

3.A AI Actions

These actions allow you to integrate the new AI tools into your processes, enabling their use to be automated.

- [3.A.1 Call LLM](#)
- [3.A.2 Call MCP Tool](#)
- [3.A.3 Call Agent](#)
- [3.A.4 Call Process Tool](#)
- [3.A.5 Examples](#)

3.A.1 Call LLM

This action allows you to send a request to an LLM that has been previously configured in Settings > AI Tools > LLM.

Once you select one of the available LLMs, all the settings configured in Settings > AI Tools > LLM are automatically loaded and can be customized as needed.

System message

Select Option... ▼

You are the virtual assistant of the vtenext application.

User message

Select Option... ▼

Rewrite the solution in a technical and professional manner.
\$2-solution
Never use titles or Markdown code.

Developer-provided instructions that the model should follow, regardless of messages sent by the user. The role of the messages author in this case is `system`.

Message sent by an end user, containing prompts or additional context information. The role of the messages author in this case is `user`.

Figure 1

In addition, you can define custom Headers, Parameters, and Output Fields, allowing you to pass dynamic values retrieved from the records involved in the process.

By default, an output field containing the model's response is already configured (Figure 2). This field can then be referenced and used in subsequent steps of the process (Figure 3).

Returned fields

answer

choices.0.message.content

Figure 2

Solution Information

☒ Solution

answer ▼

\$LLM39-answer

Figure 3

3.A.2 Call MCP Tool

This action allows you to send a request to an MCP server tool that has been previously configured in Settings > AI Tools > MCP Client.

This action provides a deterministic response from an MCP tool, unlike what could be achieved through an AI agent.

Users can therefore implement their own “agent” within a process by combining interactions between LLM calls and MCP tool calls.

Once an MCP client and the desired tool have been selected, the expected Input Parameters and Output Fields defined by the tool are loaded automatically and can be modified according to the tool response.

The screenshot shows a configuration window titled "Action: Call MCP tool". It includes a "Save" button and a "Cancel" button. The configuration is divided into several sections:

- Action title:** A text input field.
- Execution mode:** A dropdown menu with the selected option "Standard: the execution process waits for the action to complete before proceeding to the next one".
- MCP client:** A dropdown menu with the selected option "Web Search MCP - Exa".
- Available tools:** A dropdown menu with the selected option "web_search_exa". To the right of this dropdown is a detailed description of the tool: "Search the web for any topic and get clean, ready-to-use content. Best for: Finding current information, news, facts, people, companies, or answering questions about any topic. Returns: Clean text content from top search results. Query tips: describe the ideal page, not keywords. 'blog post comparing React and Vue performance' not 'React vs Vue'. Use category:people / category:company to search through LinkedIn profiles / companies respectively. If highlights are insufficient, follow up with web_fetch_exa on the best URLs."
- Input schema:** This section contains two parameters:
 - query:** A text input field with the value "What is the certified email (PEC) address of the company \$1-accountname?". To its right is a dropdown menu labeled "Select Option...". To the right of the dropdown is a description: "Natural language search query. Should be a semantically rich description of the ideal page, not just keywords. Optionally include category:<type> (company, people) to focus results — e.g. 'category:people John Doe software engineer'. minLength: 1".
 - numResults:** A text input field with the value "10". To its right is a dropdown menu labeled "Select Option...". To the right of the dropdown is a description: "Number of search results to return (default: 10)".
- Returned fields:** A text input field with the value "content.0.text". To its right is a dropdown menu labeled "Select Option...". Below this field is an "Add field" button.

Figure 1 - Example of a Web Search Tool Call

Normally, tools return results in a format suitable for interpretation by an LLM model. Therefore, the response should be reviewed and processed before being included in a workflow.

Example of a response from the `web_search_exa` tool to the question: “What is the certified email (PEC) address of VTENEXT?”

Title: Contatti - vtenext CRM + BPM
URL: <https://www.vtenext.com/it/contatti/>
Published: 2021-01-28T21:00:22.000Z
Author: N/A

Highlights:

info@vtenext.com Viale Sarca, 336/F 20126 Milano (MI) Tel (+39) 02 3790 1352 P.I. 09869110966
Registration Number MI – 2118732

Title: Fatturato Vtenext Srl (MI) – Scarica Bilanci e Visure Camerali Ufficiali
URL: <https://www.companyreports.it/vtenext-srl-09869110966>
Published: N/A

Author: N/A

Highlights:

****Indirizzo****

Viale Sarca, 336/F - [Milano](<https://www.companyreports.it/comune/milano> "Vedi le aziende del comune di Milano") [MI](<https://www.companyreports.it/provincia/milano> "Vedi le aziende della provincia di Milano"))

...

...

**Visura Camerale di Vtenext Srl**

La visura camerale di ****Vtenext Srl**** fornisce una panoramica completa dell'assetto legale e societario dell'impresa: sede legale, capitale sociale, oggetto sociale, elenco soci, cariche in corso e indirizzo PEC. La visura è disponibile in formato ordinario o storico ed è scaricabile in tempo reale da ****CompanyReports.it****.

**Bilanci e Documenti Aziendali di Vtenext Srl**

...

PEC

*

...

* Email PEC

* Codice di fatturazione

...

```
```json
{
 "@context": "https://schema.org",
 "@type": "Organization",
 "name": "CompanyReports",
 "url": "https://www.companyreports.it",
 "logo": "https://www.companyreports.it/dist/img/companyreports-logo-new.webp",
 "contactPoint": {
 "@type": "ContactPoint",
 "contactType": "sales",
 "email": "info@adcapital.it"
 }
}

...

{
 "@context": "https://schema.org",
 "@type": "LocalBusiness",
 "name": "Vtenext Srl",
 "legalName": "Vtenext Srl",
 "image": "https://www.companyreports.it/og-image-Fatturato%20Vtenext%20Srl%20%28MI%29%20%E2%80%93%20Scarica%20Bilanci%20e%20Visure%20CameraI%20Ufficiali.jpg",
 "vatID": "09869110966",
 "taxID": "09869110966",
 "foundingLocation": "MI",
 "foundingDate": "2017-04-04",
 "isicV4": "6201",
 "knowsAbout": "Produzione di software non connesso all'edizione",
 "address": {
 "@type": "PostalAddress",
 "postalCode": "",
 "addressLocality": "Milano, Italy",
 "addressCountry": "IT",
 "addressRegion": "MI",
 "streetAddress": "Viale Sarca, 336/F"
 },
 "keywords": "Vtenext Srl fatturato, utile Vtenext Srl, perdita Vtenext Srl, risultato Vtenext Srl , andamento Vtenext Srl, partita iva Vtenext Srl , bilancio Vtenext Srl , visura Vtenext Srl, 09869110966, Partita Iva 09869110966, Codice Fiscale 09869110966, DUNS Vtenext Srl, D-U-N-S Vtenext Srl, Report Aziende per fatturato"
```

---

Title: VTENEXT S.R.L. SOCIETA' BENEFIT 03641400233 - Visurissima.it - Le informazioni a portata di click

URL: [https://www.visurissima.it/aziende/VTENEXT-SRL-SOCIETA-BENEFIT\\_03641400233.html](https://www.visurissima.it/aziende/VTENEXT-SRL-SOCIETA-BENEFIT_03641400233.html)

Published: N/A

Author: N/A

Highlights:

Indirizzo completo

...

VIA ALZANA 2/C, 37040 ARCOLE (VR)

...

crmvillege@pec.it

...

Scopri Email Scopri

---

[...]

## 3.A.3 Call Agent

This action allows you to send a request to an agent that has been previously configured in Settings > AI Tools > Agents.

Once one of the available agents has been selected, all the elements configured in Settings > AI Tools > Agents are loaded automatically and can be customized.

The agent is able to perform tasks automatically based on the enabled tools and the knowledge sources available through RAG and LLM.

The **System Prompt** can be modified, while operational instructions must be entered in the **User Message** field. The Output Fields section already includes the response content field by default, allowing the agent's output to be used in subsequent process steps.

Below is an example where the agent is requested to find a solution to a customer support request received through a ticket, using the user\_manual tool, and write the solution as a comment.

The screenshot shows a configuration window titled "Action: Call agent" with a "Save" button and a "Cancel" button. The window is divided into several sections:

- Post-guardrail LLM:** A dropdown menu with "Select Option..." and "--Please select--". A description states: "Select an LLM to evaluate the output. It may affect token consumption and request latency."
- Static post-guardrail:** A text area with "Select Option..." and a description: "Define strings and regular expressions to block outputs that contain prohibited words. Use a new line to enter multiple values."
- \* User message:** A text area containing the prompt: "Add a comment to the ticket suggesting a solution to the issue by searching the user\_manual knowledge base. Do not use headings or Markdown formatting in the response. Return only plain text." A description states: "Message sent by an end user, containing prompts or additional context information. The role of the messages author in this case is user."
- Returned fields:** A section with a dropdown menu showing "content" and "messages.0.content". A description states: "Select Option...". Below this is an "Add field" button.

Figure 1

Additional information is added to the prompt written by the user to provide the context of the request, such as the entities involved in the process, including their name, module, and CRM ID (crmid).

This allows the agent to understand which CRM entity the request refers to.  
If needed, process variables can be used to make the request even clearer.  
Example: Add a comment to ticket \$15-crmid suggesting a solution...

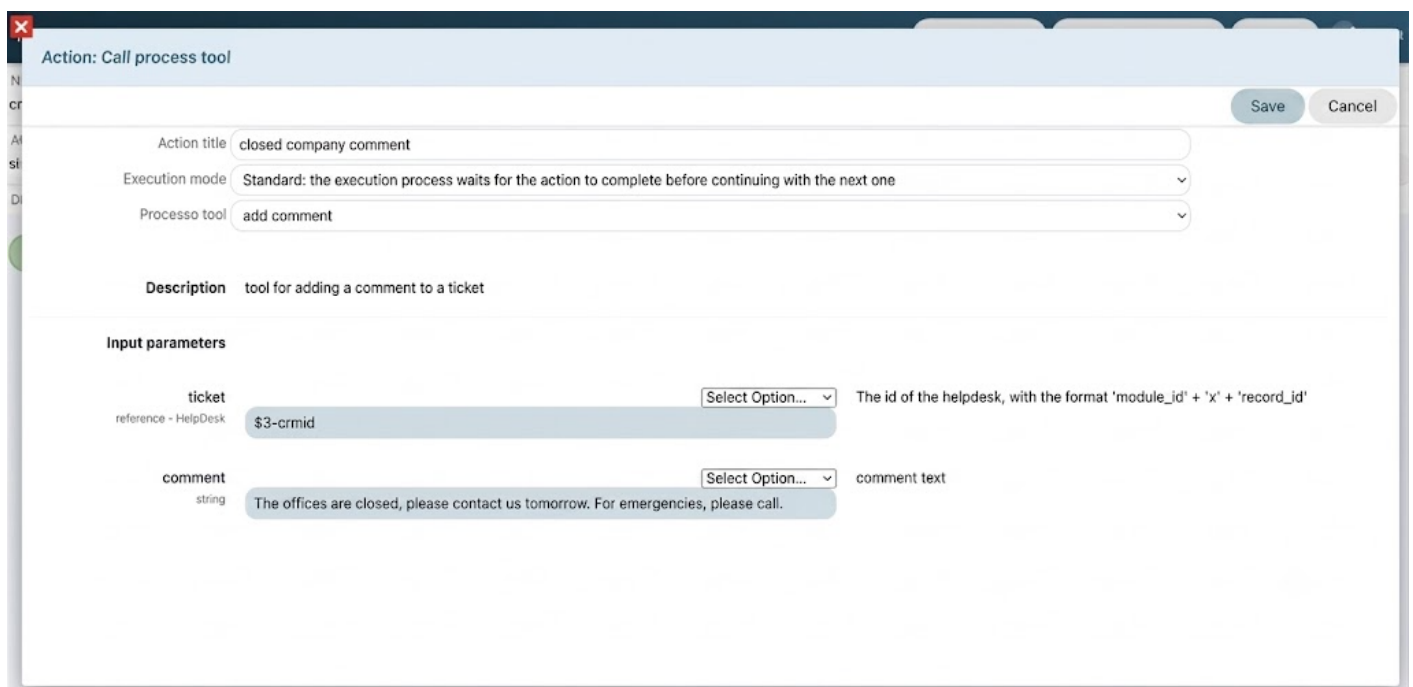
# 3.A.4 Call Process Tool

This action allows you to execute a [process tool](#).

Once the desired tool has been selected, the expected Input Parameters are loaded and can be configured using process variables, as usual.

Unlike sub-processes, this action does not have the same limitations and can be called multiple times within the same process and also from different processes.

Below is an example of a tool call used to add a comment to a ticket. The parameters are those defined in the process tool, which in this case are the ticket ID and the comment text.



The screenshot shows a configuration window titled "Action: Call process tool". It includes a "Save" button and a "Cancel" button. The form contains the following fields:

- Action title:** closed company comment
- Execution mode:** Standard: the execution process waits for the action to complete before continuing with the next one
- Process tool:** add comment
- Description:** tool for adding a comment to a ticket
- Input parameters:**
  - ticket:** reference - HelpDesk. Value: \$3-crmid. Description: The id of the helpdesk, with the format 'module\_id' + 'x' + 'record\_id'.
  - comment:** string. Value: The offices are closed, please contact us tomorrow. For emergencies, please call. Description: comment text.

# 3.A.5 Examples

Below are some example applications showing how these new tools can be integrated into processes.

## Web scraping

This process, triggered when a new company is created, calls an agent that is requested to search online for information about the company and populate the corresponding fields in the company record.

The agent must have an enabled web search tool (for example, `web_search_exa`) and the VTENEXT MCP Client tools enabled in order to access the module structure and update the record.

### Default VTE MCP Client

**create** ✓  
Create a record of the specified module. To understand the fields of the specified module, you should u...

**delete** ✓  
Delete the specified record

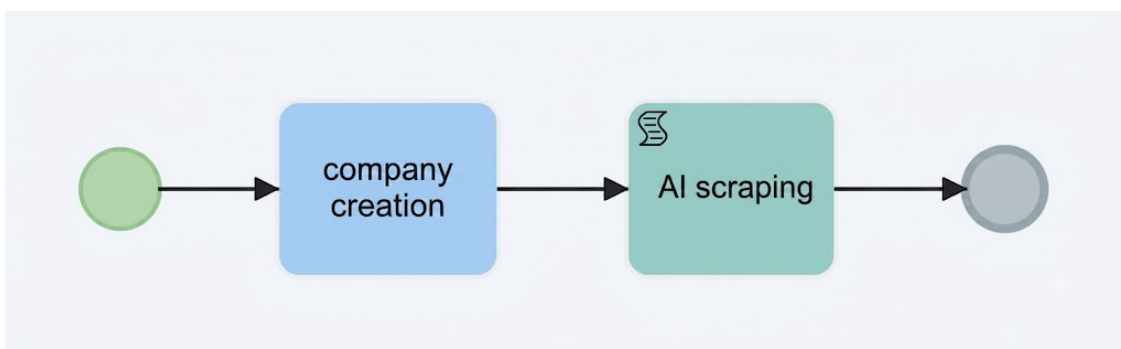
**describe** ✓  
Describe the specified module and their relative fields; you must run this tool before attempting any task, to...

### Web Search MCP - Exa

**web\_fetch\_exa** ○  
Read a webpage's full content as clean markdown. Use after `web_search_exa` when highlights ar...

**web\_search\_exa** ✓  
Search the web for any topic and get clean, ready-to-use content. Best for: Finding current information, news,...

The process starts when a company is created and executes the Call Agent action.



In the **User Message**, I request the agent to search online for the required information and populate the corresponding fields in the company record. I also request the creation of a contact for the Chief Executive Officer.

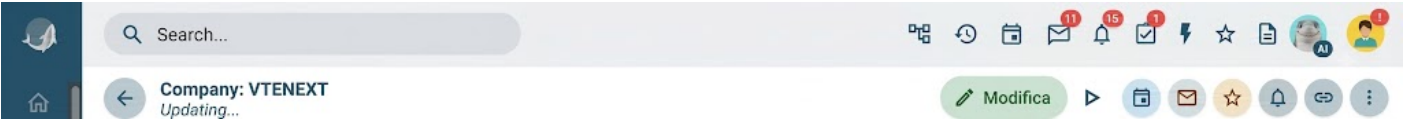
Select Option...▼

\* User message •

Search the web for the company's address, phone number, email, and VAT number, and populate them in the master record.  
Additionally, search for the name of the CEO and create the linked contact.

Message sent by an end-user, containing the prompt or additional contextual information. The role of the message author in this case is: `user`.

If the **Background** execution mode has been configured in the process action, after saving the record the system will notify you that the record is still being updated. Once the process is completed, the browser page will automatically refresh.



From the **History** tab, you can view the fields that have been updated.

Search...

Company VTENEXT

Modifica

Administrator has modified the record

Details

Field	Previous value	Current value
Phone		+39 02 3790 1352
Email		info@vtenext.com
VAT Number		09889110966
Billing Address		Viale Sarca, 336/F
Billing City		Milano
Billing Province/State		MI
Billing Postal Code		20126
Billing Country		Italia

7 days ago

Administrator has created the record

7 days ago

## Closed Customer Support

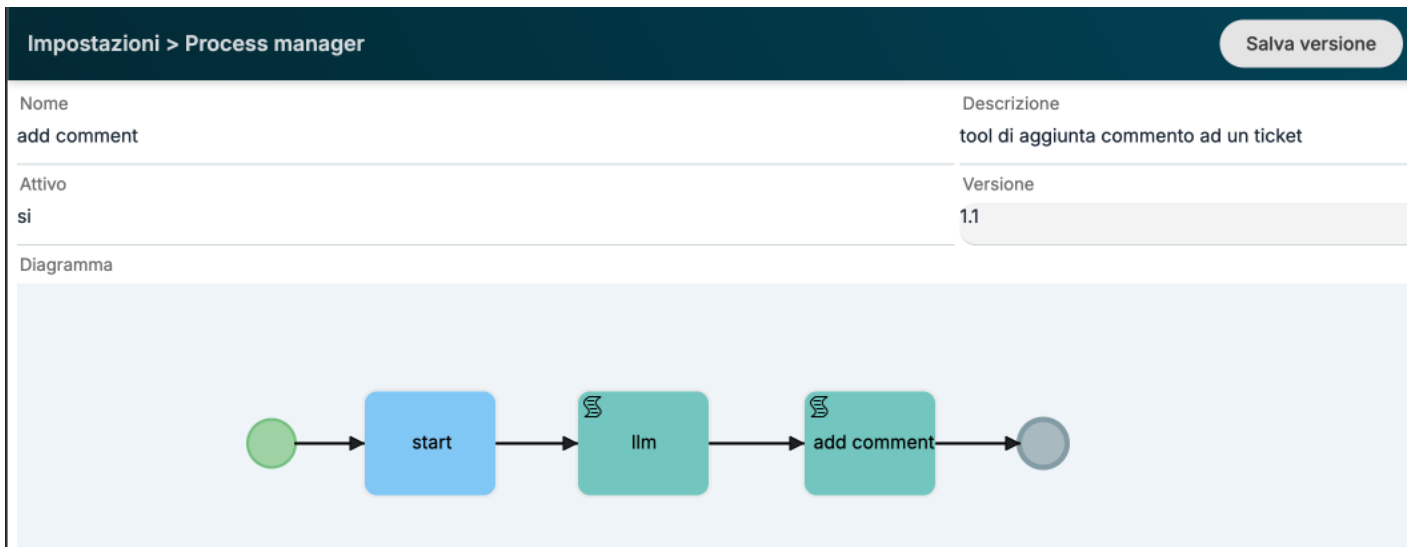
In this example, we implement a process that automatically replies to technical support tickets outside office hours, informing customers that the office is closed and attempting to provide a solution to their issue.

### Tool add\_comment

First, we create a process tool to add a comment to a ticket. The tool requires the following input parameters:

- Ticket ID

- Comment



BPMN-Task: start Activity\_1yh0zr4

Entity: --None--

When to execute the check

- ☐ on creation
- ☐ on creation and modification
- ☐ on modification
- ☐ every time the condition is true
- ☐ only the first time the process is triggered
- ☐ on launch of the subprocess
- ☒ on tool call
- ☐ in relationship with: ▼

Input parameters

Actions	Parameter name	Type	Entity	Mandatory	Description
<span>🗑️</span> <span>📄</span> <span>⬇️</span>	ticket	entity	Ticket	<input checked="" type="checkbox"/>	The id of the helpdesk, with the format 'module_id' + 'x' + 'record_id'
<span>🗑️</span> <span>📄</span> <span>⬆️</span>	comment	string	--Nessuno--	<input checked="" type="checkbox"/>	comment text

Add row

In the second step of the process, we execute an LLM call to rework the comment text by applying a specific style.



In this case, it is sufficient to use only an LLM instead of an agent, since no tools are required. We only need to leverage the language capabilities of the model.

Finally, we use the SDK Add Comment function by mapping all the required fields.

In this case, we set the ownerid parameter with the ID of the Kitt user. By setting the ai parameter to true, we force the addition of an informational note at the bottom of the comment indicating that the text was generated using AI.

Action: SDK Function

Save

Cancel

Action Title

add comment

Execution Mode

Standard: the execution process waits for the completion of the action before proceeding with the next one

SDK Function

Add comment to ticket (ticketid, comment, ownerid, ownertype[user|customer], sendPortalEmail[true|false], ai[true|false])

Additional Parameters

Parameter Value

Select Option...

\$2-crmid

Parameter Value

Select Option...

\$LLM5-answer

Parameter Value

Select Option...

4

Parameter Value

Select Option...

user

Parameter Value

Select Option...

true

Parameter Value

Select Option...

true

Add Parameter

The add\_comment tool must therefore be enabled for the agent that will be used in the following steps, together with the user\_manual tool and/or any documents configured in the RAG section to provide the knowledge required to generate a solution.

Cerca...

11

15

5

1

🔍

🕒

📅

📧

📧

🔔

📧

⚡

★

📄

🕒

👤

👤

Tool

Ricerca...

Kitt tools

Seleziona tutto

calculator

Calculate simple math expressions.

user\_manual

Search the vtenext user manual.

Default VTE MCP Client

Seleziona tutto

add\_comment

tool di aggiunta commento ad un ticket

convert\_lead

Starts the process for lead conversion

create

Create a record of the specified module. To understand the fields of the specified module, you should u...

delete

Delete the specified record

describe

Describe the specified module and their relative fields; you must run this tool before attempting any task, to...

get\_current\_context

Fetches general context information relevant for use with other MCP tools (so it does not need any parameter...

listtypes

List the modules that have fields with 'uitype' of the specified formats

process\_text

Rewrite or improve the supplied text using an optional tone of voice and complexity level.

query

Safely execute a SQL-like query and return the result's rows. Additional notes: - The "FROM" clause accept...

relate

Relate the record with all the records in a relatelist array

retrieve

Illustrate the fields of the specified

retrieveInventory

Illustrate the fields of the specified

send\_email

Send an email to one or more

summarize

Generate a summary of a VTENEXT

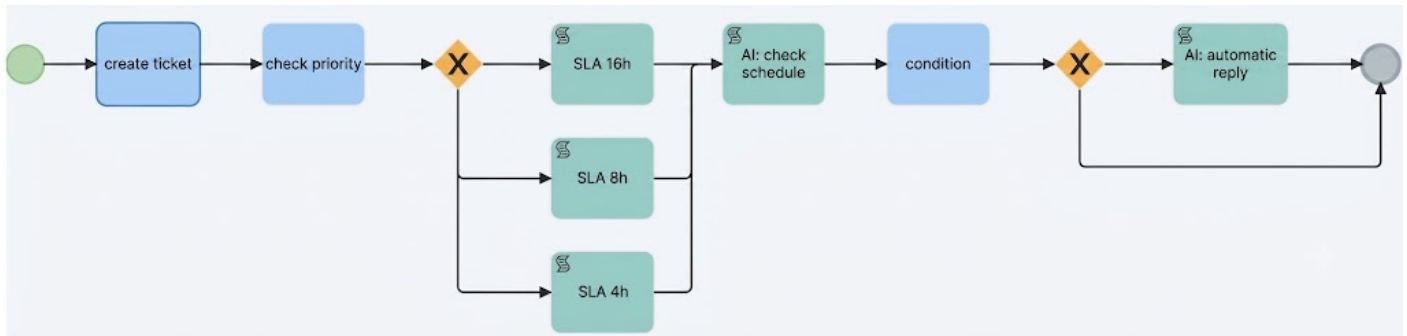
tools\_manual

Fetches additional information

# Implementing the Process on Tickets

We can either implement a new process or integrate the new AI features into an existing one.

In this example, we use an existing process that, when a ticket is created, sets the SLA based on the ticket priority.



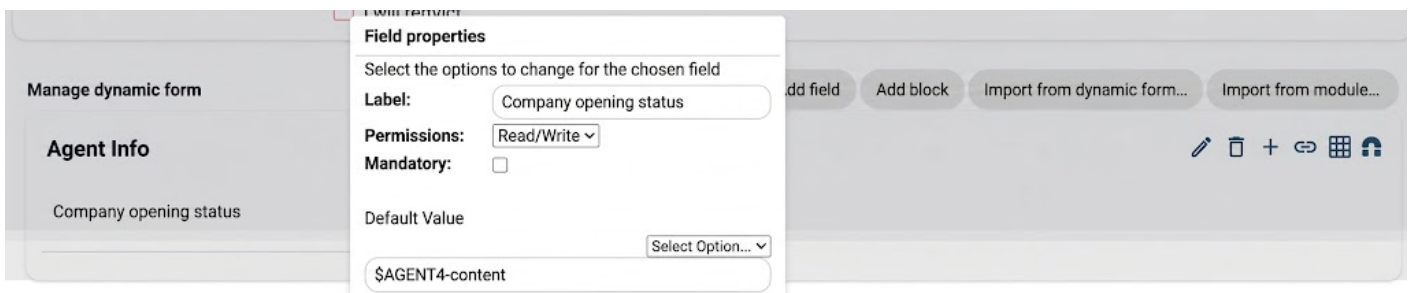
In the **AI: Check Office Hours** task, I execute a call to the agent using the following prompt:

The company is open Monday through Friday from 9:00 AM to 1:00 PM and from 2:00 PM to 6:00 PM, excluding public holidays.

If the ticket was created during business hours, return the string "open"; otherwise, return "closed".

Return exactly one of these strings and do not include any additional text, explanations, or headings.

I then configure the dynamic form with a field populated by the response returned by the LLM.



Next, I configure the condition based on the value of the dynamic form field and set up the following gateway so that, if the agent returns **"closed"**, the process proceeds to the **AI: Automatic Reply** task.

**BPMN-Task: condition** Activity\_0iifa3v

Save Cancel

Entità [DF1] Form dinamica (BPMN-ScriptTask: AI: check orario) ▼

When to esegute the control

☐ on modification

☒ every time the condition is true

Condizioni

New groupo

Company opening status ▼ equals ▼ open

Nuova Condizione

o ▼

Company opening status ▼ equals ▼ closed

New Condition

In the final task of the process, I first call the add\_comment tool to inform the customer that the office is closed. I then call the agent to generate and add a proposed solution to the issue

**BPMN-ScriptTask: AI: auto reply** Activity\_0hfdxyk

Save Cancel Advanced...

AI Actions ▼ Other actions ▼ Import actions from...

Actions

TOOLS	ACTION	DESCRIPTION
	Call process tool	closed company comment
	Call agent	Agent: propose solution

**Action: Call process tool** Save Cancel

Action title closed company comment

Execution mode Standard: the execution process waits for the completion of the action before proceeding with the next one ▼

Tool process add comment ▼

Description tool for adding a comment to a ticket

Input parameters

ticket reference - HelpDesk \$3-crmid Selezione Opzione... ▼ The id of the helpdesk, with the format 'module\_id' + 'x' + 'record\_id'

comment string \$2-answer Selezione Opzione... ▼ comment text

The offices are closed, please contact us tomorrow. For emergencies, please call.

The comment entered here will then be reworked by the LLM as part of the process tool.

## Agent Prompt:

aggiungi un commento con il tool add\_comment proponendo una soluzione al suo problema cercando in user\_manual.

Nel testo non usare mai titoli o codice markdown, restituisci solo testo semplice.

## Test

We can test the process by creating a ticket during closing hours with the title "Login Failed" and the following description:

*"Good morning, this morning I am no longer able to log in to vtenext. Could you please check this urgently?"*

Comments similar to the following will be added:

The screenshot displays a web application interface for managing tickets. On the left is a dark sidebar with navigation icons. The top header includes a search bar and various utility icons. The main content area shows a ticket titled "Ticket login failed". Below the title, there are two comments generated by an AI agent named "Kitt".

**Comment #2 (6 days ago):**

To resolve a failed login, proceed as follows: first, use the "Forgot your password?" function to start the password reset procedure; the reset email is sent to the registered email address (please also check the spam folder). If the login fails due to a lockout from incorrect attempts, the user's IP address is blacklisted after five failed attempts. In this case, the administrator must go to Settings -> Users login control and select the "Return to Whitelist" option to remove the block. The lockout st can be monitored in the User Login History section, where the administrator has the option to add the user back to the whitelist if necessary.

**Comment #1 (6 days ago):**

Our offices are currently closed. Please direct your requests to our customer service starting tomorrow. For urgent matters, you can contact us by phone at the number indicated.

On the right side of the interface, there is a sidebar with "SHORTCUTS" (Comments: 4, Conversations: 3, Notes) and "RELATIONS" (Processes: 3, Activities, Documents, Messages).