and feedback.

Beta: Beta releases are more stable than alpha releases but still may contain bugs or unfinished features. They are usually released to a wider audience for broader testing and feedback.

Labels Adicionais: Exemplos

Release Candidate (RC): Release candidates are versions that are considered almost ready for the final release. They undergo thorough testing to identify any remaining issues before being promoted to a stable release.

Snapshot: Snapshot releases are often used in projects with continuous integration or continuous delivery (CI/CD) pipelines. They represent the state of the software at a specific point in time and may be automatically generated from the latest code changes.

Labels Adicionais: Exemplos

As Labels de pré-lançamento são anexadas ao número da versão usando um hífen seguido pela label. Eles permitem que os desenvolvedores transmitam o status do lançamento e gerenciem as expectativas em relação à sua estabilidade e adequação para diferentes casos de uso.

- 1.0.0-alpha.1
- 2.1.0-beta.3
- 2.1.0-RC.1
- 2.1.0-RC.2

Labels Adicionais: Metadata

Os metadados fornecem informações adicionais sobre uma versão do software, mas não afetam a precedência da versão. Geralmente é usado para incluir informações relacionadas à compilação, como hash de commit, data de compilação ou ambiente de compilação, que podem ser úteis para depuração, rastreamento de alterações e garantia de reprodutibilidade.

Labels Adicionais: Metadata

Os metadados são anexados ao número da versão usando um sinal de mais seguido pelos metadados. Ao contrário das labels de pré-lançamento, os metadados não afetam a forma como as versões são comparadas ou classificadas; é puramente informativo.

- 1.2.3+20220320
- 2.0.0+sha.abcdef
- 1.2.3+solicitations-api-validation

Exemplo

Matheus está com um bug na lib de validação de cnpj. A versão que o projeto utiliza é a 1.1.0-stable, e a lib tem as seguintes versões disponíveis:

- 1.1.0-alpha+fix_cnpj_validation
- 1.2.0
- 1.1.1-RC.1
- 2.0.0

Vale apenas testar alguma dessas versões para ver se o erro foi corrigido?