

Jellyfish UML Modeling Assistant - User Manual

Table of Contents

1. Getting Started
2. System Requirements
3. Application Interface
4. Working with the AI Assistant
5. Supported Diagram Types
6. Diagram Management
7. Version History
8. Rendering and Export Features
9. Interaction Logging
10. Troubleshooting

Getting Started

Opening the Application

1. **Double-click** the UML_Modeling_Assistant.exe file
2. **Wait 15-30 seconds** for the application to start (first run may take longer)
3. Your **default web browser will automatically open** to the application interface
4. **If the application doesn't load in the browser after 30 seconds:** Close the console window and browser tab, then try opening the .exe file again
5. **Keep the console window open** - this runs the backend server
6. To stop the application, **close the console window** or press Ctrl+C

Important: Antivirus False Positive Warning

Your antivirus may flag this application as suspicious. This is a FALSE POSITIVE.

Why this happens:

- The executable bundles a Python interpreter and libraries using PyInstaller
- Self-extracting behavior triggers heuristic detection in antivirus software
- This is a common issue with PyInstaller applications and does not indicate malware

How to resolve:

1. **Add to antivirus exclusions** (recommended solution)
2. **Click "Allow" or "Run anyway"** when prompted
3. **Temporarily disable real-time protection** during use if necessary

This application is safe: It only runs locally on your computer and doesn't access external services except for AI chat functionality.

System Requirements

Mandatory Requirements

- **Windows 11**
- **Modern web browser** (Chrome, Firefox, Edge)
- **Internet connection** for AI assistant functionality

Optional Requirements

- **Java Runtime Environment (JRE 8 or later)** for local diagram rendering
 - **With Java:** Diagrams render locally (faster)
 - **Without Java:** Diagrams render using online PlantUML service (no impact on functionality)

Application Interface

Main Layout

- **Chat Panel :** Conversation with AI assistant
- **Diagram Panel:** Diagram editor and management (toggleable using Open Diagrams button)
- **Header:** Application controls and diagram counter
- **Footer:** Message input and send button

Interface Controls

- **"Open Diagrams" Button:** Toggle the diagram panel on/off (blank if you don't have any diagrams yet)
- **Chat Header:** Message count, download logs, clear all functionality
- **Quick Access:** Recently created diagrams appear as clickable cards (top of the chat)

Working with the AI Assistant

Starting a Conversation

1. Type your UML modeling request (check supported diagrams section) in the message input box
2. Press **Enter** or click the **Send button**
3. The AI will either:
 - Ask clarifying questions
 - Generate a complete diagram

AI Response Types

- **Questions:** The AI may ask for clarification about requirements
- **Diagram Generation:** Creates a working diagram that opens in the editor
- **Suggestions:** Provides guidance on UML modeling best practices (if prompted to clarify a generated model, after that you may ask the AI to implement these improvements)

Creating Diagrams

Option 1: AI-Assisted Creation

1. Describe your system or requirements to the AI
2. Answer any clarifying questions
3. The AI generates a complete diagram
4. The diagram automatically opens in the side panel for further editing

Option 2: Manual Creation

Use the **Quick Start buttons** to create blank diagrams:

- Class Diagram
- Use Case Diagram
- Sequence Diagram
- State Diagram

Supported Diagram Types

1. Class Diagrams

Purpose: Model system structure, classes, and relationships

Supported Features:

- **Class Types:**
 - Regular classes with attributes and methods
 - Abstract classes
 - Enums with enumeration values
- **Attributes & Methods**
 - Visibility modifiers: + public, - private, # protected, ~ package
- **Relationships:**
 - Association, Aggregation, Composition, Inheritance, Dependency, Realization
 - Cardinality (1, 0..1, 0.., 1.., *)
 - Relationship labels
- **Interactive Editing:**
 - Add/edit/delete classes, attributes, methods
 - Change class types
 - Manage relationships with cardinalities

Not Supported:

- Interfaces
- Nested classes
- Method parameter details in editor (can be written in method text)

2. Use Case Diagrams

Purpose: Model system functionality and actors

Supported Features:

- **Elements:**
 - Actors with descriptions
 - Use cases with descriptions
- **Relationships:**
 - Association (actor uses use case)
 - Extends (use case extends another)
 - Includes (use case includes another)
- **Interactive Editing:**
 - Add/edit/delete actors and use cases
 - Manage relationships between elements
 - Edit descriptions

Not Supported:

- Generalization relationships between actors
- Complex use case specifications

3. Sequence Diagrams

Purpose: Model interactions and message flows over time

Supported Features:

- **Participant Types:**
 - Participant, Actor, Boundary, Control, Entity
- **Message Types:**
 - Synchronous
 - Asynchronous
 - Return messages
 - Create/Destroy
- **Advanced Grouping:**
 - **Alt groups** (alternative flows with condition/else)
 - **Loop groups** (repetitive flows with optional count)
 - **Generic groups** (organizational grouping)
 - **Nested groups** (groups within groups for complex workflows)
- **Interactive Editing:**
 - Add/edit/delete participants and messages
 - Reorder messages (up/down arrows)
 - Assign messages to groups
 - Create nested group hierarchies
 - Move messages between groups

Not Supported:

- Combined fragments beyond alt/loop/group
- Time constraints
- Interaction operators
- Lifelines

4. State Diagrams

Purpose: Model object behavior and state transitions

Supported Features:

- **State Types:**
 - Regular states, Start state, End state
 - Fork/Join (concurrent flows)
 - Choice states (decision points with conditions)
 - Composite states (containing sub-states)
- **Advanced Features:**
 - History transitions (shallow and deep history)
 - Sub-state hierarchies (parent-child relationships)
 - Conditional transitions with actions
- **Interactive Editing:**
 - Add/edit/delete states and transitions
 - Change state types
 - Manage parent-child relationships
 - Move states between composite states
 - Edit transition conditions and actions

Not Supported:

- Deep nesting (composite states cannot have composite parents)
- Entry/exit actions on states
- Internal transitions
- Submachine states

Diagram Management

Creating New Diagrams

1. **Via AI:** Describe your requirements to the assistant
2. **Via Quick Start:** Use the diagram type buttons in chat

Selecting and Opening Diagrams

- **From Chat:** Click on diagram cards in the "Your Diagrams" section

- **From Panel:** Click the **three-line button** (≡) next to the X to see all diagrams (this button is only visible if you have more than one diagram)

Editing Diagrams

- **Element Management:** Add, edit, delete elements using the editor controls
- **Properties:** Edit names, types, descriptions, and relationships
- **Reordering:** Use up/down arrows for messages
- **Type Changes:** Convert between different element types

Deleting Diagrams

1. Open the diagram in the editor
2. Click the **"Delete Diagram"** button in the footer
3. Confirm deletion in the dialog
4. **Warning:** This action cannot be undone and removes all diagram data and history

Version History

Version history becomes available when you have multiple diagrams created. To browse all your diagrams and their versions, click the **three-line button** (≡) in the diagram editor panel. This opens the diagram list where you can see all your diagrams organized by type, along with version information for each.

How Versioning Works

- **AI-generated diagrams:** Create new versions automatically
- **User edits:** Update the **current** version
- **Version naming:** First version keeps base name, subsequent versions add "v2", "v3", etc.

Viewing Version Information

- **Version badges:** Show current version (e.g., "v2 of 3")
- **Creation indicator:** "AI-generated" badge for LLM-created versions

Version Management

- **Automatic:** New AI clarifications create new versions
- **Manual:** User edits modify **current** version

Rendering and Export Features

Diagram Rendering

1. **Open any diagram** in the editor
2. Click "**Render Diagram**" button
3. **Wait for generation** (may take a few seconds)
4. **View the visual diagram** below the controls

Rendering Options

- **Local Rendering (with Java):** Faster
- **Online Rendering (without Java):** Uses PlantUML web service

Export Options

Download Image

1. **Render the diagram** first
2. Click "**Download Image**" button
3. **PNG file** will be downloaded to your default downloads folder

Copy PlantUML Code

1. Click "**Copy Code**" button
2. **PlantUML source code** is copied to clipboard
3. **Paste into any PlantUML editor** or documentation

View/Hide Raw Code

1. Click "**View Code**" button
2. **Toggle visibility** of the PlantUML source code

Interaction Logging

What is Logged

The application automatically logs all user interactions:

- **Chat conversations** with the AI assistant
- **Model editing actions** (adding, modifying, deleting elements)
- **Workflow events** (diagram generation, type changes, clarification requests)

Log Storage

- **Local files:** Saved in a logs/ folder next to the executable
- **File naming:** interactions_YYYY-MM-DD.csv

Downloading Logs

1. Click "**Download Logs**" button in the chat header
2. **CSV file** downloads automatically
3. **Contains all session data** from current day or most recent

Additional Features

Clear Chat Functionality

1. Click "**Clear All**" button in chat header
2. **Confirm deletion** in the dialog
3. **Removes:**
 - All chat messages
 - All diagrams and their history
 - Resets conversation context
4. **Warning:** This action cannot be undone

Clarify Diagram Functionality

1. **Generate a diagram** with the AI assistant
2. Click "**Improve this diagram**" button (lightbulb icon)
3. **AI analyzes** the current diagram
4. **Provides specific suggestions** for improvement

5. **New version created** if you ask the AI to implement suggestions

Troubleshooting

Application Won't Start

- **Antivirus blocking:** Add to exclusions or click "Allow"
- **Port conflicts:** Close other applications using ports 3000 or 8000
- **Firewall warning:** Click "Allow" - the app only runs locally

Browser Doesn't Open

- **Manual access:** Go to <http://127.0.0.1:3000> in your browser

Diagram Rendering Issues

- **Java not installed:** Install Java or use online rendering
- **Java not in PATH:** Restart application after Java installation
- **PlantUML errors:** Check diagram content or use "View Code" to debug

AI Assistant Issues

- **Connection errors:** Check internet connection
- **Slow responses:** Wait patiently, complex diagrams take time

Log File Issues

- **Logs not saving:** Check application has write permissions
- **Download fails:** Logs are also stored in logs/ folder manually (if the application has write permissions)

Common Error Messages

- **"Backend unreachable":** Backend server hasn't started yet, wait a few seconds
- **"Permission denied":** Run as administrator or check antivirus