

Anexo B – Requisitos do Objeto

Natureza	Grupo	ID	Requisitos do Objeto	Peso (1 a 5)	Atendimento dos Requisitos {Sim (5); Parcial (2); Não (0)}	Pontuação {Peso * Atendimento}	Observação
Requisitos Funcionais	Consulta por Linguagem Declarativa de Grafos	1	A plataforma deve oferecer linguagem declarativa própria para consultas complexas de grafos, com sintaxe intuitiva e otimizada para expressar padrões de relacionamento complexos entre nós.	5		0	
		2	A linguagem deve ser compatível com padrões amplamente utilizados no mercado, como a linguagem Cypher, garantindo ampla portabilidade e adoção.	3		0	
		3	A linguagem deve permitir filtragem, ordenação, agregação e uso de expressões em nós, arestas e propriedades.	5		0	
		4	A linguagem de consulta deve permitir expressar padrões com profundidade variável, incluindo padrões de recursividade, ciclos, caminhos múltiplos e caminhos alternativos com controle sobre cardinalidade e direção.	5		0	
		5	A plataforma deve oferecer compatibilidade com o padrão ISO/IEC GQL (Graph Query Language), garantindo portabilidade e conformidade com as normas internacionais de consulta a grafos.	1		0	
		6	Deve assegurar evolutividade da sintaxe, suporte a funções avançadas de pattern matching e interoperabilidade com linguagens existentes.	2		0	
	Modelagem de Grafos Complexos	7	A plataforma deverá utilizar um modelo de grafo nativo, ou seja, deve implementar internamente estruturas de armazenamento e mecanismos de processamento otimizados especificamente para grafos, representando nós e relacionamentos como elementos primários no nível de disco e memória.	5		0	
		8	A plataforma deve permitir a representação de entidades e relacionamentos com múltiplos tipos de nós e arestas, com propriedades personalizadas.	5		0	
		9	Deve suportar grafos direcionados, ponderados e com alta densidade de conexões.	5		0	
		10	Deve permitir armazenamento de atributos de tipos complexos (listas, mapas etc.) em vértices e arestas, e permitir seu uso em consultas e aplicações algorítmicas.	3		0	
		11	Deve permitir a utilização de Strings como chaves internas de identificação dos vértices.	3		0	
		12	A plataforma deve implementar Grafo de Propriedades rotuladas (LPG - Labeled Property Graph).	4		0	
	Exploração Visual Interativa	13	A plataforma deverá oferecer conectores nativos para integração com ferramentas de Business Intelligence (BI) amplamente utilizadas no mercado, tais como Power BI, MicroStrategy, entre outras, possibilitando o consumo direto de dados do grafo para análises e visualizações externas.	5		0	
		14	A plataforma deve oferecer ferramenta visual nativa para exploração e navegação interativa de grafos, com possibilidade de aplicação de filtros, estilos, agrupamentos, seleção de polígonos, aplicação de algoritmos e exportação de visões.	5		0	
		15	A plataforma deverá dispor de ferramenta nativa para criação e gerenciamento de dashboards interativos, permitindo a composição de visões analíticas customizadas a partir de dados e relacionamentos do grafo.	5		0	
		16	A plataforma deverá incluir ferramenta visual nativa baseada em paradigmas no-code / low-code, que possibilite a exploração de padrões, a construção de narrativas visuais e ofereça camadas de abstração como:	5		0	
		17	a) Perspectivas: mecanismos para ocultar ou exibir seletivamente nós e relacionamentos com base em critérios definidos (ex.: tipo, classificação, domínio);	5		0	
		18	b) Cenários: visões predefinidas compostas por conjuntos de filtros, estilos, layouts e configurações visuais aplicáveis a uma determinada perspectiva analítica; Tais recursos devem facilitar a exploração e o entendimento dos dados por usuários com perfil não técnico, promovendo acessibilidade e autonomia na exploração e análise de grafos.	5		0	
	Biblioteca Nativa de Algoritmos de Grafos	19	Deve conter biblioteca nativa com algoritmos de análise de grafos já implementados, como: PageRank, centralidade, detecção de comunidades, caminhos mais curtos, predição heurística, similaridade, Louvain, Betweenness Centrality e algoritmos de embeddings (Node2Vec, HashGNN e outros).	4		0	
		20	Os algoritmos devem ser aplicáveis sobre subgrafos e resultados de consultas.	3		0	
		21	A execução dos algoritmos deve ter exemplos e SDKs nativos em Python e Java.	3		0	
		22	A plataforma deve suportar criação de novos relacionamentos a partir dos resultados dos algoritmos (ex.: clusterização), com possibilidade de persistência desses relacionamentos.	3		0	
		23	Suportar execução distribuída de algoritmos iterativos (modelo BSP/Pregel-like ou equivalente), com SDKs e exemplos práticos em Python e/ou Java.	3		0	
		24	Os algoritmos de grafos da plataforma devem permitir extensibilidade/customização (ex.: UDFs, plugins, procedures).	4		0	
	Inferência e Raciocínio Semântico	25	A plataforma deve permitir a construção de grafos de conhecimento com esquemas e dados integrados, possibilitando inferências e queries semânticas com base em hierarquias, ontologias e relações lógicas.	4		0	
	Integração com IA e Machine Learning	26	A plataforma deve possibilitar integração com pipelines de IA e ML para análise preditiva, enriquecimento de dados e assistentes inteligentes, permitindo também o uso de estrutura de grafos como fonte de features para modelos de machine learning supervisionado e não supervisionado.	5		0	
		27	A plataforma deve permitir a criação, reaproveitamento e exportação de embeddings derivados do grafo, com possibilidade de persistência direta no banco e integração posterior com modelos de ML e RAGs (Retrieval-Augmented Generation).	5		0	
	Transações ACID e Consistência	28	Deve garantir operações transacionais com propriedades ACID (Atomicidade, Consistência, Isolamento e Durabilidade).	5		0	
	Versionamento e Time Travel	29	Deve permitir a rastreabilidade histórica das modificações no grafo, com possibilidade de implementação de versionamento de nós/arestas e consultas retroativas (time travel) a partir de estrutura nativa do grafo, sem necessidade de reconstrução manual de snapshots ou logs.	3		0	
	Interface de Administração	30	Deve prover interface de administração para gerenciamento de usuários, dados, execução de jobs e monitoramento.	5		0	
	Suporte ao Desenvolvimento via APIs	31	Deve disponibilizar APIs REST e/ou drivers oficiais para diversas linguagens (ex.: Python, Java, JS) com foco em baixa latência e alto throughput.	4		0	
		32	A plataforma deve disponibilizar interfaces para integração com outras aplicações, incluindo drivers oficiais para linguagens como Python, Java, JavaScript, .NET e Go.	5		0	
		33	Deve disponibilizar API GraphQL para exposição de dados do grafo em formato moderno e consumível por aplicações web e móveis.	5		0	
		34	Deve disponibilizar ferramentas para auxiliar a migração ou integração com aplicações legadas que utilizam Neo4j embedded (community).	3		0	
		35	A plataforma deve disponibilizar APIs REST e SDKs para integração com aplicações externas, suportando linguagens como Python, Java e Go.	5		0	
		36	Deve possuir documentação completa e exemplos de uso para facilitar o desenvolvimento e automação de processos.	4		0	
	Integração de Dados	37	Oferecer integração com mecanismos de processamento distribuído e ingestão em lote/streaming (ex.: Spark, Flink, Beam ou equivalente), via conectores, APIs ou jobs.	5		0	
		38	Deve permitir integração com pipelines de ETL e streaming, garantindo consistência e performance em cenários de alta demanda.	5		0	
		39	A plataforma deve suportar leitura e gravação nativas no formato Apache Parquet, disponibilizando recursos de compressão, particionamento e leitura seletiva de colunas, visando otimização de desempenho e economia de armazenamento.	4		0	
		40	A plataforma deve integrar esse suporte em seus mecanismos de importação e exportação de dados.	4		0	
		41	A plataforma deve permitir integração com Active Directory para autenticação de usuários.	5		0	
		42	Deve permitir autenticação de aplicações via protocolo OAuth2 utilizando um provedor OpenID.	5		0	

	Autenticação Corporativa	43	Deve permitir integração com sistemas externos de LOG, como Elasticsearch, incluindo:	5		0	
		44	a) Identificação do usuário (login e IP);	5		0	
		45	b) Momento de início e fim da consulta;	5		0	
		46	c) Consulta executada e parâmetros utilizados;	5		0	
		47	d) Status da consulta e volumetria (ex.: quantidade de vértices/arestas retornados);	3		0	
		48	e) Algoritmo utilizado, parâmetros e resultados quando aplicável.	3		0	
	Controle de Acesso	49	A plataforma deve oferecer controle de acesso baseado em papéis (RBAC) nos seguintes níveis:	5		0	
		50	a) Nível da base de dados;	5		0	
		51	b) Nível dos vértices e dos relacionamentos;	5		0	
		52	c) Nível dos atributos de relacionamentos e vértices.	5		0	
		53	A plataforma deve oferecer controle de acesso baseado em papéis (RBAC) e atributos (ABAC), permitindo definir políticas por rótulos, tipos de vértices e propriedades.	5		0	
		54	Deve garantir isolamento seguro entre usuários e grupos de trabalho.	5		0	
	Governança de Dados	55	A plataforma deve integrar-se a catálogos corporativos de metadados, como Apache Atlas ou DataHub, permitindo classificação de dados, lineage e associação de políticas de acesso.	4		0	
		56	Deve apoiar estratégias de governança de dados institucionais.	3		0	
	Execução Híbrida de Consultas e Algoritmos	57	A plataforma deve permitir a execução combinada de consultas declarativas e algoritmos de análise de grafos no mesmo pipeline de processamento, sem necessidade de exportação de dados para outro ambiente ou engine.	5		0	
	Suporte a Métodos Nativos para Otimização de Consultas	58	A plataforma deve oferecer ferramentas nativas para inspeção do plano de execução de consultas (explain/profile), facilitando a otimização de travessias complexas.	5		0	
	Controle de Acesso Sensível a Contexto	59	Deve ser possível restringir ou permitir consultas com base no conteúdo semântico da consulta ou em metadados de contexto (ex.: hora do dia, tipo de operação, IP de origem ou sensibilidade dos nós envolvidos).	5		0	
	Suporte a Consultas Georreferenciadas em Grafos	60	A plataforma deve suportar operadores georreferenciadas para uso em consultas e algoritmos de grafos, com suporte a tipos de dados como coordenadas, distâncias e polígonos.	5		0	
	Indexação Vetorial Nativa em Propriedades de Nós e Arestas	61	A plataforma deve permitir criação de índices vetoriais nativos diretamente em propriedades de nós e relacionamentos, suportando busca aproximada (Approximate Nearest Neighbor - ANN) e integração com consultas declarativas de grafos no mesmo pipeline.	5		0	
	GraphRAG com Busca Vetorial e Estrutura de Grafo Unificada	62	A plataforma deve possibilitar a implementação de pipelines de GraphRAG onde a busca vetorial por embeddings e a exploração de relacionamentos no grafo ocorram no mesmo banco, sem necessidade de ferramentas externas para armazenamento vetorial.	5		0	
	Consultas Multimodais Otimizadas (Vetor e Grafo)	63	A plataforma deve possibilitar combinar, em uma única consulta, filtros relacionais (grafo), textuais (full-text) e semânticos (vetoriais) com otimização automática do plano de execução.	5		0	
	Desempenho e Escalabilidade	64	A plataforma deve ser escalável horizontalmente e lidar com grafos com centenas de milhões de nós e arestas.	5		0	
		65	Deve oferecer tempo de resposta reduzido para consultas complexas, com otimização de travessias profundas.	5		0	
		66	Deve suportar sharding para distribuição do processamento de uma mesma consulta ou algoritmo entre diferentes nós do cluster.	5		0	
		67	A arquitetura deve permitir testes comparativos de desempenho com e sem uso de sharding.	5		0	
		68	A plataforma deve suportar expansão horizontal de cluster sem tempo de inatividade, com redistribuição automática de partições de grafo e balanceamento adaptativo de recursos, assegurando performance consistente em cenários de grande volume.	5		0	
	Alta Disponibilidade e Resiliência	69	Deve prover mecanismos de alta disponibilidade, failover, e recuperação de desastres.	5		0	
		70	A plataforma deve oferecer mecanismos de alta disponibilidade, com replicação síncrona e assíncrona entre nós do cluster, failover automático e balanceamento de carga dinâmico.	5		0	
		71	Deve permitir manutenção e expansão sem interrupção do serviço, garantindo resiliência e continuidade operacional.	5		0	
	Segurança e Controle de Acesso	72	A plataforma deve implementar autenticação robusta (OAuth2, SAML, OpenID Connect).	5		0	
		73	A plataforma deverá integrar-se de forma segura com Active Directory (AD) e LDAP, permitindo autenticação centralizada e sincronização automática de grupos e perfis.	5		0	
		74	A integração deve suportar SSL/TLS e permitir o uso de múltiplas fontes de diretório em ambientes híbridos.	4		0	
		75	A plataforma deverá implementar políticas de acesso baseadas em papéis (RBAC).	5		0	
		76	A plataforma deve implementar controle de acesso baseado em papéis e permissões granulares em nível de grafo, nó ou aresta.	5		0	
	Auditoria e Rastreabilidade	77	A plataforma deve registrar todas as operações administrativas e consultas executadas, mantendo logs assinados digitalmente e armazenados de forma segura para auditoria e rastreabilidade de conformidade com a LGPD e normas do TCU.	4		0	
	Portabilidade e Multicloud	78	A plataforma deve poder ser implantada em ambiente local ou de nuvem privada, com opções de provisionamento rápido.	5		0	
		79	A plataforma deve poder ser implantada em ambientes que sigam as definições de nuvem soberana.	5		0	
		80	A plataforma deve operar em ambientes locais ou de nuvens privadas, com suporte a implantação em containers (Kubernetes, OpenShift) e integração com sistemas de CI/CD.	5		0	
	Usabilidade e Documentação	81	Deve possuir interface intuitiva e documentação abrangente para usuários técnicos e não técnicos.	5		0	

Requisitos Nao Funcionais	Extensibilidade e Integração	82	Deve permitir extensão por meio de scripts, plugins e integração com ferramentas externas de BI, ETL, ML e gestão de dados.	5		0	
		83	Deve suportar união dinâmica de duas ou mais bases de relacionamento apenas durante o momento da consulta, para execução da consulta ou aplicação do algoritmo, sem necessidade de mesclagem física de dados, mantendo alta performance.	5		0	
		84	Deve permitir carga de dados em lote, em grandes volumes (bilhões de registros), através de arquivos em formatos abertos, preferencialmente Parquet.	5		0	
		85	A plataforma deverá ser amplamente adotada e reconhecida no mercado, com base consolidada de usuários corporativos, comunidade ativa, ampla oferta de recursos educacionais, documentação, fóruns e eventos técnicos, a fim de garantir a sustentabilidade e evolução da tecnologia no longo prazo.	5		0	
	Backup e restore	86	A plataforma deve oferecer mecanismos nativos de backup completo (full) e incremental dos grafos e metadados, permitindo agendamento automatizado, armazenamento seguro criptografado e restauração seletiva em diferentes instâncias do ambiente.	5		0	
		87	Deve ser possível realizar testes de integridade dos backups e manter logs detalhados das operações de cópia e restauração para fins de auditoria.	5		0	
	Monitoramento do Ambiente	88	A plataforma deverá incluir ferramentas de monitoramento contínuo para métricas de desempenho, uso de recursos, consultas, transações e estado de cluster.	5		0	
		89	Deve permitir integração com ferramentas externas como Prometheus, Grafana ou equivalentes, além de oferecer alertas configuráveis e dashboards personalizáveis.	4		0	
	Licenciamento e Gestão de Usuários	90	O sistema deve possuir modelo de licenciamento transparente, permitindo gestão centralizada de licenças, monitoramento de uso e geração de relatórios administrativos, evitando restrições de conformidade contratual.	5		0	
	Suporte e Adoção no Mercado	91	A plataforma deverá dispor de estrutura de suporte técnico abrangente, preferencialmente com representação ou presença comercial e técnica no Brasil, visando garantir agilidade no atendimento, suporte em português e aderência a necessidades locais.	5		0	
		92	O fornecedor deve garantir suporte de longo prazo (Long-Term Support - LTS) com atualizações regulares de segurança e correções de bugs.	5		0	
		93	A atualização da plataforma deve ser realizada de forma não disruptiva e documentada, preservando compatibilidade retroativa.	5		0	
		94	Garantir suporte técnico em português, SLAs no Brasil e conformidade fiscal/contratual para fornecimento ao setor público no Brasil.	5		0	
	Migração de Código Existente	95	Disponibilizar suporte à migração de workloads existentes (consultas, modelos, rotinas), contemplando ao menos uma linguagem declarativa de grafos amplamente usada (ex.: openCypher/Cypher-like, GQL, Gremlin, SPARQL), com metodologia e ferramentas de apoio.	2		0	
		96	O processo deve gerar relatórios de conversão e logs de erro, permitindo análise e revisão assistida pelo time técnico.	1		0	
	Nuvem Privada	97	A plataforma deve suportar implantação em nuvem privada AWS Outposts.	5		0	
		98	A plataforma deve suportar implantação em nuvem privada Google Distributed Cloud (GDC).	5		0	
		99	A plataforma deve suportar implantação em nuvem privada Huawei Cloud Stack (HCS).	5		0	
	Serviços Gerenciados	100	A plataforma deve permitir implantação e operação em nuvem privada com gestão de ciclo de vida automatizada, utilizando método padronizado e suportado pelo fabricante.	5		0	
Pontuação						0	