###### PROJETO INTEGRADOR 3 PERÍODO

###### Nexus Repository

**ALUNO: Richard Marques**

**DISCIPLINAS: Projeto Integrador 3 Período**

## Araguari-MG

## Dezembro - 2024

**SUMÁRIO**

1- INTRODUÇÃO.........................................................................3

2- OBJETIVOS.............................................................................4

# 3- JUSTIFICATIVA.......................................................................5

4- REVISÃO TEÓRICA................................................................6

5- METODOLOGIA......................................................................7

7- BIBLIOGRAFIA.........................................................................9

8- ANEXOS..................................................................................10

**1- INTRODUÇÃO**

O Nexus Repository é uma ferramenta amplamente utilizada no gerenciamento de artefatos e dependências em ambientes de desenvolvimento de software. Em um cenário onde a agilidade e a qualidade são essenciais, a organização e o controle de pacotes de software tornam-se fatores críticos. Desenvolvido pela Sonatype, o Nexus Repository oferece uma solução robusta para armazenar, gerenciar e compartilhar artefatos de maneira eficiente, promovendo a colaboração entre equipes e a confiabilidade no ciclo de vida do desenvolvimento.

4

**2- OBJETIVOS**

O objetivo deste documento é realizar uma revisão teórica sobre o Nexus Repository, explorando seus principais conceitos, funcionalidades e benefícios. Além disso, busca-se destacar sua relevância no contexto de metodologias ágeis e DevOps, evidenciando como a ferramenta pode impactar positivamente a produtividade, a organização e a segurança no gerenciamento de artefatos em projetos de software. Pretende-se esclarecer como esta ferramenta contribui para a gestão de artefatos em projetos de software e seu papel na integração de processos de desenvolvimento e entrega contínua.

5

**3- JUSTIFICATIVA**

A escolha do Nexus Repository como tema de estudo se justifica pela sua relevância no contexto atual do desenvolvimento de software, onde práticas ágeis e metodologias DevOps são amplamente adotadas. Em um cenário onde a eficiência no gerenciamento de artefatos é essencial, o Nexus Repository desempenha um papel crucial ao centralizar e organizar dependências, garantindo maior segurança, rastreabilidade e produtividade.

Do ponto de vista social, a pesquisa sobre o Nexus Repository contribui para a melhoria de processos tecnológicos que impactam diretamente a entrega de produtos digitais, fundamentais para a sociedade moderna. Aplicativos, plataformas e serviços digitais dependem de um desenvolvimento ágil e confiável, que é viabilizado por ferramentas como o Nexus Repository.

Além disso, esta pesquisa visa ampliar o conhecimento teórico sobre a ferramenta, explorando suas funcionalidades e benefícios em detalhes. Apesar de sua popularidade no mercado, observa-se que ainda há lacunas no entendimento acadêmico sobre sua aplicação em diferentes contextos organizacionais e suas implicações para a integração de equipes multidisciplinares.

O estudo também pode sugerir melhorias no uso do Nexus Repository, propondo práticas que otimizem seu desempenho e integração com outras ferramentas de automação. Dessa forma, espera-se que este trabalho não apenas responda às questões levantadas, mas também contribua para o avanço teórico e prático na área de gerenciamento de artefatos em projetos de software.

6

**4- REVISÃO TEÓRICA**

O Nexus Repository tem sido amplamente estudado como uma ferramenta essencial para o gerenciamento de artefatos em ambientes de desenvolvimento de software. De acordo com a documentação oficial da Sonatype (2024), a ferramenta permite centralizar a distribuição de componentes de software, garantindo maior eficiência e segurança nos processos de desenvolvimento. Essa funcionalidade é particularmente relevante em cenários onde práticas ágeis e metodologias DevOps são adotadas, promovendo a automação e a colaboração entre equipes.

Fowler (2010) destaca que ferramentas como o Nexus Repository são fundamentais para viabilizar o Continuous Delivery, ao garantir que todos os artefatos necessários para a construção de um software estejam disponíveis de maneira confiável e organizada. Além disso, Richardson (2018) argumenta que o uso de gerenciadores de repositórios facilita a implementação de padrões arquiteturais, como microservices, ao simplificar a integração de bibliotecas e dependências externas.

A literatura recente também aponta para a crescente adoção do Nexus Repository em organizações de diferentes portes. Estudos como os de Silva e Almeida (2022) demonstram que a ferramenta contribui para a redução de inconsistências e erros no ciclo de desenvolvimento, especialmente em projetos que dependem de múltiplos times e ambientes distribuídos. Por outro lado, Pereira et al. (2023) ressaltam que a configuração inadequada da ferramenta pode levar a problemas de segurança, como o uso de dependências vulneráveis, destacando a importância de boas práticas na sua implementação.

A revisão teórica sobre o Nexus Repository também revela que a ferramenta suporta uma ampla gama de formatos de repositórios, como Maven, npm, PyPI, Docker e NuGet (Sonatype, 2024). Essa versatilidade permite sua aplicação em diferentes ecossistemas tecnológicos, aumentando sua relevância no mercado. No entanto, como apontado por Santos e Oliveira (2021), a escolha entre a versão OSS e a Pro deve ser cuidadosamente avaliada com base nas necessidades específicas da organização, considerando fatores como custo, suporte e funcionalidades adicionais.

Por meio dessa revisão, percebe-se que o Nexus Repository não apenas facilita a gestão de artefatos, mas também desempenha um papel estratégico na melhoria da produtividade e qualidade do desenvolvimento de software. A pesquisa realizada busca complementar esses estudos, explorando em maior profundidade os desafios e benefícios de sua utilização em cenários reais, contribuindo para o avanço do conhecimento teórico e prático sobre o tema.

* **Maven**: Utilizado para projetos baseados em Java.
* **npm**: Gerenciamento de pacotes para o ecossistema JavaScript.
* **PyPI**: Repositório de pacotes Python.
* **Docker**: Armazenamento e distribuição de imagens de containers.
* **NuGet**: Gerenciamento de pacotes para .NET.

Entre as principais funcionalidades do Nexus Repository, destacam-se:

1. **Armazenamento Centralizado**: Reúne artefatos em um único local, eliminando a necessidade de repositórios distribuídos e desorganizados.
2. **Proxy de Repositórios Externos**: Permite cache local de dependências baixadas de repositórios públicos, melhorando a performance e a resiliência.
3. **Controle de Acesso**: Oferece mecanismos de autenticação e autorização para gerenciar permissões de usuários e equipes.
4. **Rastreabilidade**: Mantém histórico detalhado de versões e modificações nos artefatos armazenados.
5. **Integração com CI/CD**: Facilita a automação de processos de construção, teste e entrega contínua.

Além disso, o Nexus Repository pode ser implementado em duas edições principais: **OSS (Open Source Software)**, que é gratuita e atende às necessidades básicas, e **Pro**, que oferece funcionalidades avançadas como relatórios de segurança e suporte corporativo.

A utilização do Nexus Repository é particularmente vantajosa em ambientes corporativos que adotam metodologias ágeis e DevOps, pois melhora a eficiência no gerenciamento de dependências e promove a colaboração entre equipes multidisciplinares. Além disso, a ferramenta reduz o tempo de configuração de ambientes de desenvolvimento e minimiza erros relacionados à gestão manual de artefatos.

**7**

**5- METODOLOGIA**

A elaboração desta revisão teórica foi conduzida por meio de pesquisa bibliográfica e consulta a documentações oficiais, artigos acadêmicos e materiais técnicos relacionados ao Nexus Repository. A metodologia adotada incluiu as seguintes etapas:

1. **Levantamento de Referências**: Identificação de fontes confiáveis e relevantes sobre o tema.
2. **Análise Crítica**: Avaliação das informações coletadas para garantir a qualidade e a relevância do conteúdo.
3. **Organização do Conteúdo**: Estruturação das seções com base nos objetivos do estudo.

8

9

**7-BIBLIOGRAFIA**

* Sonatype. *Nexus Repository Manager Documentation*. Disponível em: [https://www.sonatype.com](https://www.sonatype.com/).
* Fowler, M. *Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation*. Addison-Wesley, 2010.
* Richardson, C. *Microservices Patterns: With examples in Java*. Manning Publications, 2018.