



NexAIoT Co., Ltd.

# NexUA Model Maker

## User Manual

NexAIoT Co., Ltd.

Version: v1.00

Published February 2020

[www.nexaiot.com](http://www.nexaiot.com)

# CONTENTS

<b>Preface.....</b>	<b>1</b>
Disclaimer .....	1
Acknowledgements .....	1
Revision History.....	1
 <b>Chapter 1: Using the NexUA Model Maker .....</b>	<b>2</b>
1.1 Introduction.....	2
1.2 Installing the NexUA Server .....	3
1.3 Basic Functions .....	7
1.4 Instance Declaration of Each Node Class .....	15
 <b>Chapter 2: NexUA Model Maker Implementation..</b>	<b>21</b>
2.1 Launching the Application .....	21
2.2 Create and Configure a Project .....	22
2.3 Apply the Project to the NexUA Server .....	27

# PREFACE

## Disclaimer

The information in this document is subject to change without prior notice and does not represent commitment from NexAloT Co., Ltd. However, users may update their knowledge of any product in use by constantly checking its manual posted on our website: <https://www.nexaiot.com>. NexAloT shall not be liable for direct, indirect, special, incidental, or consequential damages arising out of the use of any product, nor for any infringements upon the rights of third parties, which may result from such use. Any implied warranties of merchantability or fitness for any particular purpose is also disclaimed.

## Acknowledgements

The NexUA Model Maker is a trademark of NexAloT Co., Ltd. All other product names mentioned herein are registered trademarks of their respective owners.

## Revision History

Version	Date	Description
v1.00	January 2020	Initial release

# CHAPTER 1: USING THE NEXUA MODEL MAKER

## 1.1 Introduction

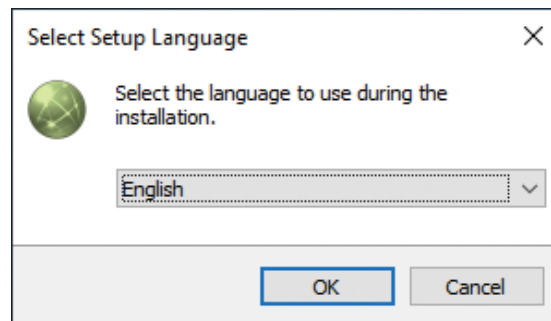
NexUA Model Maker is a graphical user interface (GUI) tool that can design the address space (model) for the OPC UA server. The model is represented by nodes, attributes and their mutual relationships. By activating the software to the standard version, it can help you to configure the numerous nodes in one model and export the model to the XML file at once. The XML file format is only compatible with NexAloT NexUA Server.

Check the operating system requirement before installing the NexUA Model Maker. The following are the supported operating systems:

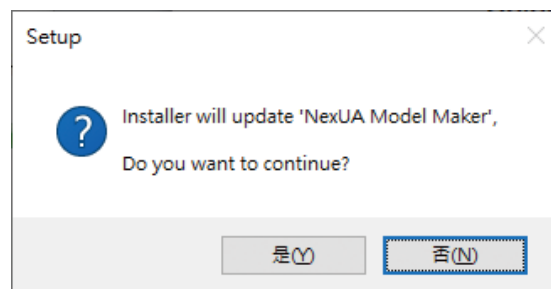
- Microsoft Windows 10
- Windows 7 with Service Pack 1

## 1.2 Installing the NexUA Server

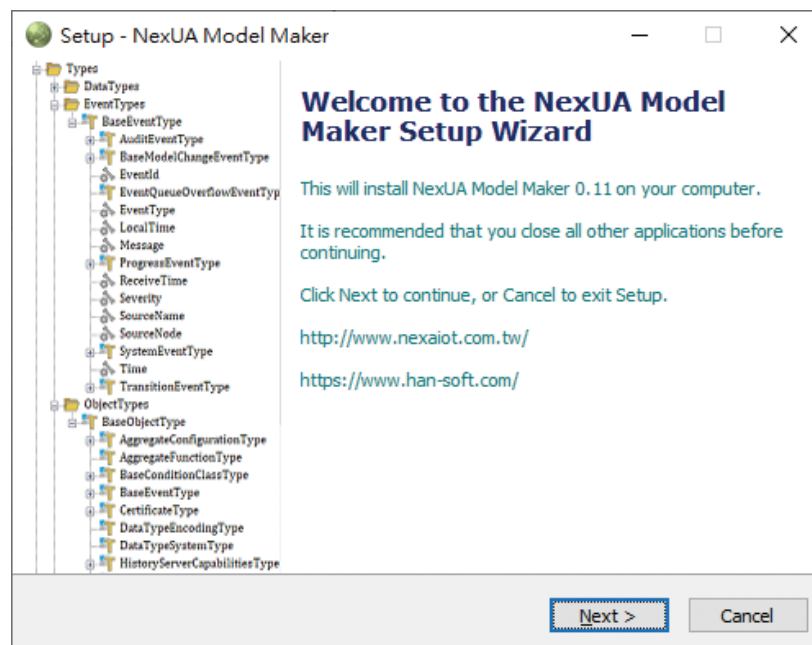
1. Double click on the filename of the NexUA Model Maker setup file, select the language for the installation, and click **OK**.



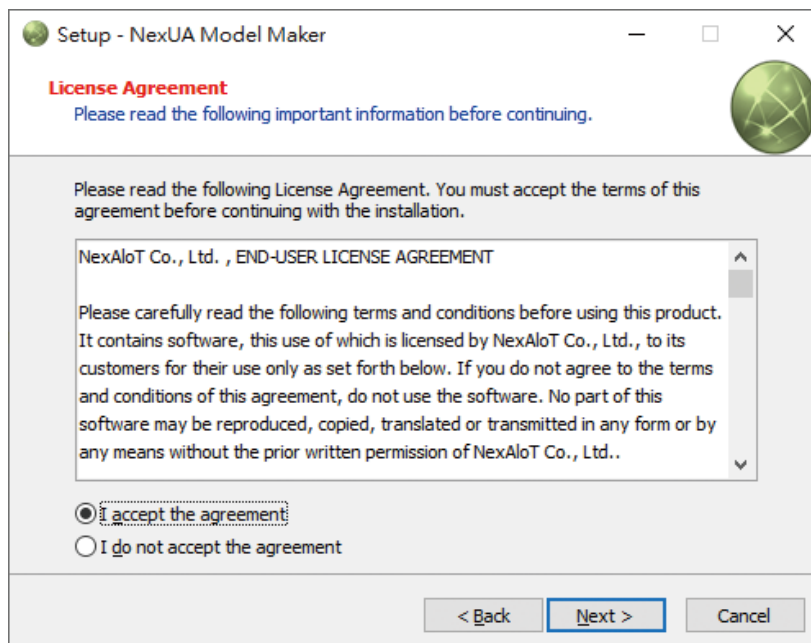
Click **Yes** to continue the installation or **No** to exit the process.



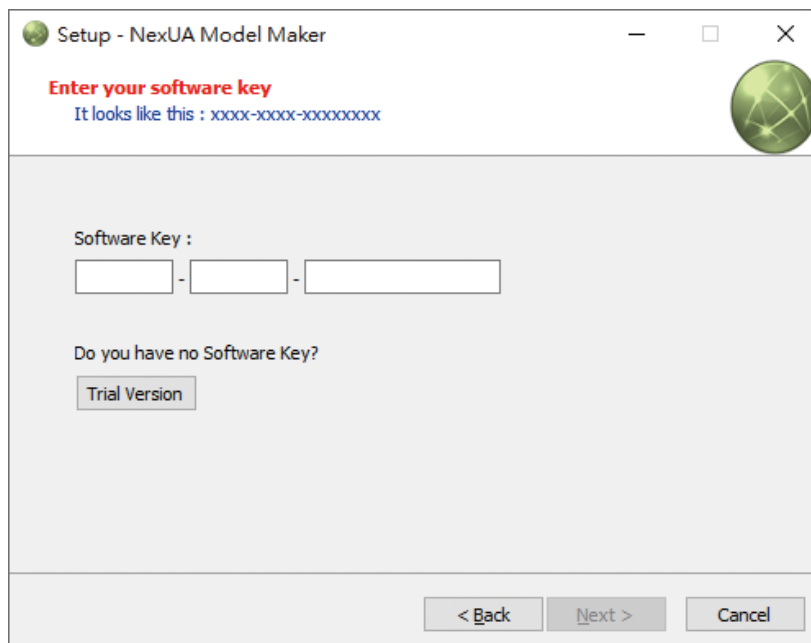
2. Click **Next** to continue, or **Cancel** to exit the setup.



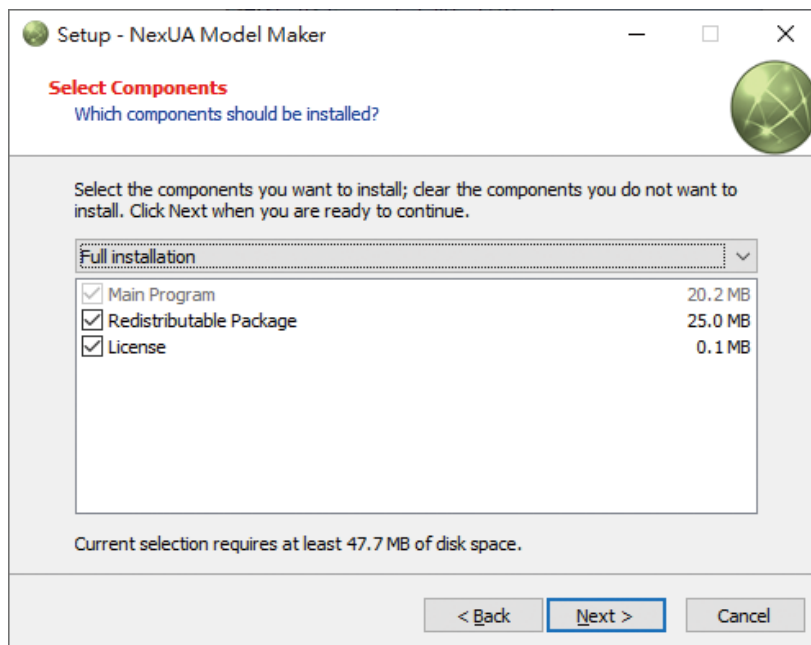
3. Check the **I accept the agreement** option, and click **Next** to proceed.



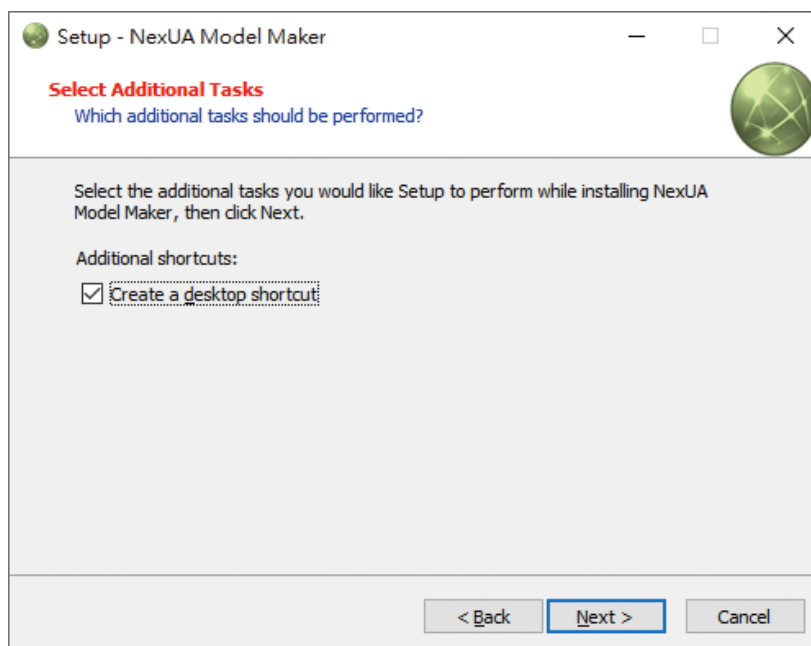
4. Enter the **Software Key** and click **Next** or click the **Trial Version** button if you don't have one.



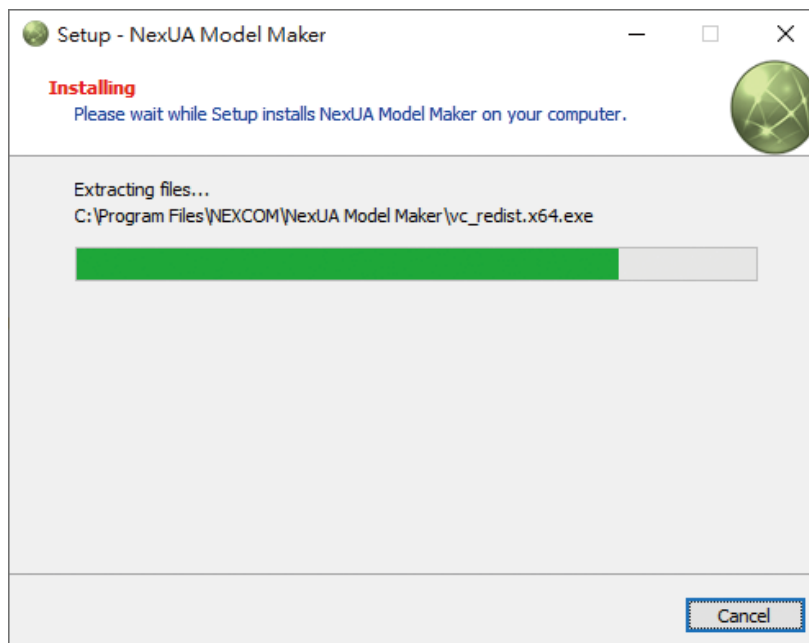
5. Select the components you would like to install, and click **Next**.



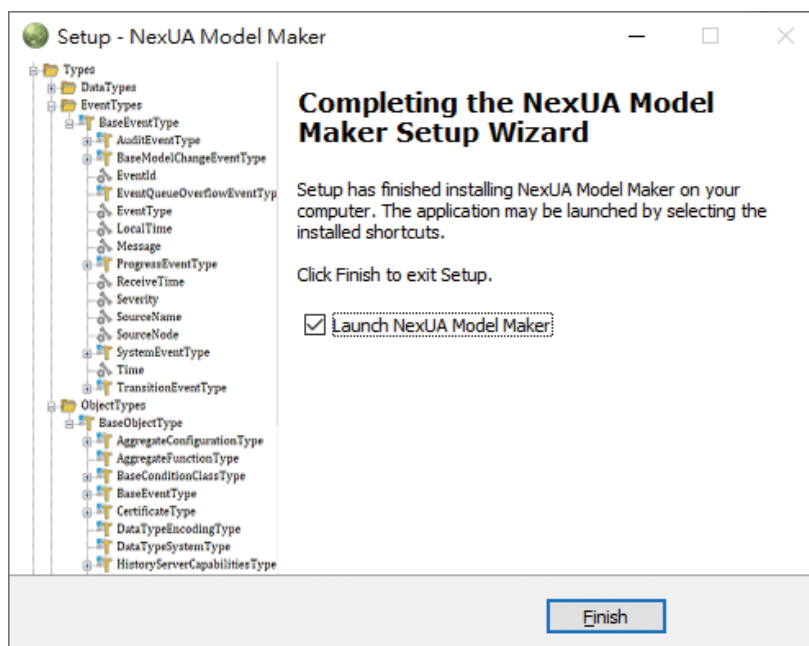
6. Click **Next** to begin installation.



The installation will begin and may take a while.



7. Once completed, click **Finish** to exit the installation wizard.

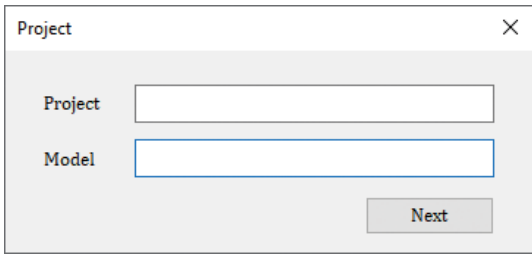
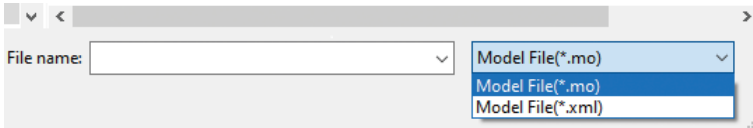


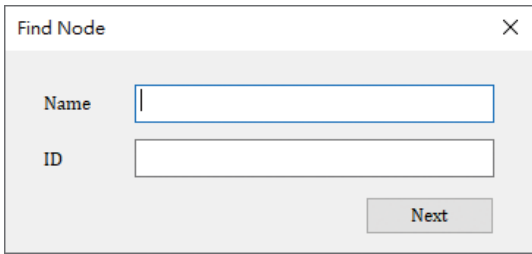
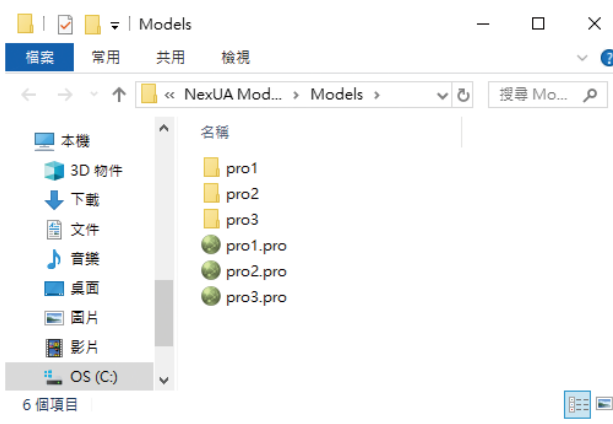
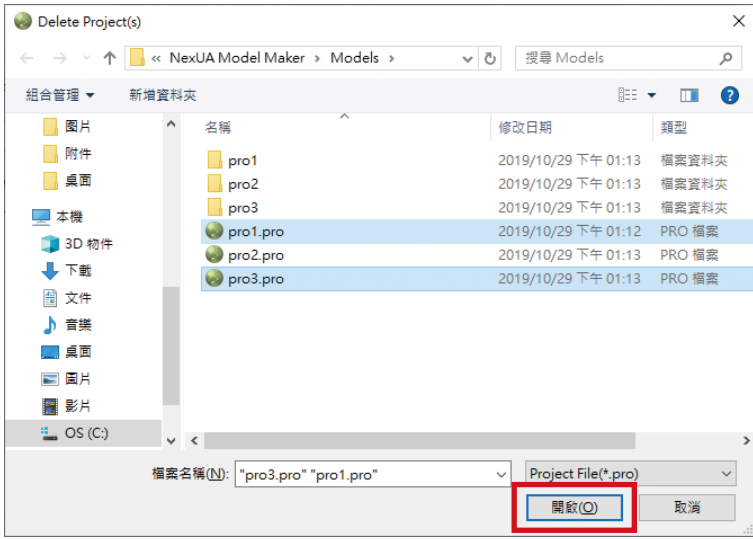


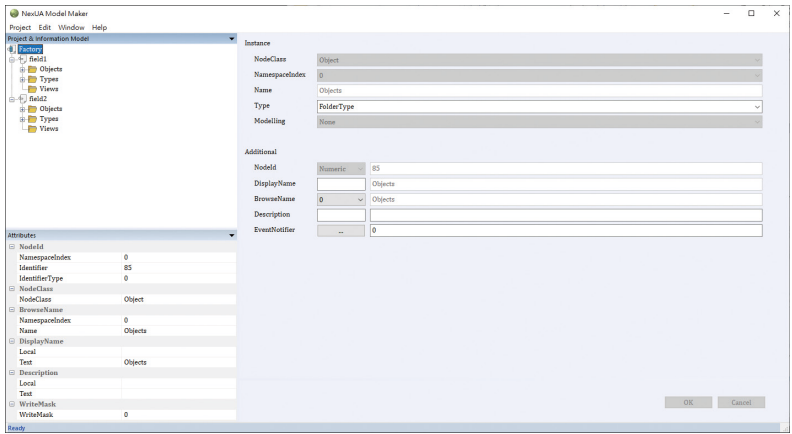
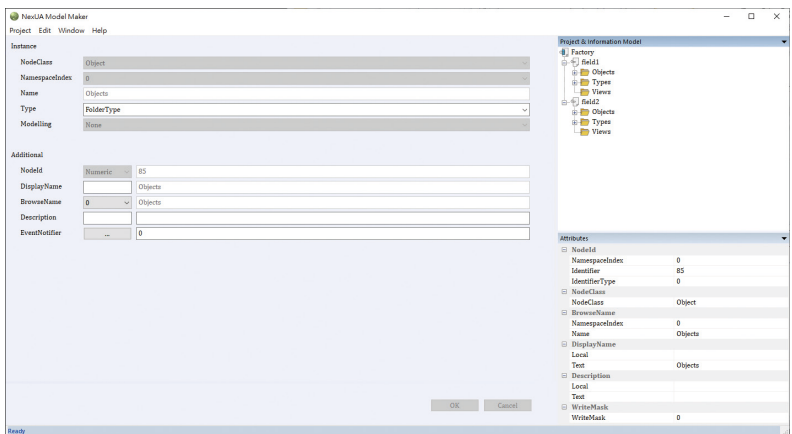
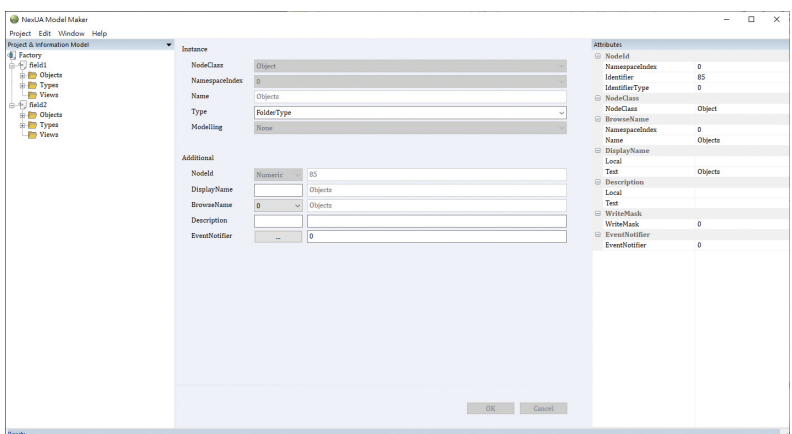
## 1.3 Basic Functions

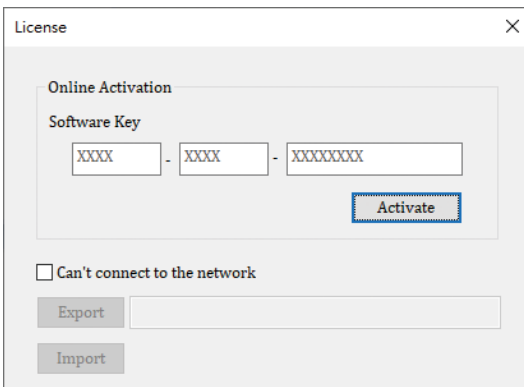
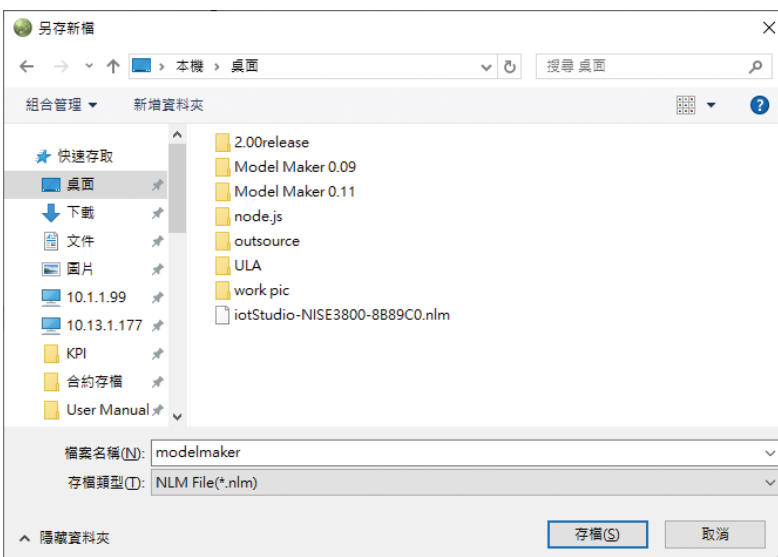
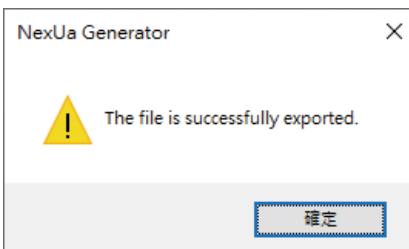
In this section you can find all the available functions on the UI, and a brief explanation of their purposes. For examples on how to use the application to create a project, please refer to **Chapter 2**.

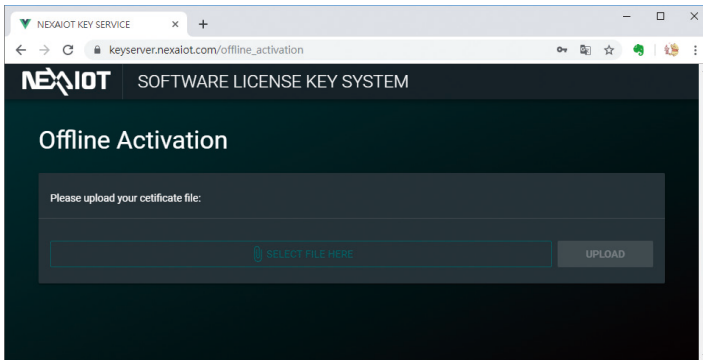
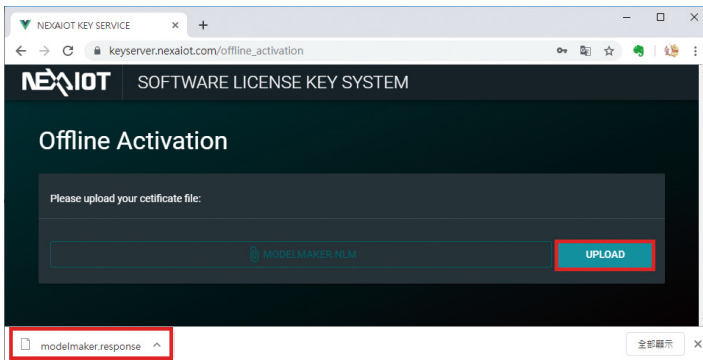
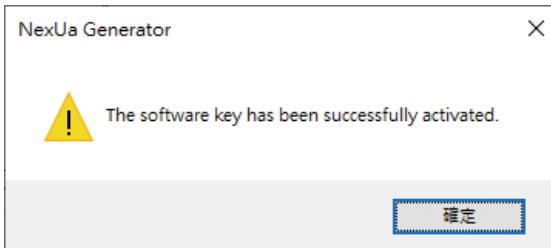
### 1. Main menu function list.

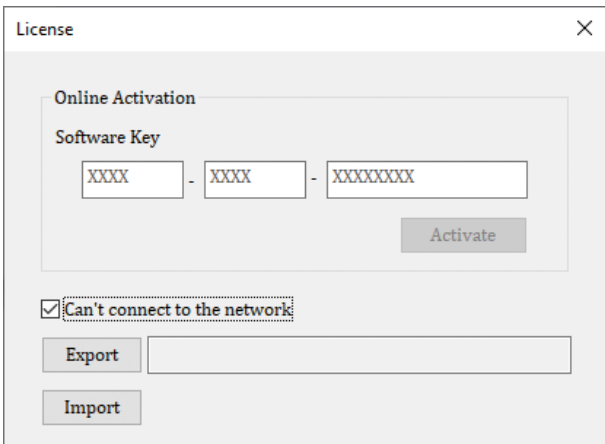
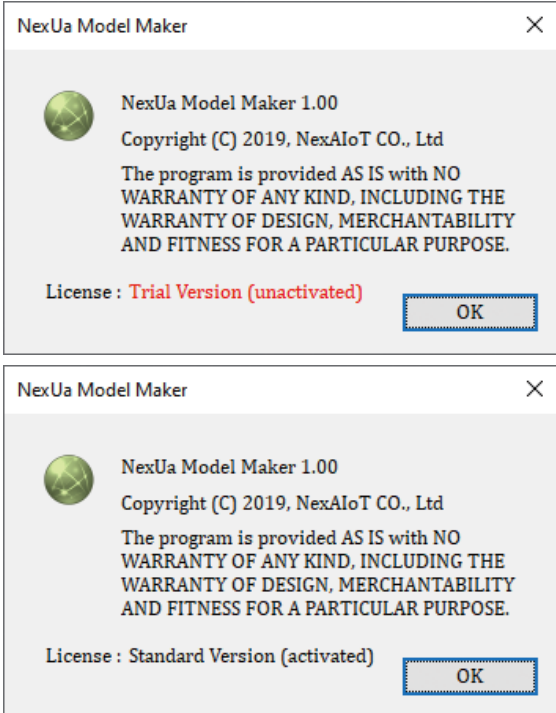
Folder	Function	Description
Project	New Project	<p>Create a new project. Select this option when you want to create a brand-new project.</p>  <p>After clicking the <b>Next</b> button, two files will be created, which are <b>.pro</b> and <b>.mo</b> files.</p>
	Open Project	Open an existing project.
	Save Project	Save the current project. Select this option to <b>save any changes</b> made to the current opened project.
	Save as	Save the current project to another one with a different name.
	New Model	Create a new model in the current project.
	Import Model	<p>Import an existing model to the current project. You can choose to import a <b>.mo</b> or <b>.xml</b> file.</p> 
	Quit	Quit the application <b>WITHOUT</b> saving.

Folder	Function	Description
Edit	Export All XMLs	Export all the models of the current project to XML files.
	Find Node	<p>Find the specified node under the current project.</p>  <p>Enter the <b>Name</b> or <b>ID</b> of the node and then click <b>Next</b>. You'll be navigated to the target.</p>
	Open Project Directory	<p>Open the file directory of the projects to check the save path of the XML files.</p> 
	Delete Project(s)	<p>Delete the selected project(s).</p>  <p>Choose the projects (.pro) you would like to delete and click the button below.</p>

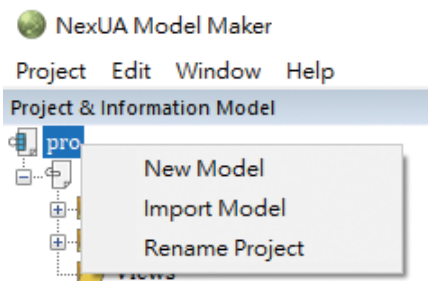
Folder	Function	Description
Window	Stack L	<p>Default layout</p> 
	Stack R	
	Horizontal	

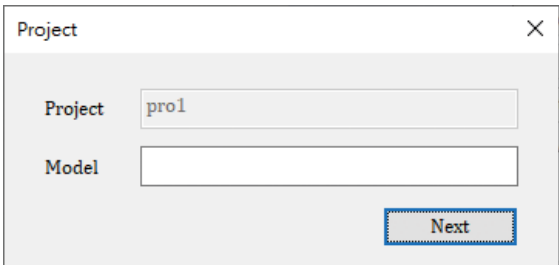
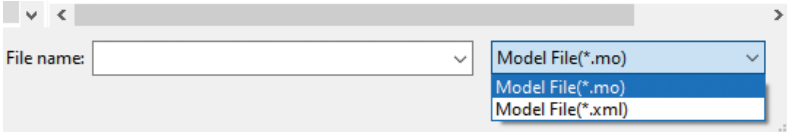
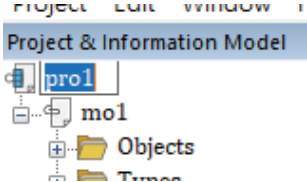
Folder	Function	Description
Help	License	<p>Software Key Online/Offline Activation</p>  <ul style="list-style-type: none"> <li>Online Activation Please enter the software key and click <b>Activate</b>.</li> <li>Offline Activation This mechanism is designed for environments where the devices are not connected to the Internet.</li> </ul> <ol style="list-style-type: none"> <li>Enter the software key.</li> <li>Check the <b>Can't connect to the network</b> checkbox.</li> <li>Click <b>Export</b> to save the key file (.nlm).</li> </ol>
		 

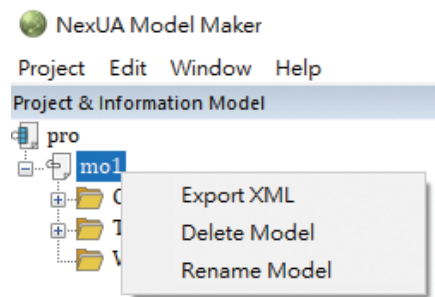
Folder	Function	Description
Help	License	<p>4. Go to NexAloT Software License Key System:  <a href="https://keyserver.nexaiot.com/offline_activation">https://keyserver.nexaiot.com/offline_activation</a></p>  <p>5. Click <b>SELECT FILE HERE</b> to upload the key file you've just exported. Then click <b>UPLOAD</b> to download the activated key file (.response).</p>  <p>6. Go back to the utility and click <b>Import</b> to put the file you've just downloaded on the Key Server. You'll see the below message box when the activation succeeds.</p>  <p>Please note that the License dialog will no longer appear as soon as the software license has been activated.</p>

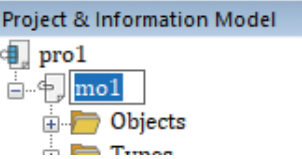
Folder	Function	Description
	License	<p><b>Note:</b> “Can’t connect to the network” will be forcibly checked and the “Activate” button will be unable to click when the host machine cannot connect to the Key Server.</p> 
Help	About	<p>Software, license and copyright information.</p> 

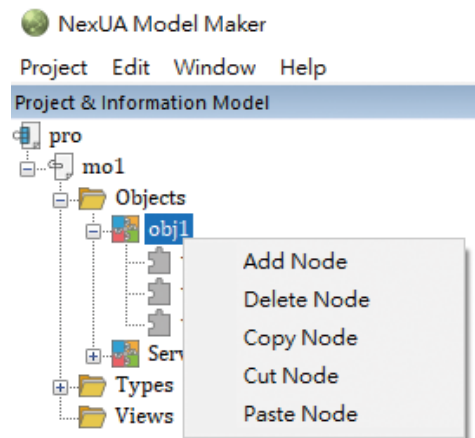
## 2. Pop-up menu list under **pro** (project):



Function	Description
New Model	<p>Create a new model in the current project.</p> 
Import Model	<p>Import an existing model to the current project. You can choose to import a <b>.mo</b> or <b>.xml</b> file.</p> 
Rename Project	<p>Rename the selected project.</p> 

3. Pop-up menu list under **mo1** (model):

Function	Description
<b>Delete Model</b>	Delete the selected model.
<b>Export XML</b>	Export the XML file of the selected model.
<b>Rename Model</b>	Rename the selected model. 

4. Pop-up menu list under **obj1** (node):

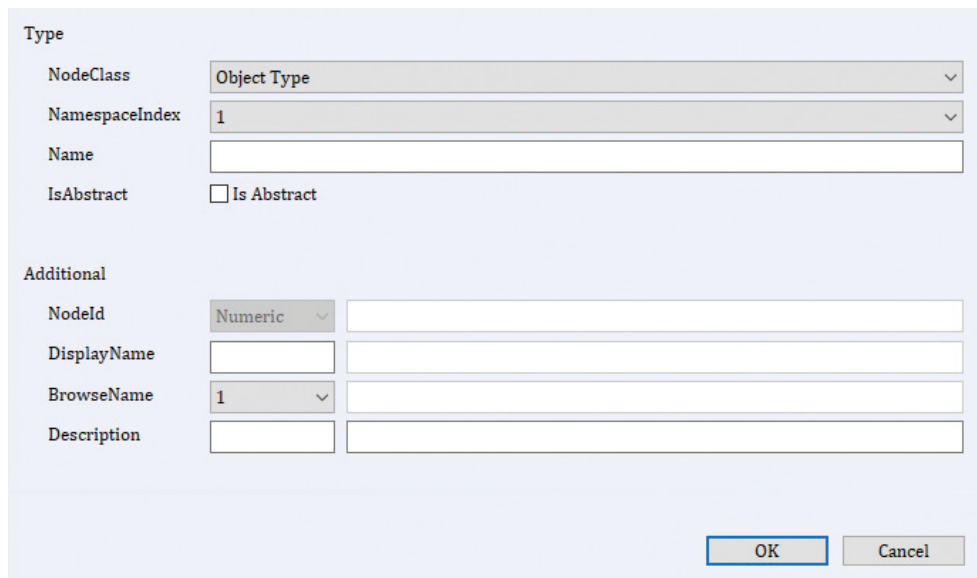
Function	Description
<b>Add Node</b>	Add a child node under the selected node.
<b>Delete Node</b>	Delete the selected node.
<b>Copy Node</b>	Copy the selected node.
<b>Cut Node</b>	Cut the selected node.
<b>Paste Node</b>	Paste the node that had been copied or cut earlier.

**Note:** Copy, Cut and Paste Node options can only operate within the same model.



## 1.4 Instance Declaration of Each Node Class

### 1. Instance declarations of ObjectType.



**Type**

NodeClass: Object Type

NamespaceIndex: 1

Name:

IsAbstract: ☐ Is Abstract

**Additional**

NodeId: Numeric

DisplayName:

BrowseName: 1

Description:

OK Cancel

Attribute	Description
<b>NamespaceIndex</b>	This index is the index of the namespace in the local Server's NamespaceArray. The client may read the NamespaceArray Variable to access the string value of the namespace.
<b>IsAbstract</b>	<p>This attribute indicates whether the ObjectType is concrete or abstract and therefore cannot directly be used as type definition. It's mainly used in Number and abstract sub-types of Number. Server may choose to use an abstract DataType if the concrete DataType for a Variable value is not known or even varies. A Boolean Attribute can have one of the following values:</p> <ul style="list-style-type: none"> <li>• <b>TRUE</b>: This is an abstract ObjectType. For example, no Objects of this type shall exist, only Objects of its subtypes.</li> <li>• <b>FALSE</b>: This is not an abstract ObjectType. For example, Objects of this type can exist.</li> </ul>
<b>NodeId</b>	The nodes in the server are identified by a constructed identifier called the NodeId. NodeId can uniquely identify a Node within a system, or across systems (e.g. GUIDs). There are three IdentifierTypes of NodeId and only string type can be defined by users.

Attribute	Description
<b>DisplayName</b> <b>BrowseName</b>	<p><b>BrowseName</b> Attribute is used as a non-localized human-readable name when browsing the AddressSpace to create paths out of BrowseNames.</p> <p><b>DisplayName</b> Attribute contains the localized name of the Node. Clients should use this Attribute if they want to display the name of the Node to the user.</p> <p>Both of the two attributes are mandatory, and in order to reduce unnecessary errors, the application automatically generates these two names according to the <b>Name</b> for users.</p>
<b>Description</b>	Optional field to describe the application or the meaning of the node.

## 2. Instance declarations of Object Node.

Instance

NodeClass

Object

NamespaceIndex

1

Name

Type

Modelling

None

Additional

NodeId

Numeric

DisplayName

BrowseName

1

Description

EventNotifier

...

0

OK

Cancel

Attribute	Description
<b>Type</b>	It is known as the ObjectType NodeClass. It provides definitions for Objects.
<b>Modelling</b>	It's known as Modelling Rule, only available when adding the node under the object type. You can define the node's modelling rule as <b>Optional</b> or <b>Mandatory</b> . The instances of the type can selectively ( <b>Optional</b> ) or forcibly ( <b>Mandatory</b> ) have a child with the same name as the instance declaration.
<b>NodeId</b>	Same description as the one in 1. Instance declarations of ObjectType.
<b>DisplayName</b> <b>BrowseName</b>	Same description as the one in 1. Instance declarations of ObjectType.
<b>Description</b>	Optional field to describe the application or the meaning of the node.

### 3. Instance declarations of Variable Node.

Instance

NodeClass

Variable

NamespaceIndex

1

Name

Humidity

Type

BaseVariableType

Modelling

Mandatory

Data Type

Double

Additional

NodeId

Numeric

6001

DisplayName

Humidity

BrowseName

1

Humidity

Description

AccessLevel

...

3

Alarm

☐ Has Alarm

Historizing

☐ Is Historizing

ValueRank

Scalar

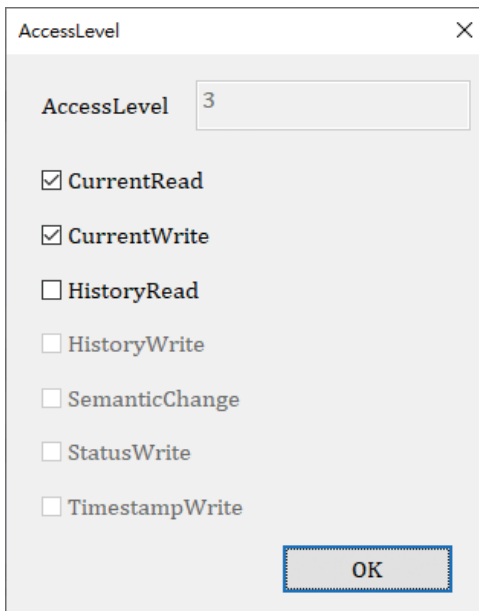
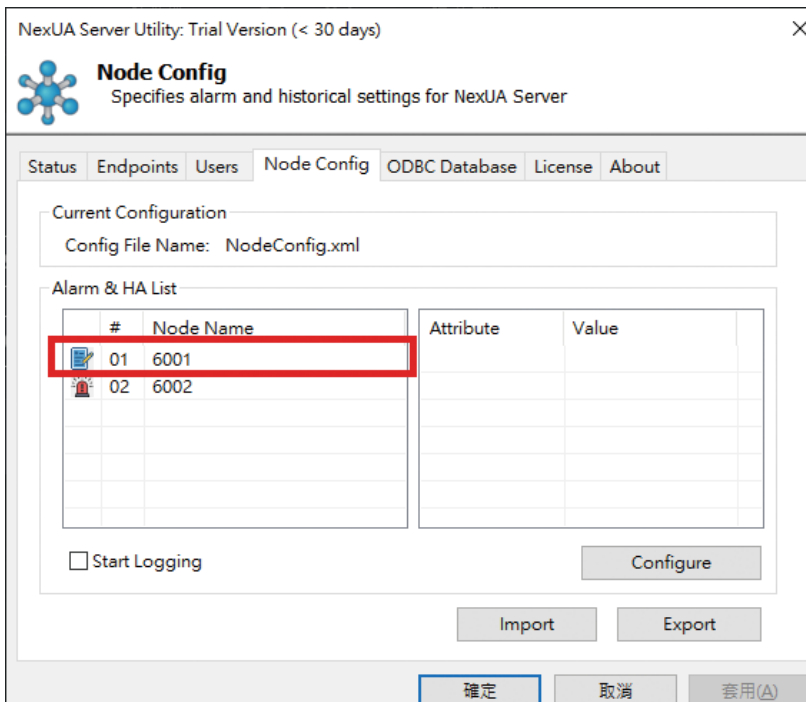
Value

0.000000

OK

Cancel

Attribute	Description
<b>Type</b>	It is known as the VariableType NodeClass. It provides definitions for Objects.
<b>Modelling</b>	Same description as the one in 2. Instance declarations of Object Node.
<b>Data Type</b>	Data types are used to describe the structure of the Value Attribute of Variables and their VariableTypes.
<b>NodeId</b>	Same description as the one in 1. Instance declarations of ObjectType.
<b>DisplayName</b> <b>BrowseName</b>	Same description as the one in 1. Instance declarations of ObjectType.
<b>Description</b>	Optional field to describe the application or the meaning of the node.

Attribute	Description												
AccessLevel	<div data-bbox="602 283 1078 890">  <p>AccessLevel dialog box showing the following options:</p> <ul style="list-style-type: none"> <li>AccessLevel: 3</li> <li><input checked="" type="checkbox"/> CurrentRead</li> <li><input checked="" type="checkbox"/> CurrentWrite</li> <li><input type="checkbox"/> HistoryRead</li> <li><input type="checkbox"/> HistoryWrite</li> <li><input type="checkbox"/> SemanticChange</li> <li><input type="checkbox"/> StatusWrite</li> <li><input type="checkbox"/> TimestampWrite</li> <li>OK button</li> </ul> </div> <p>Select the <b>HistoryRead</b> option to enable the “Historical Data” property in the NexUA Server, as shown in the red frame below.</p> <div data-bbox="602 1008 1403 1705">  <p>NexUA Server Utility: Trial Version (&lt; 30 days)</p> <p><b>Node Config</b> Specifies alarm and historical settings for NexUA Server</p> <p>Current Configuration Config File Name: NodeConfig.xml</p> <p>Alarm &amp; HA List</p> <table border="1"> <thead> <tr> <th>#</th> <th>Node Name</th> <th>Attribute</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>01</td> <td>6001</td> <td></td> <td></td> </tr> <tr> <td>02</td> <td>6002</td> <td></td> <td></td> </tr> </tbody> </table> <p><input type="checkbox"/> Start Logging</p> <p>Buttons: Configure, Import, Export, 確定 (red frame), 取消, 套用(A)</p> </div>	#	Node Name	Attribute	Value	01	6001			02	6002		
#	Node Name	Attribute	Value										
01	6001												
02	6002												

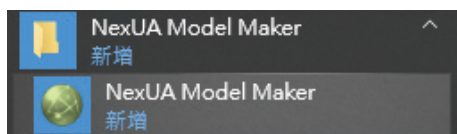
Attribute	Description												
Alarm	<div><div>Alarm</div><div><div>HighHigh</div><div>High</div><div>Low</div><div>LowLow</div></div><div>OK</div></div> <p>Check the <b>Alarm</b> checkbox and enter the alarm value to enable the “Alarm Event” property in the NexUA Server, as shown in the red frame below.</p> <div><div>NexUA Server Utility: Trial Version (&lt; 30 days)</div><div><div>Node Config</div><div>Specifies alarm and historical settings for NexUA Server</div></div><div><div>Status</div><div>Endpoints</div><div>Users</div><div>Node Config</div><div>ODBC Database</div><div>License</div><div>About</div></div><div><div>Current Configuration</div><div>Config File Name: NodeConfig.xml</div></div><div><div>Alarm &amp; HA List</div><div><table><thead><tr><th>#</th><th>Node Name</th><th>Attribute</th><th>Value</th></tr></thead><tbody><tr><td>01</td><td>6001</td><td></td><td></td></tr><tr><td>02</td><td>6002</td><td></td><td></td></tr></tbody></table></div></div><div><div><input checked="" type="checkbox"/> Start Logging</div><div>Configure</div></div><div><div>Import</div><div>Export</div></div><div><div>確定</div><div>取消</div><div>套用(A)</div></div></div>	#	Node Name	Attribute	Value	01	6001			02	6002		
	#	Node Name	Attribute	Value									
	01	6001											
	02	6002											
ValueRank	This attribute indicates whether the Value Attribute of the VariableType is an array and how many dimensions the array has.												
Value	The default Value for instances of this type.												

## CHAPTER 2: NEXUA MODEL MAKER IMPLEMENTATION

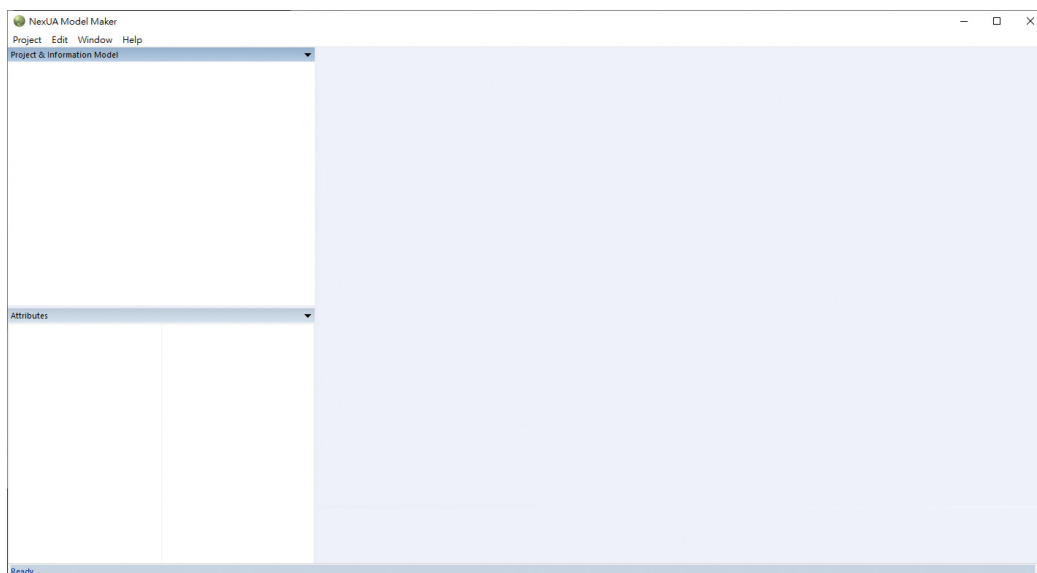
In this chapter, we will walk you through on how to create a project from the application and import the model of the project to the NexUA Server.

### 2.1 Launching the Application

Once the application is installed, you should be able to find the program named **NexUA Model Maker** in the **All Programs** list as shown below.

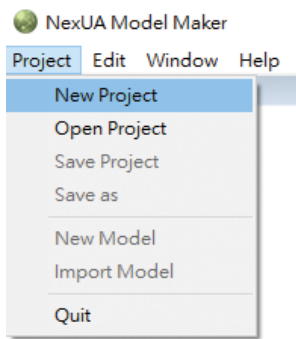
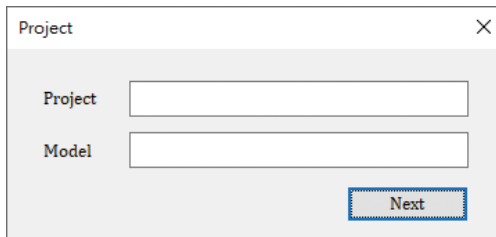
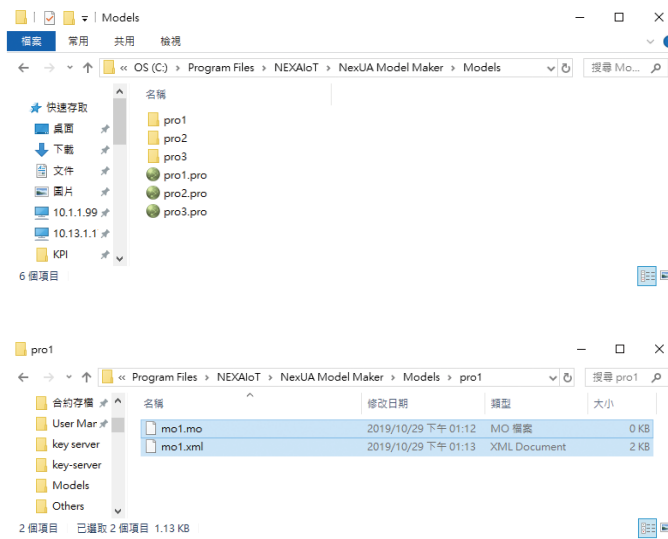


The GUI of the NexUA Server will be displayed on the screen.



## 2.2 Create and Configure a Project

1. Create a project and model.

Step	Description	Screenshot
1-1	Click <b>New Project</b> in the <b>Project</b> menu.	
1-2	Name the project and model. Please follow the file naming conventions.	
1-3	<p>As soon as the project is created, the associated files will be created in <a href="C:\Program Files\NEXAIoT\NexUA Model Maker\Models">C:\Program Files\NEXAIoT\NexUA Model Maker\Models</a></p> <p>The project's configuration file (.pro) will be saved in the root directory. Each project folder will have its model file (.mo) and the corresponding XML file.</p>	 <p><b>Note:</b> Unpredictable results may occur if you delete or change the name of the model's folder and its included files directly on the File Explorer, or use any other compilation software to modify the file content.</p>



- Design the address space according to the requirement – create the user-defined ObjectType.

Create the user-defined ObjectType		
Step	Description	Screenshot
2-1	Navigate to the <b>Type/ObjectType</b> folder and right-click the <b>BaseObjectType</b> and click <b>Add Node</b> .	
2-2	Define the node class as <b>Object Type</b> , then specify a name for it and click <b>OK</b> .	
2-3	Right-click the node you just created and click <b>Add Node</b> .	

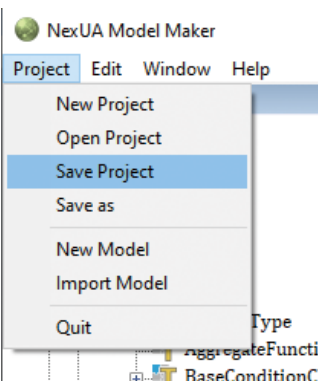
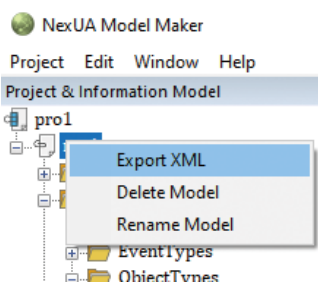
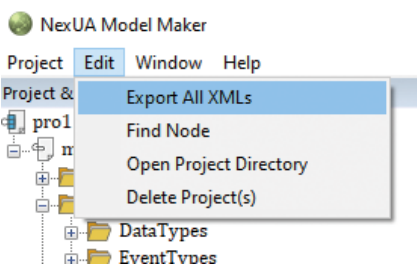
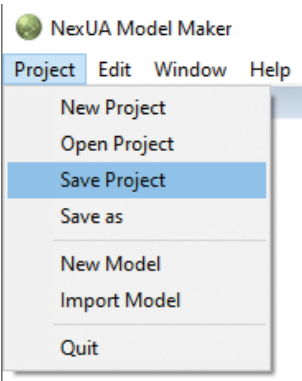
Create the user-defined ObjectType		
Step	Description	Screenshot
2-4	Configure the node and click <b>OK</b> .	

- Design the address space according to the requirement – create nodes under the address space.

Create nodes under the address space		
Step	Description	Screenshot
3-1	Right-click the <b>Objects</b> folder or the existing <b>object/variable</b> node and click <b>Add Node</b> .	<ul style="list-style-type: none"> <li>Add nodes under the Object folder:            </li> <li>Add nodes under the existing node:            </li> </ul>

Create nodes under the address space		
Step	Description	Screenshot
3-2	<p>Choose the NodeClass you'd like to add.</p> <p>For Object Type, you can choose the self-defined type or the native type from the server.</p> <p>Configure the other attributes and click <b>OK</b>.</p>	
3-3	<p>Repeat the first two steps.</p> <p>Tip: You can flexibly use the copy &amp; paste function to save time.</p>	

## 4. Save the project and export the model to XML file.

Step	Description	Screenshot
4-1	Click <b>Save Project</b> in <b>Project</b> menu.	
4-2	Export the model to XML file.	 <p>Or</p> 
4-3	Remember to save the project before quitting the application.	

## 2.3 Apply the Project to the NexUA Server

5. Import the XML file to the NexUA Server.

Step	Description	Screenshot
5-1	Open the NexUA Server and change the tab to <b>Node Config</b> .	
5-2	Click <b>Import</b> and select the XML file exported from the NexUA Model Maker. Default file path: <a href="#">C:\Program Files\NEXA IoT\NexUA Model Maker\Models</a>	
5-3	After completing the entire configuration, click <b>Apply</b> to restart the server and apply the changes.  For instructions on operating the NexUA Server, please refer to the NexUA Server user manual.	