



UML diagramok készítése - példák

Rendszerfejlesztés I.

UML modellezés

<https://www.uml-diagrams.org/>

Sokféle eszközzel lehet UML diagramokat készíteni:

- Rajzoló eszközök – sokszor csak az ismertebb diagramokat ismerik, kinézetre megfelelnek, de a rajzelemek mögöttes jelentését nem vagy csak részben ismerik
 - Javasolt eszköz: draw.io
- Modellező eszközök – a diagram kinézetén felül „ismerik” az elemek jelentését, a háttérben egy modell is készül
 - Javasolt eszközök: Visual Paradigm, ArgoUML(?)

A továbbiakban a Visual Paradigm eszköz használatának néhány lépését képenyőképek segítségével kísérik végig

Visual Paradigm

Sok funkciója van, nem csak UML modellezésre alkalmas

Elérhető a Community Edition változat, melyet magáncélra ingyenesen használhatunk

<https://www.visual-paradigm.com/download/community.jsp>

Megj: Személyes adatok megadása nélkül is használható

Készítsünk új UML diagramot:

The screenshot shows the Visual Paradigm Community Edition software interface. The main window displays a 'Welcome to Visual Paradigm Community Edition' screen with several tabs: System Modeling, Cloud Arch. Design, Web Diagram, Team, and More Features. The 'System Modeling' tab is active, showing a UML class diagram on the left and a SysML diagram on the right. The 'UML +' icon is circled in red, and a list of UML diagram types is visible below it.

UML +
 Design software systems by drawing UML diagrams

- Use Case
- Class
- Sequence
- Communication
- State Machine
- Activity
- Component
- Deployment
- Package
- Object
- Composite Structure
- Timing
- Interaction Overview

ERD +
 Create conceptual, logical or physical data model that aids the communication and implementation of database design.

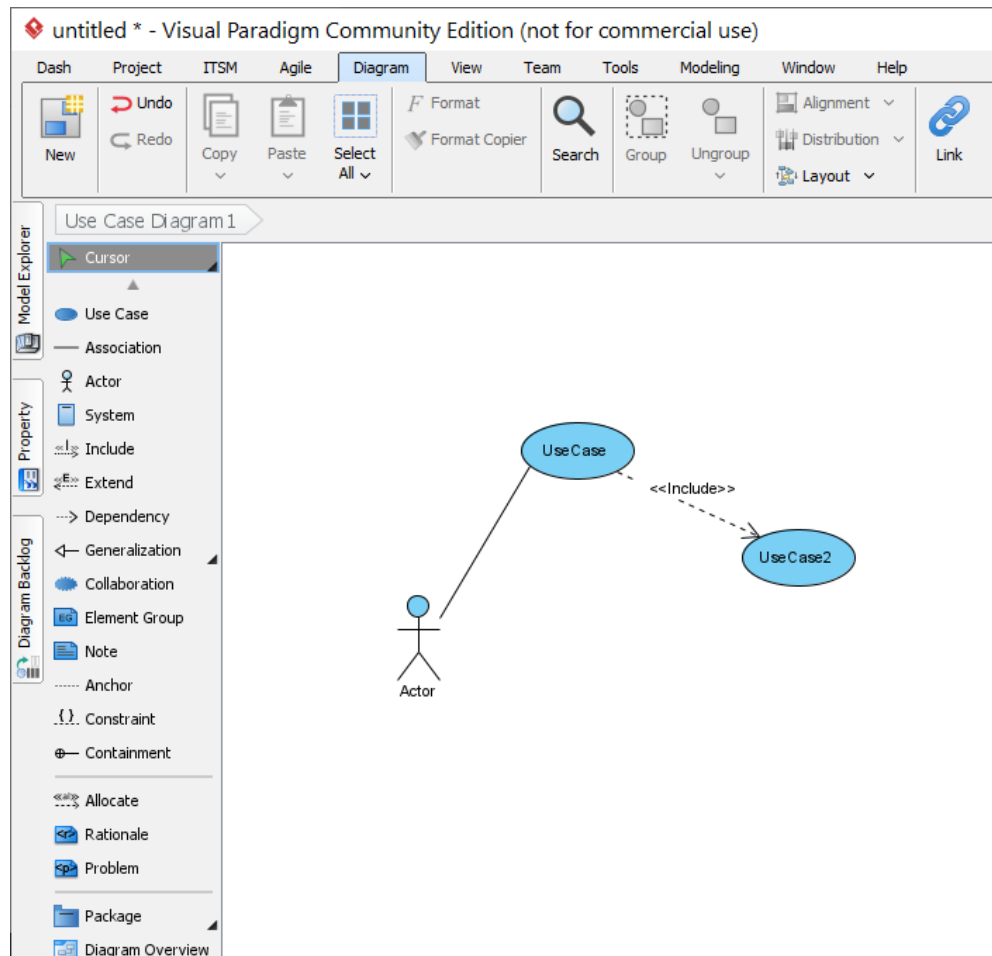
SysML +
 Create SysML diagrams for system engineering applications.

Textual Analysis +
 Document project, user problem or other information in a rich-text editor and extract elements from it.

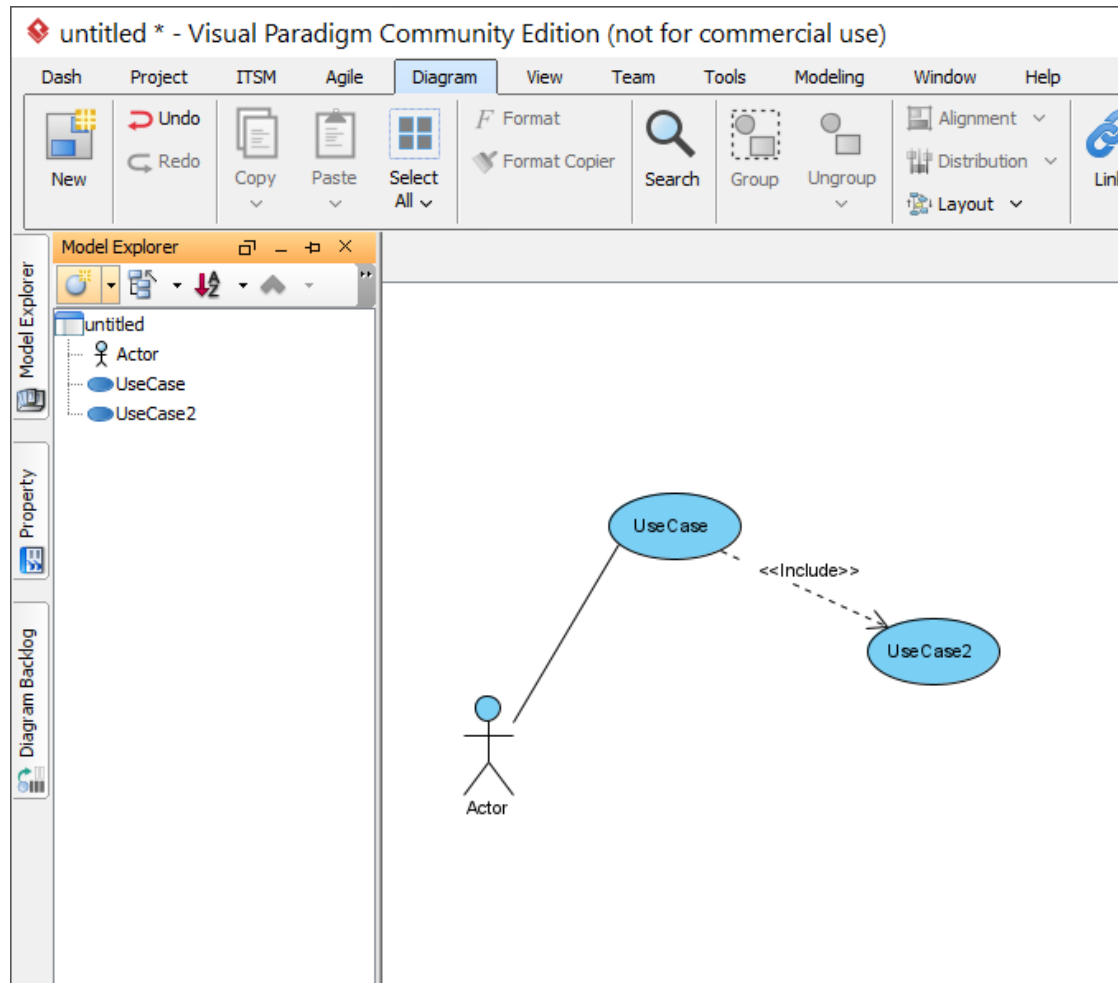
At the bottom of the interface, there are two promotional banners:

- Store and share your design on your cloud workspace**
 Enjoy additional features like web diagrams, SWOT, and more.
- Jump start your design with diagram templates**
 Draw your own diagrams with predefined templates. Hundreds of templates are available. Visit our Community Circle for details.

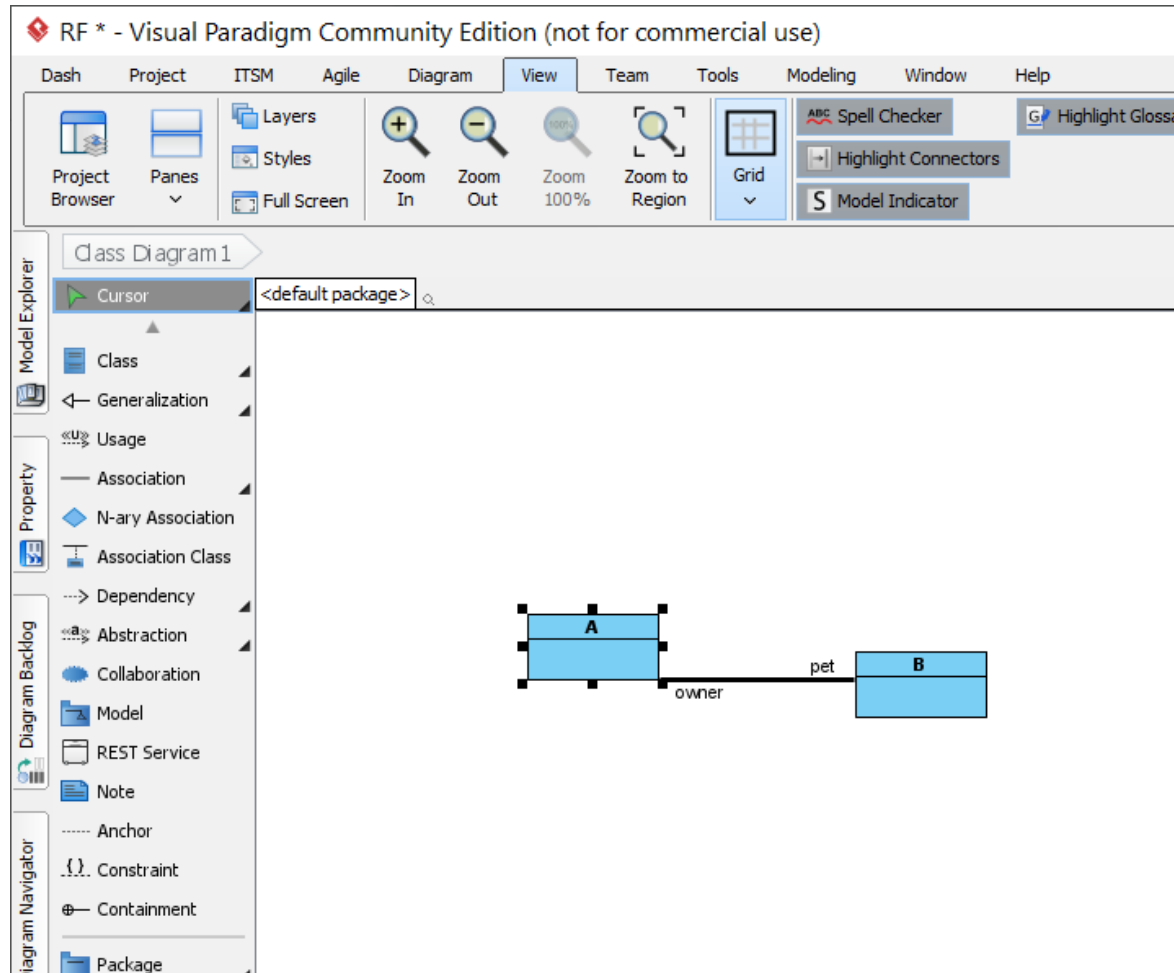
Válasszuk ki a Use case diagramot, majd vegyük fel az Aktorokat, használati eseteket, illetve azok kapcsolatait



A Model Explorer ablakban látható a diagram rajzolás közben automatikusan készülő mögöttes modell



Készítsünk új osztálydiagramot: osztályok, kapcsolatok, attribútumok, opeációk



A Property ablakban sok tulajdonság állítható, pl. az attribútum neve és típusa. A típusok között természetesen kiválasztható a másik osztály (B) is, mivel a mögöttes modell azt típusként nyilvántartja

The screenshot shows the Visual Paradigm Community Edition interface. The main window displays a class diagram with two classes, A and B, connected by an association line labeled 'owner' and 'pet'. The Property window is open, showing the properties of the selected element. The 'Attributes' section is expanded, showing the 'name' attribute with its 'Type' set to 'B'. The 'Type' dropdown menu is open, showing a list of available types: 'short', 'string', 'void', and '<default pack>'. The '<default pack>' section is expanded, showing the classes 'A' and 'B' as available options for the attribute type.

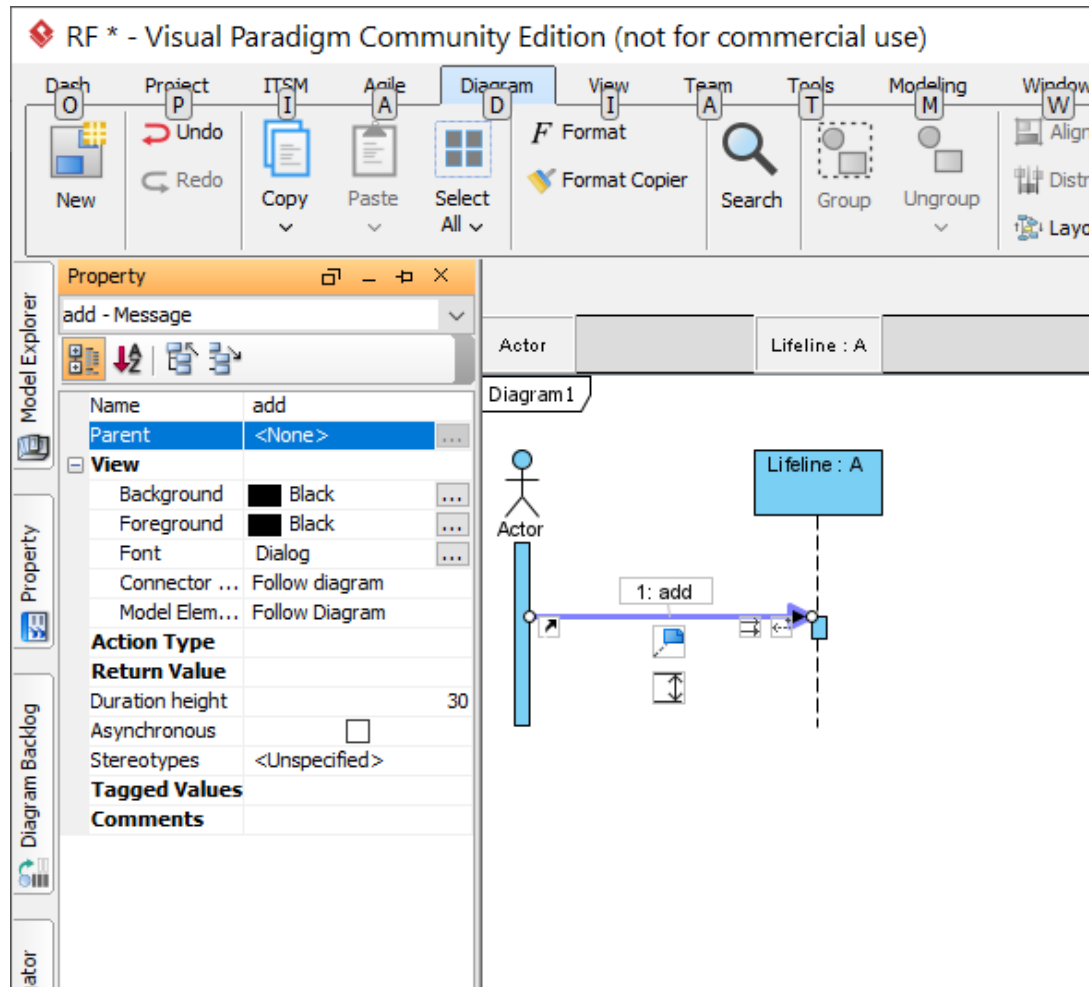
A háttérben a modell tovább bővül...

The screenshot displays the Visual Paradigm Community Edition interface. The top menu bar includes options like Dash, Project, ITSM, Anale, Diagram, View, Team, Tools, Modeling, Window, and Help. The View menu is currently open, showing options such as Zoom In, Zoom Out, Zoom 100%, Zoom to Region, and Grid. The Model Explorer on the left shows a project named 'RF' with a class 'A' containing a property '-name : B'. Below the Model Explorer, a UML class diagram is shown with two classes, 'A' and 'B', connected by a relationship line labeled 'pet'. Class 'A' has a property '-name : B' and class 'B' is empty.

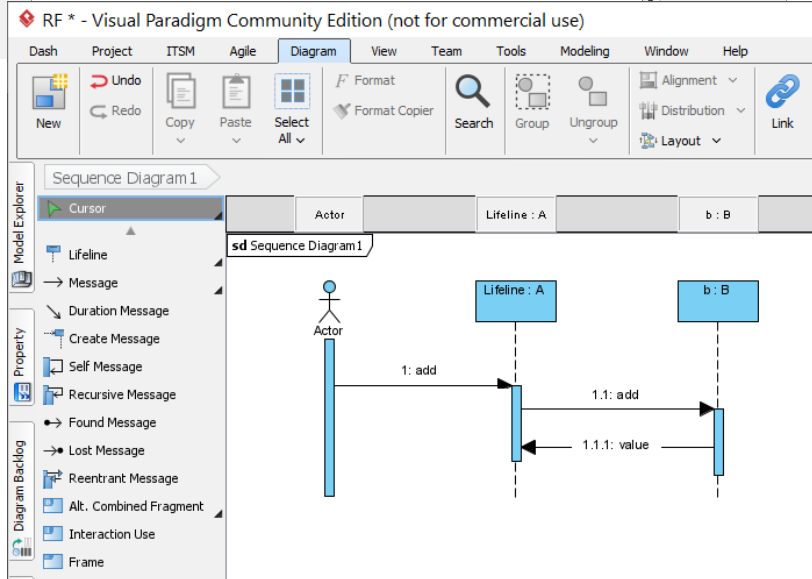
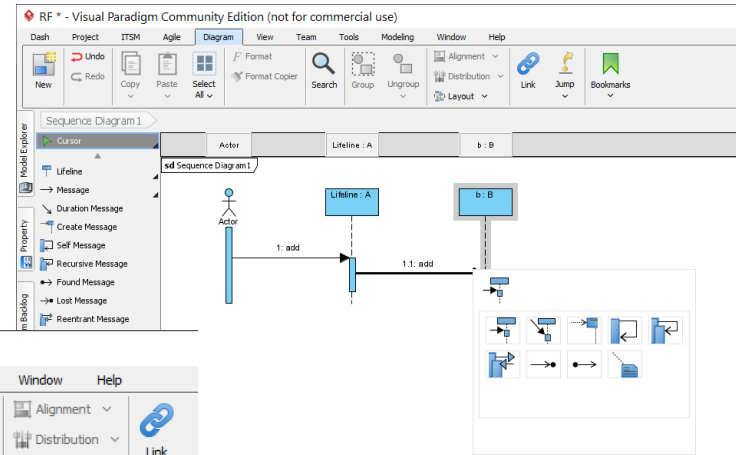
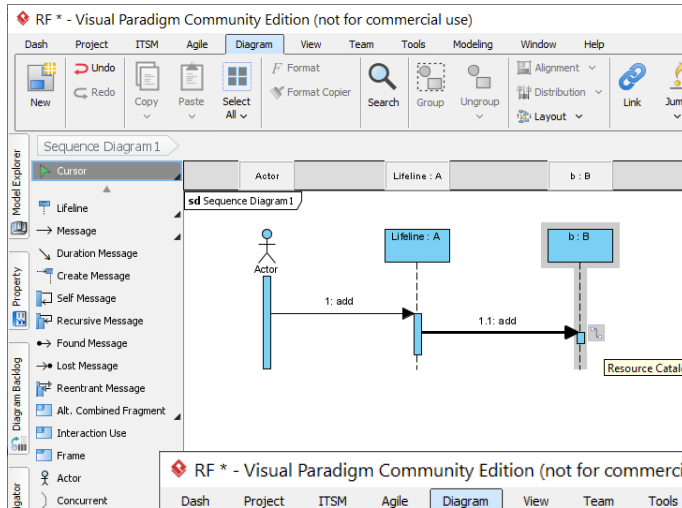
```

classDiagram
    class A {
        -name : B
    }
    class B
    A -- B : pet
    
```

Új Szekvencia diagram készítése: a használati esetről létrehozott Aktor felhasználható, illetve egy idővonal esetén ki tudjuk választani, hogy az melyik meglévő osztály objektumához tartozik. Az üzenetek tulajdonságai is a szokásos módon adhatóak meg.



Az üzenetek további objektumokhoz juthatnak el és vissza is térnek



Ne felejtsük el menteni a projektet. Az egyes diagramok képként is exportálhatók a Diagram Navigator ablakban.

