

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE  
SÃO PAULO**

**MATHEUS DI GIACOMO**

**SOFTWARE DRAWIO**

**CAMPOS DO JORDÃO  
2025**

## RESUMO

Este trabalho apresenta uma análise detalhada do Draw.io (também conhecido como diagrams.net), uma ferramenta gratuita e de código aberto desenvolvida para criação de diagramas técnicos e modelagem visual. O objetivo principal é demonstrar as funcionalidades, vantagens e aplicações práticas desta ferramenta no contexto acadêmico e profissional, com ênfase especial em sua utilização para modelagem de bancos de dados. A metodologia utilizada baseou-se em pesquisa bibliográfica e análise técnica da plataforma. Os resultados mostram que o Draw.io é uma solução eficiente, multiplataforma e gratuita para criação de diversos tipos de diagramas, incluindo modelos conceituais, lógicos e físicos de bancos de dados, oferecendo interface intuitiva e recursos adequados para o ensino e desenvolvimento de projetos. Conclui-se que a ferramenta representa uma alternativa versátil e acessível para modelagem de dados e criação de diagramas técnicos, especialmente no ambiente educacional e em projetos de código aberto.

**Palavras-Chave:** Draw.io; Diagrams.net; Modelagem de Dados; Diagramas Técnicos; Banco de Dados; Ferramentas de Diagramação.

## ABSTRACT

This work presents a detailed analysis of Draw.io (also known as diagrams.net), a free and open-source tool developed for creating technical diagrams and visual modeling. The main objective is to demonstrate the functionalities, advantages and practical applications of this tool in academic and professional contexts, with special emphasis on its use for database modeling. The methodology used was based on bibliographic research and technical analysis of the platform. The results show that Draw.io is an efficient, cross-platform and free solution for creating various types of diagrams, including conceptual, logical and physical database models, offering an intuitive interface and adequate resources for teaching and project development. It is concluded that the tool represents a versatile and accessible alternative for data modeling and technical diagram creation, especially in educational environments and open-source projects.

**Keywords:** Draw.io; Diagrams.net; Data Modeling; Technical Diagrams; Database; Diagramming Tools.

## LISTA DE ILUSTRAÇÕES

**FIGURA 1** – Interface principal do Draw.io

10

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b> _____	<b>12</b>
<b>1.1</b>	<b>Objetivos</b> _____	<b>12</b>
<b>1.2</b>	<b>Justificativa</b> _____	<b>12</b>
<b>1.3</b>	<b>Aspectos Metodológicos</b> _____	<b>12</b>
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b> _____	<b>14</b>
<b>2.1</b>	<b>Modelagem Visual</b> _____	<b>14</b>
<b>2.2</b>	<b>Diagramas Técnicos em Engenharia de Software</b> _____	<b>14</b>
<b>2.3</b>	<b>Ferramentas de Diagramação</b> _____	<b>14</b>
<b>3</b>	<b>SOFTWARE DRAWIO</b> _____	<b>15</b>
<b>3.1</b>	<b>Características Gerais</b> _____	<b>15</b>
<b>3.2</b>	<b>Funcionalidades Principais</b> _____	<b>15</b>
<b>3.3</b>	<b>Vantagens e Limitações</b> _____	<b>16</b>
<b>4</b>	<b>CONCLUSÃO</b> _____	<b>23</b>
	<b>REFERÊNCIAS</b> _____	<b>24</b>

## 1 INTRODUÇÃO

A representação visual de informações constitui uma das práticas fundamentais no desenvolvimento de sistemas de informação e na documentação de processos organizacionais. A capacidade de criar diagramas técnicos claros e padronizados é essencial para a comunicação efetiva entre equipes de desenvolvimento, analistas de sistemas e stakeholders.

O Draw.io, atualmente conhecido também como diagrams.net, surgiu como uma solução gratuita e de código aberto para criação de diagramas técnicos diversos, incluindo fluxogramas, diagramas UML, diagramas de rede e, especialmente relevante para este estudo, modelos de bancos de dados. Desenvolvido inicialmente por JGraph.Ltd e posteriormente mantido pela comunidade open-source, o Draw.io se consolidou como uma alternativa robusta às ferramentas comerciais de diagramação.

Diferentemente de soluções proprietárias que exigem licenciamento oneroso, o Draw.io oferece recursos profissionais sem custos, executando diretamente no navegador web ou através de aplicações desktop, garantindo acessibilidade e flexibilidade de uso. Esta característica tornou a ferramenta particularmente popular no ambiente educacional e em projetos de tecnologia com recursos limitados.

### 1.1 Objetivos

Este trabalho tem por objetivo apresentar e analisar o Draw.io como ferramenta de modelagem e diagramação técnica, destacando suas principais características, funcionalidades e aplicações no contexto acadêmico e profissional, com ênfase em modelagem de bancos de dados.

Para a consecução deste objetivo foram estabelecidos os objetivos específicos:

- Apresentar as características do software Draw.io;

- Analisar as principais funções oferecidas pela ferramenta;
- A Identificar as vantagens e limitações do software;
- Demonstrar a aplicabilidade do Draw.io no ensino de banco de dados e desenvolvimento de sistemas;
- Comparar os recursos da ferramenta com outras soluções disponíveis no mercado.

## **1.2 Justificativa**

A relevância deste trabalho justifica-se pela necessidade de conhecer ferramentas acessíveis para modelagem de dados e criação de diagramas técnicos no contexto acadêmico. O Draw.io representa uma alternativa gratuita e multiplataforma que democratiza o acesso a recursos profissionais de diagramação, atendendo demandas educacionais e empresariais.

No contexto educacional brasileiro, onde instituições enfrentam limitações orçamentárias para aquisição de licenças de software, o Draw.io emerge como uma solução viável que não compromete a qualidade do ensino nem a preparação profissional dos estudantes.

## **1.3 Aspectos Metodológicos**

Cite aqui de maneira bastante sucinta a metodologia utilizada neste trabalho. Um exemplo é apresentado no parágrafo a seguir. Depois haverá uma seção (3) na qual cada etapa da metodologia será explicitada detalhadamente.

O presente estudo fez uso das pesquisas de natureza bibliográfica, para o que remete à parte escrita, e de campo, para o que se refere à parte prática.

## **2 FUNDAMENTAÇÃO TEÓRICA**

Nesta seção será apresentada uma revisão de textos, artigos, documentação técnica e demais materiais pertinentes à fundamentação teórica necessária ao desenvolvimento do trabalho.

### **2.1 Modelagem de Dados**

A modelagem visual é o processo de representar conceitos, estruturas e processos através de elementos gráficos padronizados. Segundo Fowler (2004), a modelagem visual é fundamental para o desenvolvimento de software, pois facilita a comunicação de ideias complexas de forma clara e concisa, reduzindo ambiguidades e facilitando o entendimento compartilhado entre diferentes stakeholders.

A representação visual de sistemas de informação permite que desenvolvedores, analistas e usuários compartilhem uma visão comum da estrutura e comportamento do sistema. Esta prática tornou-se indispensável na engenharia de software moderna, onde a complexidade dos sistemas exige mecanismos eficientes de documentação e comunicação.

No contexto de bancos de dados, a modelagem visual permite representar entidades, relacionamentos, atributos e restrições de forma intuitiva, facilitando o processo de design e a identificação precoce de problemas estruturais. A qualidade da modelagem visual impacta diretamente na qualidade do produto, na manutenibilidade do sistema e na eficiência da comunicação técnica.



## 2.2 Diagramas Técnicos em Engenharia de Software

Os diagramas técnicos constituem a base da documentação em engenharia de software. Pressman (2016) destaca que diferentes tipos de diagramas servem a propósitos específicos no ciclo de vida do desenvolvimento de software, desde a captura de requisitos até a implementação e manutenção.

Entre os principais tipos de diagramas utilizados em projetos de software, destacam-se:

**Diagramas UML:** A Unified Modeling Language (UML) estabelece um padrão para modelagem de sistemas orientados a objetos, incluindo diagramas de classes, casos de uso, sequência, atividades, entre outros. Segundo Booch, Rumbaugh e Jacobson (2005), a UML fornece uma linguagem visual padrão para especificar, visualizar, construir e documentar artefatos de sistemas de software.

**Diagramas de Banco de Dados:** Incluem o Modelo Entidade-Relacionamento (MER), diagramas de modelo lógico e diagramas de modelo físico. Estes diagramas representam a estrutura de dados do sistema, definindo entidades, atributos, relacionamentos e restrições de integridade.

**Fluxogramas:** Representam algoritmos e processos através de símbolos padronizados, facilitando a compreensão da lógica de processamento.

**Diagramas de Arquitetura:** Ilustram a estrutura de alto nível do sistema, incluindo componentes, módulos e suas interações.

A padronização destes diagramas é fundamental para garantir que diferentes profissionais possam interpretar corretamente as representações visuais, independentemente de sua formação ou experiência específica.

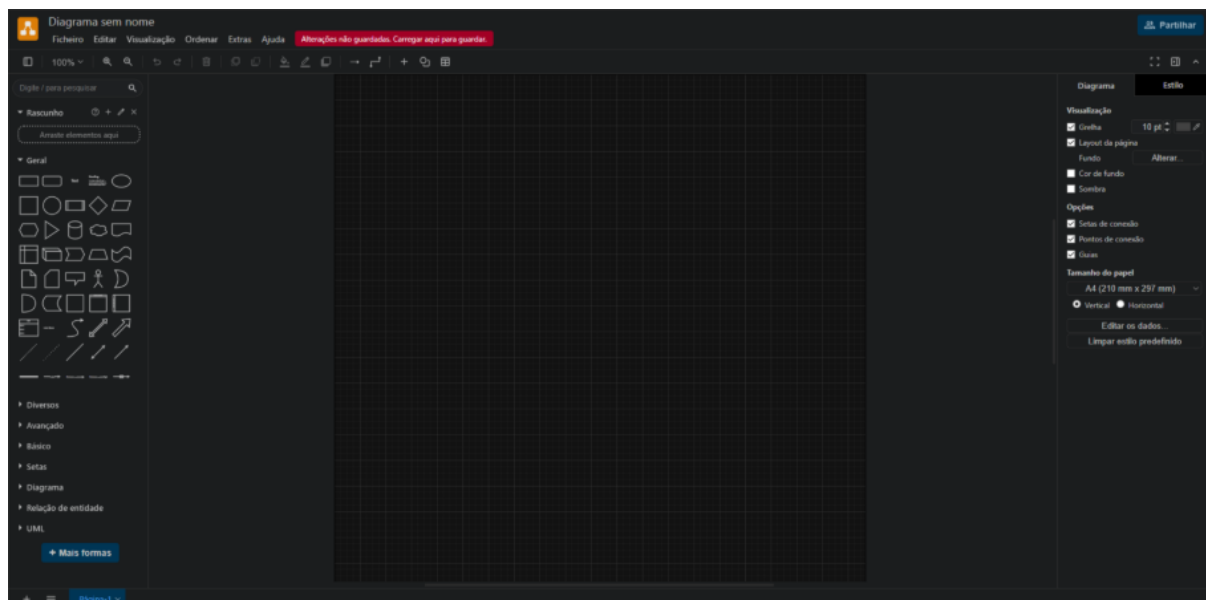
## 2.3 Ferramentas de Diagramação

As ferramentas de diagramação evoluíram significativamente desde os primeiros sistemas CAD (Computer-Aided Design). Atualmente, estas ferramentas podem ser classificadas em diferentes categorias baseadas em sua arquitetura, modelo de licenciamento e público-alvo.

As principais categorias incluem: ferramentas desktop tradicionais (como

Microsoft Visio) instaladas localmente; ferramentas web-based que executam no navegador oferecendo acessibilidade de qualquer dispositivo; e ferramentas open-source que permitem inspeção e modificação do código-fonte sem custos de licenciamento.

Sommerville (2011) observa que a escolha de ferramentas adequadas impacta significativamente na produtividade das equipes de desenvolvimento. O Draw.io se posiciona na intersecção destas categorias, sendo uma ferramenta web-based de código aberto que também oferece versões desktop, combinando as vantagens de diferentes abordagens arquiteturais.



**Figura 1** – Interface principal do Draw.io (O Autor, 2025)

### **3 SOFTWARE DRAWIO**

O Draw.io, oficialmente renomeado para diagrams.net em 2020, é uma ferramenta de diagramação gratuita e de código aberto desenvolvida para criar diversos tipos de diagramas técnicos. A ferramenta se destaca por sua versatilidade, acessibilidade e pela capacidade de operar tanto online quanto offline.

#### **3.1 Características Gerais**

O Draw.io foi desenvolvido pela JGraph Ltd e posteriormente teve seu código aberto à comunidade. A ferramenta é executada como aplicação web no navegador, mas também está disponível como aplicação desktop para Windows, macOS e Linux através do framework Electron.

Uma característica distintiva do Draw.io é sua filosofia de privacidade e segurança. A ferramenta não requer criação de conta obrigatória e não armazena diagramas em servidores proprietários. Os arquivos podem ser salvos localmente, em serviços de nuvem escolhidos pelo usuário (Google Drive, OneDrive, Dropbox) ou em repositórios Git, garantindo controle total sobre os dados.

A interface gráfica é intuitiva e organizada, com barra de ferramentas superior, painéis laterais com bibliotecas de formas e área de trabalho central. O software suporta extensas bibliotecas de formas pré-definidas, incluindo diagramas de fluxo, UML, Entidade-Relacionamento, rede, BPMN, mockups de interfaces e diagramas de engenharia. A capacidade de trabalhar offline é particularmente valiosa em contextos em que a conectividade é limitada.

### 3.2 Funcionalidades Principais

O Draw.io oferece um conjunto abrangente de funcionalidades para criação e gerenciamento de diagramas técnicos:

**Modelagem de Banco de Dados:** O Draw.io possui bibliotecas específicas para modelagem de dados, incluindo formas para representação de entidades, atributos e relacionamentos seguindo o padrão do Modelo Entidade-Relacionamento. A ferramenta permite criar modelos conceituais, lógicos e físicos, com suporte a notações variadas como crow's foot, UML e Chen. Oferece formas de tabelas que permitem representar estruturas de banco de dados com colunas, tipos de dados, chaves primárias e estrangeiras.

**Interface Intuitiva e Personalizável:** A interface é altamente customizável, permitindo ocultar ou exibir painéis, reorganizar elementos e personalizar atalhos de teclado. O sistema de arrastar-e-soltar facilita a adição de formas, e conectores inteligentes se ajustam automaticamente quando elementos são movidos.

**Bibliotecas Extensas:** O Draw.io inclui centenas de bibliotecas pré-definidas que podem ser ativadas conforme necessário. As bibliotecas UML são particularmente completas, incluindo todos os tipos de diagramas da especificação UML 2.5.

**Colaboração e Compartilhamento:** Oferece mecanismos de colaboração através de integração com plataformas de armazenamento em nuvem (Google Drive, OneDrive, Dropbox) e repositórios Git (GitHub, GitLab, Bitbucket), permitindo versionamento de diagramas junto com o código-fonte.

**Exportação e Importação:** Uma das funcionalidades mais valiosas é a capacidade de exportação para múltiplos formatos: PNG, JPEG, SVG, PDF, XML e VSDX (Microsoft Visio). O Draw.io também pode importar arquivos de outras ferramentas, incluindo Microsoft Visio, Lucidchart e Gliffy, facilitando migração de projetos existentes.

### 3.3 Vantagens e Limitações

O Vantagens do software:

Gratuidade: O software é completamente gratuito, sem versões premium ou limitações de recursos;

Multiplataforma: Funciona em qualquer sistema operacional através do navegador web e oferece aplicações desktop para Windows, macOS e Linux;

Código Aberto: Por ser open-source, o código pode ser auditado e modificado conforme necessidades específicas;

Privacidade e Segurança: Não requer criação de conta obrigatória e permite trabalho completamente offline;

Versatilidade: Suporta dezenas de tipos diferentes de diagramas, tornando-se uma ferramenta única para diversas necessidades;

Integração com Serviços Populares: Integra-se nativamente com Google Drive, OneDrive, Dropbox, GitHub, GitLab e outros serviços;

Exportação Flexível: Suporta exportação para múltiplos formatos profissionais, incluindo PDF, SVG, PNG e VSDX.

Limitações Identificadas:

Ausência de Conversão Automática: Não oferece conversão automática entre modelos conceituais, lógicos e físicos;

Sem Geração de Código: Não possui capacidade nativa de gerar scripts SQL ou código de implementação;

Colaboração em Tempo Real Limitada: Não oferece edição colaborativa simultânea como algumas ferramentas comerciais;

Validação Limitada: Não possui mecanismos robustos de validação automática de modelos de banco de dados;

Performance: Em diagramas extremamente complexos, a performance pode degradar, especialmente na versão web;

Suporte Técnico: Por ser gratuito e open-source, não oferece suporte técnico comercial dedicado.

Apesar das limitações identificadas, o Draw.io representa uma solução eficiente para a maioria das necessidades de diagramação técnica no contexto acadêmico e profissional.

## 4 CONCLUSÃO

O Draw.io demonstra ser uma ferramenta valiosa e versátil para modelagem de dados e criação de diagramas técnicos no contexto brasileiro. Através da análise realizada, foi possível identificar que o software atende adequadamente aos objetivos propostos, oferecendo funcionalidades profissionais sem custos de licenciamento. Sua interface intuitiva, extensas bibliotecas de formas, capacidade multiplataforma e filosofia de privacidade fazem dele uma opção atrativa tanto para o ensino quanto para uso profissional.

A característica de código aberto representa um diferencial significativo, garantindo transparência e independência de fornecedores comerciais. Para o contexto acadêmico brasileiro, o Draw.io representa uma alternativa viável para ensino de modelagem de dados e documentação técnica. A ausência de conversão automática entre níveis de modelagem, embora represente uma limitação, pode ser vista como oportunidade pedagógica ao promover compreensão mais profunda dos conceitos.

Conclui-se que o Draw.io constitui uma ferramenta essencial para profissionais e estudantes da área de tecnologia da informação, oferecendo recursos profissionais, flexibilidade e total liberdade de uso, representando uma democratização do acesso a ferramentas de qualidade para modelagem e documentação técnica.

## REFERÊNCIAS

SOBRENOME, Autor. **Título:** subtítulo (se houver). Disponível em: <endereçoonlinecompleto>. Acesso em: dia mês ano.

**BOOCH, Grady; RUMBAUGH, James; JACOBSON, Ivar.** UML: Guia do Usuário. 2. ed. Rio de Janeiro: Elsevier, 2005.

**DIAGRAMS.NET.** Official Documentation. Disponível em: <https://www.diagrams.net/doc/>. Acesso em: 23 out. 2024.

**FOWLER, Martin.** UML Essencial: Um Breve Guia para Linguagem Padrão de Modelagem de Objetos. 3. ed. Porto Alegre: Bookman, 2004.

**HEUSER, Carlos Alberto.** Projeto de Banco de Dados. 6. ed. Porto Alegre: Bookman, 2009.

**JGRAPH LTD.** draw.io - About. Disponível em: <https://about.draw.io/>. Acesso em: 23 out. 2024.

**PRESSMAN, Roger S.** Engenharia de Software: Uma Abordagem Profissional. 8. ed. Porto Alegre: AMGH, 2016.

**SILBERSCHATZ, Abraham; KORTH, Henry F.; SUDARSHAN, S.** Sistema de Banco de Dados. 6. ed. Rio de Janeiro: Elsevier, 2012.

**SOMMERVILLE, Ian.** Engenharia de Software. 9. ed. São Paulo: Pearson, 2011.

**TEOREY, Toby J.; LIGHTSTONE, Sam; NADEAU, Tom.** Database Modeling and Design: Logical Design. 4. ed. San Francisco: Morgan Kaufmann, 2006.