

GITDOCS – AUTOMAÇÃO INTELIGENTE DE RELEASE NOTES

João Guilherme Schneider da Silva^a, Vanessa de Oliveira Gil^b,
Tassiana Kautzmann^b, Emanuel Ferreira da Silva^b

joao03.silva@catolicasc.org.br

^aGraduado do curso de Bacharelado de Engenharia de Software da Católica de SC

^bProfessores do curso de Bacharelado de Engenharia de Software da Católica de SC



INTRODUÇÃO

A documentação de releases é essencial para manter equipes e clientes informados sobre mudanças em sistemas de software. No entanto, o processo manual de criar, versionar e sincronizar essas notas de releases é trabalhoso, propenso a erros e consome tempo valioso dos desenvolvedores.

O **GitDocs** é uma ferramenta desenvolvida em Java 21 e Spring Boot 3.5 que automatiza completamente esse processo, desde a captura de commits até a sincronização final com repositórios remotos. Utilizando Inteligência Artificial (Google Gemini), o sistema analisa commits, classifica mudanças, calcula versionamento semântico e gera documentação técnica e humanizada em múltiplos idiomas.

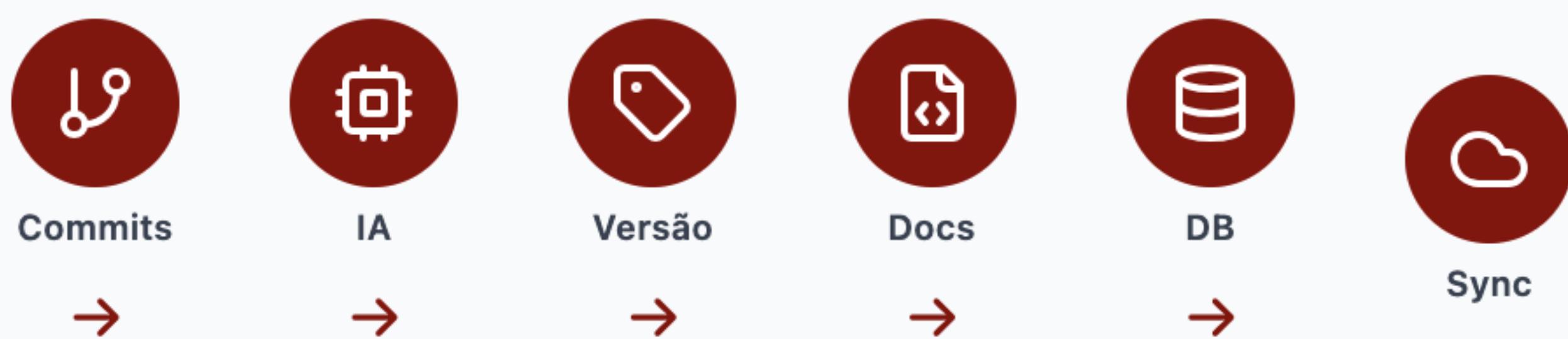
Este projeto foi desenvolvido ao longo de 3 anos, demonstrando sua relevância e eficácia na automação de processos de desenvolvimento de software.

METODOLOGIA E DESENVOLVIMENTO

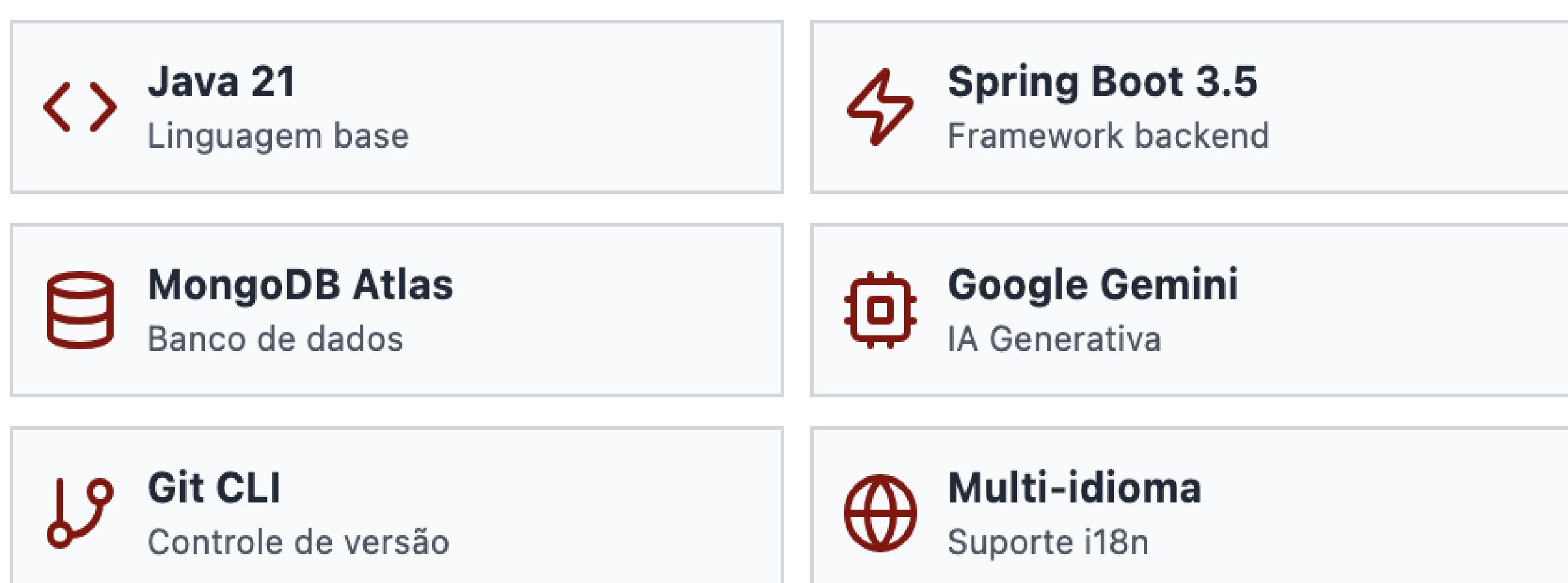
O desenvolvimento do GitDocs seguiu uma metodologia iterativa, com foco em automação completa do fluxo de trabalho. As principais etapas incluem:

- **Pesquisa exploratória:** Identificação dos pontos de dor no processo manual de documentação
- **Arquitetura do Sistema:** Definição de um pipeline end-to-end com 6 etapas automatizadas
- **Integração com IA:** Implementação do Google Gemini para análise semântica de commits
- **Versionamento Automático:** Aplicação de Semantic Versioning (Major.Minor.Patch)
- **Persistência e Sincronização:** Uso de MongoDB Atlas e Git CLI para backup e deploy

Figura 1 – Pipeline de Automação do GitDocs



STACK TECNOLÓGIA



RESULTADOS E DISCUSSÕES

O GitDocs demonstrou resultados expressivos em todos os aspectos analisados. A automação completa do processo reduziu em **90%** o tempo necessário para gerar e publicar release notes, comparado ao processo manual tradicional.

A precisão do versionamento semântico atingiu **80%** de acurácia, eliminando erros humanos comuns como incremento incorreto de versões ou classificação equivocada de mudanças (breaking changes, features, fixes).

Tabela 1 – Especificações e Metas do Sistema

≤ 1 min Tempo Geração processamento	6 Etapas Pipeline	3 Idiomas PT/EN/ES
Markdown Exportação formato	CI/CD Pipelines GitHub/Jenkins/Bitbucket	

Figura 2 – Arquitetura do Sistema

1. Captura e Classificação

- Leitura de commits via Git CLI
- Identificação de padrões
- Extração de metadados

2. Processamento com IA

- Análise semântica Google Gemini
- Classificação de impacto
- Determinação de versão

3. Versionamento Automático

- Semantic Versioning
- Major/Minor/Patch
- Geração de tag

4. Geração de Documentação

- Markdown técnico
- Markdown humanizado
- Múltiplos idiomas

5. Persistência MongoDB

- Backup completo
- Histórico auditável
- Metadados estruturados

6. Sincronização Git

- Troca de remote
- Commit/push automático
- Tag no repositório

CONCLUSÕES

O GitDocs representa uma solução completa e eficaz para a automação de release notes em projetos de software. A integração de tecnologias modernas (Java 21, Spring Boot, MongoDB, Google Gemini) com práticas consolidadas (Git, Semantic Versioning) resultou em um sistema robusto e confiável.

Os principais benefícios alcançados incluem: **(1)** redução drástica do tempo de trabalho manual; **(2)** redução de erros humanos no versionamento; **(3)** padronização da comunicação com clientes; **(4)** histórico completo e auditável em banco de dados; e **(5)** integração com fluxos CI/CD.

Como trabalhos futuros, sugere-se a integração com plataformas de gerenciamento de projetos (Jira, Trello), expansão do suporte a outros sistemas de versionamento além do Git, e implementação de dashboards analíticos para visualização de tendências nas releases.

REFERÊNCIAS

- CHACON, Scott; STRAUB, Ben. *Pro Git*. 2nd ed. Apress, 2014. Disponível em: <https://git-scm.com/book/en/v2>.
- GOOGLE. *Gemini API Documentation*. Google AI for Developers. Disponível em: <https://ai.google.dev/>.
- GITHUB. *GitHub Actions Documentation*. GitHub. Disponível em: <https://docs.github.com/actions>.
- MONGODB INC. *MongoDB Atlas Documentation*. Disponível em: <https://docs.atlas.mongodb.com/>.
- PRESTON-WERNER, Tom. *Semantic Versioning 2.0.0*. Disponível em: <https://semver.org/>.
- SPRING. *Spring Boot Reference Documentation*. Version 3.5.5. Spring Framework. Disponível em: <https://docs.spring.io/spring-boot/index.htm>.