

IKAVA DataService

User Manual

Version 1.0.0

For DataService Version 1.0.0

© IKAVA AG 2022

System Requirements	4
Microsoft Windows.....	4
.NET Framework	4
UiPath Platform.....	4
MongoDB.....	4
Quickstart Tutorial	5
Installation	5
MongoDB (Database).....	5
Install Robo3T (GUI for the Database).....	9
DataService NuGetPackage	12
Usage.....	13
Add DataService Package to your project.....	13
Insert “DataService Configuration” Activity into your project.....	13
Create a unique “MailEntity”	13
Use the newly created “MailEntity” in your Studio Project	14
Run your Automation.....	16
Debugging with Robo3T (GUI for MongoDB).....	16
Activities	20
Configuration	20
DataService Configuration.....	20
Entity	20
Create Entity	20
EntityList	20
Add Entity to List	20
Create Entity List.....	20
Delete Entity from List by ID	20
EntityRecord	21
Create Entity Record.....	21

Delete Entity Record by ID	21
Get Entity Record by ID	21
Update Entity Record.....	21
EntityRecordList	21
Create Entity Record List.....	21
Delete Entity Record List	21
Get Entity Record List	21

System Requirements

Microsoft Windows

Our product has passed all quality tests with the following Windows Versions:

- Windows 10

.NET Framework

Our product has passed all quality tests with the following .NET Framework Versions:

- .NET Framework 4.8

UiPath Platform

Our product has passed all quality tests with the following UiPath Platform Versions:

- UiPath Platform 22.4.7 with Compatibility Mode Legacy
- UiPath Platform 22.10.3 with Compatibility Mode Legacy and Windows

MongoDB

Our product has passed all quality tests with the following MongoDB Versions:

- MongoDB 4.2.23
- MongoDB 6.0.3

Quickstart Tutorial

Installation

MongoDB (Database)

Please download and install a currently supported MongoDB version on your local machine.

You will find supported versions in chapter “[MongoDB](#)” in this manual.

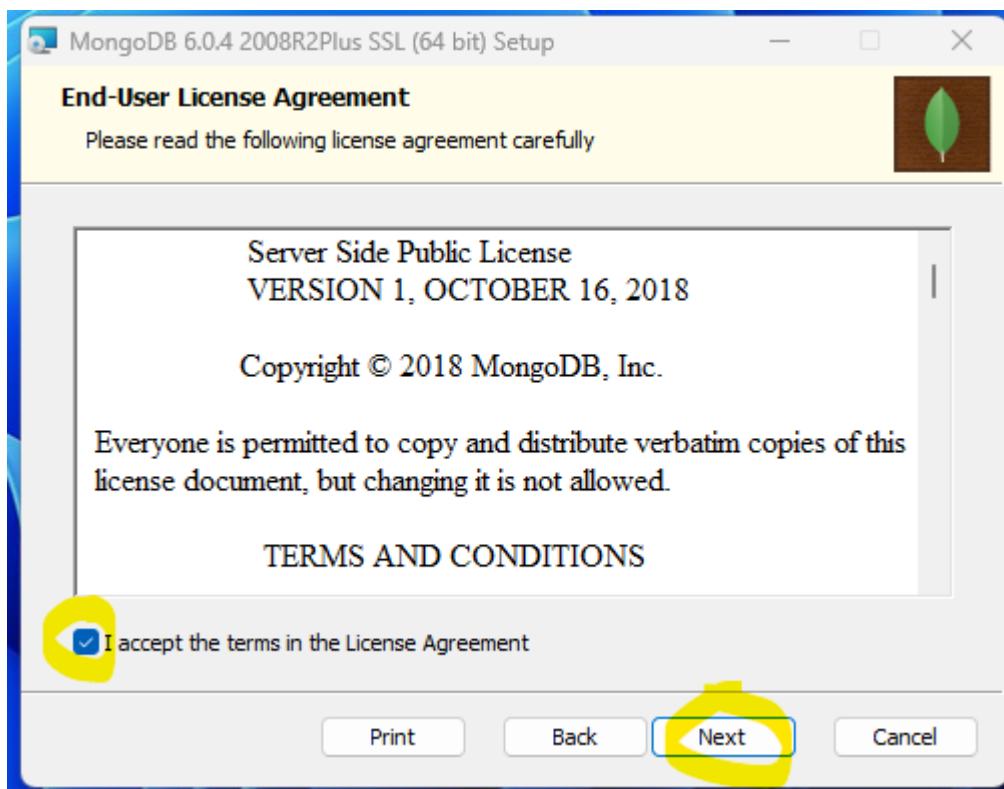
Download the free community edition of MongoDB as MSI Package from this page:

<https://www.mongodb.com/try/download/community>.

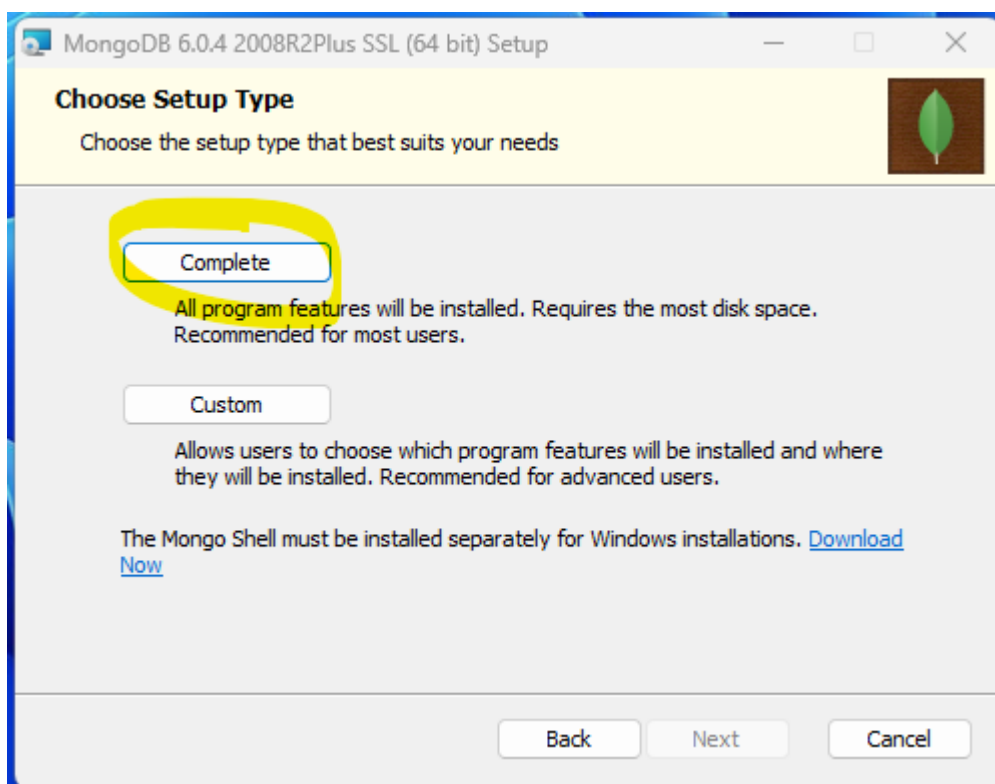
Then install it with the following settings:



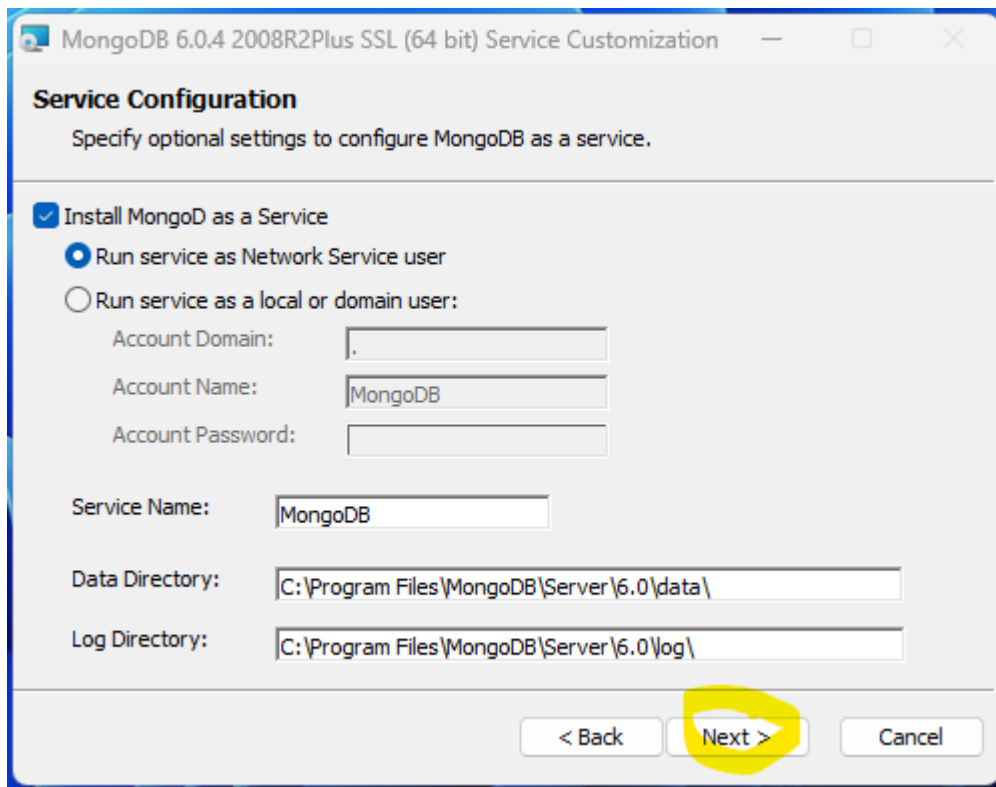
Click “Next”.



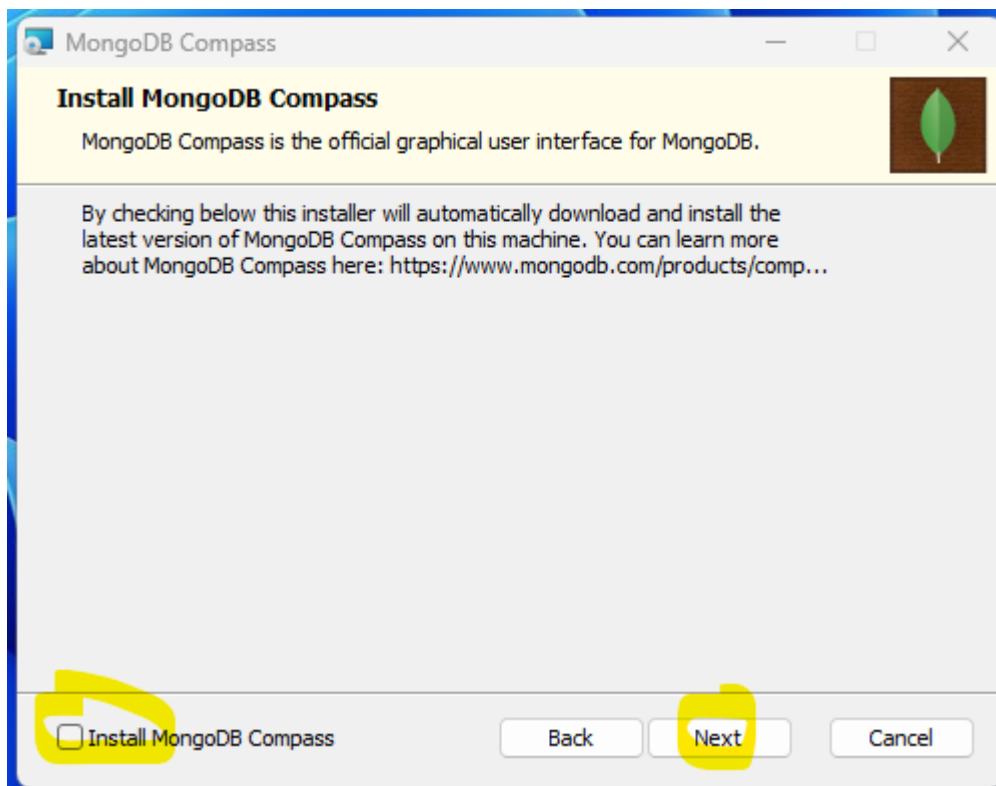
Accept the terms.
Click "Next".



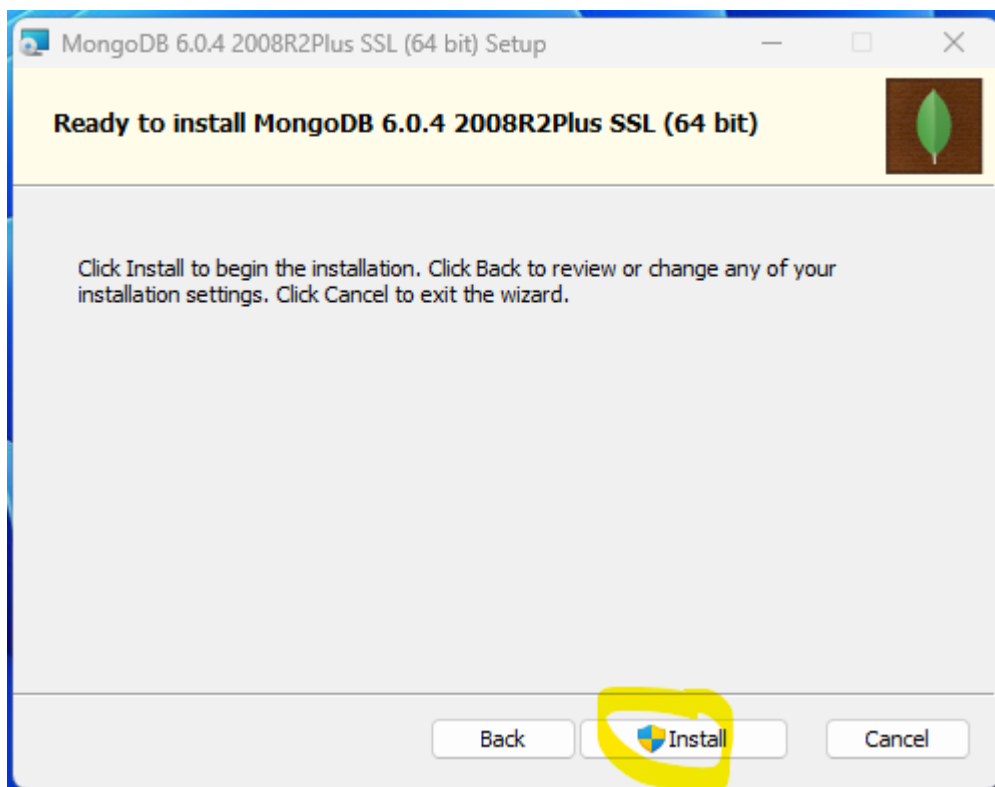
Click "Complete"



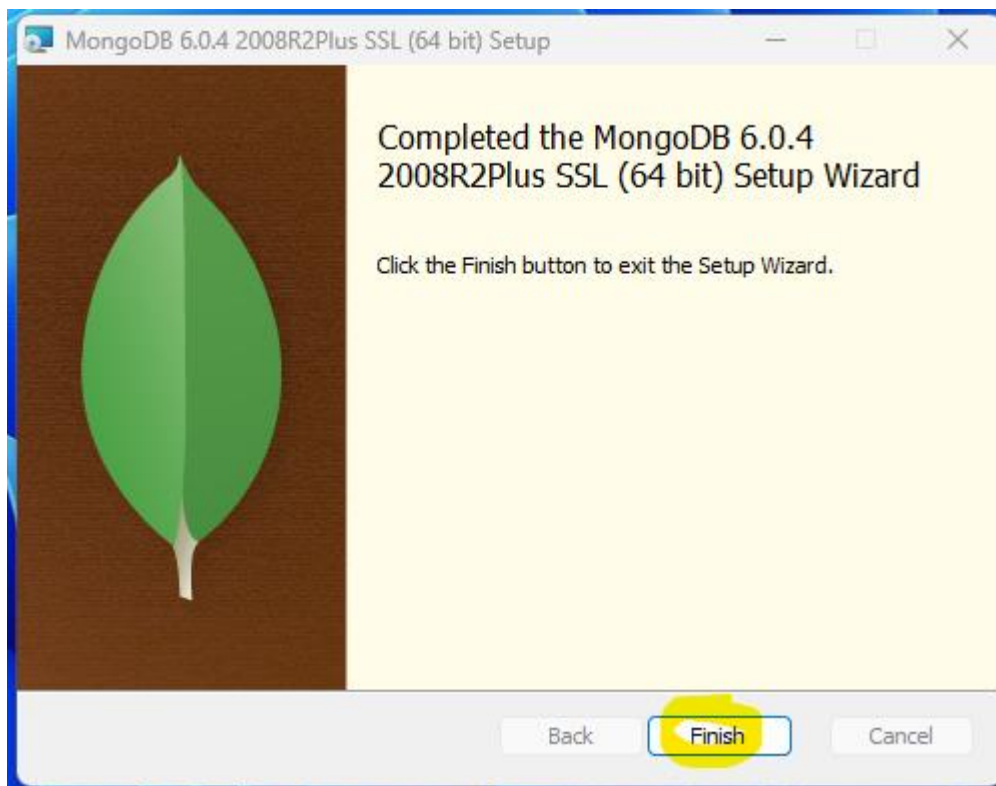
Click "Next"



MongoDB Compass must not be installed, because we will use Robo3T as GUI.
Click "Next".



Click "Install".



Click "Finish".

Install Robo3T (GUI for the Database)

Please download and install any GUI for your MongoDB.

We recommend Robo3T, because it will be used in this Tutorial.

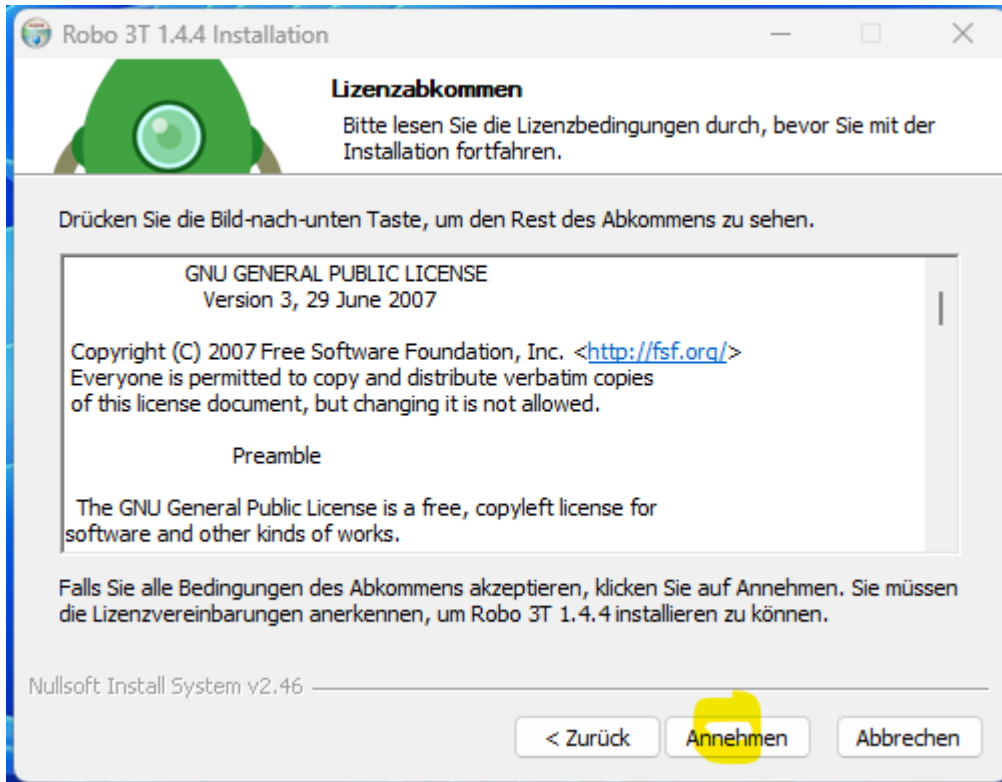
Download Robo3T from this page:

<https://github.com/Studio3T/robomongo/releases>.

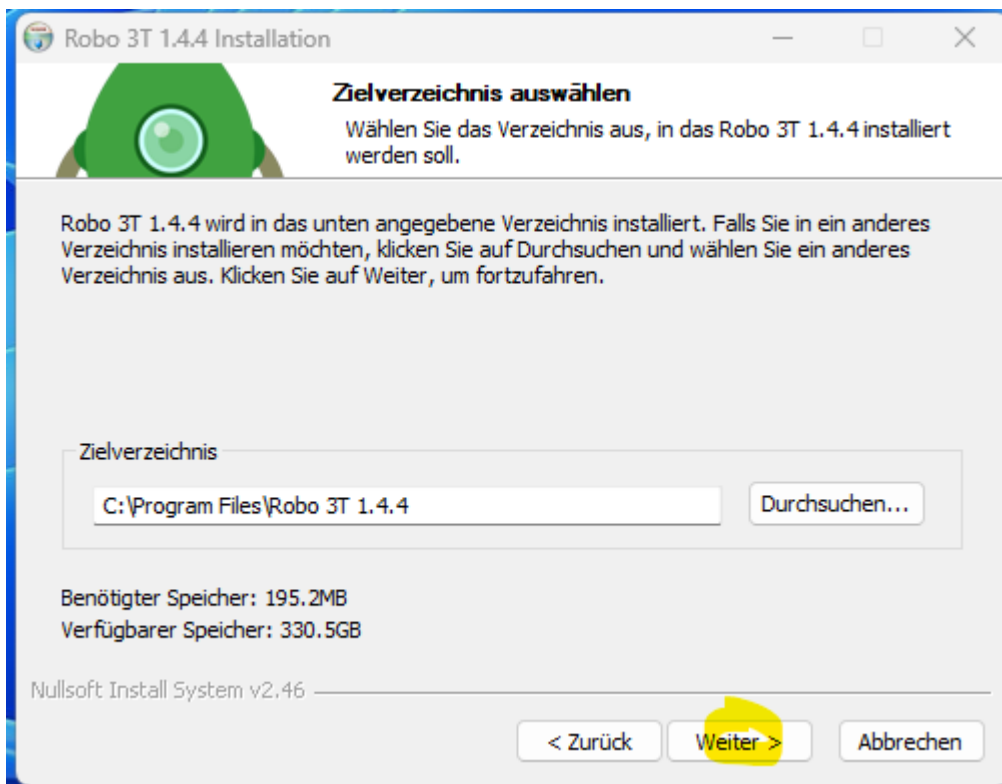
Then install it with the standard settings:



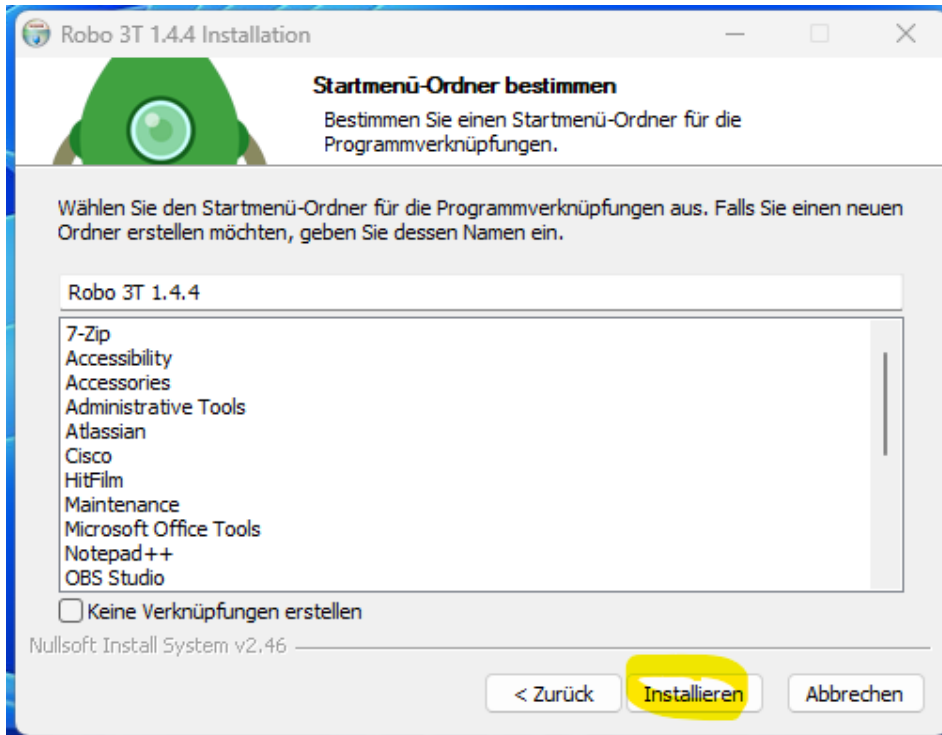
Click "Next".



Click "Accept".



Click "Next".



Click "Install".



Click "Finish".

Then robo3T should start.

How to connect it with MongoDB will be explained in the "Usage" Chapter.

DataService NuGetPackage

1.Download

You can order the IKAVA.DataService directly from the UiPath Marketplace:

https://cloud.uipath.com/ikavabnpvnvu/marketplace_/listings/ikavadataservice-on-premises-dataservice5757

If the direct link is broken, search for IKAVA.DataService from the UiPath Marketplace:

<https://marketplace.uipath.com/>

You can also directly download the DataService from our website:

<https://www.ikava.ch/downloads/>

If you download the package directly, please contact IKAVA AG for a license key.

2.Installation

Please copy your downloaded IKAVA.DataService NuGetPackage into your preferred NuGetPackages folder or alternatively upload to your Orchestrator.

The Windows standard NuGetPackage folder is “**%userprofile%\nuget\packages**”.

Usage

Add DataService Package to your project

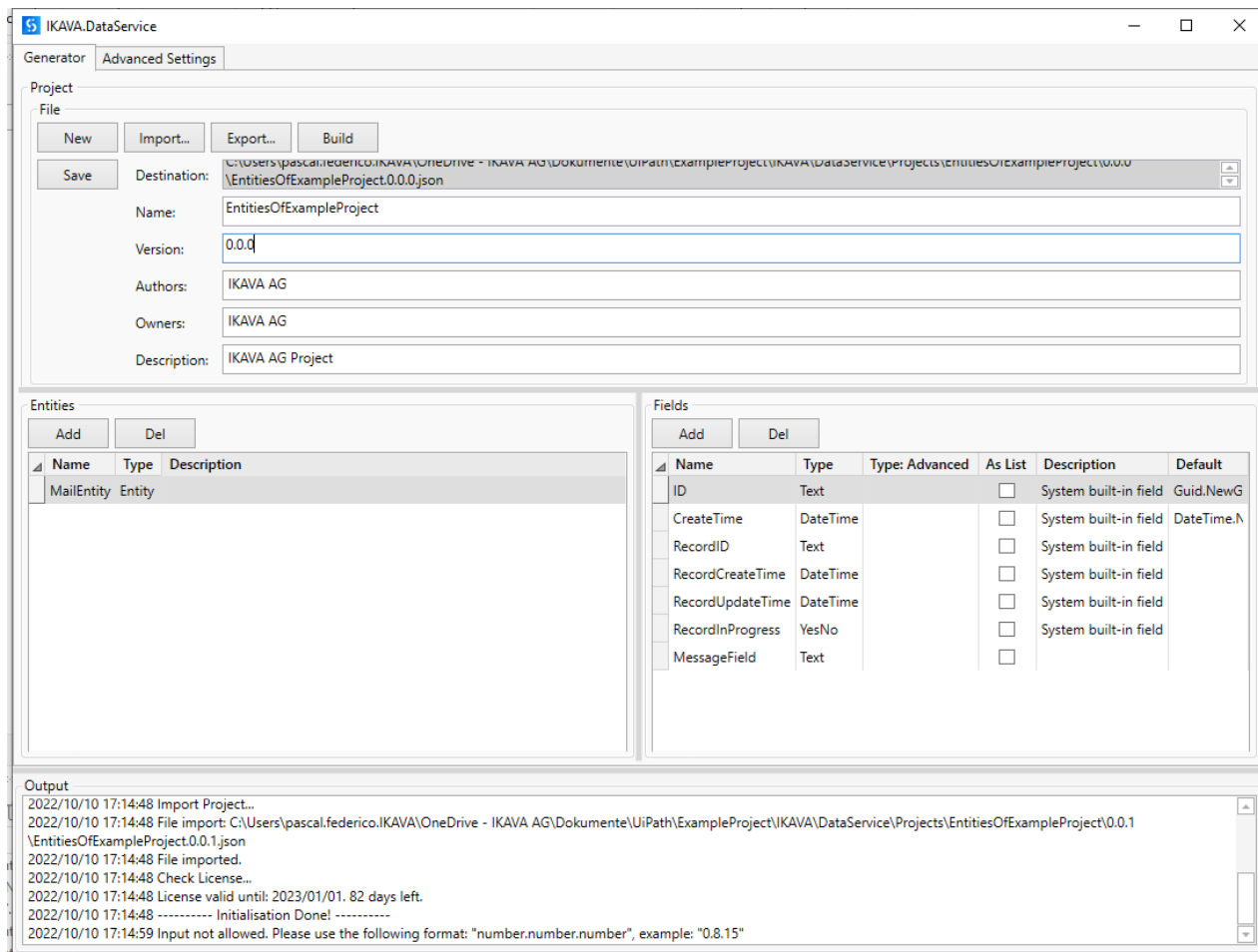
1. Open a UiPath Studio Project.
2. Go to Manage Packages.
3. Select "All Packages" and search for IKAVA.DataService.
4. Click install.
5. Click save.

Insert "DataService Configuration" Activity into your project

1. Go to the Activities-Tab in Project-Explorer.
2. Navigate to IKAVA/DataService/Configuration.
3. Select the "DataService Configuration" Activity and place it somewhere in your UiPath Studio project.
4. Click "Start..."

Create a unique "MailEntity"

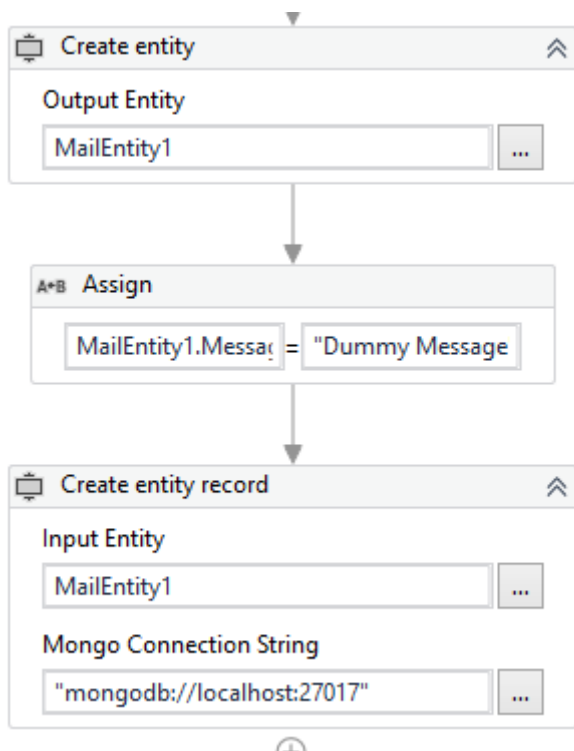
We are now creating the following DataService-Project step-by-step:



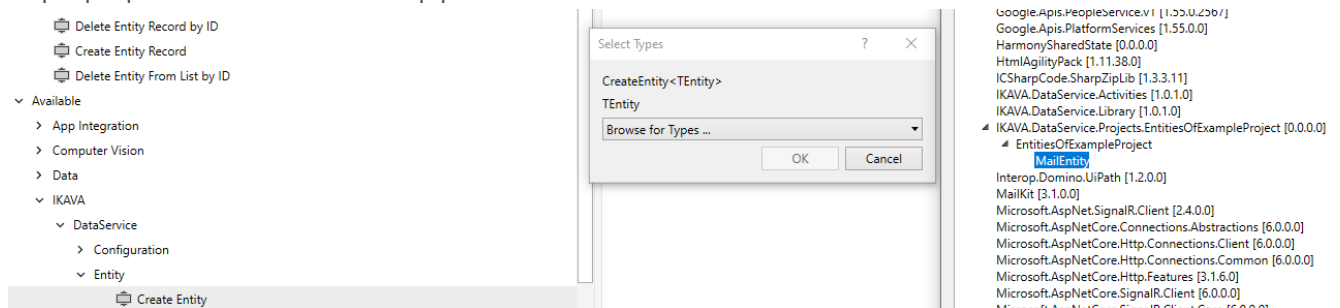
1. Move your cursor to the "Entities" GroupBox and click "Add".
2. Now "Entity1" should appear in the "Entities" Groupbox.
3. Double-Click Entity1 and rename it to "MailEntity".
4. Move your cursor to the "Fields" GroupBox and click "Add".
5. Now "Field1" should appear in the "Fields" Groupbox.
6. Double-Click Field1 and rename it to "MessageField".
7. Move your cursor to the "Project" GroupBox and click "Build".
8. You will be asked if you wish to save. Please click "Yes".
9. Building might take some time. When everything went fine, then a "Build Successful" message should appear.
10. Insert the newly created NuGetPackage to your project according to the popped up instructions window.

Use the newly created "MailEntity" in your Studio Project

We are now modelling the following sequence step by step:



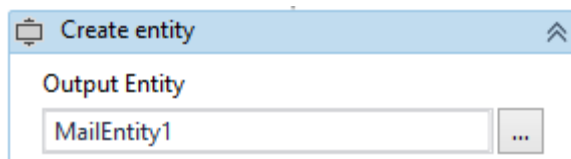
1. Go to the Activities-Tab in Project-Explorer.
2. Navigate to IKAVA/DataService/Entity.
3. Select the “Create Entity” Activity and place it in your UiPath Studio project.
4. A pop-up window should appear.



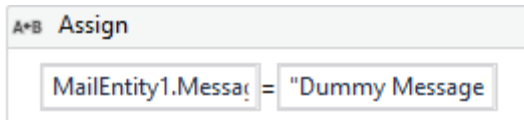
Click “Browse for Types” and select your newly created “MailEntity” as type.

(Note: The selected type can never be changed again unless you reinsert the whole activity)

5. Click the Output Entity Field in your newly inserted activity and press ctrl + k to create a new variable. Name the new variable “MailEntity1” and press enter.



6. Now add an assign Activity to your project and write:
Left side: “MailEntity1.MessageField”.
Right side: “Dummy Message”.



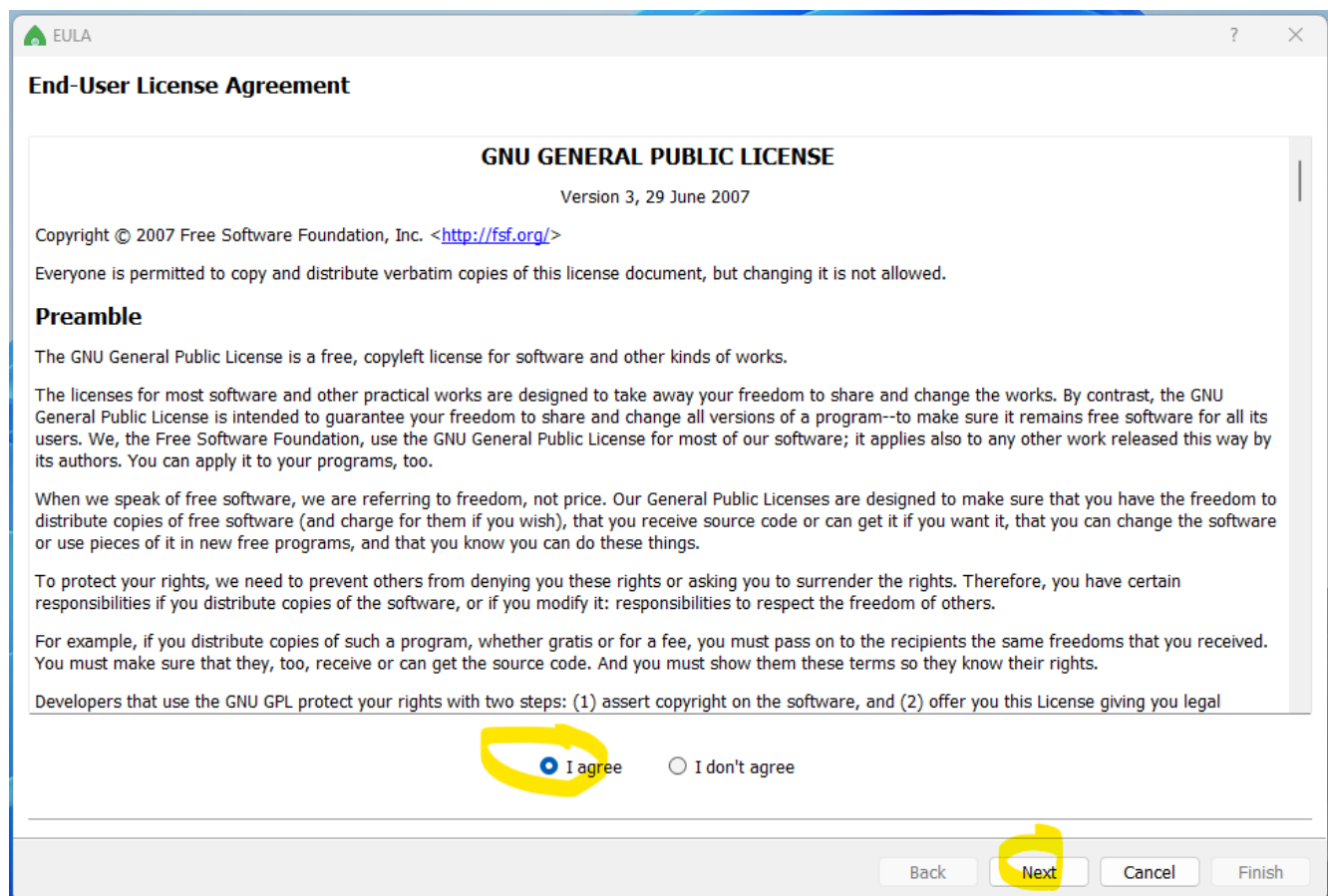
7. Select the “Create Entity Record” Activity and place it in your UiPath Studio project.
8. A popup window should appear. Again select the type.
9. Write “MailEntity1” into the “Input Entity” field and we are done.

Run your Automation

1. Make sure that the MongoDB-Service has been started in the background. To check this, go to the Windows Taskbar and enter “Services”. The MongoDB service has to be in the running state.
2. “Run” your UiPathStudio Project.
3. After termination without errors, you can open Robo3T (or any other GUI for MongoDB) and check the results of your first DataService usage.

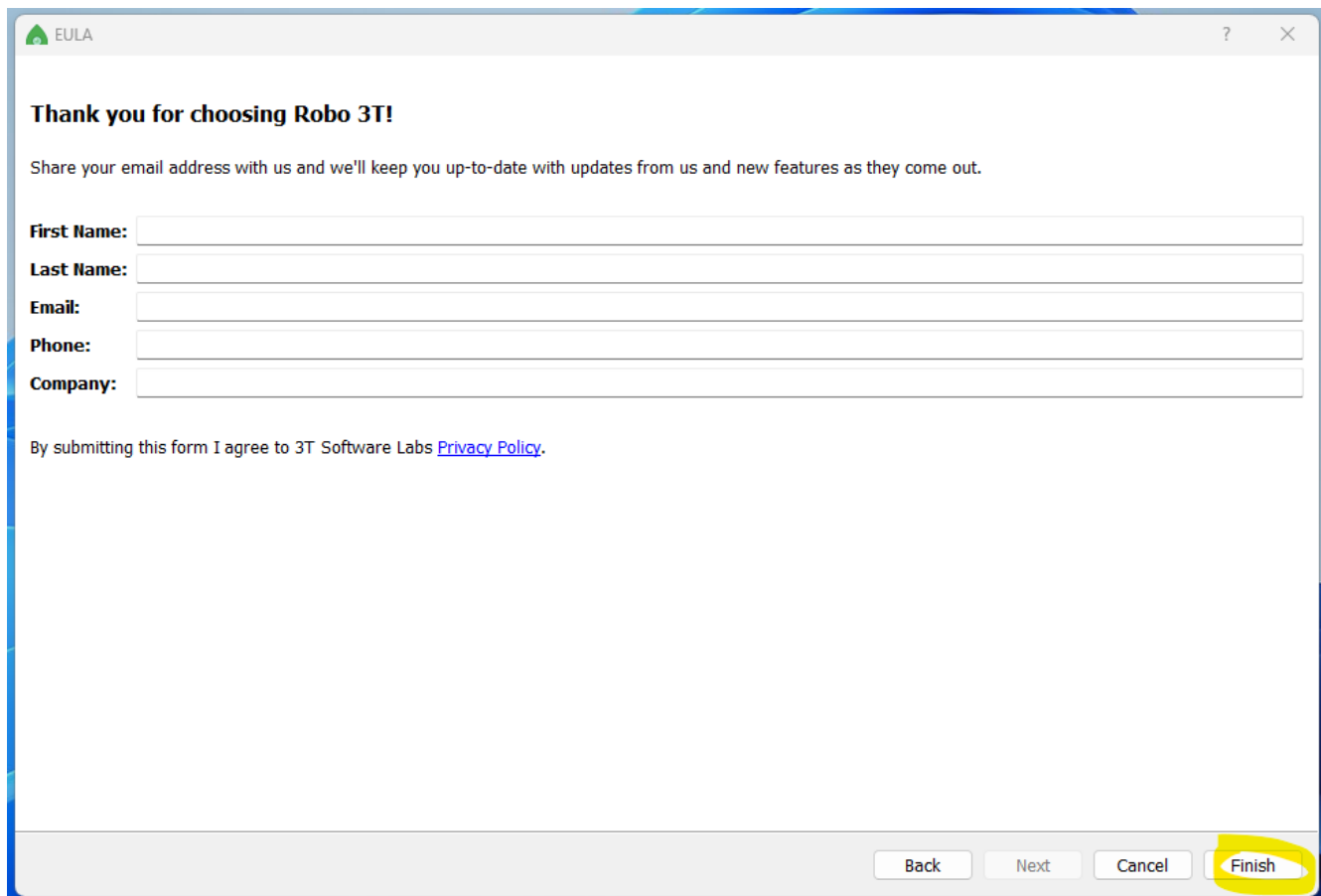
Debugging with Robo3T (GUI for MongoDB)

Start Robo3T.



Select “I Agree”.

Click "Next".



The image shows a software window titled "EULA" with a standard macOS-style title bar (green, yellow, red buttons and window controls). The window content includes a heading "Thank you for choosing Robo 3T!", a paragraph about sharing an email address for updates, and a form with five input fields labeled "First Name:", "Last Name:", "Email:", "Phone:", and "Company:". Below the form is a line of text: "By submitting this form I agree to 3T Software Labs [Privacy Policy](#).". At the bottom right, there are four buttons: "Back", "Next", "Cancel", and "Finish". The "Finish" button is highlighted with a yellow circle.

EULA

Thank you for choosing Robo 3T!

Share your email address with us and we'll keep you up-to-date with updates from us and new features as they come out.

First Name:

Last Name:

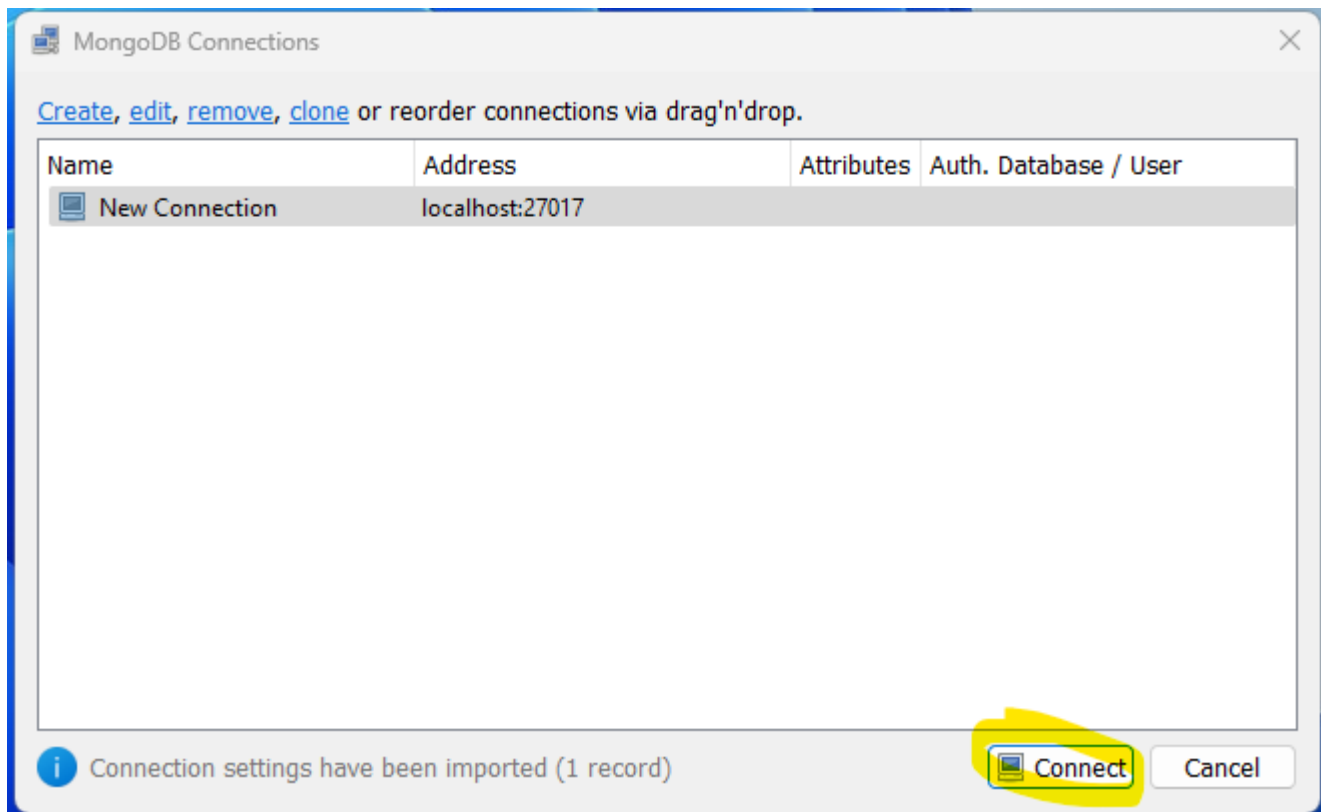
Email:

Phone:

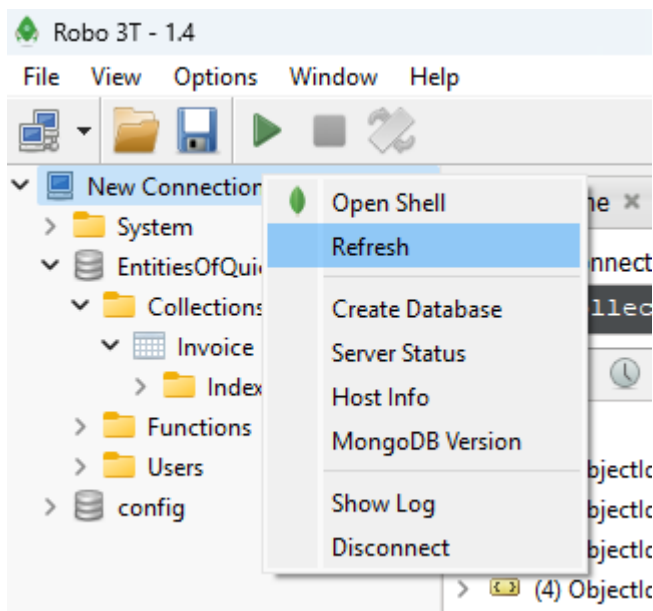
Company:

By submitting this form I agree to 3T Software Labs [Privacy Policy](#).

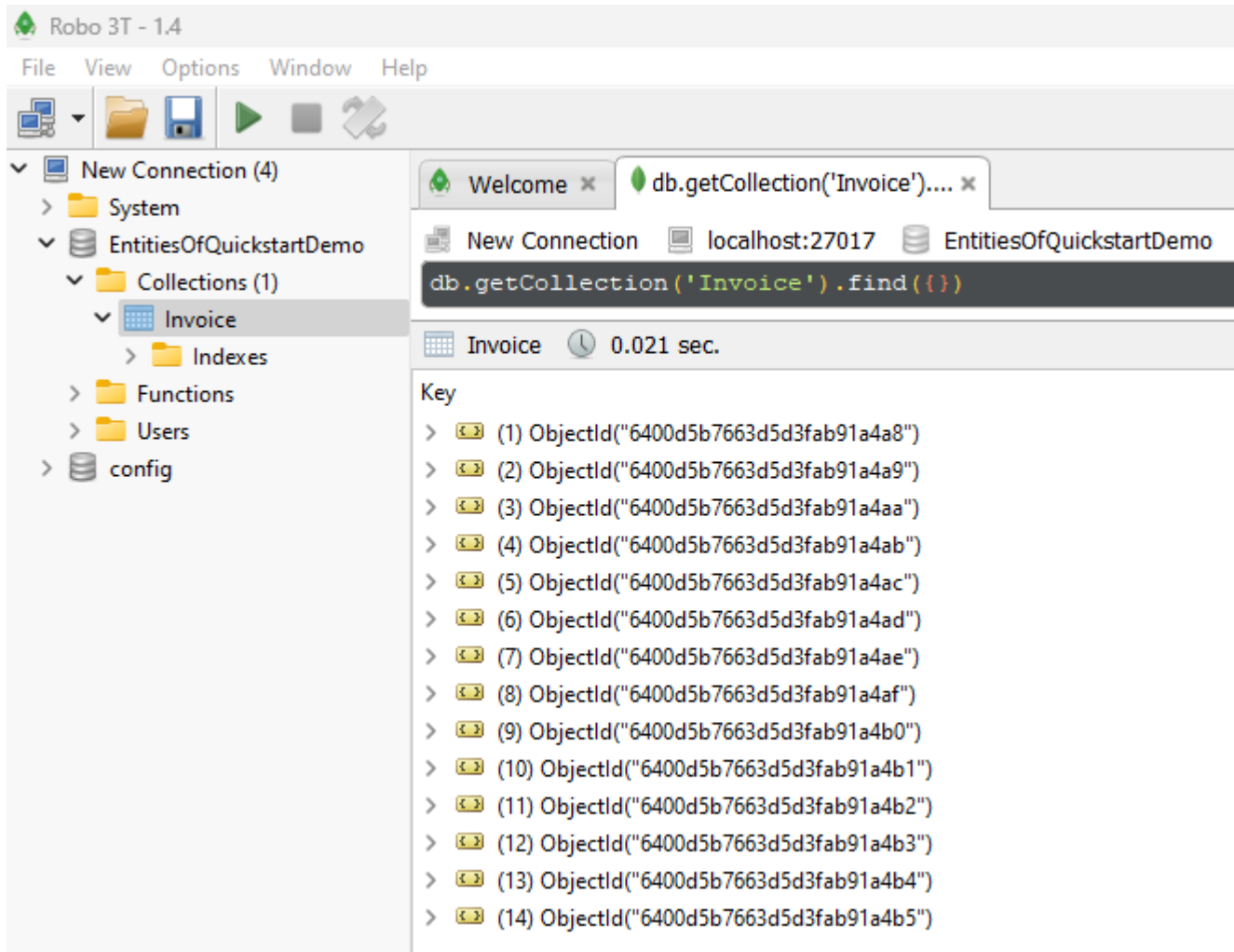
Enter your data or leave it blank.
Click "Finish".



Select "localhost:27017".
Click "Connect".



Select your connection and right click with your mouse.
Click "Refresh".



Now your collection with the created objects should appear.
(They might be different than in the screenshot)

Activities

Configuration

DataService Configuration

This activity does absolutely nothing during automation runtime.

You need it to create and manage Entities that you can use within our project.

Project

File

New Import... Export... Build

Save

Destination: C:\Users\pascal.federico.IKAVA\OneDrive - IKAVA AG\Dokumente\UIPath\ExampleProject\IKAVA\DataService\Projects\EntitiesOfExampleProject\0.0.0\EntitiesOfExampleProject.0.0.0.json

Name: EntitiesOfExampleProject

Version: 0.0.0

Authors: IKAVA AG

Owners: IKAVA AG

Description: IKAVA AG Project

Entities

Add Del

Name	Type	Description
Entity1	Entity	

Fields

Add Del

Name	Type	Type: Advanced	As List	Description	Default
ID	Text		<input type="checkbox"/>	System built-in field	Guid.NewGuid().ToString()
CreateTime	DateTime		<input type="checkbox"/>	System built-in field	DateTime.Now
RecordID	Text		<input type="checkbox"/>	System built-in field	
RecordCreateTime	DateTime		<input type="checkbox"/>	System built-in field	
RecordUpdateTime	DateTime		<input type="checkbox"/>	System built-in field	
RecordInProgress	YesNo		<input type="checkbox"/>	System built-in field	

Output

2022/10/10 15:17:19 ----- Initialisation Started... -----

2022/10/10 15:17:19 Import last Session...

2022/10/10 15:17:19 File import: C:\Users\pascal.federico.IKAVA\OneDrive - IKAVA AG\Dokumente\UIPath\ExampleProject\IKAVA\DataService\Session.json

2022/10/10 15:17:19 File not Found.

2022/10/10 15:17:19 Import Advanced Settings...

2022/10/10 15:17:19 File import: C:\Users\pascal.federico.IKAVA\AppData\Roaming\IKAVA\DataService\AdvancedSettings.json

2022/10/10 15:17:20 File imported.

Entity

Create Entity

Creates a new Entity during automation runtime.

EntityList

Add Entity to List

Adds an Entity to a List of type Entity during automation runtime.

Create Entity List

Creates an empty List of type Entity during automation runtime.

Delete Entity from List by ID

Deletes an Entity from a List of type Entity during automation runtime.

EntityRecord

Create Entity Record

Takes an Entity and stores/records it on the database during automation runtime.

Delete Entity Record by ID

Takes the ID of an Entity and deletes its corresponding record (by ID) from the database during automation runtime.

Get Entity Record by ID

Takes the ID of an Entity and gets its corresponding record from the database during automation runtime.

Update Entity Record

Takes an Entity and updates its corresponding record (by ID) in the database during automation runtime.

EntityRecordList

Create Entity Record List

Takes a List of type Entity and stores/records it on the database during automation runtime.

Delete Entity Record List

Takes a List of type Entity and deletes its corresponding records (by ID) on the database during automation runtime.

Get Entity Record List

Gets a List of type Entity from the Database during automation runtime. Queries can be used to filter the Entities.