GENERARE DIAGRAME

INGINERIE SOFTWARE

# diagrams.net (ex-draw.io)



## Introducere

diagrams.net (cunoscuta drept draw.io) este o aplicatie cu predilectie web de desenare si diagramare gratuita, open-source care permite crearea de diagrame, organigrame, diagrame de flux etc.

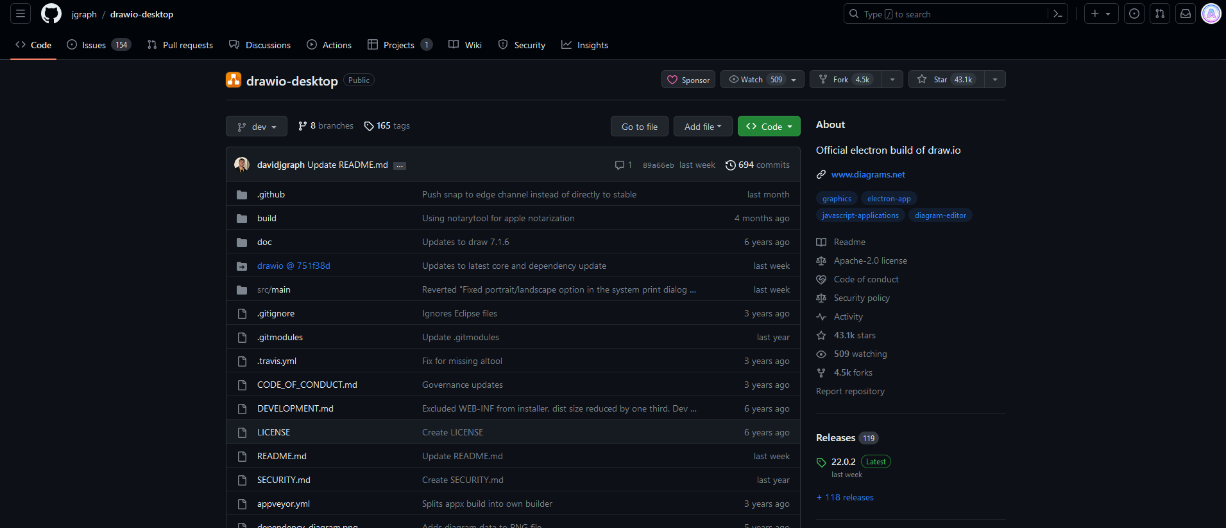
drawio-desktop este o aplicatie desktop pentru diagrame si whiteboarding bazata pe Electron care integreaza editorul de baza draw.io.

## Instalare

Initial, in contextul in care instalarea nu ar fi benefica utilizatorului, se poate opta pentru varianta online care se afla la adresa: <https://diagrams.net/>.

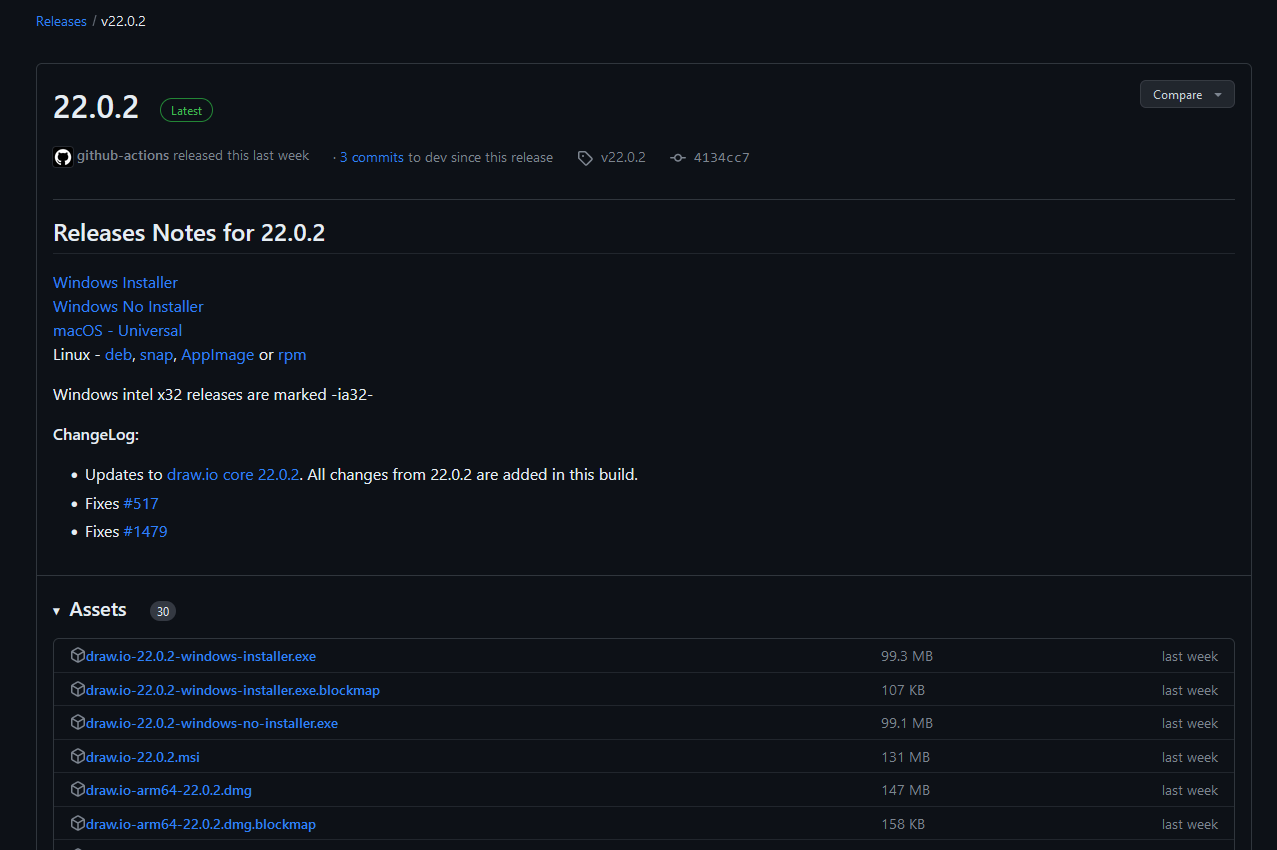
1. **Identificare site web**

Se va accesa host-ul de pe GitHub la adresa: https://github.com/jgraph/drawio-desktop/ si se va cauta sectiunea **Releases**.

****

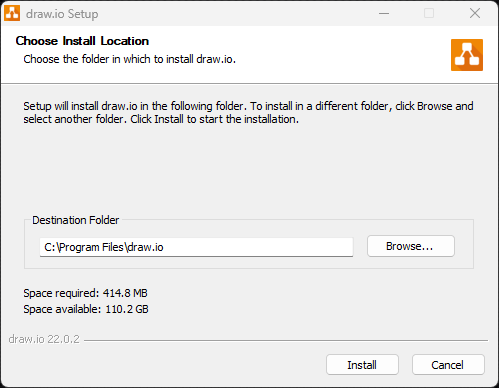
1. **Descarcare**

Se va descarca ultima versiune pentru sistemul de operare specific, din bara **Assets**.



**3. Instalare**

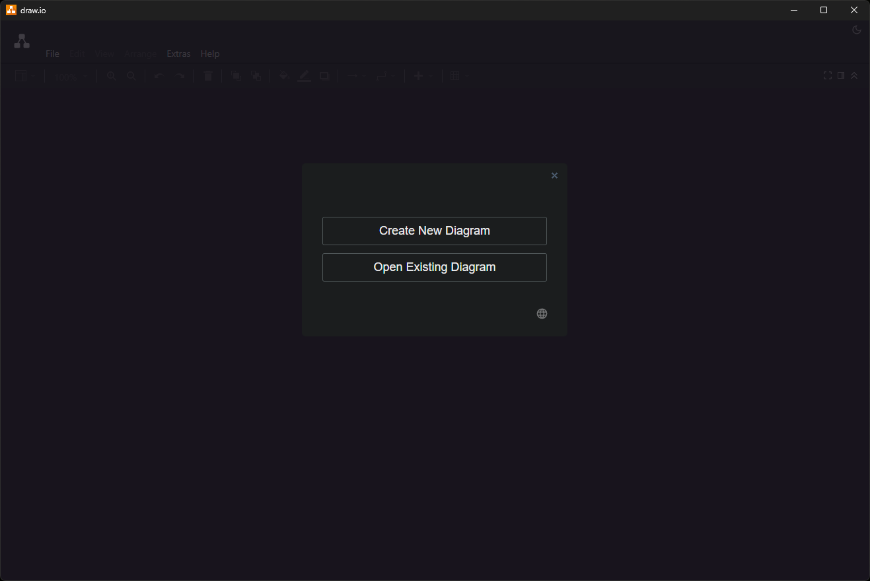
Instalarea se va executa intr-un mod foarte simplu. Se va selecta calea de instalare si apoi aceasta va fi finalizata.



## Utilizare aplicatie

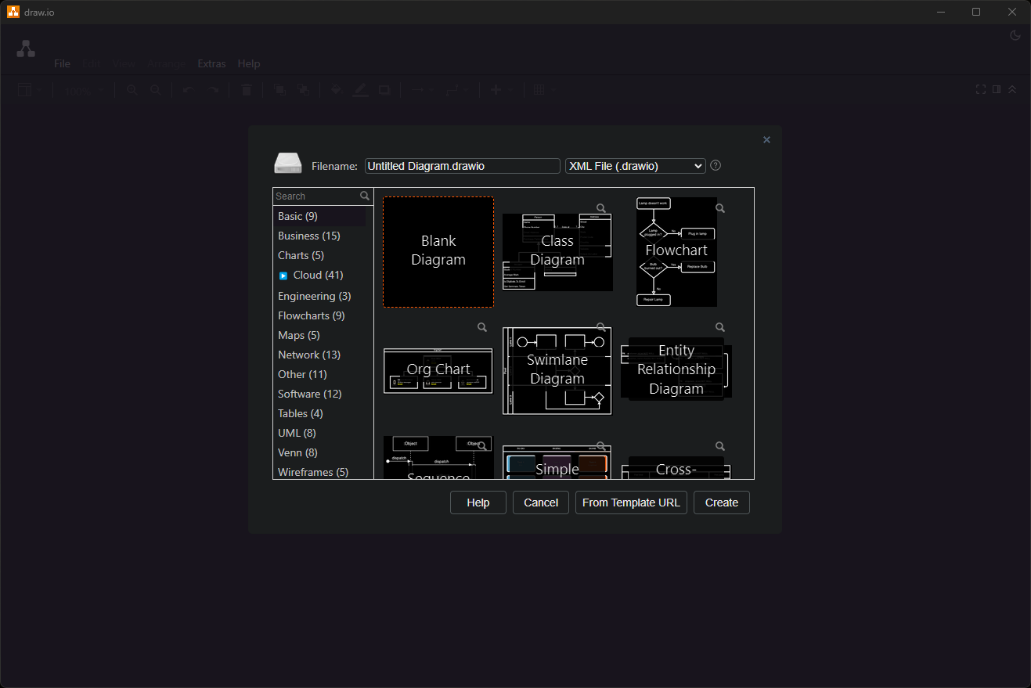
**1. Rulare initiala**

Aplicatia se va lansa. Aceasta va permite rularea unei diagrame pre-existente sau deschiderea unui sablon nou.



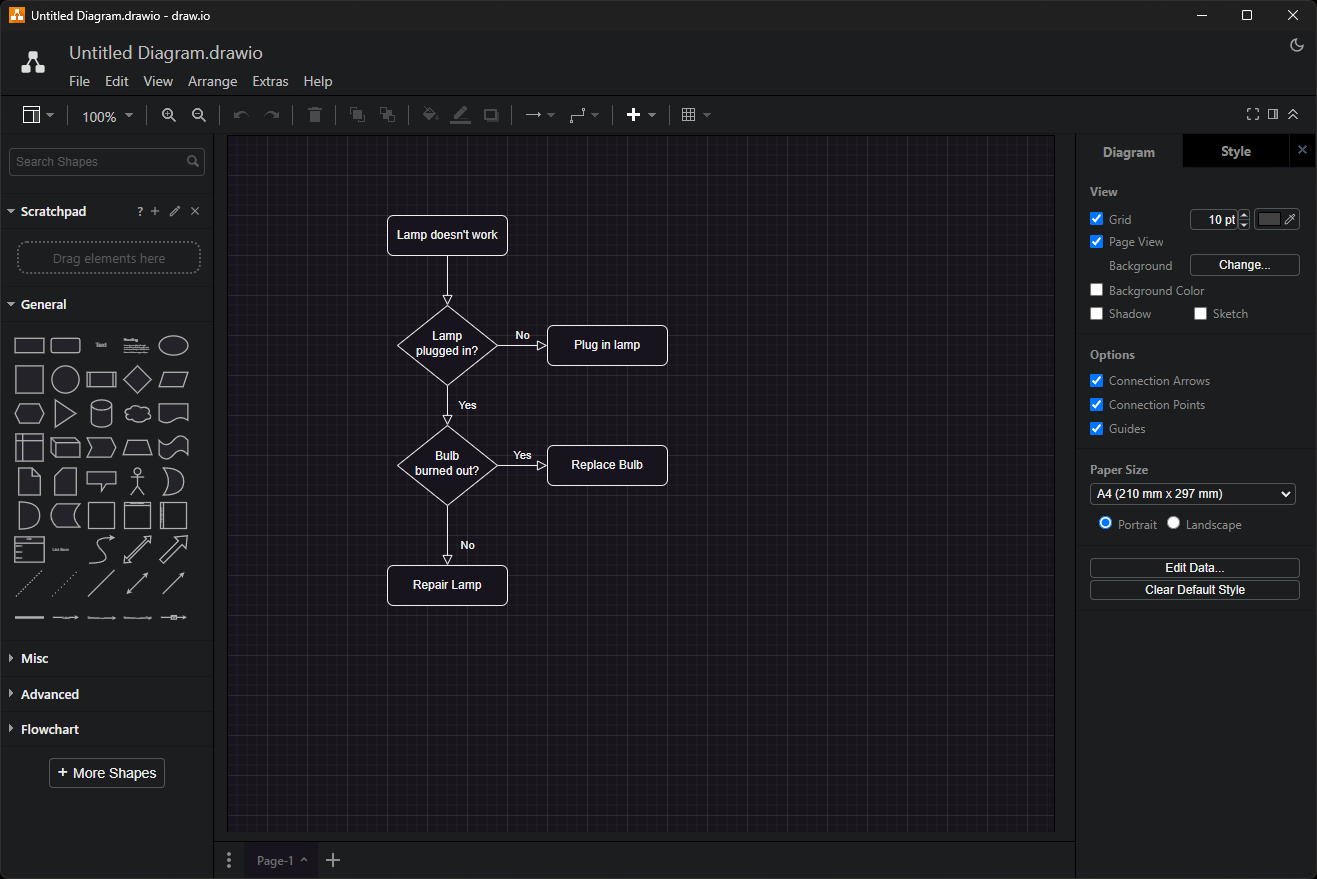
**2. Sabloane**

In urma selectarii unui sablon nou (**New diagram**), utilizatorului i se va prezenta o serie de sabloane pre-definite din care poate alege.



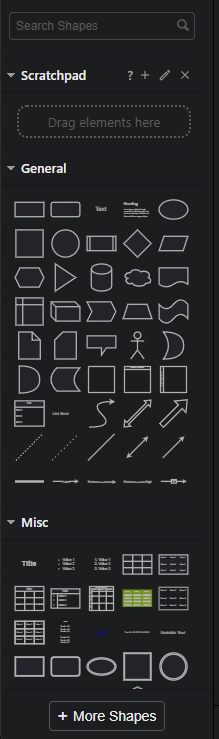
**3. Spatiu de lucru**

Aplicatia va lansa sablonul iar interfata si spatiul de lucru vor fi afisate.



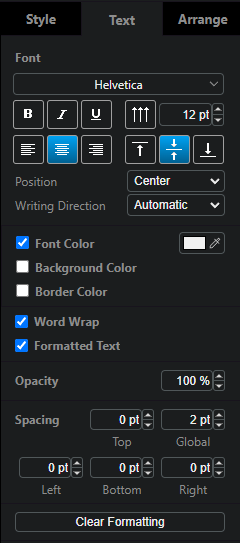
**3. Bara de instrumente**

Din acest loc, utilizatorul poate sa se foloseasca de instrumentele de desenare, precum inserarea de forme si linii in interiorul spatiului de lucru.



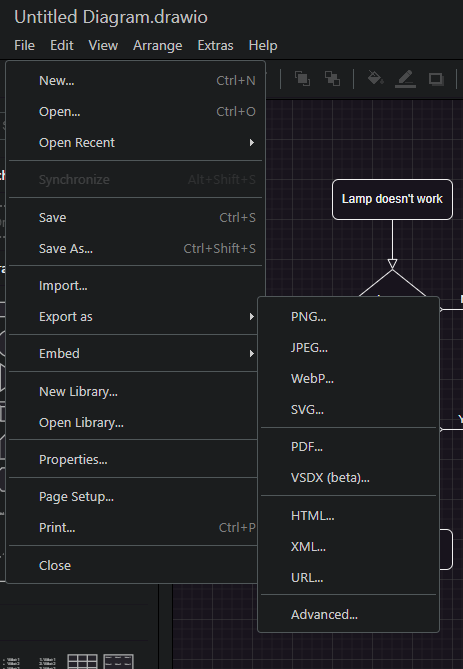
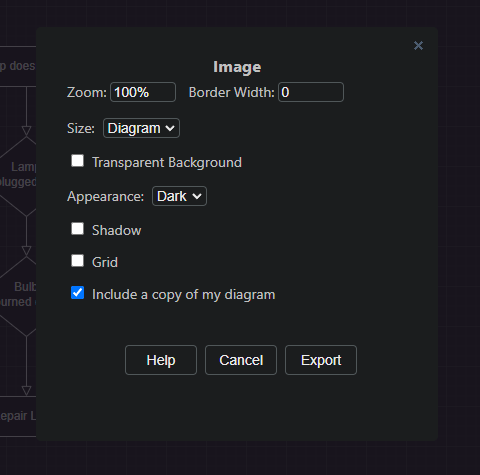
**4. Panoul de proprietati**

Acesta permite modificarea formelor sau a textului, in sensul dimensiunilor, marimii, culorilor, formatului etc.



**4. Finalizare lucru**

In momentul in care diagrama va fi finalizata, aceasta poate fi salvata (fisierul tip **\*.drawio**) si chiar exportata.



## Tipuri diagrame

Exista numeroase tipuri de diagrame care pot prezenta date, procese, relatii si concepte. Cateva dintre acestea sunt de tip:

**Flowchart** (Diagrama de flux): O diagrama care ilustreaza un proces sau un flux de lucru cu diferite forme si sageti care reprezinta etape, decizii si fluxul de informatii.

**Organizational Chart** (organigrama): O diagrama care descrie structura unei organizatii, aratand ierarhia posturilor si relatiile de raportare.

**Venn Diagram** (Diagrama Venn): O diagrama care utilizeaza cercuri suprapuse pentru a arata elementele comune si distincte ale mai multor seturi sau grupuri.

**Mind map** (Harta mentala): O diagrama ierarhica utilizata pentru a reprezenta vizual idei, concepte sau sarcini, centrata de obicei in jurul unui subiect sau al unei idei centrale.

**Gantt Chart** (Diagrama Gantt): O diagrama cu bare orizontale utilizata in gestionarea proiectelor pentru a reprezenta sarcinile sau activitatile, datele de inceput si de sfarsit ale acestora si dependentele lor.

**Diagrama de retea** (Network): O diagrama utilizata pentru a reprezenta conexiunile si relatiile dintre diverse elemente, cum ar fi noduri, dispozitive sau procese, intr-o retea.

**Diagrame UML (Unified Modeling Language):** Un set de diagrame utilizate in ingineria software si in proiectarea sistemelor pentru a reprezenta clase, obiecte, relatii si altele.

* Class Diagram
* Use Case Diagram
* Sequence Diagram
* Statechart Diagram
* Activity Diagram

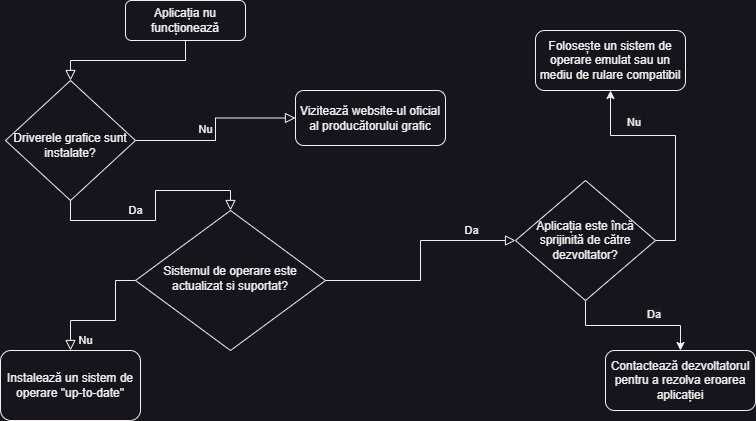
**ERD (Entity-Relationship Diagram):** O diagrama utilizata pentru a reprezenta structura si relatiile dintre datele dintr-o baza de date sau dintr-un sistem informatic.

**Bar Chart:** Folosit pentru a compara datele și valorile pentru diferite categorii.

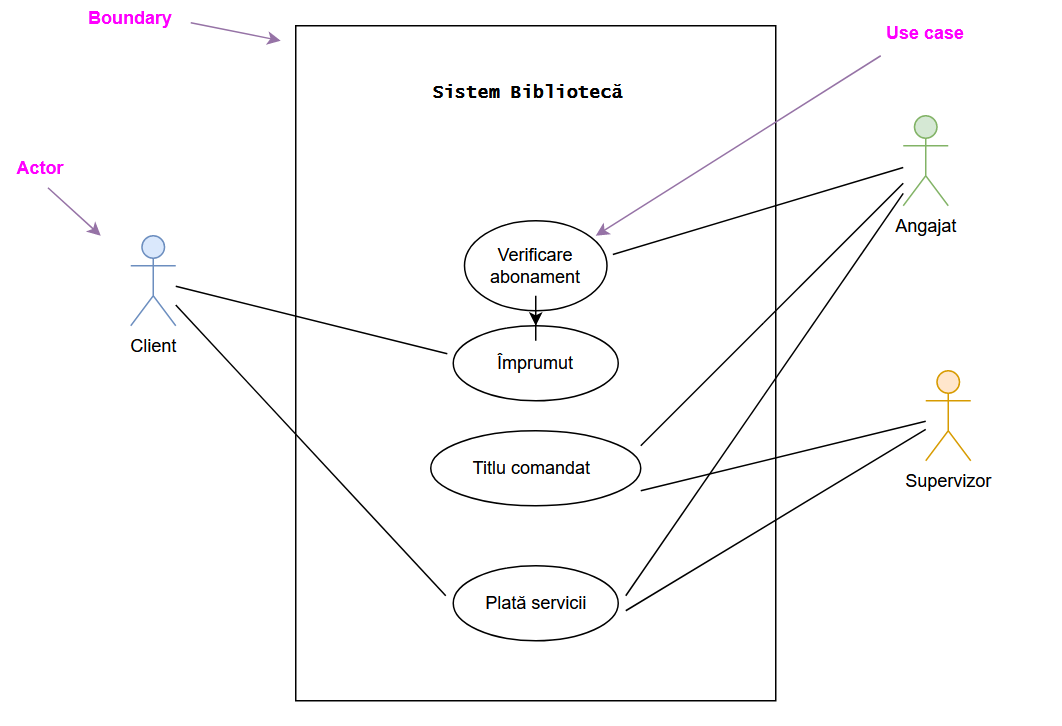
**Pie Chart:** Afișează proporțiile sau procentele unor părți în raport cu întregul.

**Line Chart:** Utilizat pentru a evidenția tendințele sau schimbările în date pe parcursul timpului.

**Exemple diagrame:**

****

Flowchart



UML – Use case