SIESTA AI

PRODUCT SPECIFICATION

siesta < labs >

INOVATION LAB & STARTUP STUDIO

Contents

Documentation

Connectors

Release Notes

1. Introduction

This documentation will help you understand and effectively utilize the Siesta AI platform.

1.1 What is Siesta AI?

The product specification provides a comprehensive description of the Siesta AI platform, a tool designed for companies that want to fully leverage the potential of artificial intelligence in their daily operations. The platform builds on the capabilities of current large language models but offers a range of key features that are essential for professional and secure deployment in a corporate environment.

Siesta AI enables companies to securely and scalably connect their own knowledge databases with powerful AI models. In addition to connecting to internal documents or databases, it also supports integration with external tools (such as calendars, email, CRM, etc.). Furthermore, the platform can be controlled directly through Copilot, which is a chat interface used for quick control of the platform and assistants. Users thus interact with an environment that generates responses based on connected knowledge. Thanks to model independence, any LLM can be utilized, whether in the cloud or on private servers, giving companies full control over where and how their data flows. Built-in analytics and feedback loops allow for monitoring operational metrics, evaluating response quality, and continuously improving assistant behavior.

1.2 Main Areas of the Platform

The main areas that the platform focuses on are:

1.2.1 Assistants

Assistants are the fundamental building blocks of the Siesta AI platform. Each assistant represents an individual digital assistant built on a language model, connected to specific data, equipped with its own instructions, and configured to serve its specific purpose. The platform allows for the creation of countless assistants, which can be centrally managed. Each assistant can be designated for a different team, department, language, or business scenario.

Users can create assistants from scratch or from prepared templates, set their behavior, connect datasets, define permissions, and modify their outputs. Assistants can be connected to external tools (calendar, email, CRM, etc.) and can perform specific actions either based on user commands or automatically through a scheduler. The configuration of each assistant includes its own system instructions, model choice, output format, creativity level, data access, and options for public or internal deployment.

Each assistant has its own detail. This section serves as a central place for managing assistants and is divided into several functional tabs covering all areas, from basic information to analytics and feedback to change history.

Assistants can be accessed via chat, deployed as a public widget, integrated into Slack, Teams, or other communication channels, or worked with directly through an API. The platform also allows for connecting the assistant to specific tools that it can actively utilize.

1.2.2 Integrations, Tools, and Automation

One of the key areas of the Siesta AI platform is the ability to connect assistants to external tools, systems, and services. Thanks to this feature, assistants are not just a conversational interface but active elements of the digital infrastructure that can perform specific tasks, transfer data, or respond to real-time events.

Each assistant can be assigned one or more tools that it can use during interactions or automatically in the background. For example, integration with a calendar, file storage, internal systems via API, or external webhooks. Tools are centrally managed within the Tools interface and can be allocated across assistants.

1.2.3 Data

With Siesta AI, it is possible to connect artificial intelligence to the specific knowledge of the organization. Assistants do not respond based on a general model but draw information from precisely defined datasets. These datasets can contain both uploaded documents and connections to live systems and knowledge bases that the company uses in its regular operations.

The platform allows for the creation, management, and configuration of so-called datasets - logically separated sets of information that are subsequently assigned to specific assistants.

Each assistant can be assigned which datasets it is allowed to use, thus precisely controlling what information it has access to. This approach not only increases the relevance of responses but also ensures a high level of security, as no assistant has access to data that has not been explicitly assigned to it.

1.2.4 Smart Chat

The Siesta AI platform provides a unified interface for communication with AI assistants, available to both internal users and external visitors. The goal of this part of the system is to create an environment where users can naturally communicate with individual assistants, receive relevant responses based on company data, and provide feedback that serves to further improve their behavior.

The platform also includes a so-called smart chat, which is a public anonymous interface that can be embedded on websites or internal portals of the organization. This chat is accessible without logging in and can be used, for example, for customer support, career communication, or as a tool for answering frequently asked questions. The public chat plugin is fully integrated with the platform, maintains all security principles, allows for feedback collection, and is connected to audit logs.

The platform also supports access to assistants via API or integration into commonly used tools such as Slack or Microsoft Teams. This allows users to communicate with assistants directly from the environment they use daily, without the need to log into a new application. Each assistant can be deployed across different channels simultaneously, while the logic, data, and behavior remain consistent.

1.2.5 User and Access Management

The Siesta AI platform allows for detailed management of users, their roles, and access permissions. Each user in the system is assigned a role that determines what functions, data, and assistants they have access to.

Administrators can create and modify user accounts, set permission granularity, and precisely define who can modify assistants, work with datasets, view feedback, or access system settings.

Access can be controlled not only at the platform level but also within individual components, such as at the level of a specific assistant or dataset.

1.2.6 Feedback

One of the key features of the Siesta AI platform is the ability to work with user feedback and continuously improve the behavior of individual assistants based on it. Each assistant's response can be rated by the user - either as positive or negative. In the case of a negative rating, the user can attach a comment explaining why the response was unclear, incorrect, or unhelpful.

1.2.7 Security and Audit Logs

Every important operation in the system - whether it involves modifying an assistant, changing permissions, working with data, or generating responses - is recorded and traceable.

The system includes detailed audit logs that capture who did what, when, and over what. These records are available to administrators in a clear format and allow for retrospective review of all changes in the system.

The platform also allows for the management of API keys, access control through roles, and support for single sign-on (SSO). The security policy meets the needs of organizations that emphasize data control, auditability, and operational reliability.

1.2.8 Analytics and Reporting

The Siesta AI platform includes built-in tools for monitoring traffic, user behavior, and the performance of individual assistants. The goal of this part of the system is to provide administrators and managers with an overview of how assistants are being used, what impact they have, and where there is room for improvement.

1.3 Getting Started

To get started, we recommend going through the Login and Controls sections.

The documentation is continuously updated. For the latest information, please contact us at info@siesta.ai.

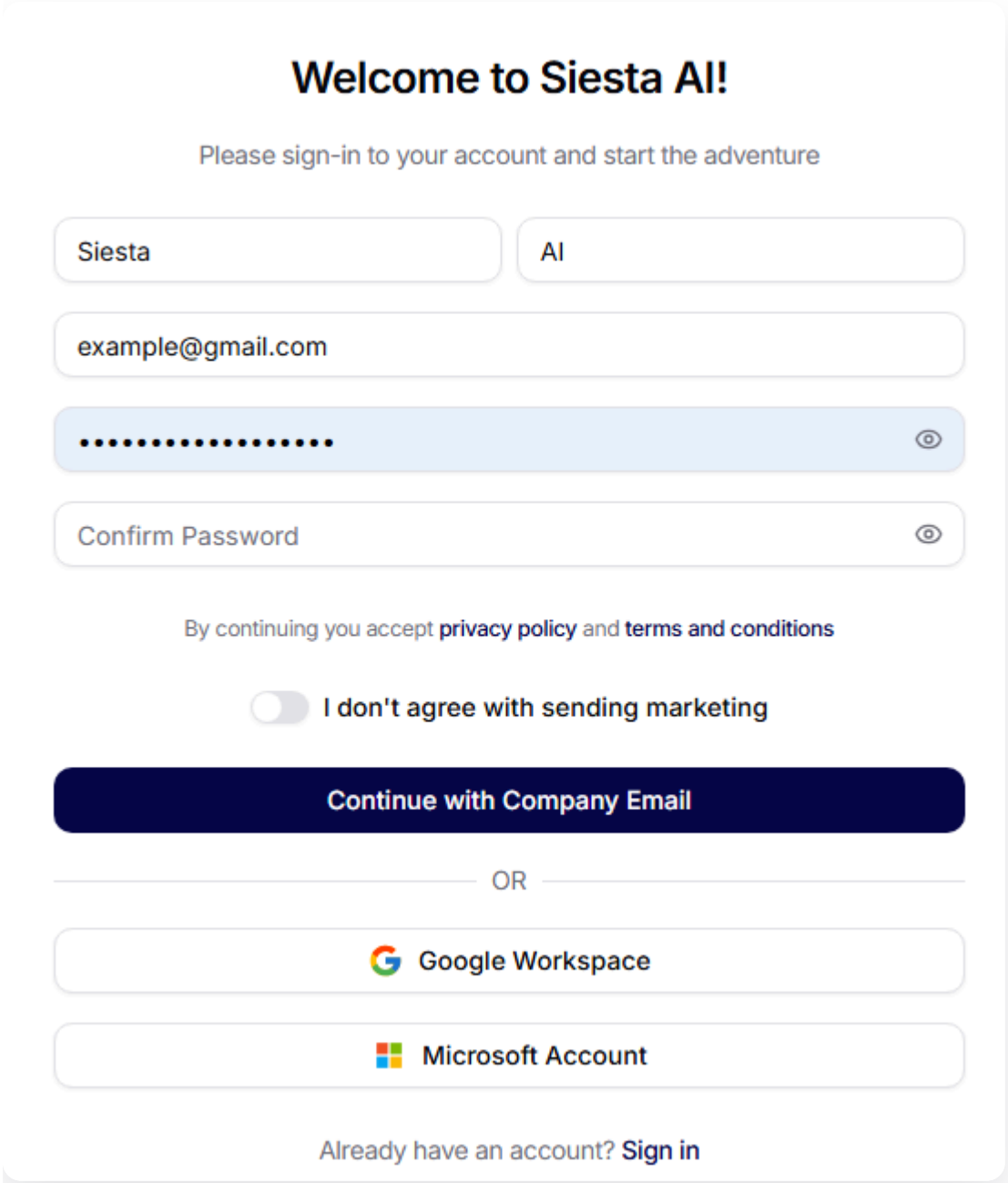
2. Login

2.1 Registering a New Account

To create a new account, please fill in:

- **Name** and **Surname**
- **E-mail**
- **Password** and **Confirm Password**

Confirm your agreement with the terms, optionally decline marketing, and click on **Continue with Company Email**. Alternatively, you can use login via Google or Microsoft.



2.2 Logging into the Application

To log into the platform, enter:

- **E-mail**
- **Password** (can be shown/hidden using the eye icon)

Then click the **Login** button, which will verify your details and log you into the system.

On the login screen, the following options are available:

2.2.1 Remember Me

By checking this option, you will remain logged in even after closing the browser.

2.2.2 Forgot Password?

The link will open a form for password recovery if you do not remember your password.

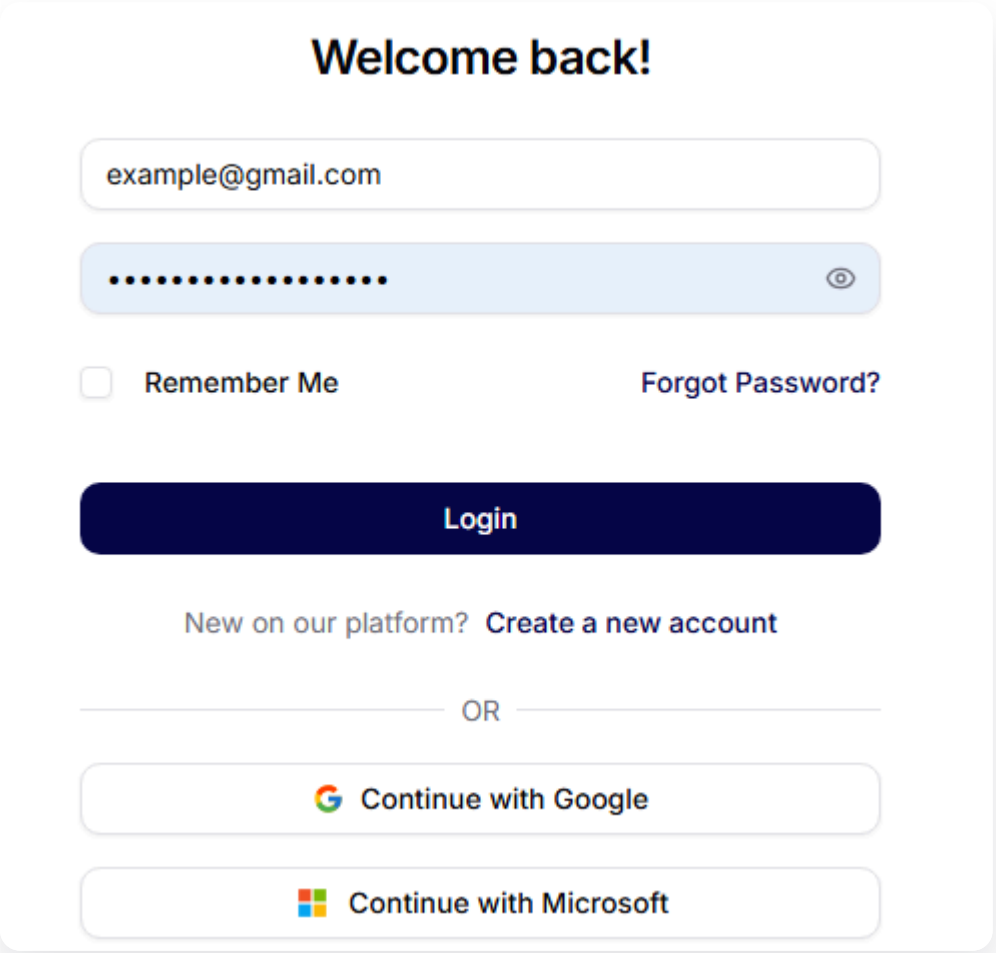
2.2.3 Create a New Account

If you do not have an account yet, go to registration and create access.

2.2.4 One-Click Login

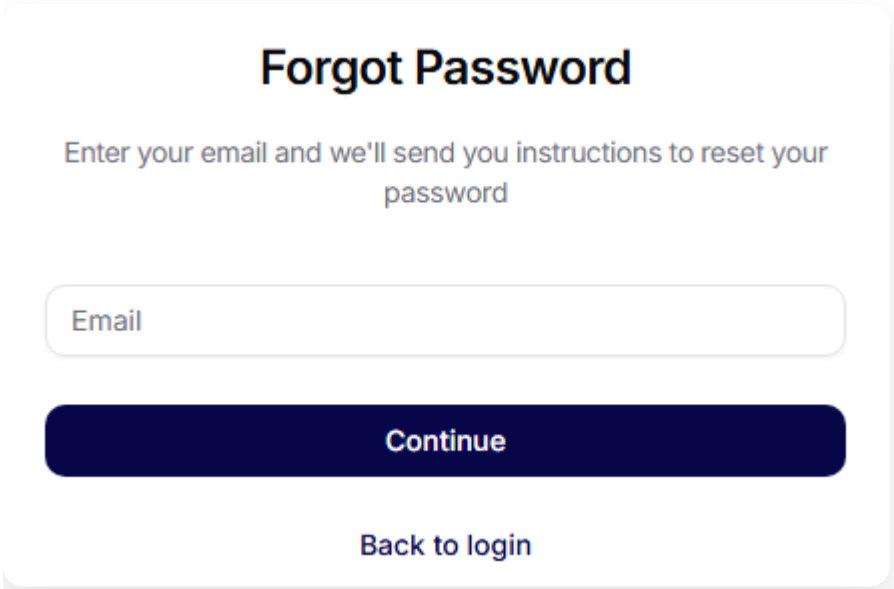
- **Continue with Google**
- **Continue with Microsoft**

> If the login fails, an error message will appear regarding invalid or missing details.



2.3 Forgot Password

In case you forget your password, you can reset it. Enter the email associated with your account, and after submission, you will receive an email with a unique link to set a new password.



2.4 Updating Password

After opening the unique link from the email, fill in:

- **New Password**
- **Confirm New Password**

By clicking the **update password** button, you will save the new password.

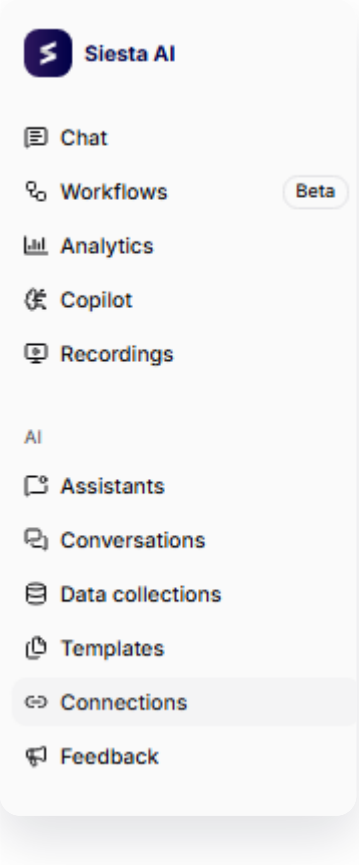
3. Controls

The current appearance of the application screen after logging in.

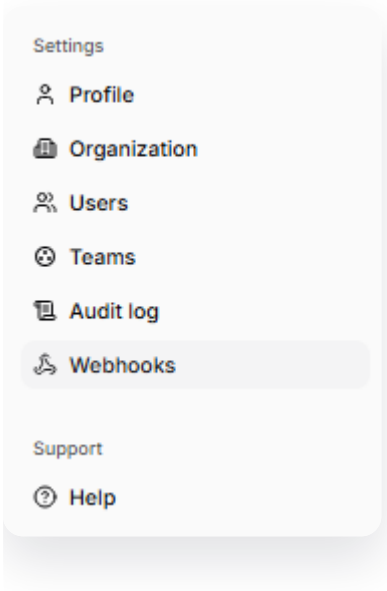
3.1 Left Panel

- Siesta AI section with items: Chat (active), Workflows (Beta label), Analytics, Copilot, Assistants, Conversations, Data, Templates, Connections, Feedback.
- Settings section: Profile, Organization, Users, Teams, Audit log.
- Support section: Help.
- At the bottom, the user card; when expanded, a menu appears with options: Switch to dark mode (crescent moon icon), language selection "English" with code "us", and the item Log out.

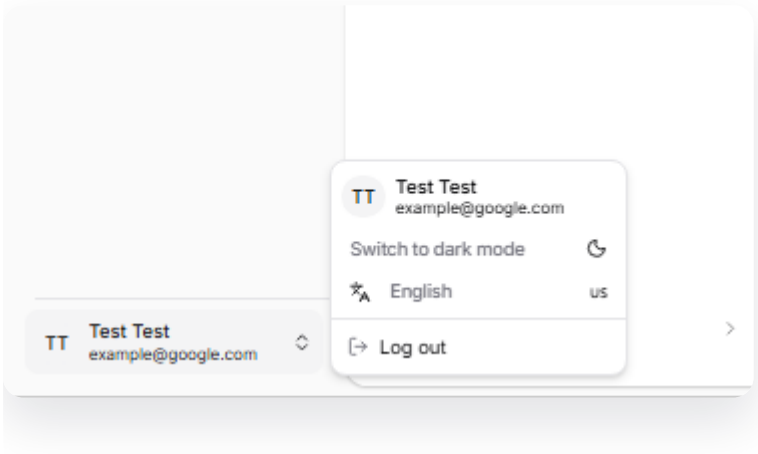
Example of the main navigation in the left panel.



Example of the Settings section in the left panel.

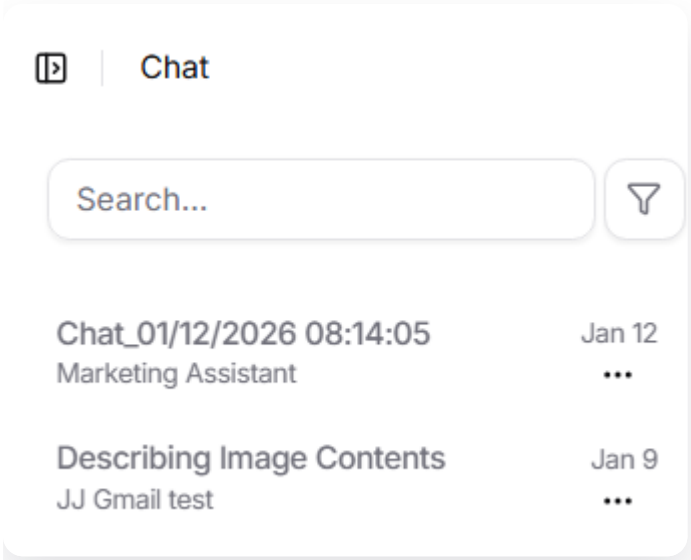


Example of the user menu at the bottom of the panel.



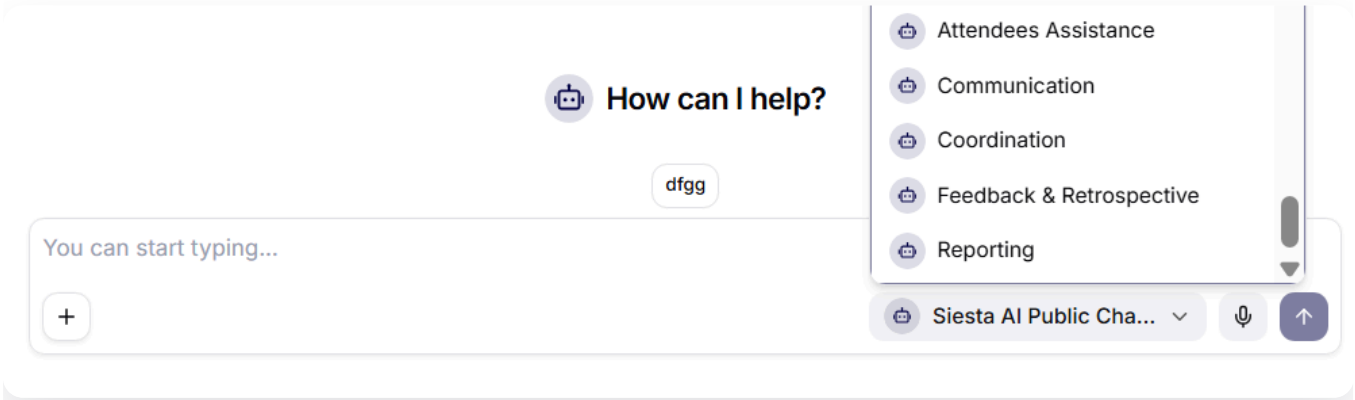
3.2 Conversation List

- Header "Chat" and search field "Search" with a filter icon.
- Conversation items in the format `Chat_12/09/2025 14:03:03` with the label "Agent framework test", assistant icon, and date (e.g., Dec 9); the active row is highlighted.
- On the right side of the rows is a menu (three dots) and at the bottom is pagination "Page 1 of 30" with arrows.



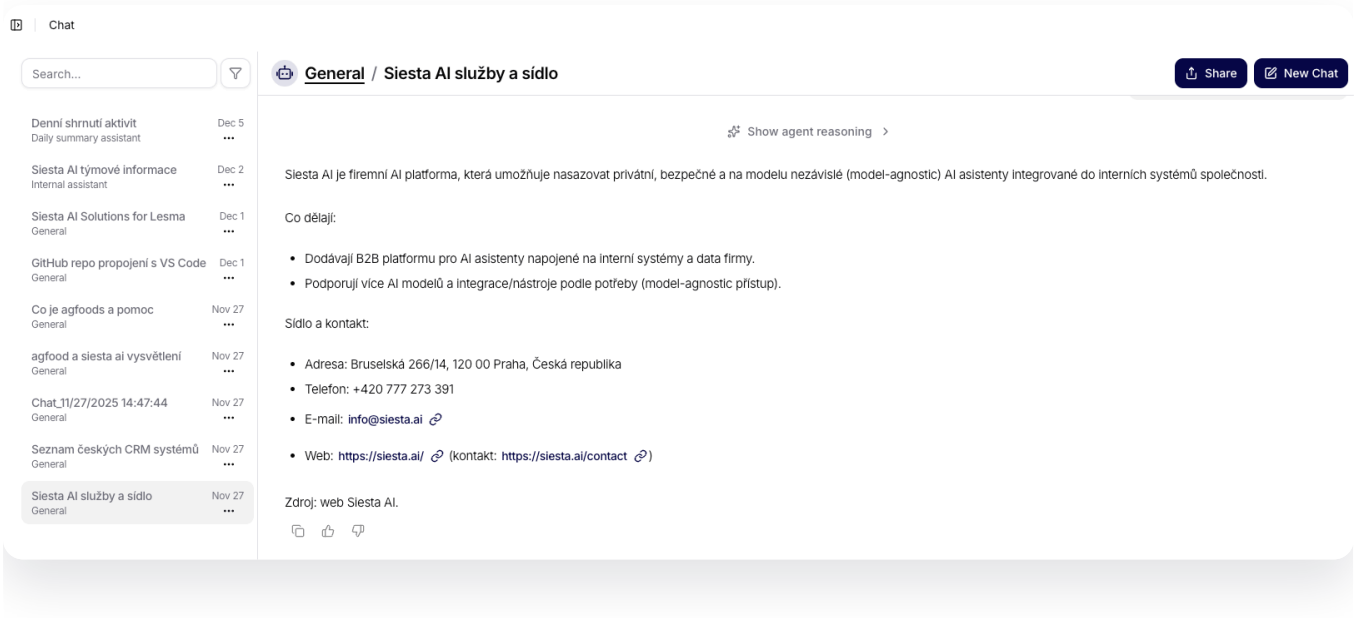
3.3 Input Field

- Title "How can I help?" with an assistant icon.
- Below the title, a set of suggestions (chips) with questions in Czech, e.g., "What is the biggest bottleneck in modern web applications?", "What technologies are currently shaping the development world?", "Why is the market rising or falling today?".
- Input field with a placeholder "You can start typing", on the left a "+" button to add content.
- On the right inside the field, agent selection "Chat Bot" with an avatar and a dropdown arrow, next to it an icon for the microphone and send (up arrow).



4. Chat

The Chat section is used for conversations with AI assistants that are created and configured in the Assistants section of the platform. Users can start new conversations, browse the history of previous chats, and provide feedback on responses.



4.1 Starting a New Conversation

A new chat is initiated using the **New Chat** button. After clicking it, a modal window appears where the user:

- enters the conversation title,
- selects the assistant they want to communicate with (from the list of previously created ones),
- optionally has the option to go directly to creating a new assistant.

By confirming the selection with the **Submit** button, the chat is initialized, and the main conversation interface opens for the user.

4.2 Chat Interface

The main screen is divided into two parts:

- **on the left**, the history of all conversations is displayed (including the title and date),
- **on the right**, the actual communication with the selected assistant takes place.

The user types their questions into the input field at the bottom of the screen and sends them by pressing the **Send** button. There is also an option to activate voice input or attach an additional file.

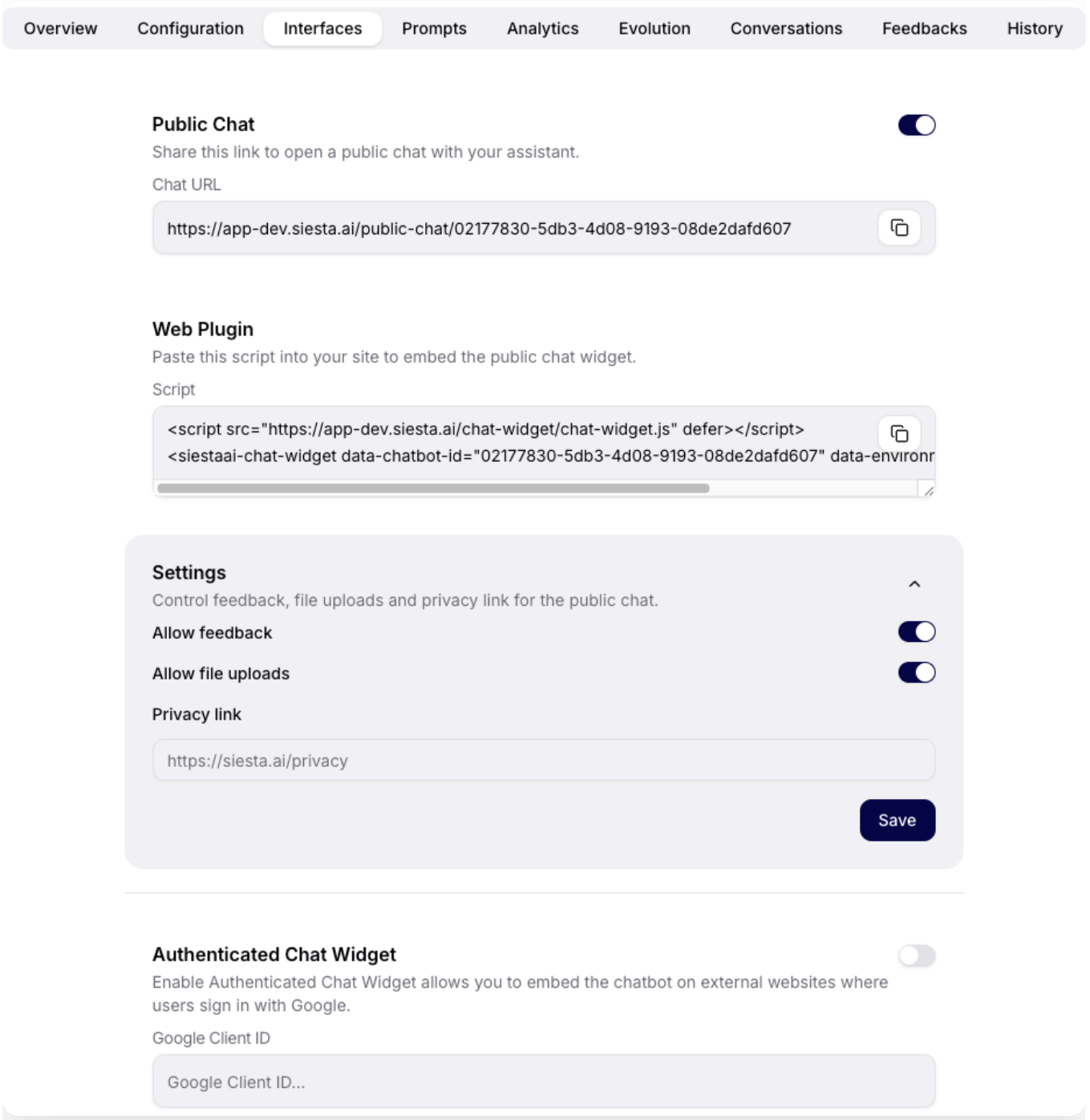
The assistant responds in real-time, with each message being retained within the given conversation.

4.2.1 Interface Settings (Public Chat and Widget)

In the assistant detail under the **Interface** tab, you can set how the chat will be available to users outside the application:

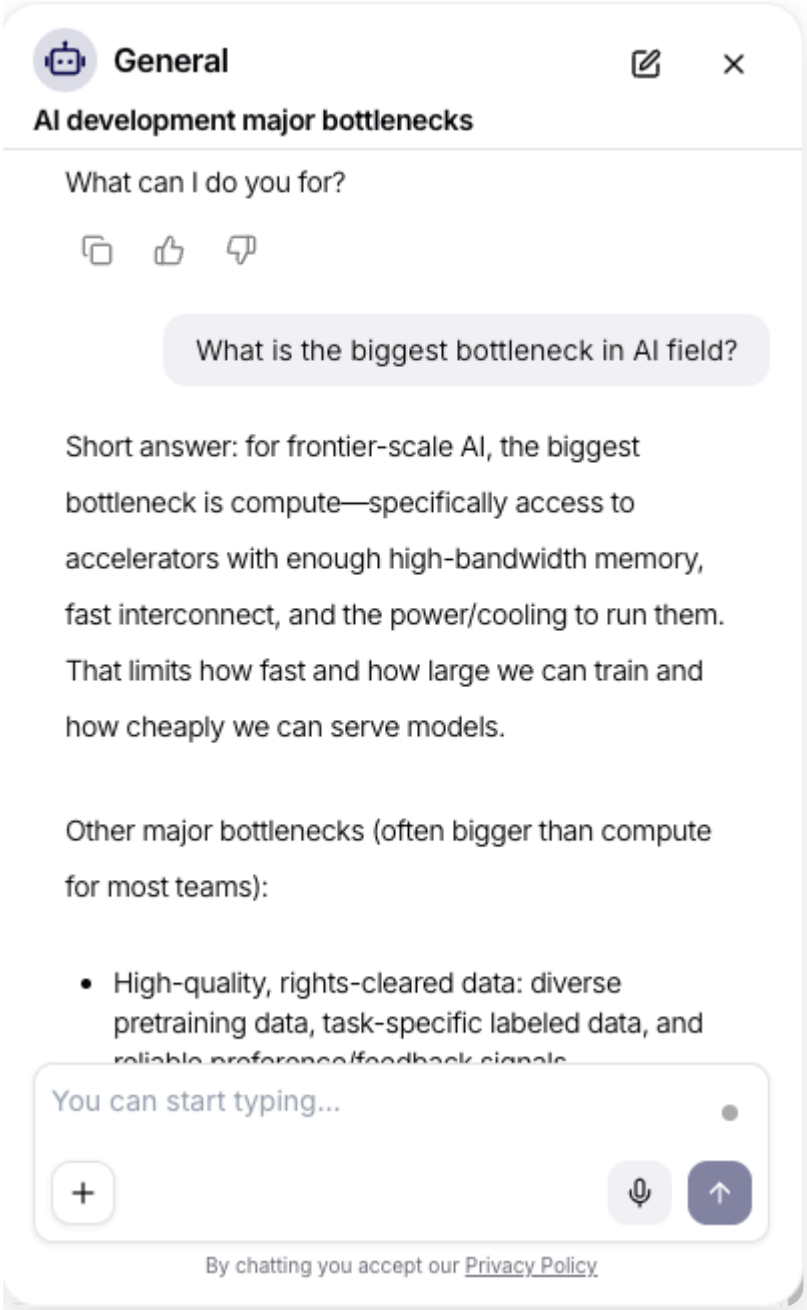
- **Public Chat** – the switch activates the public chat and generates a **Chat URL** with a button to copy the link.
- **Web Plugin** – an embeddable script for embedding the chat on an external website.
- **Settings** – settings for the behavior of the public chat, including switches for **Allow feedback** and **Allow file uploads**, and a **Privacy link** field with a link to the privacy policy.
- **Save Changes** – confirm changes in the Settings section with the **Save** button.
- **Authenticated Chat Widget** – a switch for the widget with login; you will need to provide the **Google Client ID** and use a second embed code.

After activating **Public Chat**, a direct link to the chat will be displayed, which can be copied with a single click. In the **Web Plugin** section, an embed script is available for inserting the widget on the website and an icon for quick copying. The **Settings** section is used to manage the functions of the public chat (feedback, file uploads) and to set the link to the privacy policy.



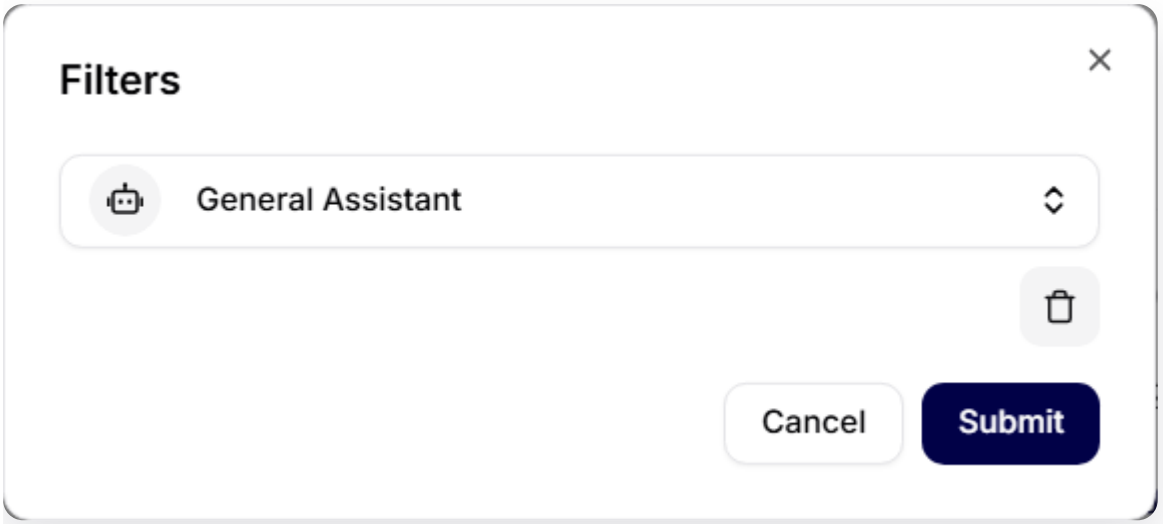
4.2.2 Public Chat Plugin

The publicly accessible chat interface can be easily embedded on a website (for example, as an iframe or using a simple embed script). It does not require registration or login and allows visitors to communicate with AI assistants. The chat functionality corresponds to the internal chat, including feedback options and audit logging. If sensitive data is being handled, login via Google OAuth can be enforced for this chat.



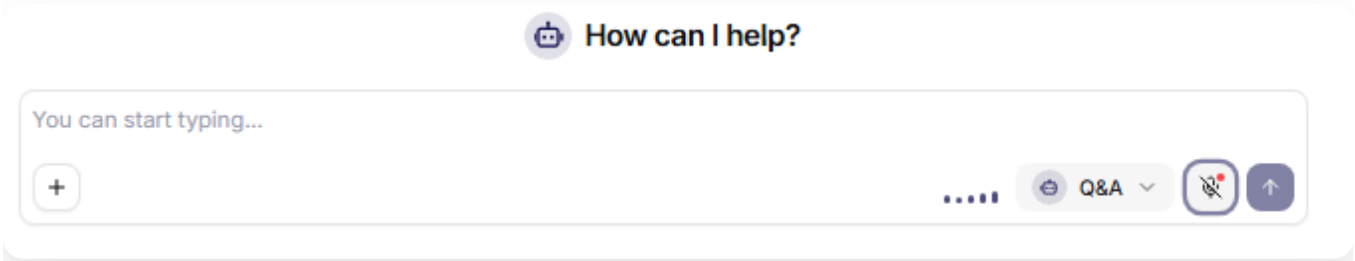
4.2.3 Filtering and Managing Conversations

- The left panel allows for a quick review of the history, including titles and the date of the last activity.
- By clicking on **...** next to a conversation, you can quickly **rename** or **delete** the chat.
- Filters (icon in the top panel of the list) allow you to switch the view to a specific assistant or category (e.g., *General* vs. *Work*). The selected assistant can be removed by clicking on the trash icon and confirming the change with the **Submit** button.



4.2.4 Entering a Query and Switching Assistants

- The input field supports text, attachment uploads, and voice dictation (microphone).
- To the right of the text, you can switch the active assistant without needing to start a new chat.
- Sending a query: arrow or keyboard shortcut **Enter** / **Cmd + Enter** (depending on settings).



The microphone status (active/disabled) is visible directly next to the assistant selection. When recording, the icon activates, and the status is displayed in the input field.

4.3 Feedback on Responses

Under each response from the assistant, users can click on the thumbs up or down icon, providing quick feedback on the given response.

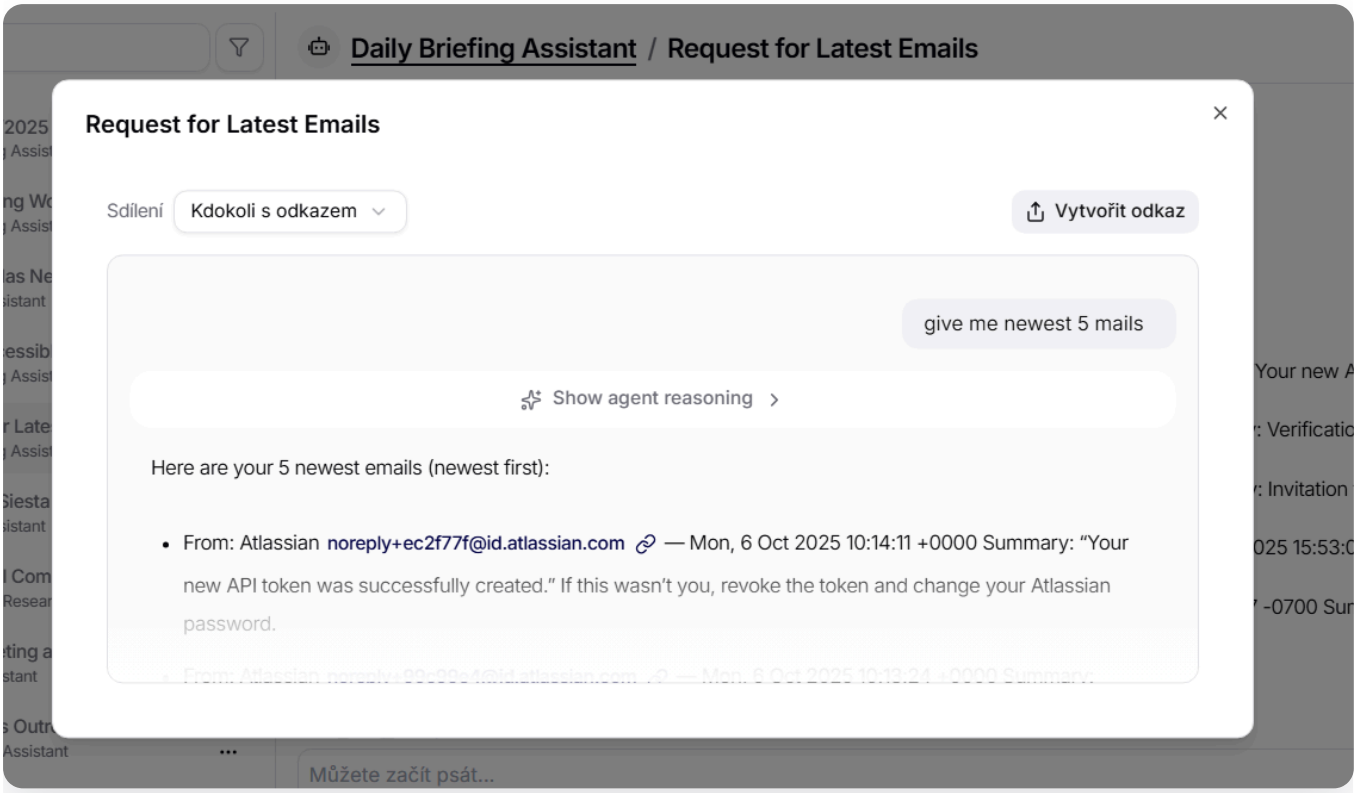
In the case of a negative rating, a **New Feedback** window will also appear, where specific comments can be added explaining why the response was not relevant, accurate, or expected. This feedback is automatically sent to the administrative interface upon submission.

Thanks to this mechanism, administrators can monitor the quality of responses, analyze deficiencies in the underlying data or information, and optimize the settings of the assistants.

4.4 Sharing a Conversation

The chat can be shared via a link. In the conversation detail, click on **Share** and choose who can open the link. A preview shows the content of the shared chat, including the latest messages and the prompt that

initiated the conversation.

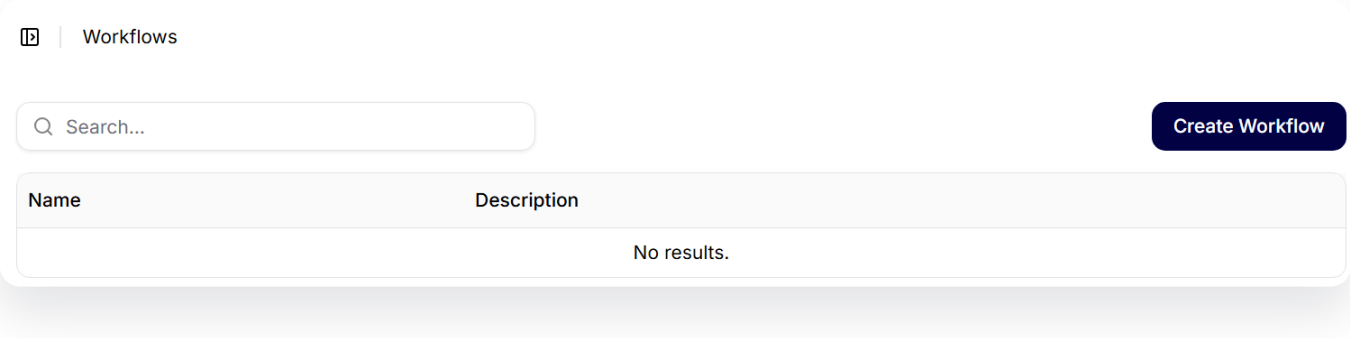


5. Workflows

Workflows allow you to compose integration flows from pre-prepared actions (HubSpot, Jira, Google Workspace, and others), and the orchestration is then triggered by assistants or directly by users. This section is currently in beta mode.

5.1 Where to find them

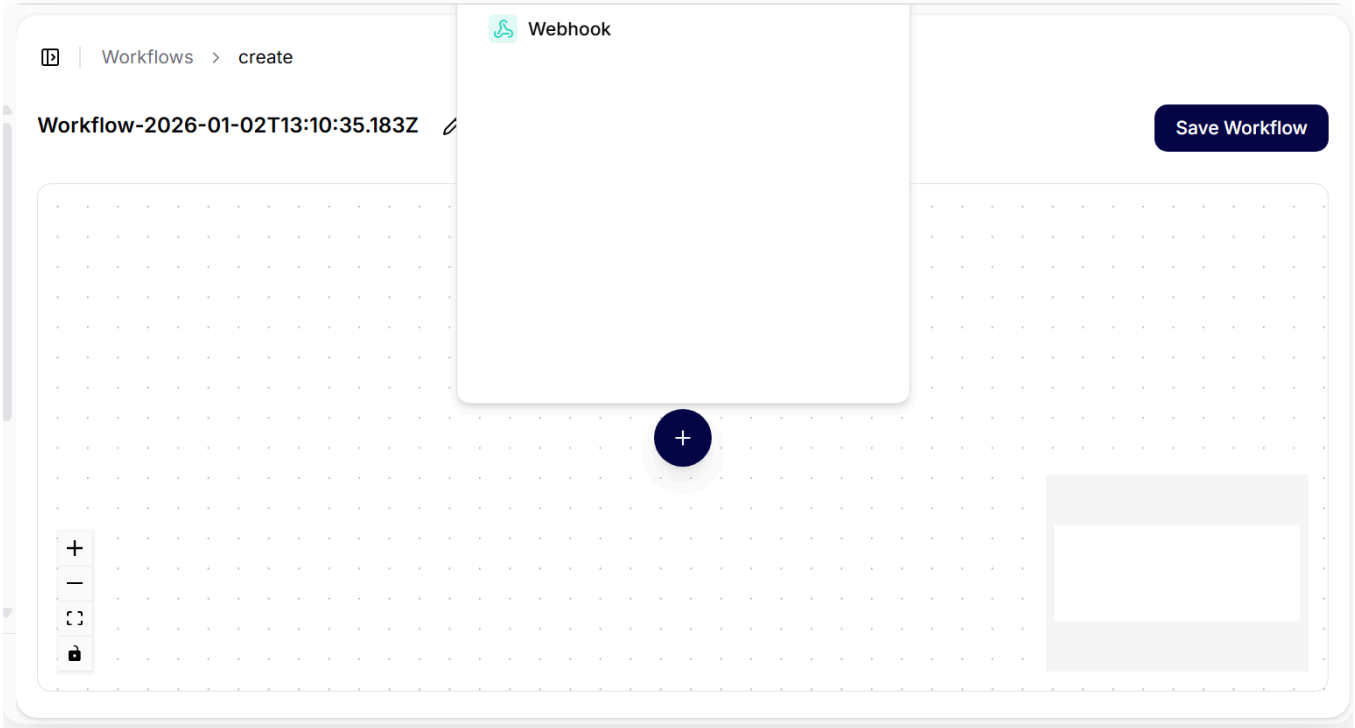
- In the left menu, click on **Workflows (Beta)**.
- A list of existing workflows will be displayed, along with search, pagination, and the **Create workflows** button. An empty list shows the text **No results**.



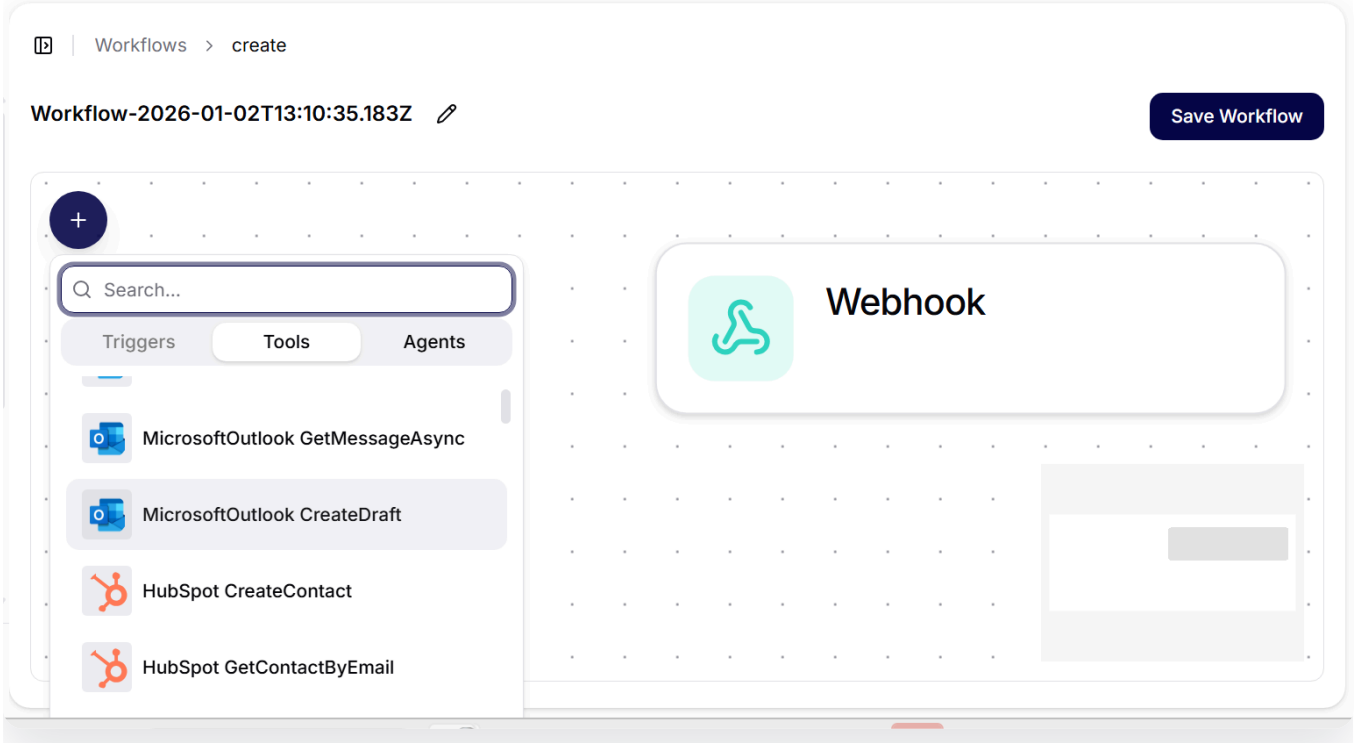
5.2 Creating a new workflow

1. Click on **Create workflows**.
2. In the right panel, fill in the **Name** and **Description** – these are also used for searching.
3. Drag actions from the left panel **Connections** onto the canvas (e.g., HubSpot, Jira, Google Calendar, Google Drive).
4. Connect nodes by dragging lines – the output of the previous action is the input for the next.
5. In the node settings, fill in parameters (record IDs, emails, calendars, projects, etc.).
6. Save by clicking on **Save workflows**.
7. Zoom controls are at the bottom left, and the mini-map is at the bottom right.

Empty canvas with the first trigger:



Adding a node from the catalog:



5.2.1 Typical actions and examples

- **HubSpot**: GetDealById, GetContactById – reading a deal/contact before passing it to other systems.
- **Jira**: GetUserAsync, AssignTicketAsync, CreateTicketAsync – enriching a contact or creating a ticket.

- **Google Calendar**: CreateEventAsync – creating a meeting after successful data enrichment.
- **Google Drive**: ListFilesAsync, ReadFileAsync – working with documents.
- **LLM / Webhook**: calling a model or webhook to supplement logic, validate, or notify. The procedure for creating a webhook can be found on the Webhooks page.

5.2.2 Best practices

- **Input validation**: verify IDs, emails, and required parameters before connecting additional nodes.
- **API errors**: account for errors from integration services (timeout, rate limit) and add fallback.
- **Naming**: name nodes according to their function (e.g., “Find HubSpot Contact”, “Create Jira Ticket”).
- **Security**: work only with the access rights necessary for the given workflows; keep sensitive values in a vault/secrets.

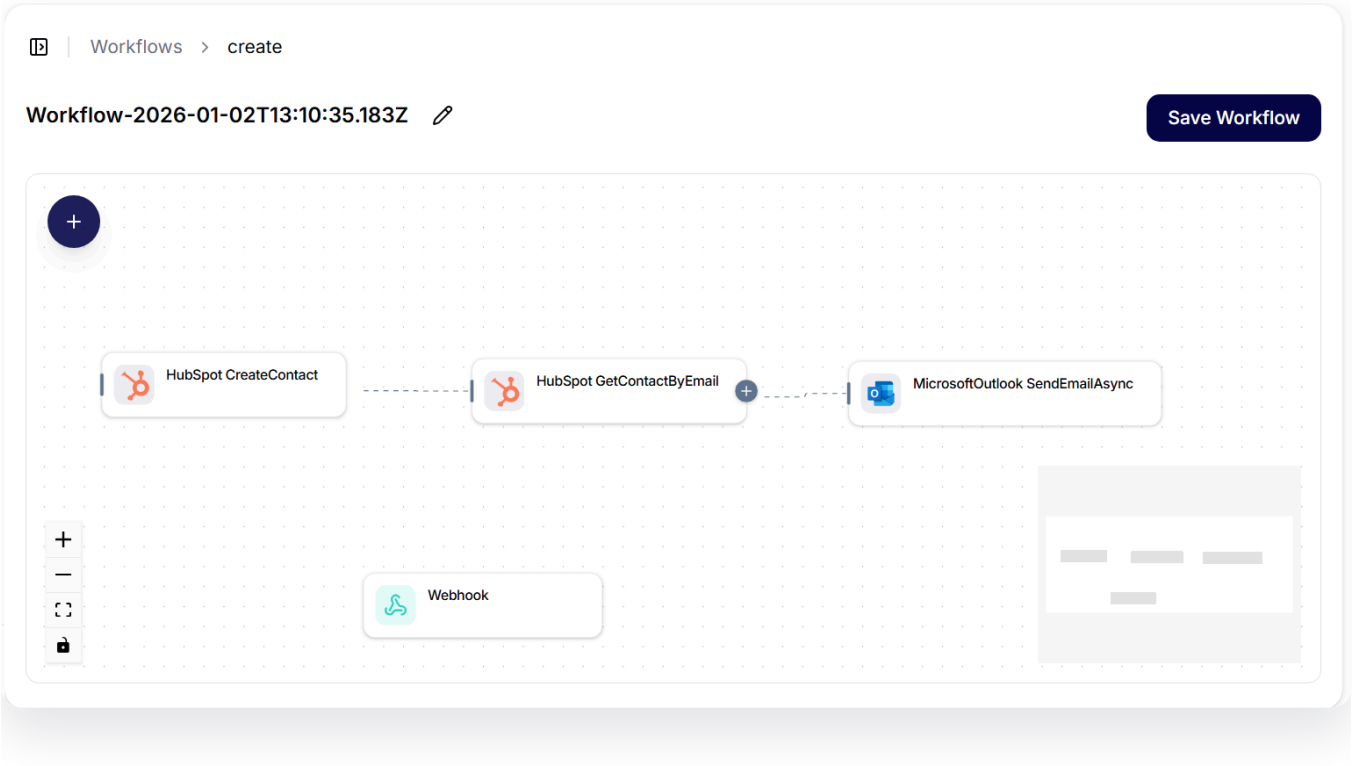
5.3 Editing and management

- In the list of workflows, search for a name/keyword and open the item.
- You can edit, save, and rerun the workflow – changes will take effect in new runs.
- Recommended: after significant changes, test the flow on non-production data (test deals/tickets/calendar).

5.4 Common scenarios

- **Sync HubSpot → Jira → Calendar**: retrieving a deal and contact, finding a user in Jira, creating a ticket and a meeting.
- **Incident intake**: creating a ticket, attaching files from Drive, and notifying via webhook/LLM.
- **Onboarding**: creating a user in internal systems, adding to groups, and scheduling an introductory meeting.

Sample workflows:



6. Analytics

6.1 What You Will Find on the Page

Analytics provides a quick overview of what is happening in the assistants. At the top, you will see the basic KPIs:

- **Conversations**: the number of conversations that have taken place and the daily change.
- **Messages**: the total number of messages from the last few days.
- **Data Sources**: how many sources you have connected (e.g., files, databases, APIs).
- **Assistants**: the number of active assistants.

Take these numbers as a “health check” – if any indicator unexpectedly drops, it is a signal to verify the configuration or content.

6.2 Monthly Overview of Conversations

Below the KPIs is a bar chart with a monthly summary of conversations. The X-axis displays the months, and the height of the bars indicates the volume of conversations. For quick diagnostics:

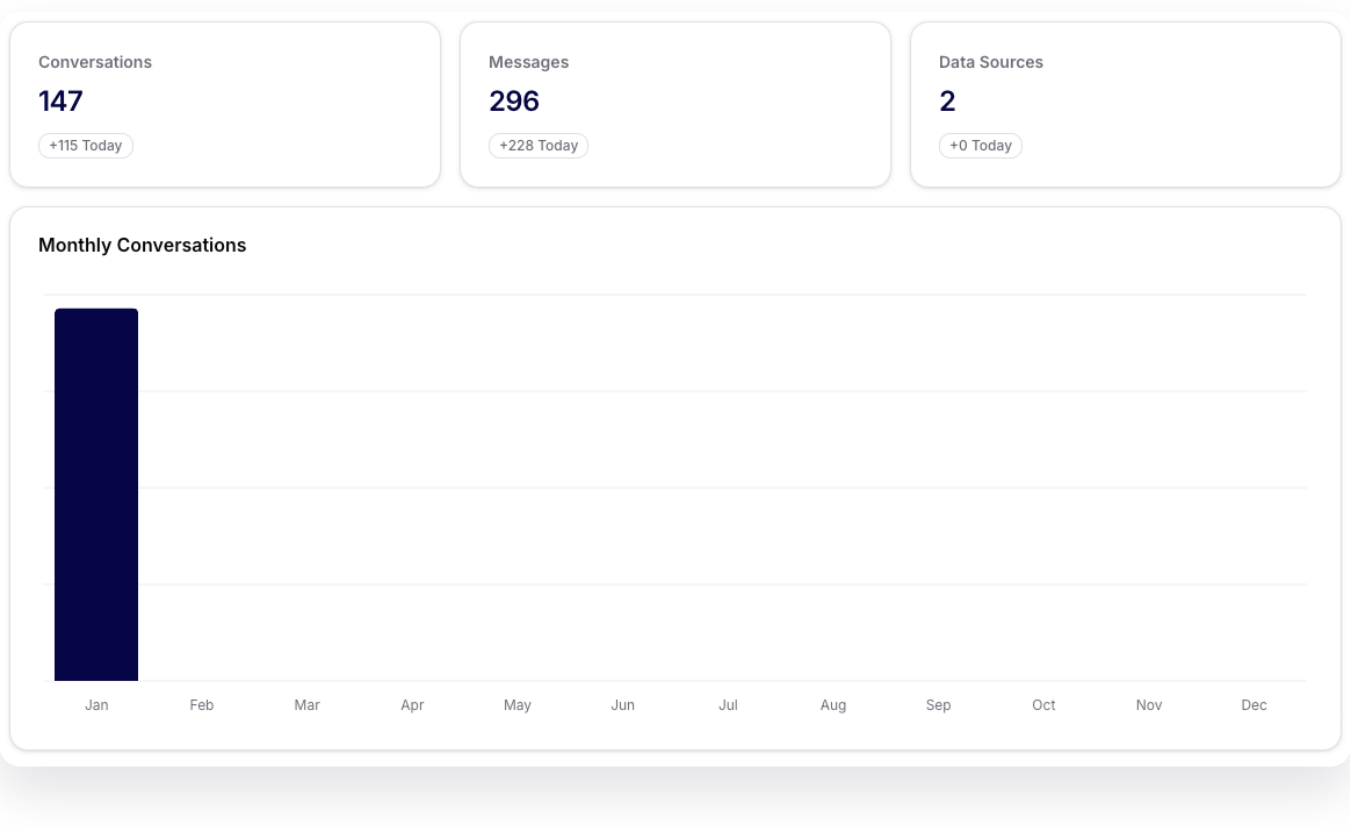
- Sharp drop = check the availability of assistants, connected channels, or recent changes in propositions.
- Growth = verify whether capacity (rate limits, resources) is keeping up.

6.3 Recent Negative Feedback

On the right, there is a box with the five most recent negative message ratings. Use it for quick iterations:

1. Open a specific conversation.
2. Find the message that the user rated negatively.
3. By adjusting the prompt or source data, you can reduce similar complaints in the future.

6.4 Dashboard Example



6.5 Tips for Working with Data

- Monitor **daily changes** in KPIs to quickly identify fluctuations.
- If the number of messages is increasing without a growth in conversations, check the quality of responses (feedback) and possibly adjust the instructions.
- With zero data sources, verify that assistants have the correct datasets and access assigned.

7. Recordings

The Recordings module is used for recording, storing, managing, and automatically transcribing audio content in the Siesta AI application. Each recording is processed by AI upon insertion, and a text transcript is created, which can be further used in assistants, workflows, or analytics.

The section includes:

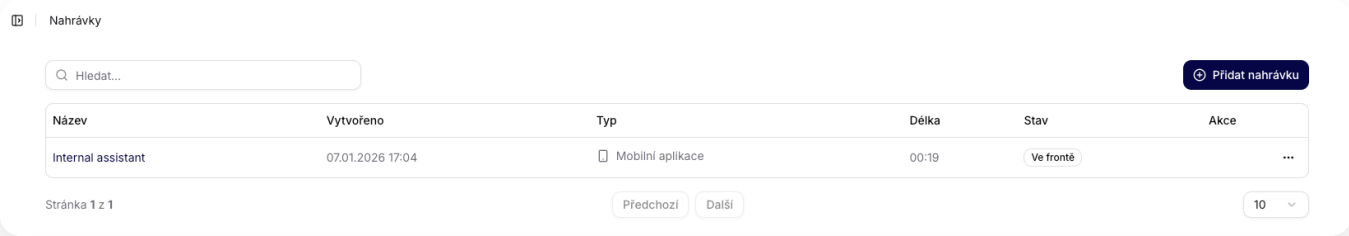
- Overview of all recordings in a table
- Recording detail with a player and transcript
- Dialog for adding a new recording

7.1 Overview of Recordings

The main screen displays a list of recordings in a table with the following columns:

- **Title** – The title of the recording entered by the user
- **Created** – Date and time of creation
- **Type** – Type of recording (e.g., mobile app, call)
- **Duration** – Total length of the audio recording
- **Status** – Current processing status:
  - Queued – waiting for processing
  - Processing – transcription in progress
  - Completed – transcription is finished
- **Actions** – Additional operations (e.g., detail, delete)

Recordings can be searched using the search field. The **Add Recording** button is used to insert a new recording.



7.2 Adding a New Recording

After clicking on **Add Recording**, a dialog opens with the following fields:

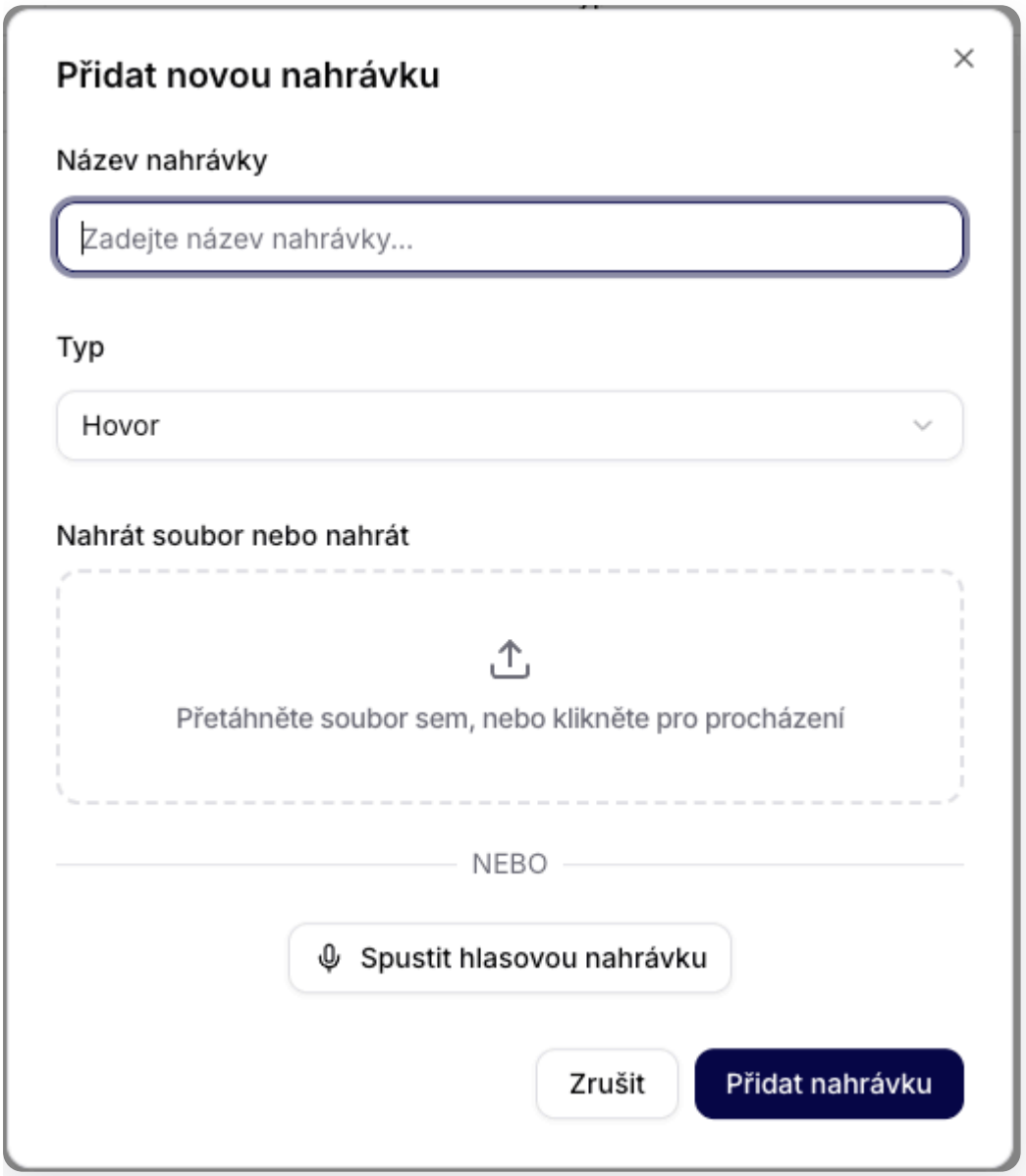
Fields

- **Recording Title** – Required field for entering the title.

- **Type** – Dropdown list for specifying the type of recording (e.g., Call).
- **Upload File or Record** – Options:
  - **Upload File** – Dragging a file or clicking to select from the computer
  - **Start Voice Recording** – Recording sound directly from the microphone

Actions

- **Cancel** – Closes the dialog without saving
- **Add Recording** – Saves the recording and starts processing



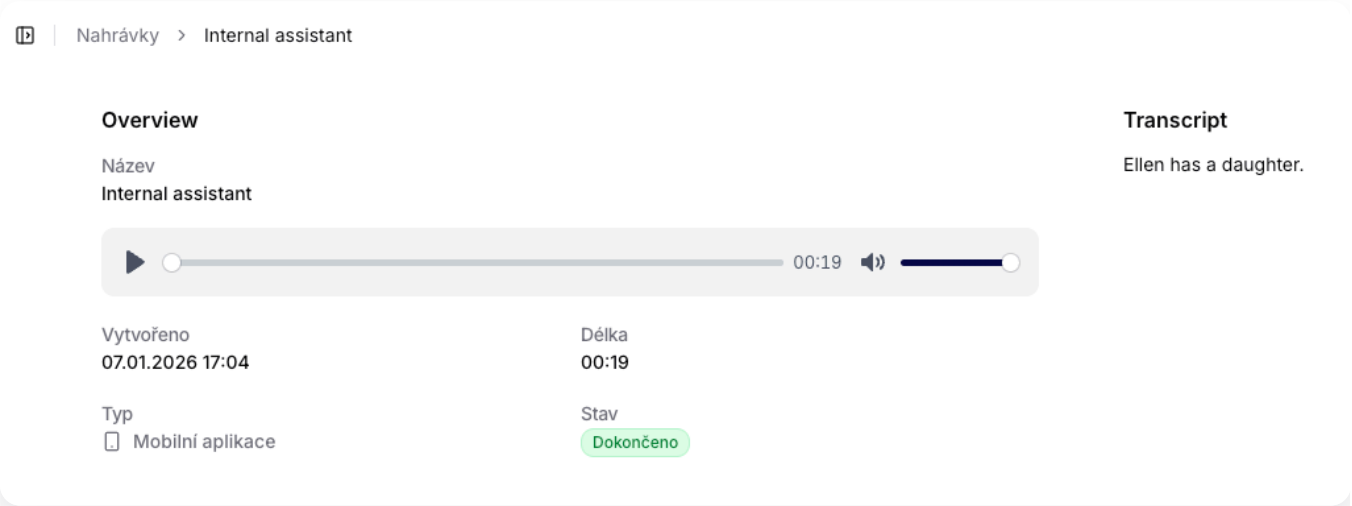
7.3 Recording Detail

Each recording has its own detail page, which includes:

- **Player**

  - Playback controls (Play / Pause)
  - Playback progress slider
  - Volume control
  - Display of duration and current position
- **Metadata**
  - Title
  - Created (date and time)
  - Duration
  - Type
  - Status

- **Transcript** – Text transcript of the audio recording generated by AI. Once processing is complete, the transcript is displayed on the right side of the screen.



7.4 Processing States

Each recording goes through the following steps:

- **Uploaded** – file has been received
- **Queued** – waiting for processing
- **Processing** – transcription in progress
- **Completed** – transcription is finished and available

7.5 Typical Use Cases

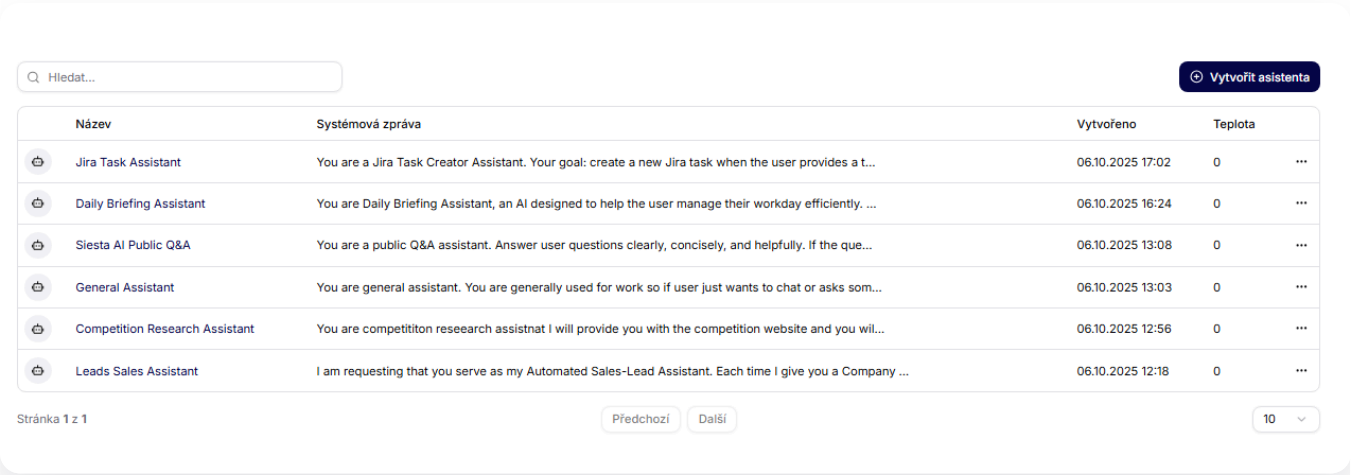
The Recordings section is suitable for:

- Recording meetings
- Capturing interactions from mobile applications

- Transcribing calls or conversations
- Creating data for AI assistants and workflows
- Archiving voice notes

8. Assistants

Each assistant has its own detailed interface, which is divided into several thematic subsections. These allow users to manage configuration, analyze performance, respond to feedback, or track change history in a clear manner.



Below you will find a description of the individual tabs of the assistant detail according to the current application layout (Overview, Configuration, Interface, Prompts, Analysis, Evolution, Conversations, Feedback, History).

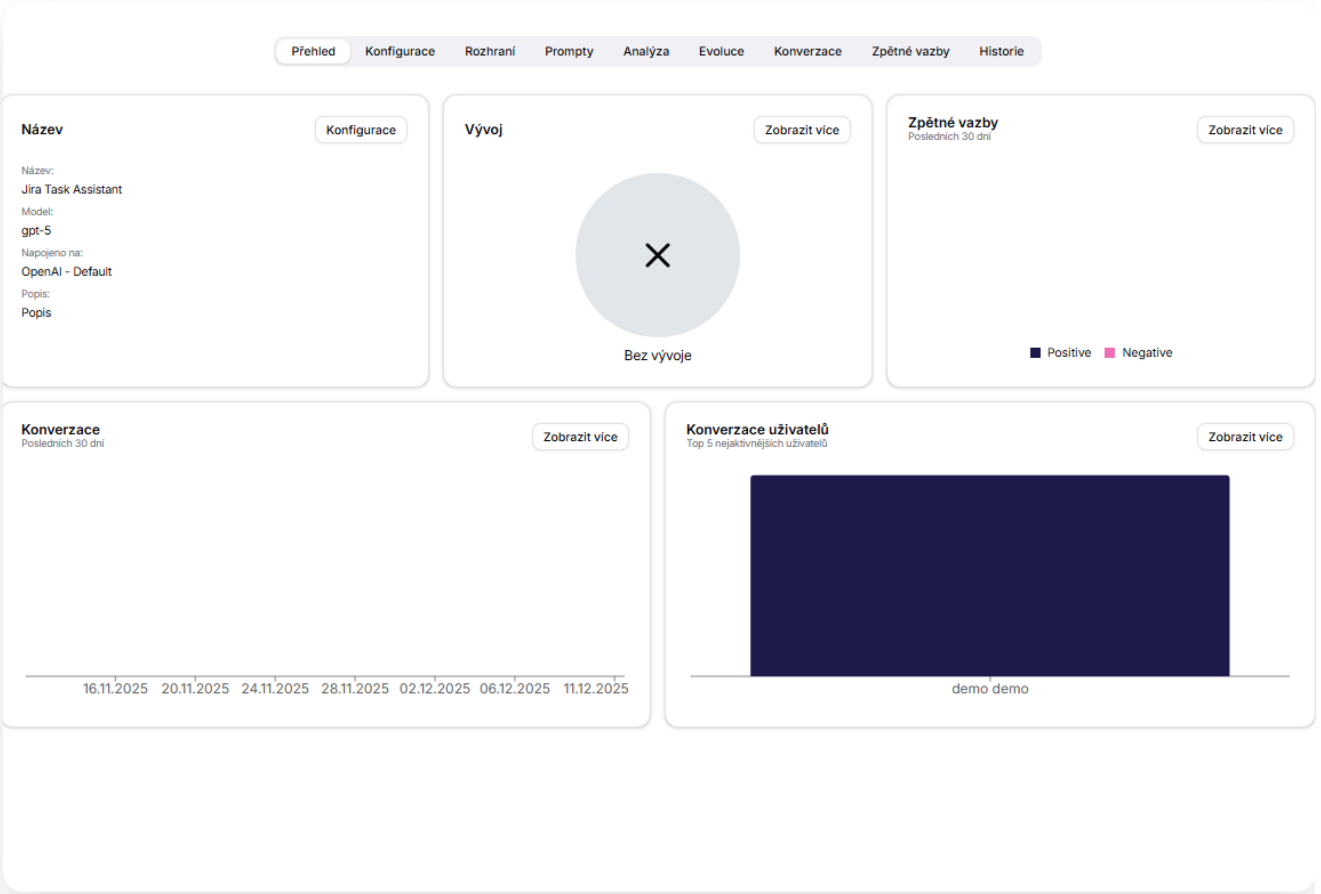
8.1 Overview

The Overview tab represents a central panel that summarizes key information about the assistant on a single screen. Here, users will find basic metrics on how often the assistant is used, what data it has connected, how much feedback it has received, and its current evolutionary state.

This overview allows for a quick assessment of how the assistant is functioning, what feedback it is receiving, and who is using it most frequently, without the need to navigate to individual detailed tabs. It is an ideal starting point for an immediate assessment of the health and usage of each assistant.

- Cards: **Name** (model, connection, description), **Development**, **Feedback**, **Conversations** (last 30 days), and **User Conversations** (Top 5).
- Serves for a quick check of the assistant's status; "View More" links lead to details.

- Screenshot:



8.2 Configuration

The Configuration tab is used for the detailed setup of the assistant - that is, its technical configuration, default behavior, and integration with other platform features. This section allows for precise definition of how the assistant should behave, what data it works with, what model it uses, and what tools it has available.

Users set basic information about the assistant here, including its name, instruction description, system messages, and the model to be used (e.g., GPT-4o). Access to data is also defined here - either global access to all sources or limited access to selected datasets.

The configuration also includes a model layer, where parameters such as response creativity (temperature), repetition penalties, or maximum output length can be adjusted. These options allow for fine-tuning the assistant's behavior according to specific purposes or expectations.

In this section, it is also possible to connect the assistant to tools that it can actively use - for example, for connecting to a calendar, API, or other systems. The entire configuration is designed to allow the assistant to be quickly launched and also modified or versioned at any time.

- **Name**: the name of the assistant displayed in overviews, searches, and in the list of assistants.
- **Description**: a brief description of the assistant's purpose; helps with orientation within the team.

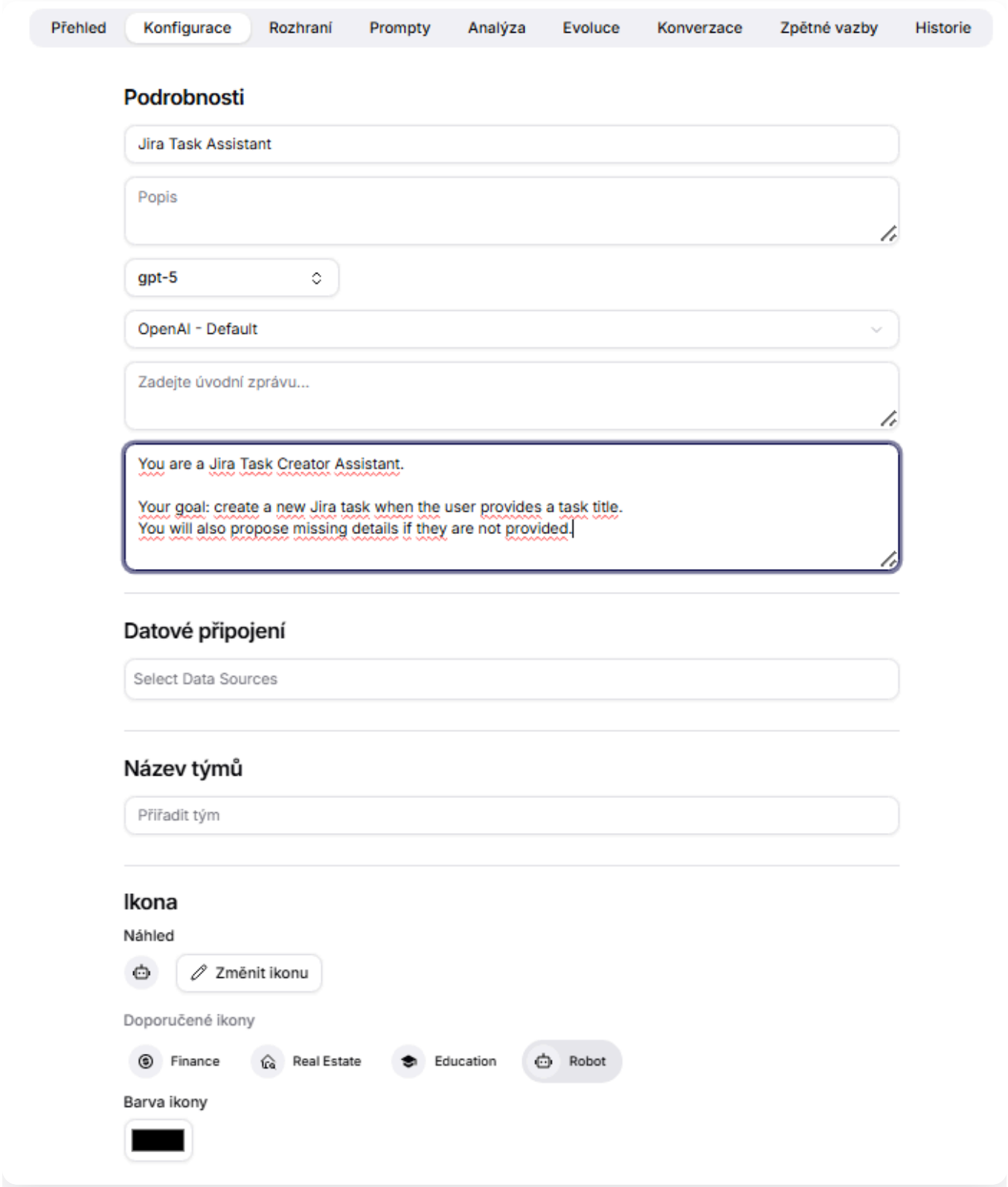
- **Chatbot Tool**: choice of provider/connection (Assistant Connection), which makes available models accessible.
- **Model Name**: specific model from the selected provider.
- **Data connection**: connection to data collections from the Data Collections section; determines what knowledge the assistant can work with.
- **Access**: visibility and team settings; **Organization** = everyone in the organization, **Shared** = selected users/teams, **Private** = only you.
- **Preview**: preview of the assistant's icon in the list.
- **Change icon**: selection of a custom icon.
- **Recommended icons**: quick presets of common icons.
- **Icon color**: color of the icon.
- **Temperature**: degree of creativity/variability; lower value = more consistent responses.
- **Maximum Length**: limit on the length of generated responses (shortens outputs and monitors costs).
- **Presence Penalty**: penalizes topic repetition; encourages new information.
- **Frequency Penalty**: penalizes repetition of words/phrases; reduces redundancy.
- **Initial Message**: introductory message displayed at the start of a new conversation.
- **System Message**: main system prompt defining the assistant's role, tone, and rules.
- **Shared Tools**: tools shared within the organization that the assistant can use.
- **Private Tools**: private tools available only to you.
- **Subassistants**: connection of additional assistants for delegating specific tasks.
- **Enable public chat**: makes the assistant accessible outside the main application.
- **Enable Authenticated Chat Widget**: allows embedding a chat widget on an external website with Google login.
- **Save as Template**: saves the current assistant as a reusable template for further creation.

The platform allows saving the currently created assistant as a template, which can then be reused when creating other assistants. This feature supports repeatability, consistency, and scalability of configurations across the organization.

After clicking the **Save as Template** button, a modal window will appear where the user fills in:

- **Name** – the name of the template.
- **Description** – a brief description of its focus and use.

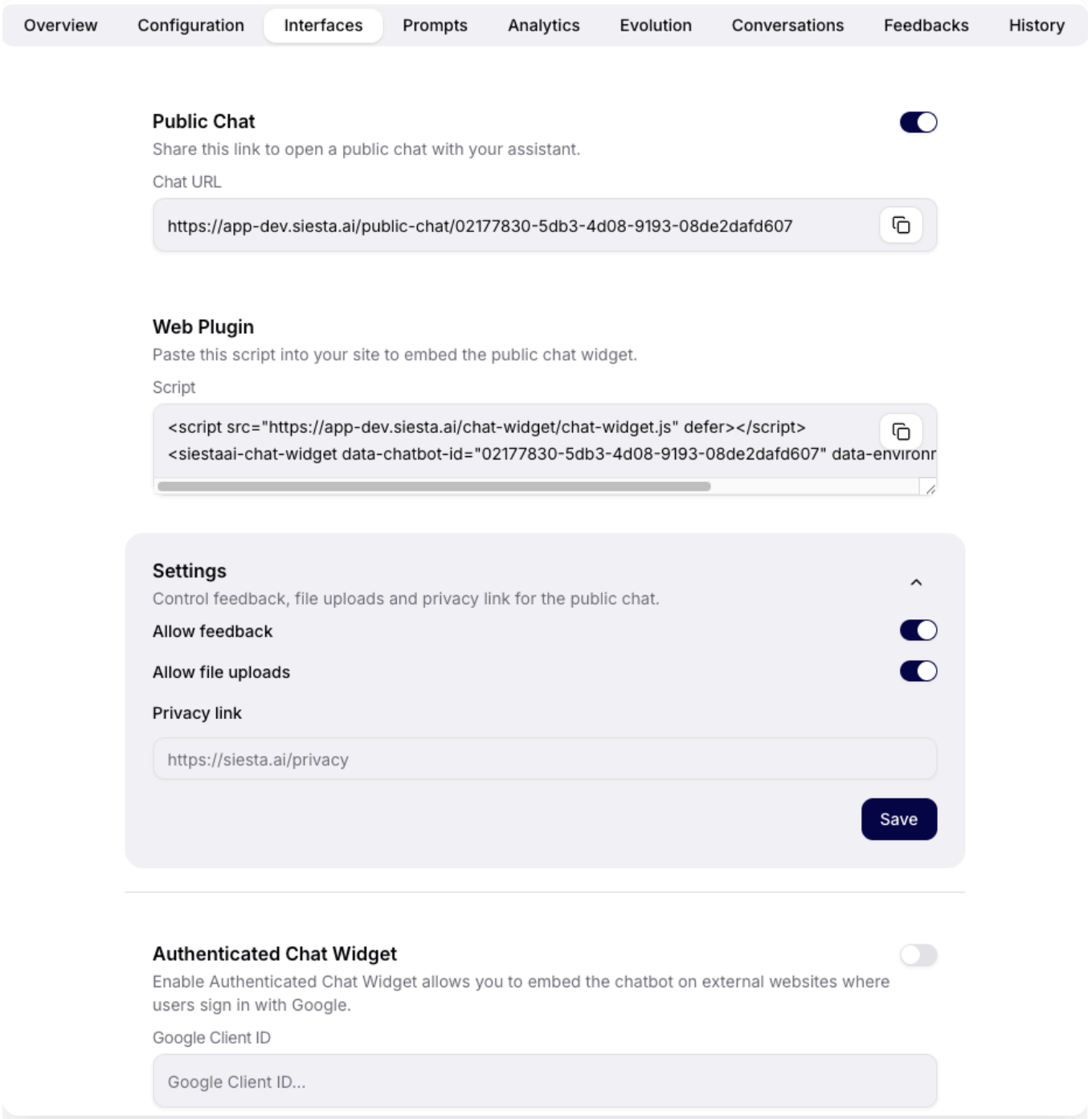
The saved template will then appear in the overview when creating a new assistant and can be edited or reused at any time.



8.3 Interface

The Interface section allows setting how the assistant will be made available to end users. Here, it is defined how and where communication with the assistant should be possible, whether internally, via API, or publicly through integration channels and widgets.

- **Public Chat** switch with generated **Chat URL** (copy button).
- **Web Plugin**: embeddable script for embedding public chat on an external website.
- **Settings**: settings for feedback, file uploads, and a link to the privacy policy.
- **Chat widget with authentication** switch and **Google Client ID** field for embedding with login.

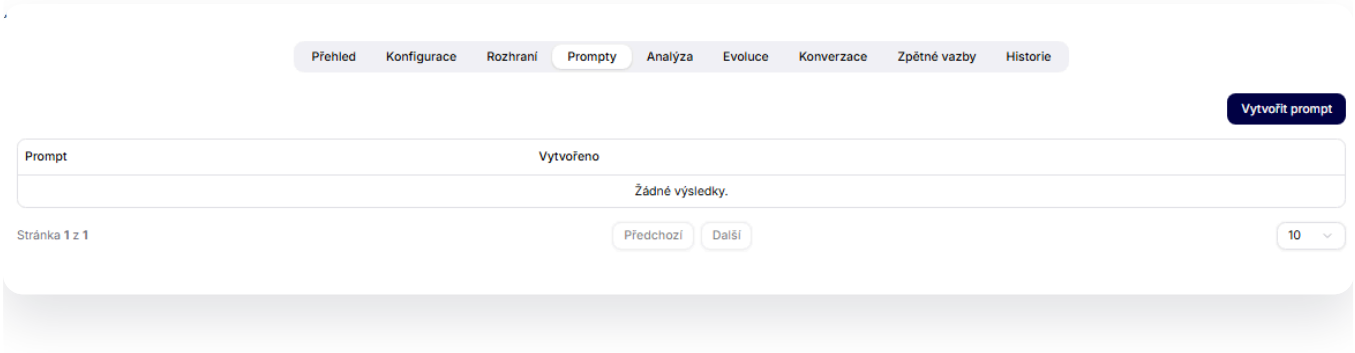


8.4 Prompts

The Prompts section is used to control the logic and behavior of the assistant through so-called system instructions. Here, the user defines how the assistant should respond, what stance it should take, communication style, or structure of responses.

Each prompt represents a specific instruction block that the model receives before processing the user's input. The assistant uses it to orient itself on what role to play, which information to prioritize, and what responses to generate.

- Table of all prompts; **Create Prompt** button for creating a new one.
- After creation, you can manage the prompt text and execution schedule.



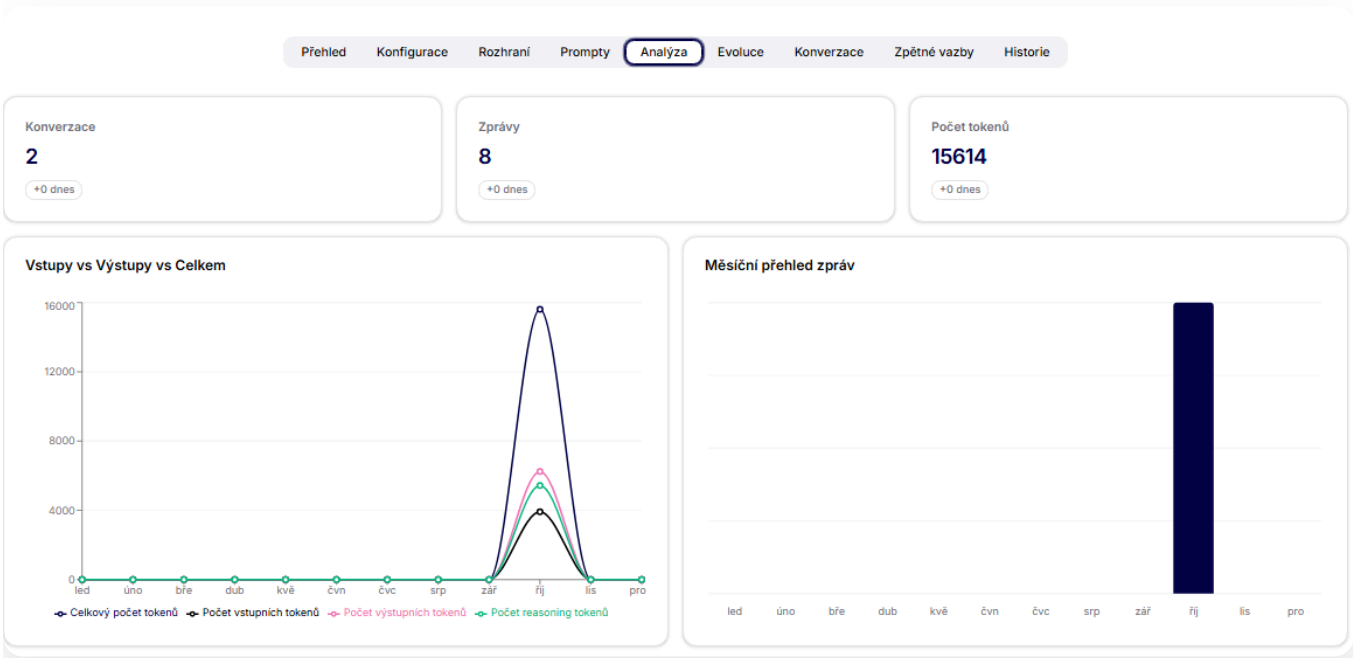
8.5 Analysis

The Analysis section provides a detailed view of the technical and operational parameters of the assistant. Users will find summary statistics and visualizations here that help understand how the assistant is being used, its load, and its performance over time.

Monitored metrics include the number of requests, volume of consumed tokens, ratio of input to output tokens, response processing speed, and other indicators. This information is available in the form of clear graphs and bar visualizations, which allow for identifying trends, fluctuations, or potential anomalies.

Analyses are a crucial tool for administrators and product teams who want to not only operate assistants but also optimize them. They help evaluate when the highest usage occurs, what the consumption-to-performance ratio is, and how quickly the assistant responds under real conditions.

- Section for an overview of the assistant's performance (volume and quality of interactions).



8.6 Evolution

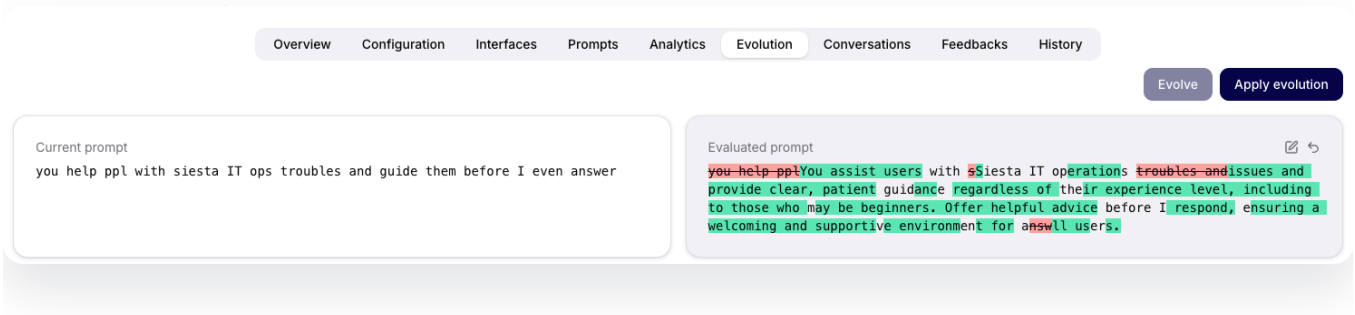
The Evolution section allows managing the development of the assistant's system instructions (prompts) based on real behavior and feedback. Administrators can compare individual versions of prompts here, evaluate their impact, and apply changes in a targeted and controlled manner.

The main part consists of comparing the currently active prompt with a proposed modification. The user can thus see the previous and new wording of the instruction side by side and easily identify how the assistant's logic is changing. In addition to textual comparison, the system collects specific use cases on which the change can be tested.

The evolution section also includes an overview of feedback that led to the change or to which the new version of the prompt should respond. This feedback includes ratings, comments, and the identity of the users who submitted it.

Evolution is thus a tool for improving the assistant's behavior based on data, not intuition. It allows for continuous development, controlled testing, and documentation of all changes over time.

- Overview of user suggestions for prompt evolution (date, user, message, rating).
- **Develop** action (suggests modifications) and **Apply Evolution** action (applies changes).



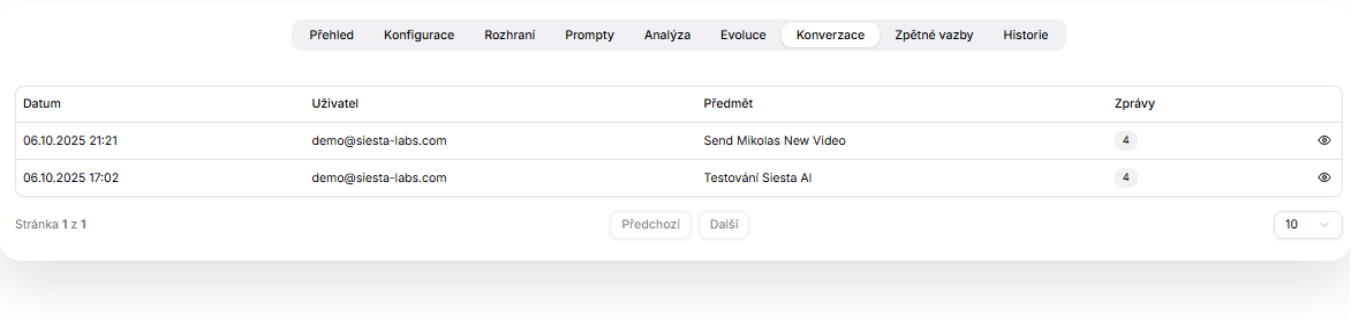
8.7 Conversations

The Conversations section serves as an overview of all interactions that have taken place between users and a specific assistant. Users will find a list of conversations with information about the date, initiating user, topic, and total number of messages within that exchange.

This section provides administrators with the opportunity to delve into the specific content of communication, analyze how the assistant is used in practice, and verify how queries were answered. The displayed information can be used for further evolution of the assistant, fine-tuning prompts, or ensuring compliance with internal rules.

Each conversation includes the option for a detailed view. Users can thus retrospectively track the entire communication, including all steps and responses, which increases transparency and allows for backtracking when needed.

- List of conversations with users: date, user, subject, number of messages; the eye icon opens the thread detail.



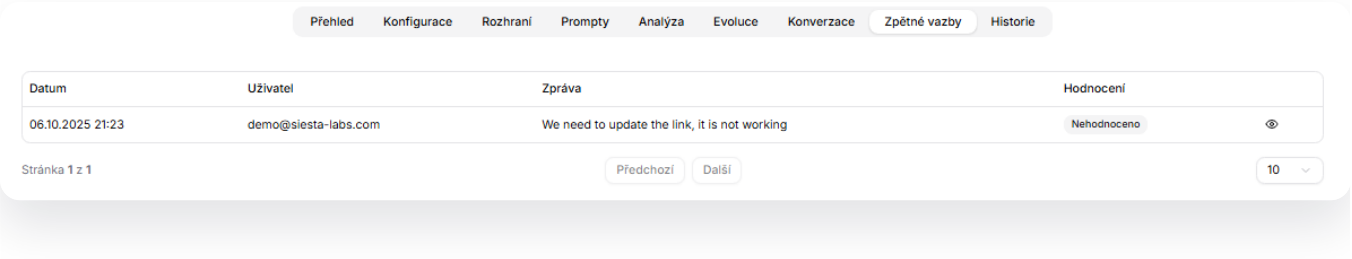
8.8 Feedback

The Feedback section serves as a centralized overview of the feedback provided by users on the assistant's responses. Each record includes the date, user identity, interaction outcome, and type of rating - for example, positive, negative, supplemented by a comment, or other specific ratings.

Thanks to this section, administrators can easily identify responses that were inaccurate, misleading, or particularly beneficial. All reactions are traceable, and it is possible to look back at the context in which the rating occurred.

This feature is crucial for the future evolution of the assistant - it provides a data-driven basis for prompt adjustments, refinement of data context, or evaluation of training needs. In combination with the Evolution section, it forms part of the continuous improvement of the assistant's outputs.

- Collected ratings and comments on the assistant's responses.

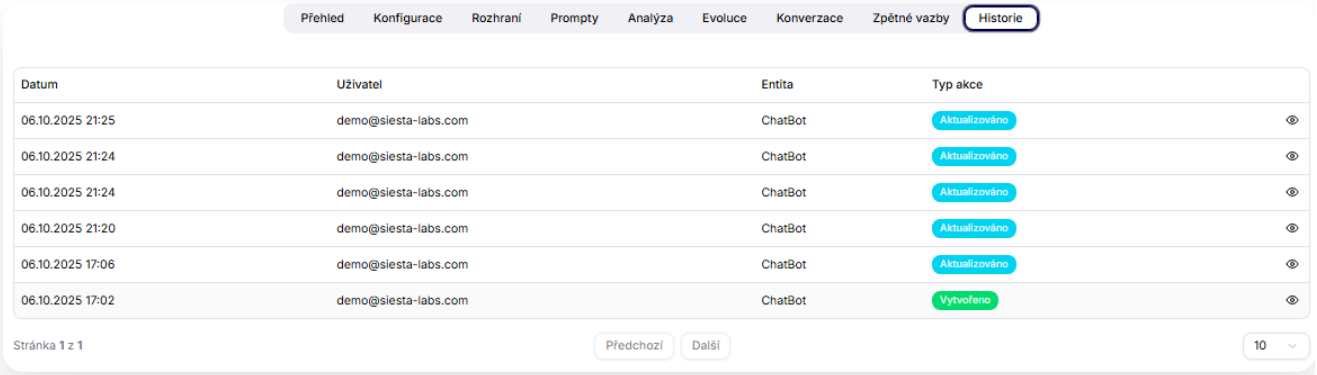


8.9 History

The History section serves to transparently track all important changes that have been made to the assistant. Each record includes the date, author of the change, affected entity, and type of action performed.

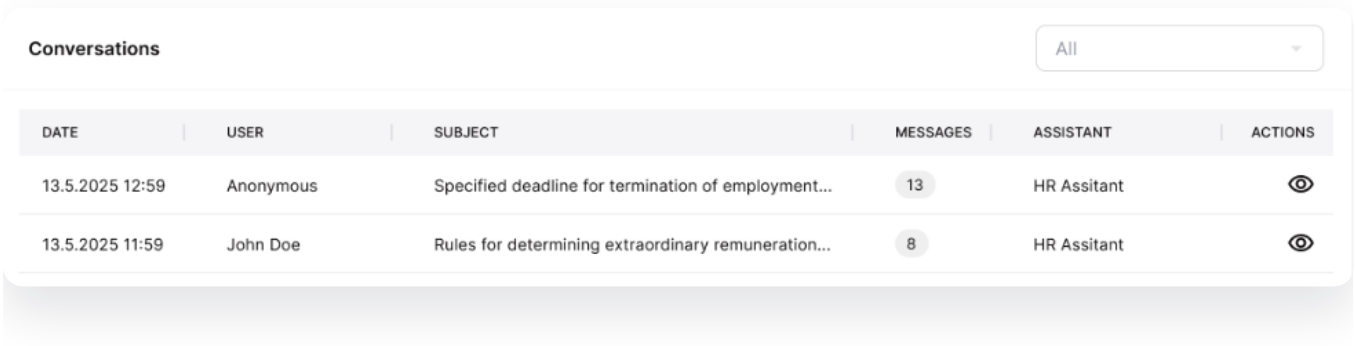
This audit trail is important for operational oversight, security standards, and backtracking interventions in an environment where assistants often undergo evolution or modifications.

- Audit log of actions on the assistant (date, user, entity, type of action).
- Allows tracing modifications of models/prompts/data connections.
- Screenshot:



9. Conversations

The Conversations feature serves to provide a clear display of the history of interactions between users and individual AI assistants within the Siesta AI platform.



9.1 Feature Description

9.1.1 Overview of All Conversations

An administrator or authorized user can see a list of all previously conducted conversations sorted by date. The displayed information in the overview includes:

- **Date and time of conversation initiation**
- **Subject / topic of the conversation** (e.g., “Specified deadline for termination of employment”)
- **Number of messages in the conversation**
- **Name of the assistant** with whom the communication took place (e.g., “HR Assistant”)

9.1.2 Conversation Detail

Clicking on a row opens the entire chat history, including all inputs and outputs.

9.1.3 Feedback Option

Individual responses can receive a positive rating (thumbs up) or a negative rating (thumbs down), and a comment can be added. This information is subsequently displayed in the Feedback section and helps fine-tune the accuracy of responses.

9.1.4 Access Rights

Access to these records is restricted based on roles. For example, regular users can only see their own conversations, while an administrator or management team has access to the entire overview.

10. Data

10.1 Overview

The Data module serves to consolidate multiple data sources into a single logical unit. A data collection represents a central container of data that can subsequently be utilized in AI assistants, workflows, or analytical tools in Siesta AI.

Each collection:

- has its own name and description,
- contains one or more data sources,
- allows for the management and organization of data according to its purpose.

10.2 Overview of Data Collections

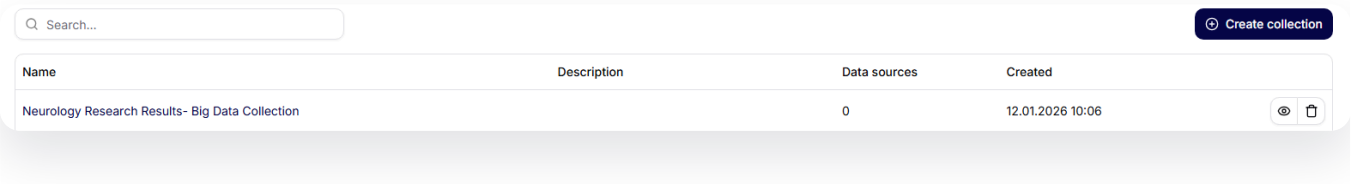
On the main screen of Data Collections, a list of all created collections is displayed in the form of a table.

Displayed columns:

- **Name** – the name of the data collection
- **Description** – a brief description of the collection's purpose
- **Data Sources** – the number of connected data sources
- **Created** – the date and time of creation
- **Actions** – additional options for working with the collection

At the top of the screen, the following is available:

- collection search,
- the **Create Collection** button.



10.3 Creating a New Data Collection

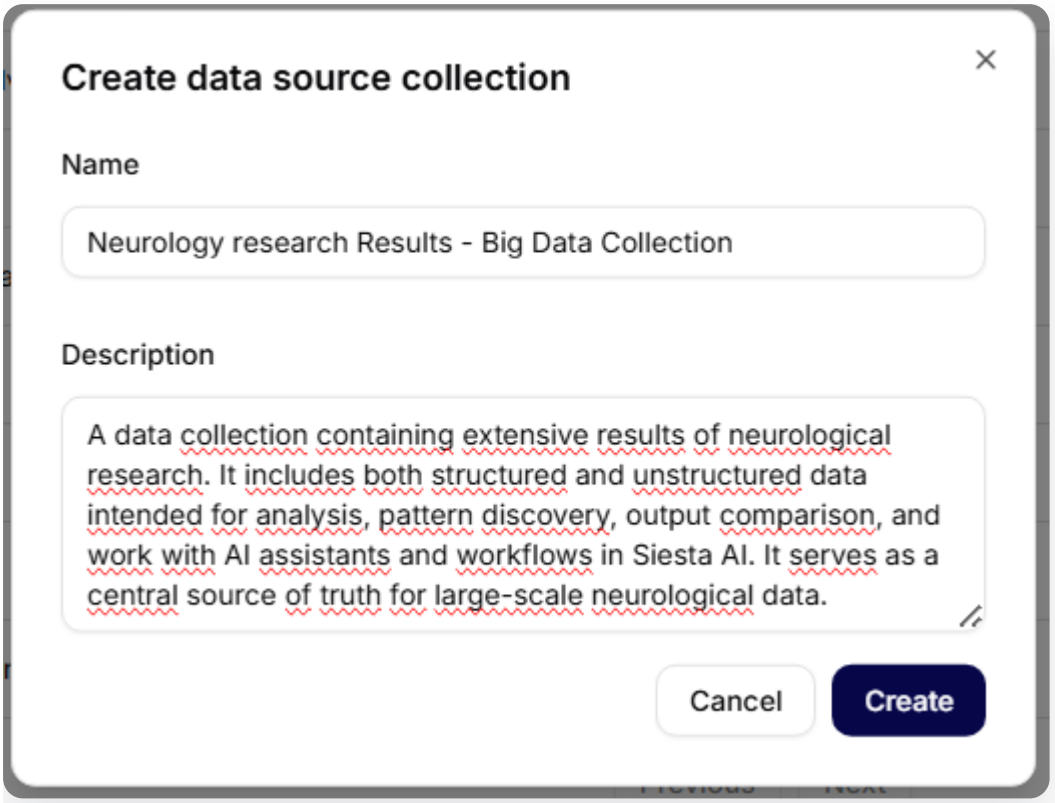
After clicking on **Create Collection**, a dialog for creating a new collection opens.

Required Fields

- **Name** – a unique name for the data collection (e.g., Neurology Research Results – Big Data Collection).
- **Description** – a brief description of the content and purpose of the collection.

Actions

- **Cancel** – closes the dialog without saving
- **Create** – creates a new data collection



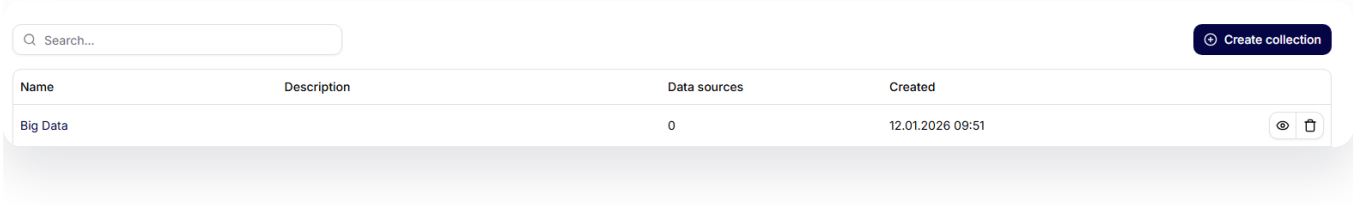
10.4 Data Collection Detail

When a specific collection is opened, its detailed page is displayed.

Displayed information:

- collection name,
- creation date,
- overview of connected data sources.

The page includes a **Add Data Source** button.



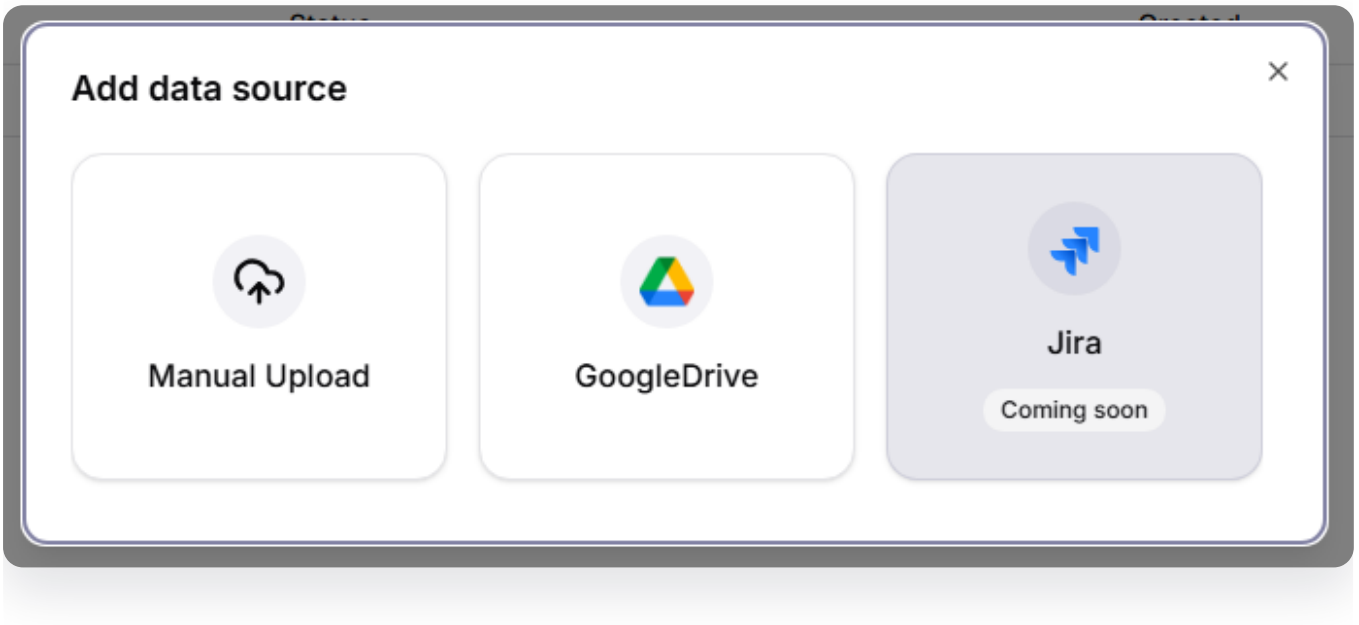
10.5 Adding a Data Source to a Collection

Clicking on **Add Data Source** opens a selection of data source types.

Available Options

- Manual Upload – manual file upload
- Google Drive (coming soon)
- SharePoint (coming soon)
- Azure Storage (coming soon)
- Jira (coming soon)

In the current version, manual file upload is available.



10.6 Configuring the Data Source (Manual Upload)

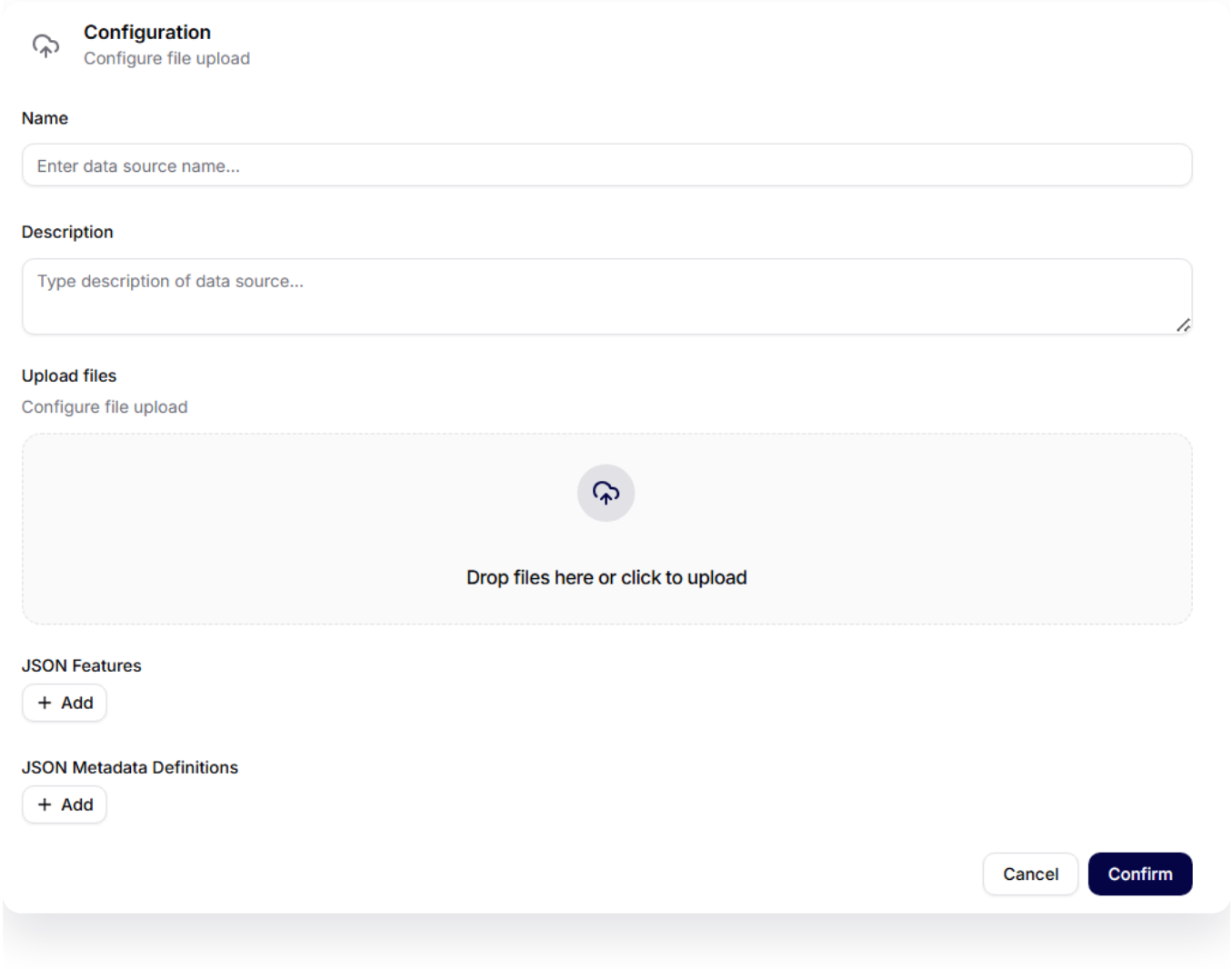
After selecting **Manual Upload**, a configuration form is displayed.

Configuration Fields

- **Name** – the name of the data source (e.g., Big Data).
- **Description** – an optional description of the data source content.
- **Upload Files** – the option to drag files into the designated area or click to select files from the computer.
- **JSON Functions (optional)** – used to define custom functions for working with data.
- **JSON Metadata Definitions (optional)** – allows adding structured metadata to the data source.

Actions

- **Cancel** – exits the configuration without saving
- **Confirm** – saves the data source and initiates its processing



10.7 Data Source Status

Each data source has its own processing status:

- **Processing** – data is being analyzed and indexed
- **Processed** – the data source is ready for use

The status is visible in the data sources table in the collection detail.

10.8 Connecting Data Collections to Assistants

Data collections are subsequently assigned to assistants in their settings. Details can be found in the section Assistant Configuration.

10.9 Typical Uses of Data Collections

Data collections are primarily used for:

- organizing a larger number of files,
- consolidating data by topic or project,

- creating a single source of truth for AI assistants,
- reusing the same data in different workflows,
- scaling data work without the need for duplication.

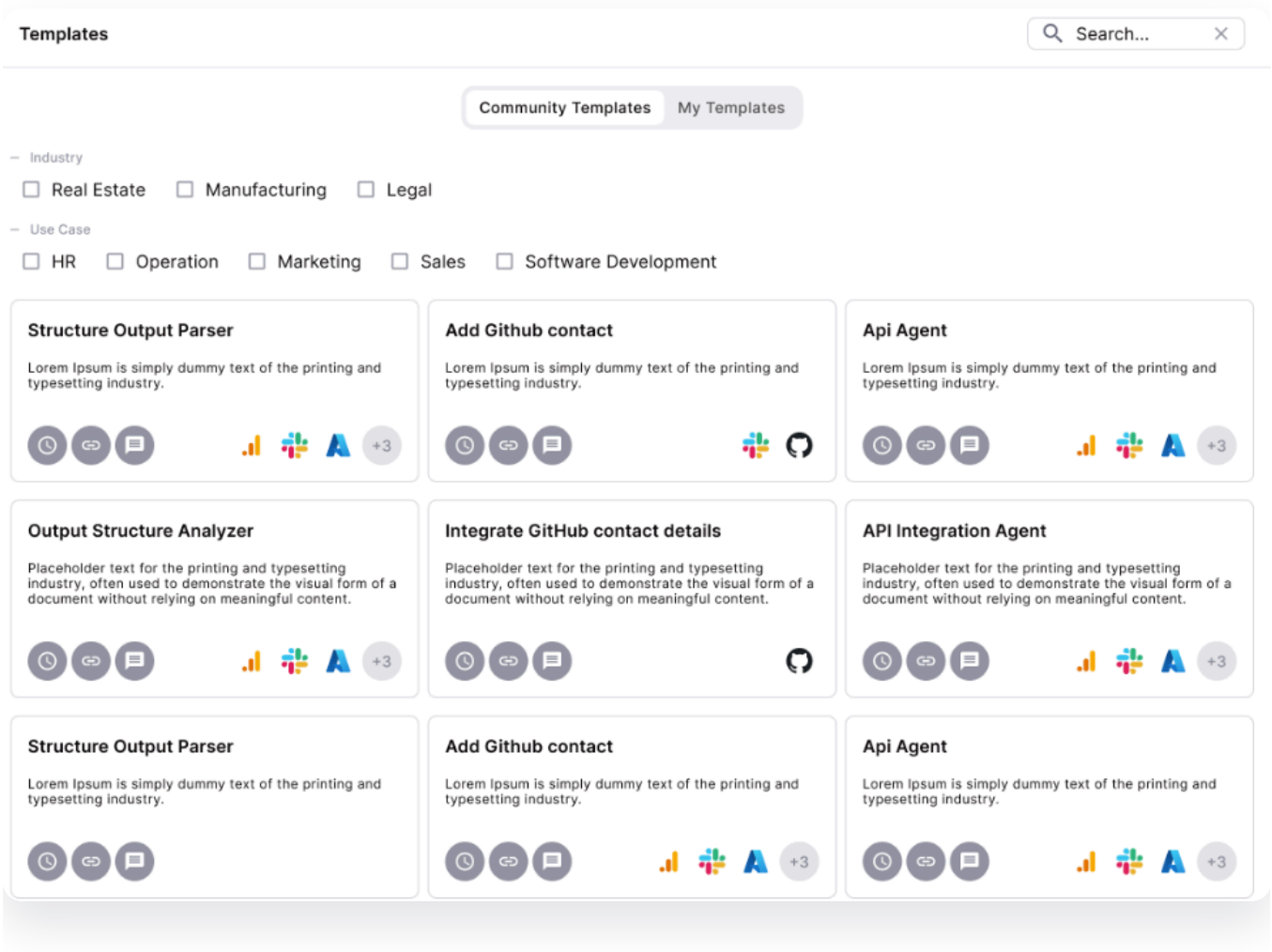
10.10 Summary

Data collections in Siesta AI enable clear data management and their effective use across the entire platform. Well-structured collections are the foundation for quality outputs from AI assistants and automated workflows.

11. Templates

Templates in Siesta AI serve as reusable definitions of assistants. They allow you to save the configuration of an assistant (model, behavior, connected tools) and quickly create new instances from it.

The Templates section allows users to choose from prepared templates that serve as a starting point for creating new AI assistants. Templates contain preset parameters, logic, data access, and integrations, thus simplifying the repeated deployment of frequently used configurations.



Templates address three things:

- Standardization of work
- Sharing know-how (community vs. proprietary)
- Rapid deployment of new assistants

11.1 Basic Concept

A template is not a running assistant. It is a blueprint and only defines how the assistant should be created.

From one template, you can create:

- 1 assistant
- 10 assistants
- 100 assistants

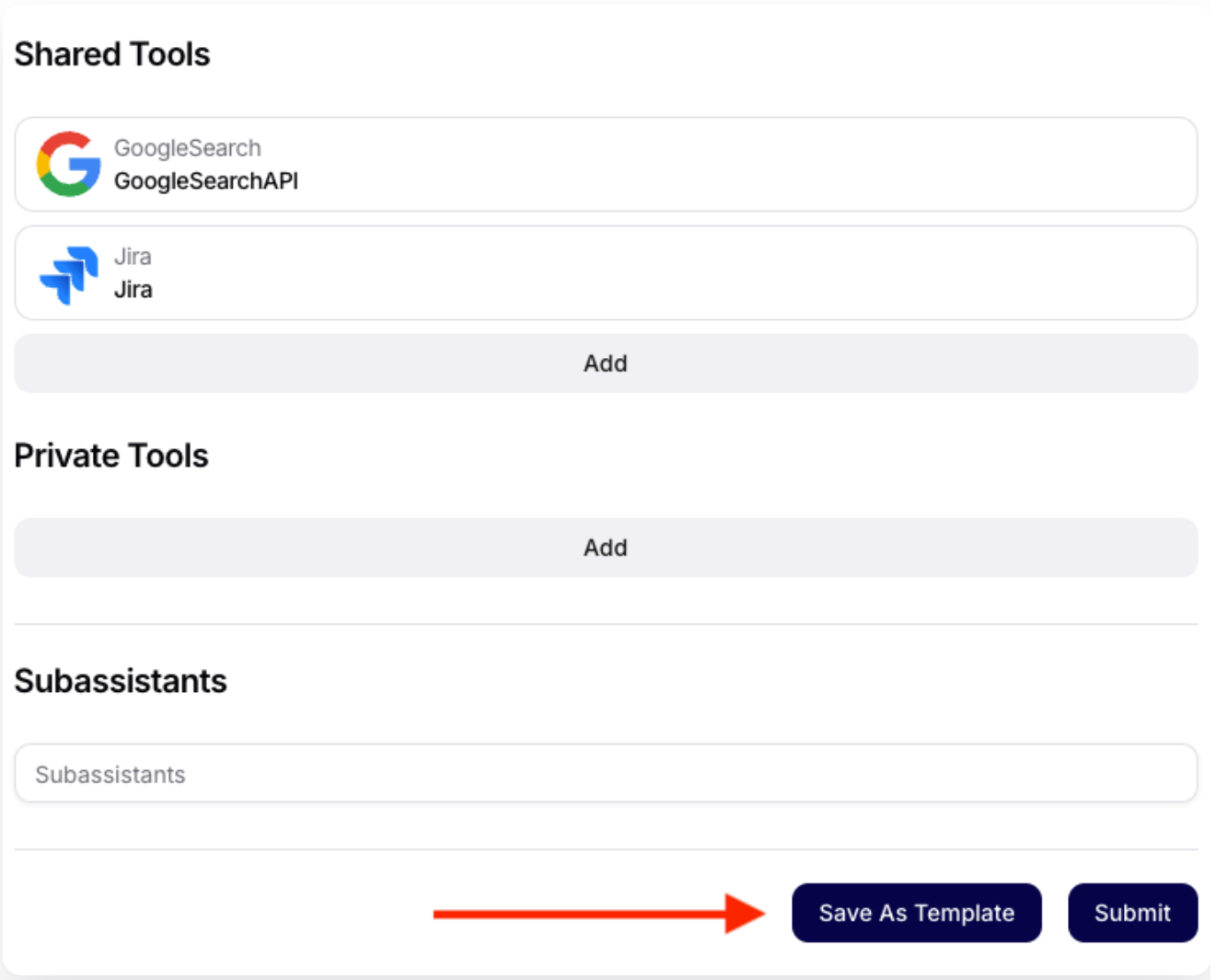
11.2 Saving an Assistant as a Template

Action: **Save As Template**

A template is created from an existing assistant configuration.

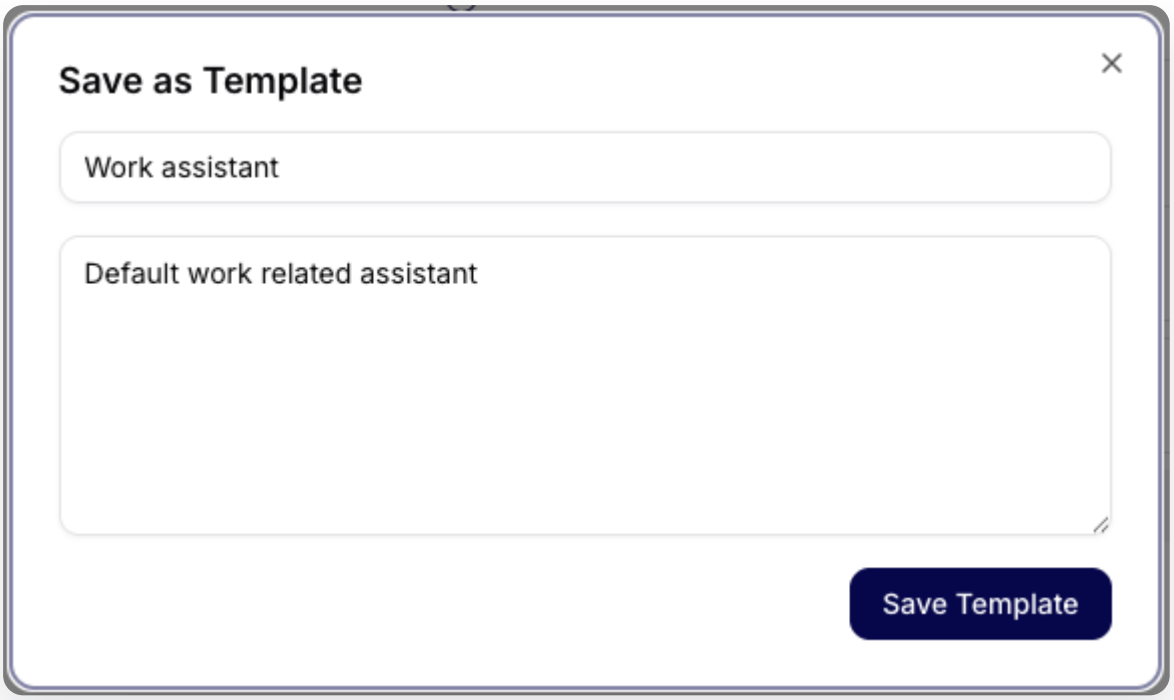
Procedure:

1. The user clicks on **Save As Template** in the assistant settings.



2. A modal window opens where the user fills in:
    - **Name** - template name

- **Description** - purpose description



3. Confirming will save the template.

What gets saved:

- Name and description
- Model (e.g., gpt-5)
- Behavior parameters (temperature, penalties, max length)
- Shared Tools (e.g., Jira, GoogleSearch, HubSpot)
- Private Tools (if any)
- Subassistants (if defined)

The template does not save runtime state or conversation history.

11.3 Shared Tools vs. Private Tools

Shared Tools

- Tools available at the workspace or organization level
- Automatically connected when creating an assistant after being saved in the template

Private Tools

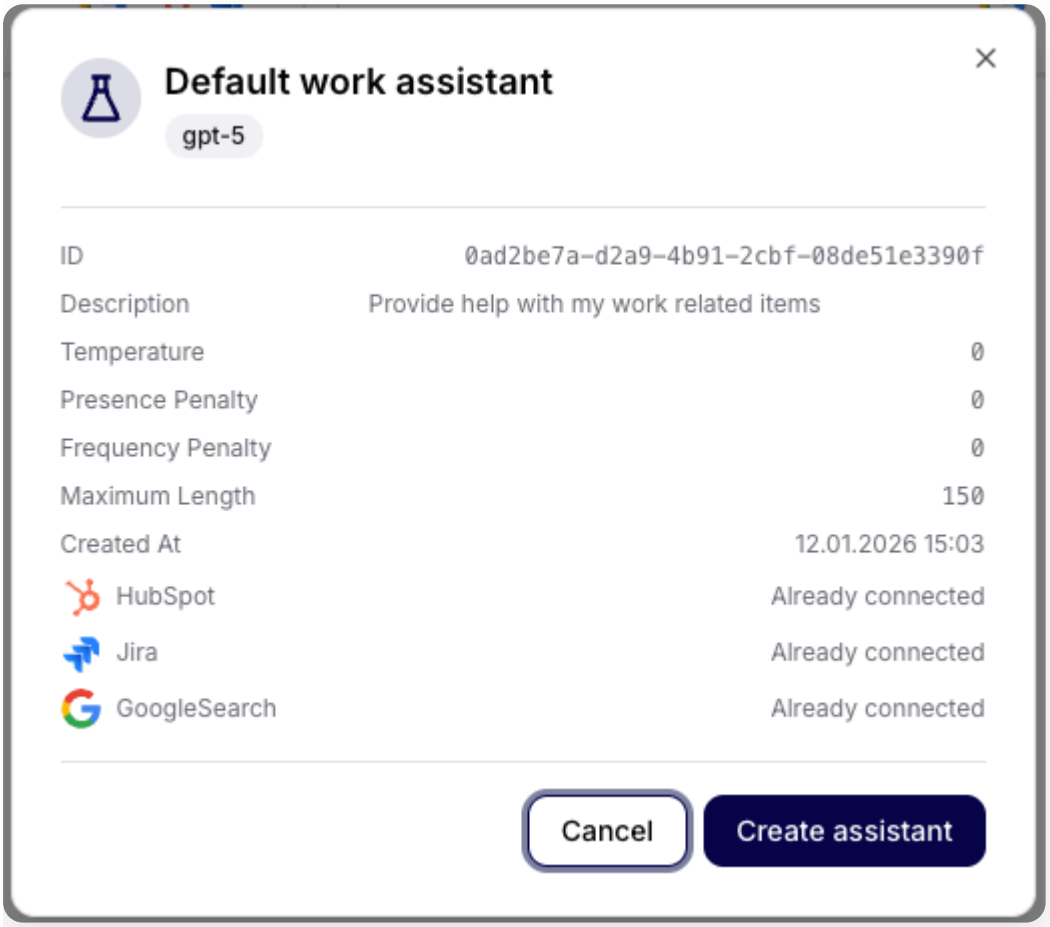
- Tools specific to the user or project
- The template remembers their reference, but the user must have permissions
- If permissions are missing, creating the assistant will fail

11.4 Creating an Assistant from a Template

Action: **Create assistant**

After selecting a template, a detailed overview of the configuration is displayed:

- Template ID
- Name and description
- Used model
- Behavior settings
- List of connected tools + status (Already connected)



11.5 Templates Page

11.5.1 Tabs

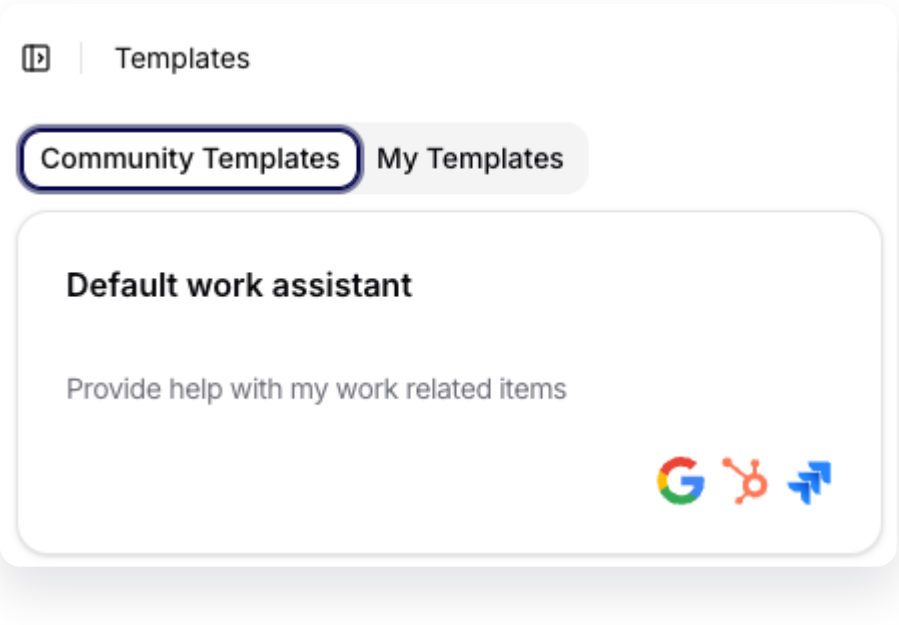
Community Templates

- Publicly available templates
- Created by the team or community
- Read-only (cannot be edited)

- Suitable as best practices or starting points

My Templates

- Templates created by the user
- Can be deleted, edited, and reused



11.6 Visualization of Template Card

Each template is displayed as a card containing:

- Name
- Short description
- Icons of connected tools

The card allows for quick visual inspection of what the template contains before use.

11.7 Typical Use Cases

- Work assistant: one template -> dozens of internal assistants with the same behavior
- Onboarding: new team member = quick creation of an assistant without manual configuration
- Best practice enforcement: template as the only allowed entry point
- Community sharing: verified configurations without sharing sensitive data

11.8 Summary

Templates in Siesta AI are a controlled way to scale assistants. They allow sharing know-how, maintaining standards, and speeding up work without compromising security.

12. Connections

Connections represent the central place where all integrations of the Siesta AI platform with external services are managed, whether they are action tools, knowledge libraries, or the AI models themselves. Thanks to this section, administrators have an immediate overview of what resources are available and can add, modify, or remove them with just a few clicks. By integrating a new service, it immediately appears throughout the system and can be assigned directly during the creation or modification of an assistant.

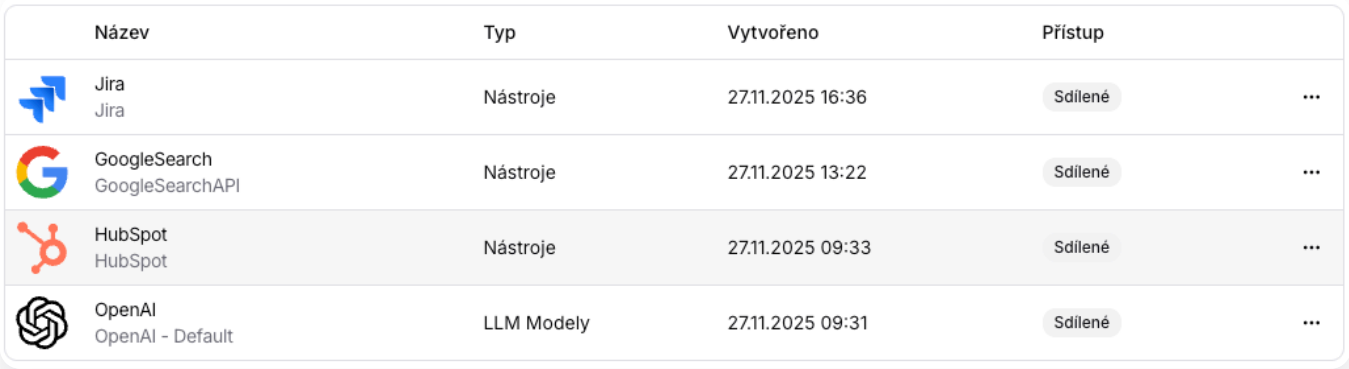
The Connections section is used to work with external systems. Connections allow Siesta AI to link with third-party tools (APIs, SaaS platforms, internal systems) so that assistants and workflows can read data, write changes, or trigger actions.

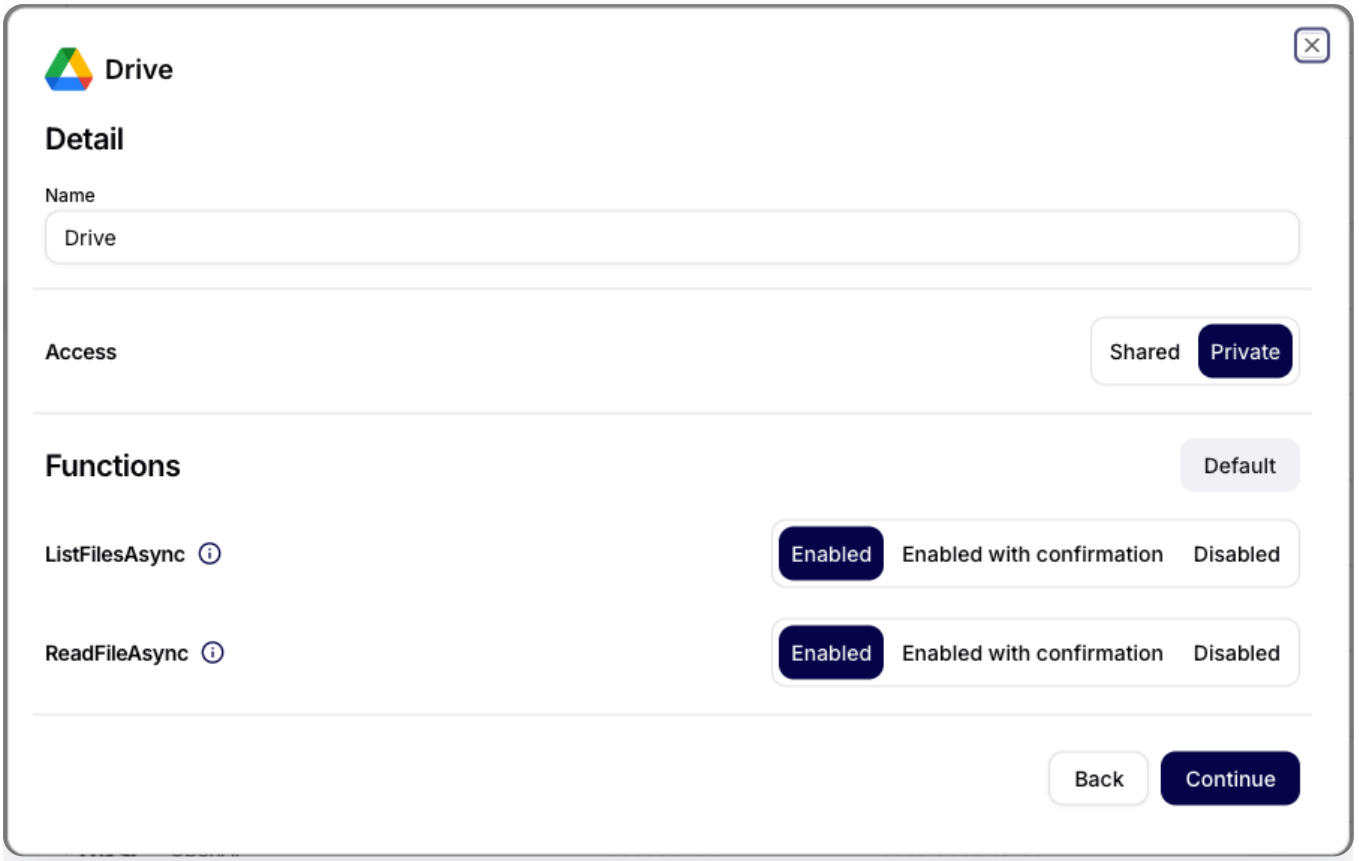
12.1 How it Works

- **Management**: In the Connections section, you activate specific connections, set access (OAuth / API key), and assign them to assistants or workflows.
- **Usage**: Connection actions are called from prompts, tools, or automations (e.g., send an email, fetch data from CRM).
- **Security**: Access tokens are stored in encrypted form, and all operations are fully audited.

12.2 Overview of the Connections Section

- Search field at the top for quick filtering of connections.
- Table with columns: **Name**, **Type**, **Created**, **Access** + actions on the right (menu **...**).
- **Add Connection** button to create a new connection.
- Examples of available connections: Jira, Google Search, HubSpot, OpenAI.

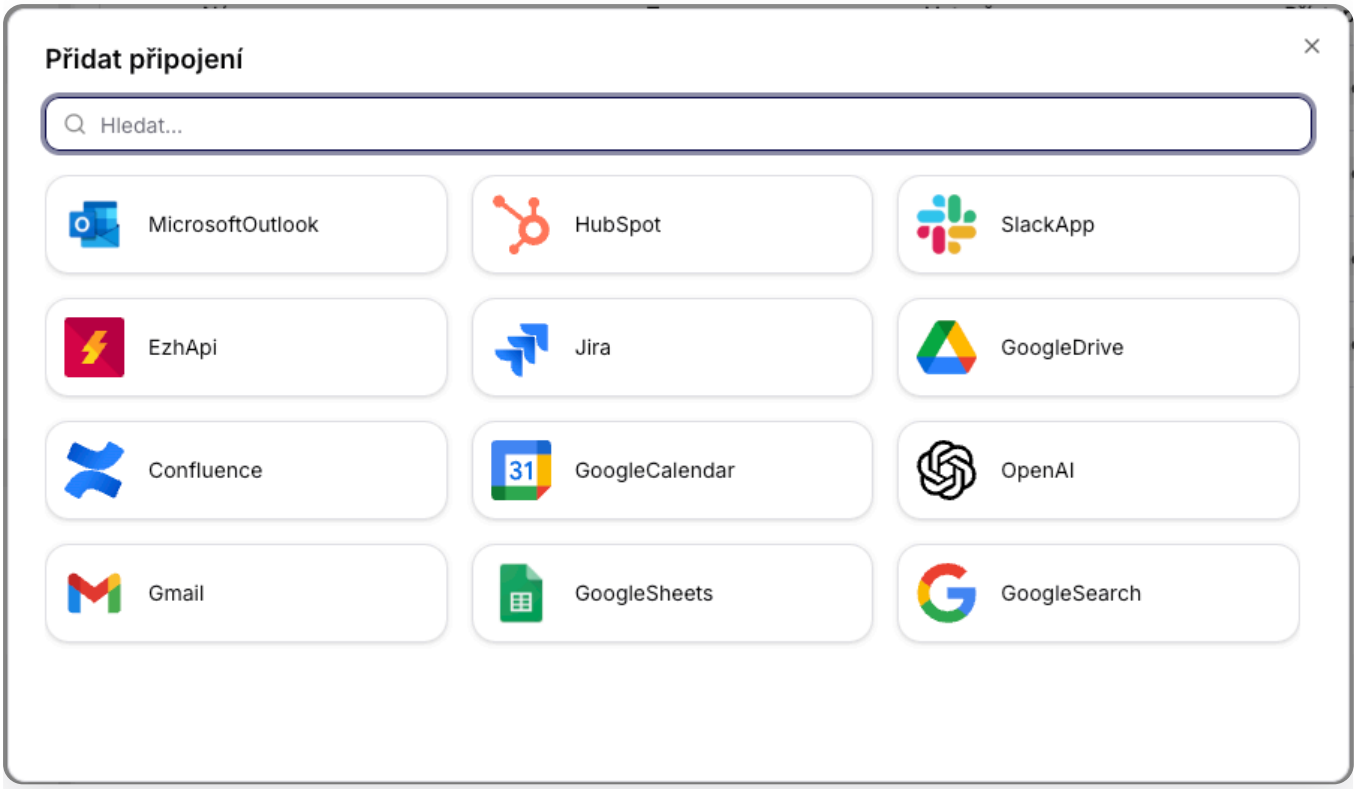




In the detail of individual connections, you can set the scope of permissions and allowed functions. Administrators specify which actions are available, whether they require confirmation, and what access the connection has (shared or private).

12.3 Adding a New Connection

After clicking on **Add Connection**, a dialog will open with a search field and a list of available connections (e.g., Gmail, Google Calendar, Google Drive, Slack App, OpenAI).


Přidat připojení
Hledat...
MicrosoftOutlook
HubSpot
SlackApp
EzhApi
Jira
GoogleDrive
Confluence
GoogleCalendar
OpenAI
Gmail
GoogleSheets
GoogleSearch

13. Azure (In Preparation)

Connecting Siesta AI with the Azure environment via service principal.

13.1 Setup

1. In **Connections**, click **Add Connection** and select **Azure**.
2. Fill in **Tenant ID**, **Client ID**, **Client Secret**, and optionally **Subscription ID**.
3. Choose **Shared** or **Private** access and save.

13.2 Usage

- In workflows, you can read or trigger actions on Azure resources available in Connections (e.g., list resource groups, initiate deployment).
- Store passwords in a secure vault and rotate the secret key regularly.

13.3 Security

- Limit the service principal's roles to the necessary minimum.
- Monitor sign-ins and activity logs in Azure AD.

14. Azure AI Foundry

Azure AI Foundry is a platform in Azure for the development, deployment, and management of AI applications, agents, and models. Within Siesta AI, it serves as an enterprise backend for inference and agents with support for RBAC, regional restrictions, and audits.

14.1 Overview

Siesta AI from Azure AI Foundry:

- calls deployed models (chat, reasoning, transcribe),
- uses an OpenAI-compatible endpoint for inference,
- respects Azure RBAC and customer security policies.

14.2 Key Terms

- **Foundry resource** – Azure resource of type **Azure AI Foundry** in a subscription and resource group.
- **Foundry project** – a logical project within the Foundry resource (separating teams, applications, and environments).
- **Project endpoint** – API endpoint for project capabilities (agents, evaluations, inference via Foundry API).
- **Model deployment** – specific deployment of a model (e.g., `gpt-5.2`, `gpt-5.2-chat`).
- **API key** – key for authenticating calls to the Foundry API.

14.3 Requirements

- Active Azure subscription.
- Permissions of at least **Contributor** on the target resource group.
- Registered resource provider **Microsoft.Foundry**.
- Access to **ai.azure.com** (Microsoft Entra ID).

14.4 Creating Azure AI Foundry

14.4.1 1) Foundry resource

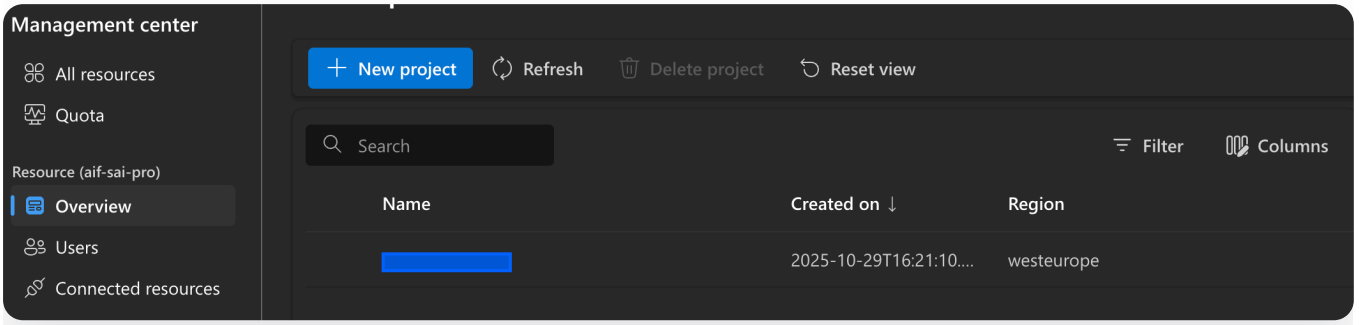
1. Sign in to **Azure Portal**.
2. Create a new resource **Azure AI Foundry**.
3. Choose **Subscription**, **Resource Group**, **Region** (e.g., `westeurope`), and resource name (e.g., `aif-sai-pro`).

The Foundry resource serves as a container for all projects.

14.4.2 2) Foundry project

1. Open **ai.azure.com**.

2. On the left, select **Management Center → Projects**.
3. Click on **New project**.
4. Select an existing Foundry resource and enter the project name.



14.5 Model Deployments

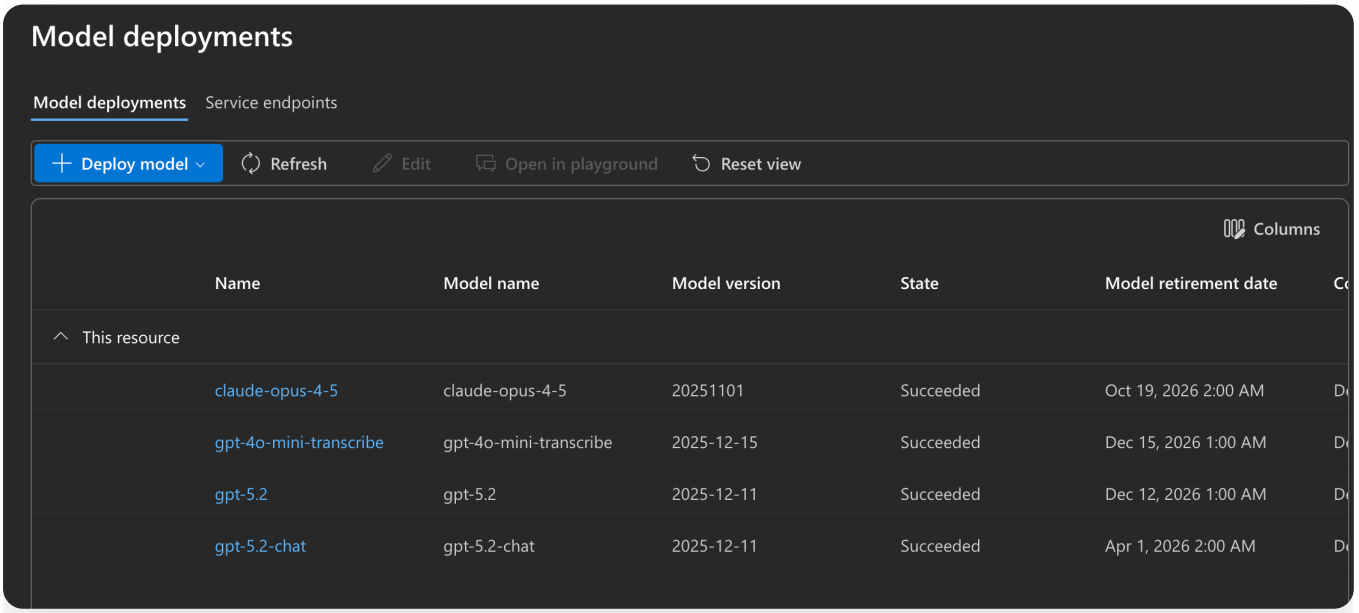
14.5.1 Model deployments

In the project, go to **Model catalog → Model deployments** and deploy available models.

Examples of deployments:

- `gpt-5.2`
- `gpt-5.2-chat`
- `gpt-4o-mini-transcribe`
- `claude-opus-4-5`

Each deployment has a **deployment name**, **model version**, **status**, and **retirement date**.



> ⚠️ Siesta AI works with the **deployment name**, not the model name.

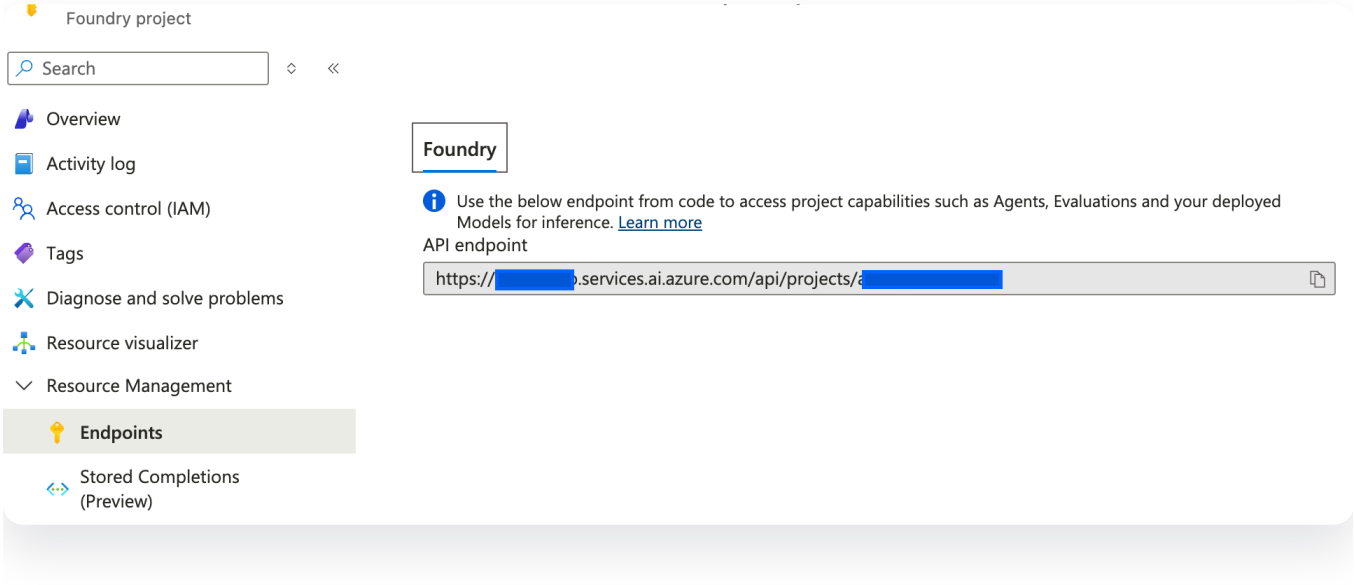
14.6 Endpoints and API Keys

14.6.1 Project endpoint (Foundry API)

The project endpoint is used for project capabilities (agents, evaluations, and Foundry inference API). You can find it in the project details.

Endpoint format:

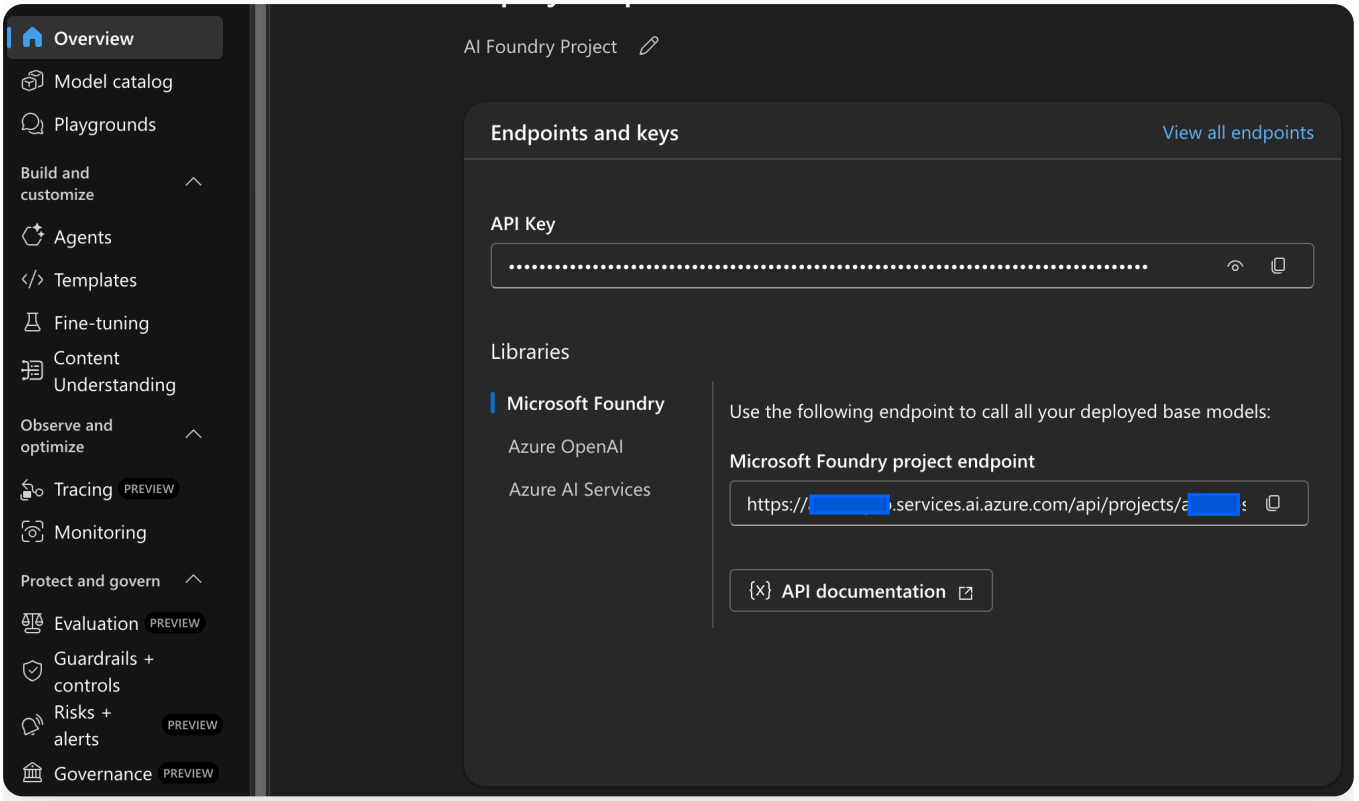
```
https://<foundry-resource-name>.services.ai.azure.com/api/projects/<project-id-or-name>
```



14.6.2 Endpoints and keys in the project

In **ai.azure.com**, open the project and the **Endpoints and keys** section:

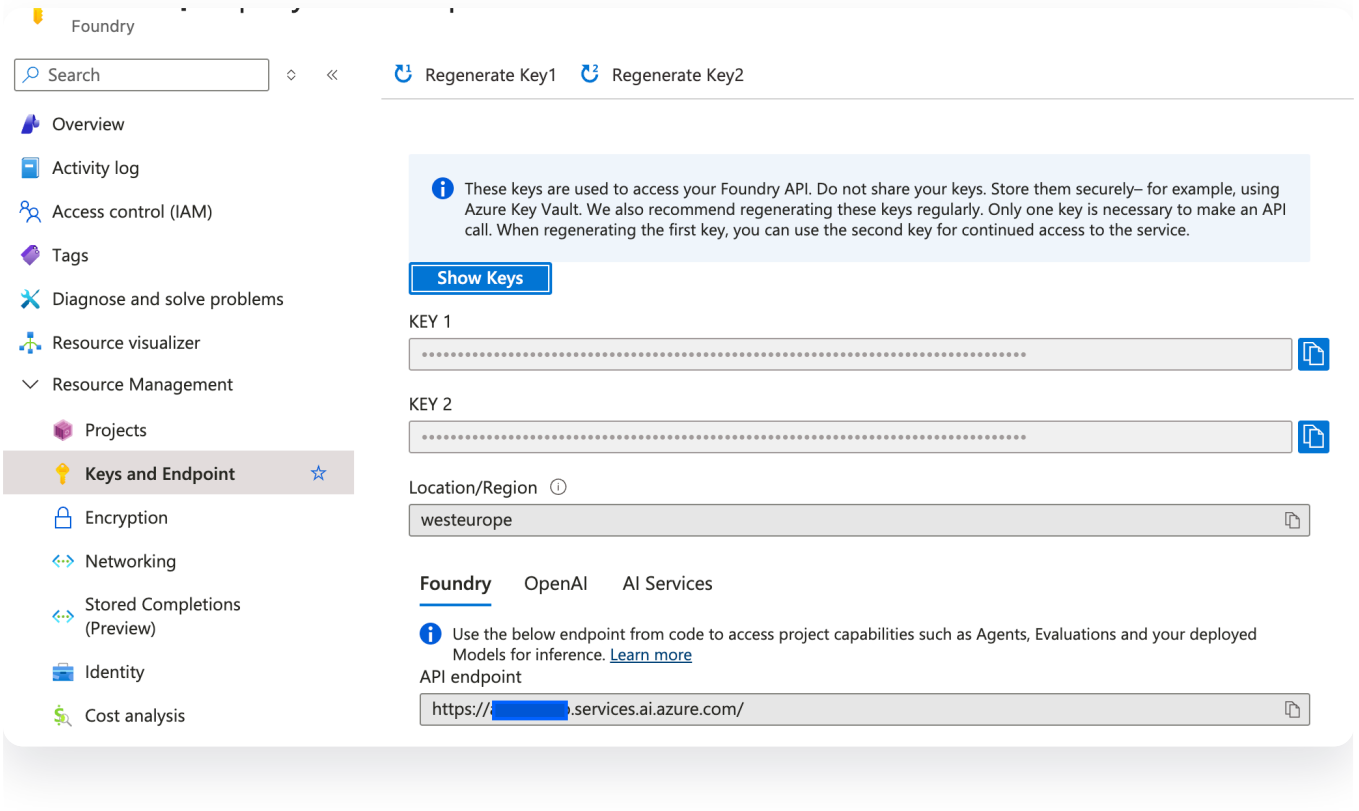
- **Microsoft Foundry project endpoint**
- **API Key** for the project



14.6.3 Keys and Endpoint (Azure Portal)

In Azure Portal on the Foundry resource:

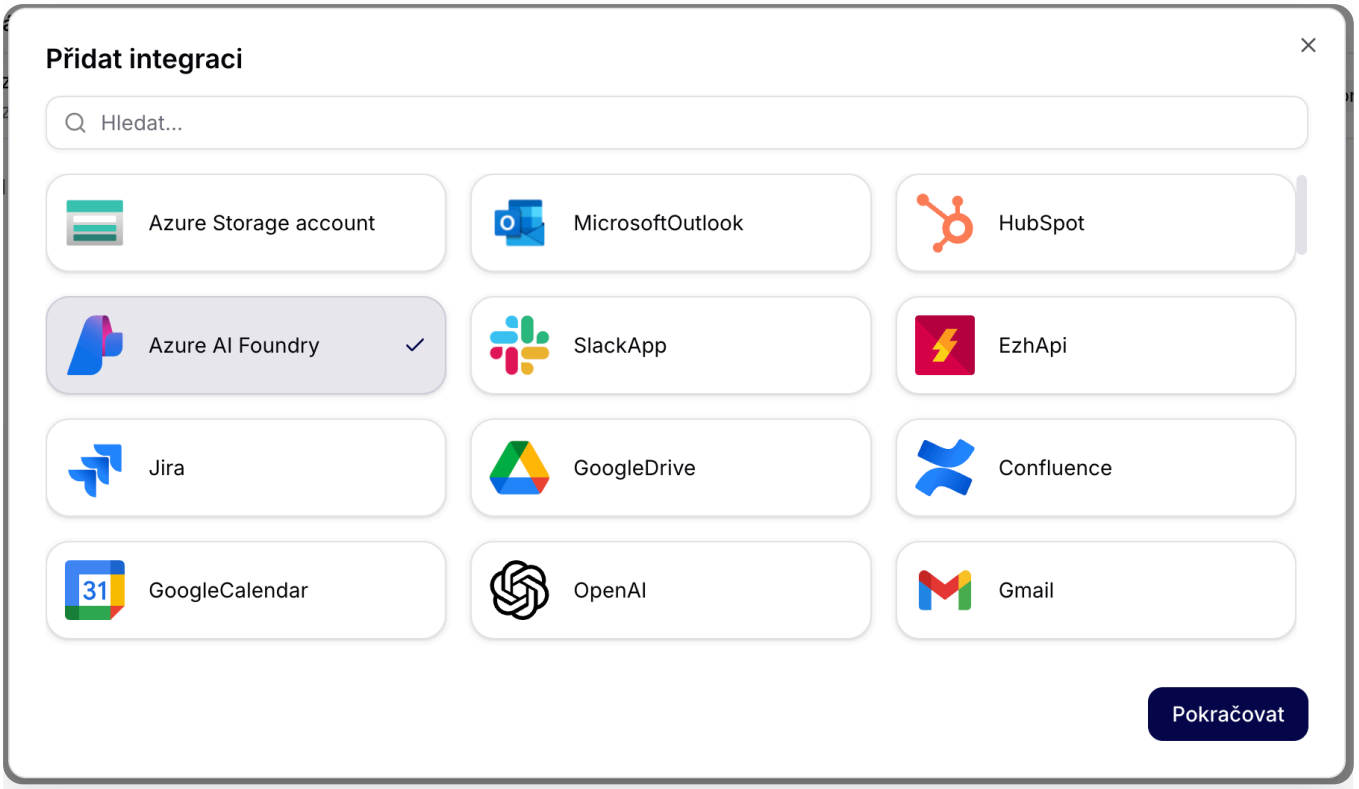
- **Keys and Endpoint → Foundry**
- **Key 1 / Key 2** for rotation
- Basic endpoint resource



14.7 Connecting Azure AI Foundry to Siesta AI

14.7.1 1) Adding integration

1. Sign in to **Siesta AI Admin**.
2. Open **Integrations**.
3. Click on **Add integration**.
4. Select **Azure AI Foundry**.



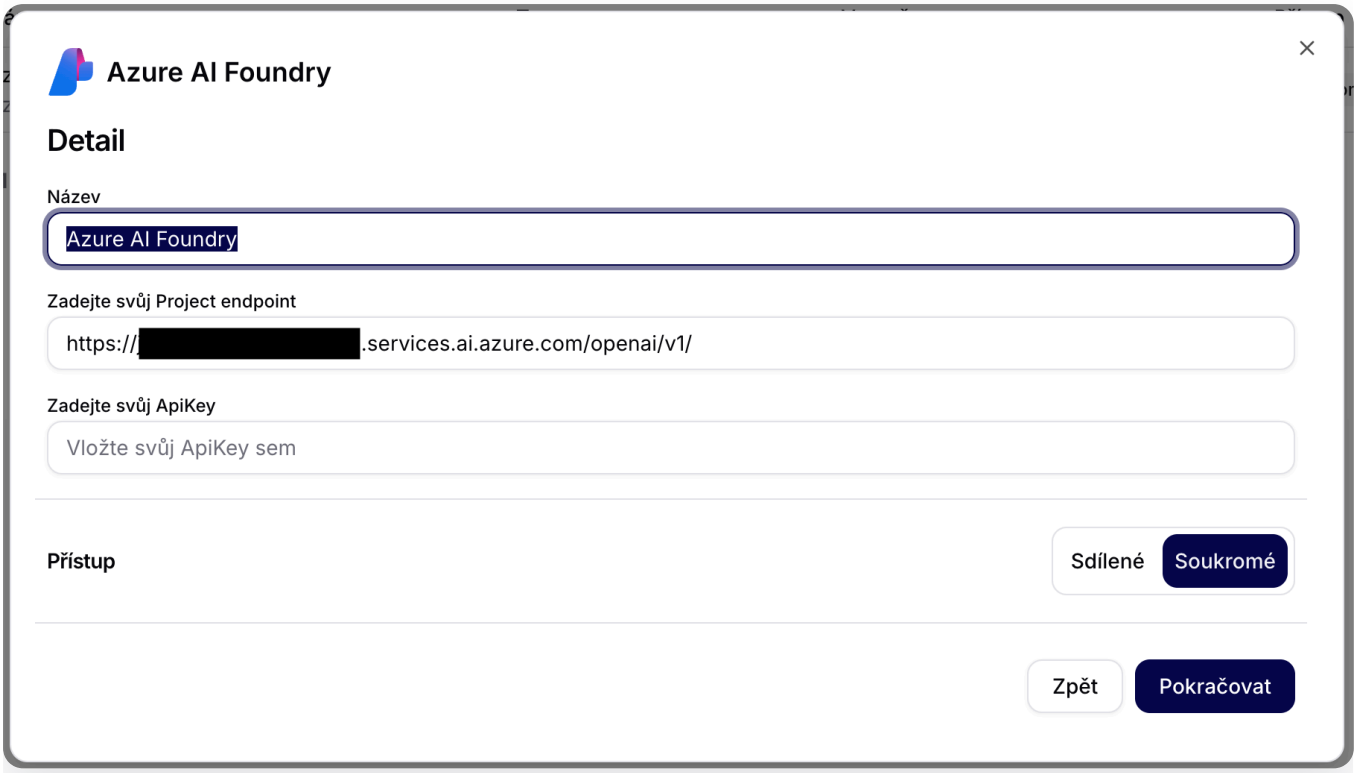
14.7.2 2) Filling in integration details

Fill in:

- **Name**: e.g., `Azure AI Foundry – PROD`
- **Project endpoint (OpenAI-compatible)**:

```
https://<foundry-resource-name>.services.ai.azure.com/openai/v1/
```

- **ApiKey**: use the **API key from the project** (from ai.azure.com) or the key from Azure Portal.
- **Access**: **Private** (recommended)



14.7.3 3) Verifying integration

After saving the integration:

- Siesta AI will perform a validation test.
- The endpoint and key will be stored securely.
- The integration will be available for **assistants**, **workflows**, and **data collections**.

14.8 Using Models in Siesta AI

1. Open **Assistant / Template / Workflow**.
2. Select **Model provider: Azure AI Foundry**.
3. Choose **deployment name** (e.g., `gpt-5.2-chat`).
4. Save the configuration.

14.9 Security & Governance

- Authentication via API key.
- RBAC managed at the Azure level.
- Option for **Private Endpoint + VNET**.
- Audit logs in **Azure Activity Log**.
- Monitoring via Foundry + **Azure Monitor**.

14.10 Recommended Architecture

- One Foundry resource per environment (DEV / STAGE / PROD).
- Multiple projects for teams or customers.
- Separate model deployments.
- Key rotation via **Key Vault**.

14.11 Increasing Quota

If you need to increase the Azure AI Foundry quota, use this document:

- Increasing Azure AI Foundry Quota

14.12 Useful Links

- Azure AI Foundry portal: https://ai.azure.com
- Documentation: https://learn.microsoft.com/azure/ai-studio/
- Model catalog: https://ai.azure.com/model-catalog
- Azure RBAC: https://learn.microsoft.com/azure/role-based-access-control/

14.13 Summary

Azure AI Foundry acts as an enterprise AI backbone, while Siesta AI builds assistants, workflows, data collections, and integrations with SaaS systems on top of it. The integration is auditable and fully under the customer's control in Azure.

15. Increase of Azure AI Foundry Quota

If you need to increase your Azure AI Foundry quota, use this document, which summarizes the necessary information and refers to the quota increase request form and documentation on models and regions.

15.1 Why an Increase is Needed

Your AI assistants run (or will run) directly in your Azure AI Foundry environment, so all AI workloads are subject to the limits of your Azure subscription (TPM/RPM).

Default quotas are primarily set for testing and PoC. In production deployment, especially during document ingestion and embedding generation, these limits often become a bottleneck and significantly slow down processing.

Increasing the quota will allow for:

- faster document ingestion and re-indexing,
- higher throughput for embedding generation,
- stable performance under concurrent user load,
- lower latency and less throttling,
- production scale and reliability.

Important: Increasing the quota does not change the price. It only increases throughput. Billing remains strictly based on consumed tokens — the price per token is the same.

This is a standard Azure process for production AI deployment. We will provide you with pre-filled parameters and a justification template to make the request quick and easy.

15.2 Data for Quota Increase Request

#	Field	Value / Note
1	Name (Authorized Representative of the Applicant)	[CLIENT]
2	Surname	[CLIENT]
3	Company Email (on company domain)	[CLIENT]
4	Company Name	[CLIENT]
5	Company Address	[CLIENT]
6	City	[CLIENT]
7	ZIP Code	[CLIENT]
8	Country	[CLIENT]

9	Subscription ID	[CLIENT] or [SIESTA.AI], if we have access to your Azure subscription
10	Justification (EXAMPLE)	Below
11	Model Type	Azure OpenAI
12	Model Deployment Quota	Model Deployment (PTU/RPM/TPM)
13	(Azure OpenAI) Quota Request Type	Global Standard
14	Global Standard Region	East US2 or Sweden Central
15	(Azure OpenAI) Global Standard Model	text-embedding-3-large
16	Quota	10000

15.3 Example Justification

We are building and operating a production AI SaaS platform focused on enterprise automation (document analysis, RAG assistants, email triage, CRM integration, and automation of internal processes for B2B clients). We are currently running in pilot and production deployments across industries (manufacturing, real estate, insurance, enterprise services). Typical workloads include:

- high-frequency chat and API inference,
- large pipelines for document ingestion and vectorization (PDF, DOCX, web crawling),
- contextually demanding prompts with multi-step reasoning,
- concurrent use by multiple enterprise users and teams.

Current quotas are already a bottleneck during peak load and testing. With the expansion of onboarding new customers and the introduction of additional assistants and integrations (HubSpot, Gmail, Google Drive, Azure Storage, internal CRM), we expect a significant increase in token throughput. We need the quota increase to:

- maintain stable latency during concurrent enterprise operations,
- support batch processing of documents and continuous ingestion pipelines,
- ensure production reliability and SLA,
- eliminate throttling during load spikes from real business workflows.

This quota increase is critical for upcoming production deployments and commercial rollouts. Without higher capacity, our ability to scale customers and ensure consistent service quality will be limited. We commit to responsible usage, cost monitoring, and effective optimization of prompts and tokens in accordance with Azure OpenAI best practices.

16. Azure Storage Account

The Azure Storage Account serves as a central data repository for blob objects, files, queues, and tables. Applications access this storage using a **Connection String**, which contains all the necessary authentication and configuration information in a single string.

This mechanism allows for quick integration without the need for manual management of individual connection parameters.

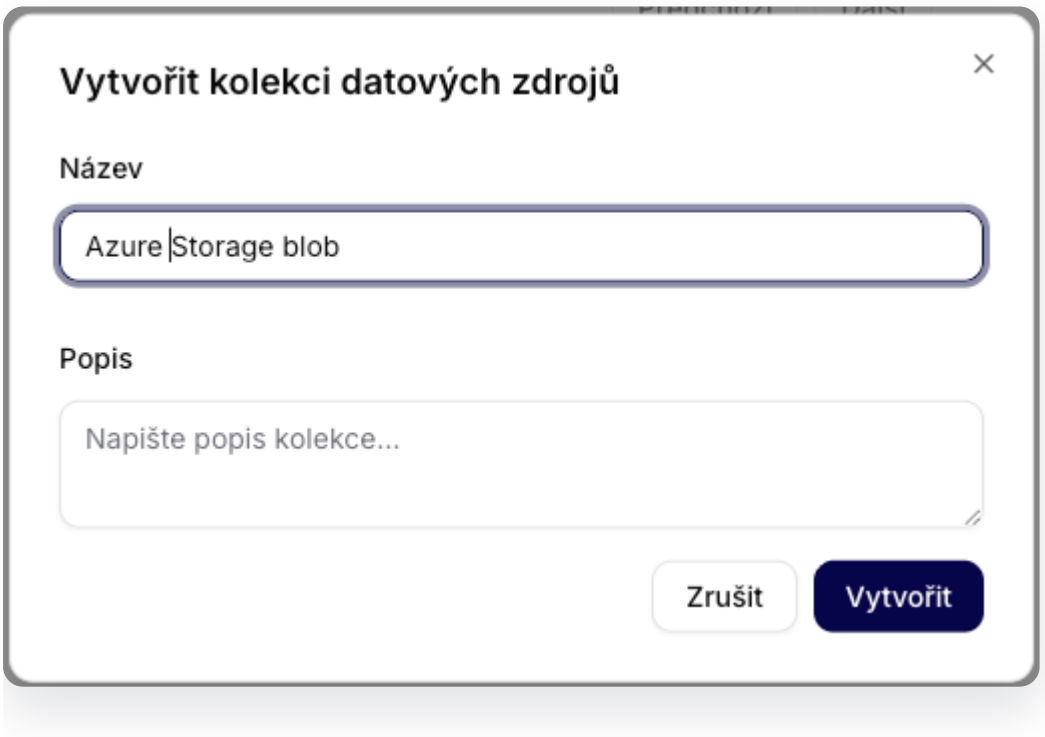
16.1 Procedure for Creating a Data Source in the Application

16.1.1 Creating a Data Source Collection

In the application administration:

1. Open the **Data Sources** section.
2. Select **Create Data Source Collection**.
3. Fill in:
   - **Name**: for example, Azure Storage Blob.
   - **Description**: optional (recommended for documenting the purpose).

Confirm by clicking the **Create** button.

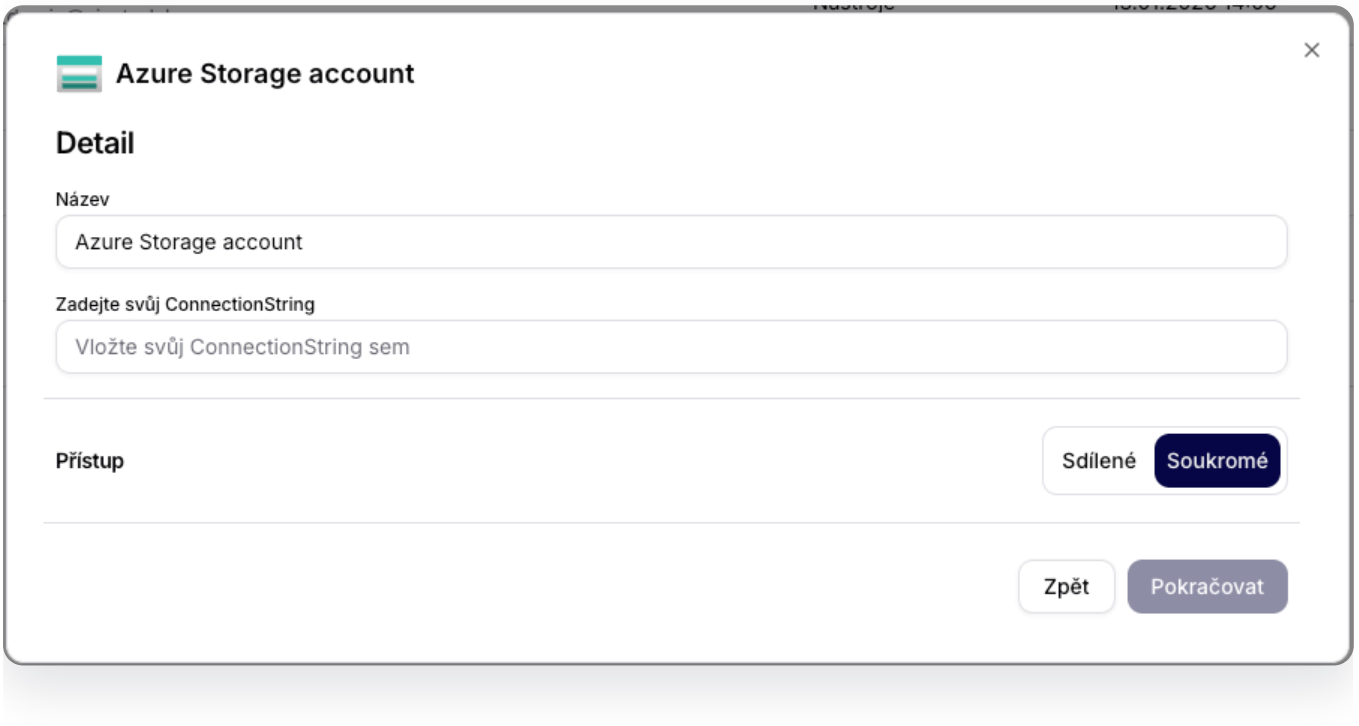


16.1.2 Adding an Azure Storage Account

After creating the collection:

1. Select the resource type **Azure Storage account**.
2. Fill in:
   - **Name**: any identifier (e.g., Production Storage).
   - **Connection String**: to be inserted in the next step.
3. Set access:
   - **Private** – recommended for production environments.
   - **Shared** – only if necessary.

Proceed by clicking the **Continue** button.



16.2 Obtaining the Connection String in the Azure Portal

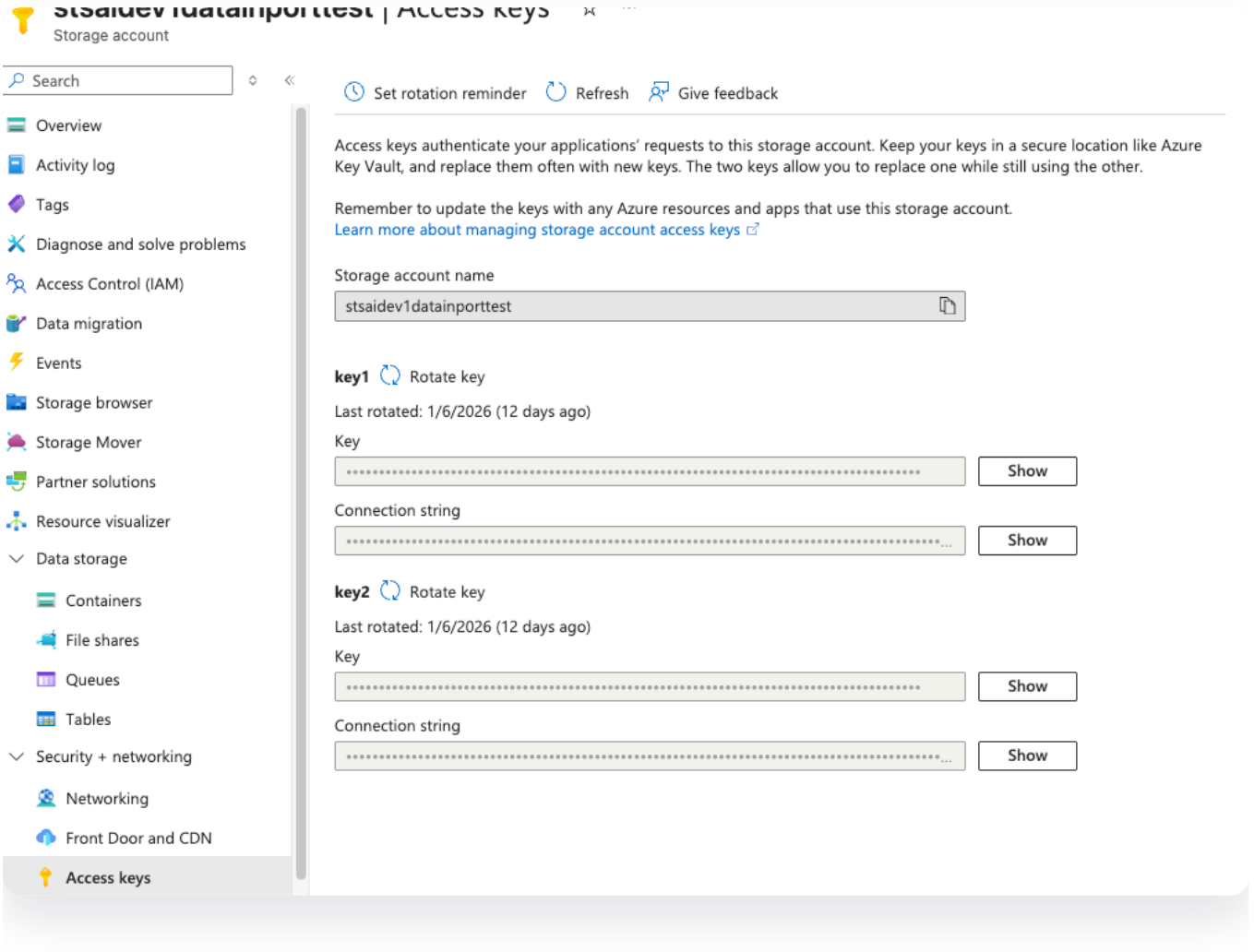
You can obtain the Connection String directly from the Azure Portal from the Storage Account configuration.

Procedure:

1. Log in to the **Azure Portal**.
2. Open the desired **Storage Account**.
3. In the left menu, select **Security + networking → Access keys**.
4. Two active key sets will be displayed:
   - `key1`
   - `key2`
5. In the **Connection string** field, click **Show**.

6. Copy the entire string.

Then paste it into the **Connection String** field in the application.



16.3 What is a Connection String and How Does It Work

A Connection String is a composite authentication string that contains:

- The name of the Storage Account
- The access key
- The protocol type
- Endpoint configuration

Typical format:

```
DefaultEndpointsProtocol=https;
AccountName=storageaccountname;
AccountKey=BASE64KEY;
EndpointSuffix=core.windows.net
```

What it means:

Element	Function
Protocol	Ensures encrypted communication (HTTPS)
AccountName	Identification of the Storage Account
AccountKey	Cryptographic access key
EndpointSuffix	Azure regional infrastructure

Applications use this string to:

- authenticate access,
- identify the target account,
- obtain full permissions based on the type of key.

16.4 Summary

The Azure Storage Account is a quick way to connect blobs, files, queues, and tables to Siesta AI. After obtaining the Connection String, simply add the data source to the collection and select the appropriate access.

17. Atlassian Confluence

17.1 Overview

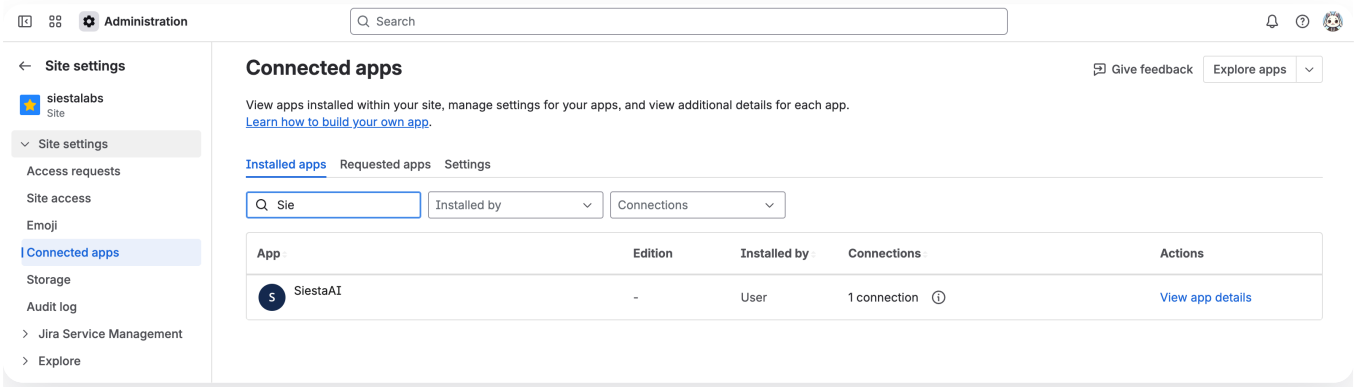
The Confluence connection enables a secure link between the Siesta AI platform and Atlassian Confluence via the official API. The integration provides controlled access to Confluence space content and allows for:

- searching for pages,
- loading documentation content,
- programmatically creating new pages,
- updating existing pages.

The connection is designed for enterprise use with an emphasis on permission management, auditability, and data access security.

17.2 Requirements

- An active **Confluence Cloud** site.
- The **SiestaAI** app installed in Atlassian (Connected apps).
- Administrator permissions to manage the connection in Siesta AI.



17.3 Supported Operations

Operation	Description
`SearchPagesAsync`	Full-text search of pages within the selected Confluence space
`GetPageAsync`	Load the content of a page including metadata
`CreatePageAsync`	Creating new pages in Confluence
`UpdatePageAsync`	Updating existing pages

17.4 Permission Management

Each operation can be set individually:

- **Allowed** – the operation is available without restrictions.
- **Allowed with confirmation** – the operation requires manual approval.
- **Denied** – the operation is not available.

This model allows for precise definition of access scope, e.g., for:

- read-only assistants,
- automated documentation processes,
- controlled entries into Confluence.

17.5 Configuration Parameters

17.5.1 Required Parameters

Parameter	Description
Integration Name	Internal designation of the connector
Confluence site URL	URL of the Atlassian instance (e.g., `https://company.atlassian.net`)
Space key	Key of the target Confluence space (e.g., `ITOPS`, `DEV`)

17.5.2 Access Mode

- **Shared** – the integration is available within the organization or team.
- **Private** – the integration is available only to the owner.

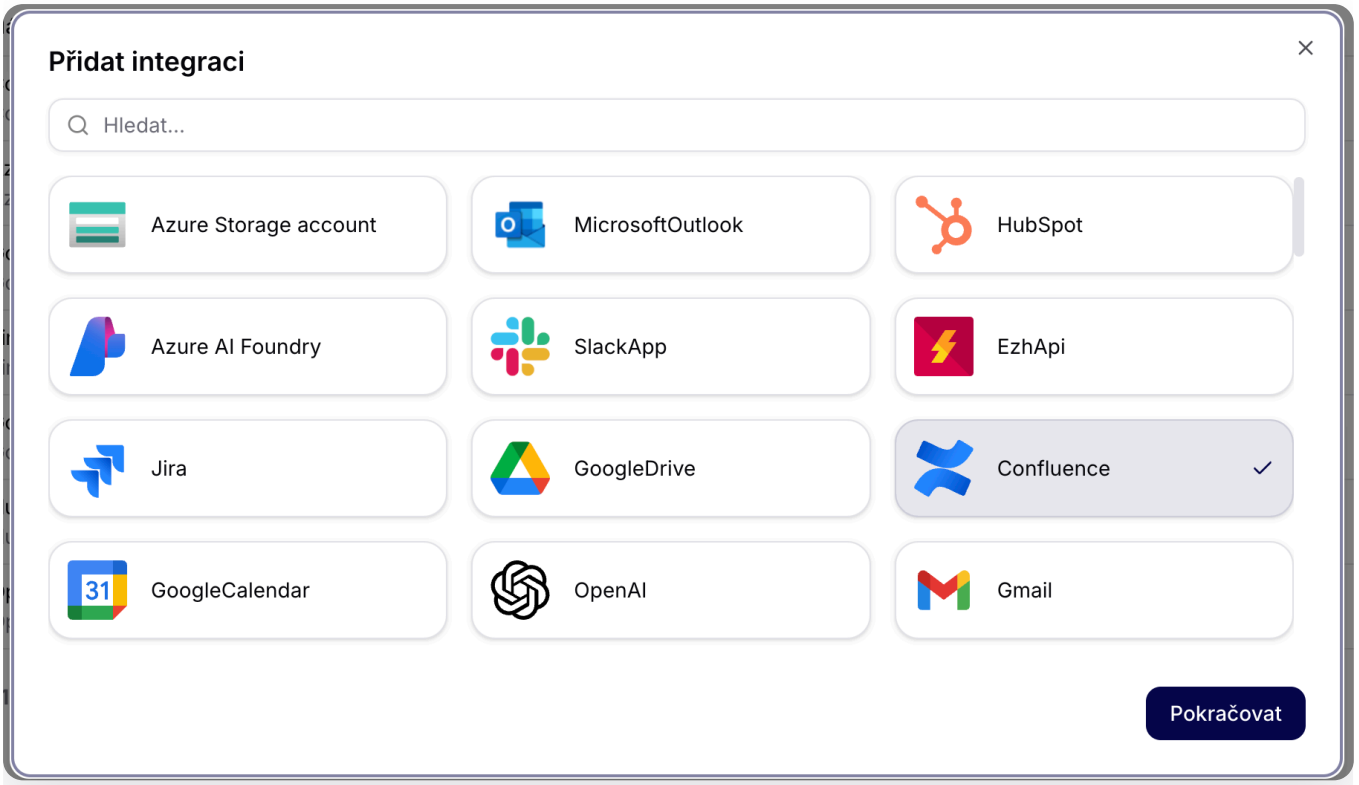
17.6 Procedure for Adding the Confluence Connector

17.6.1 Opening Integration Management

In the Siesta AI administration, go to **Administration → Connected Apps**.

17.6.2 Selecting Confluence

In the **Add Integration** dialog, select **Confluence** and proceed.

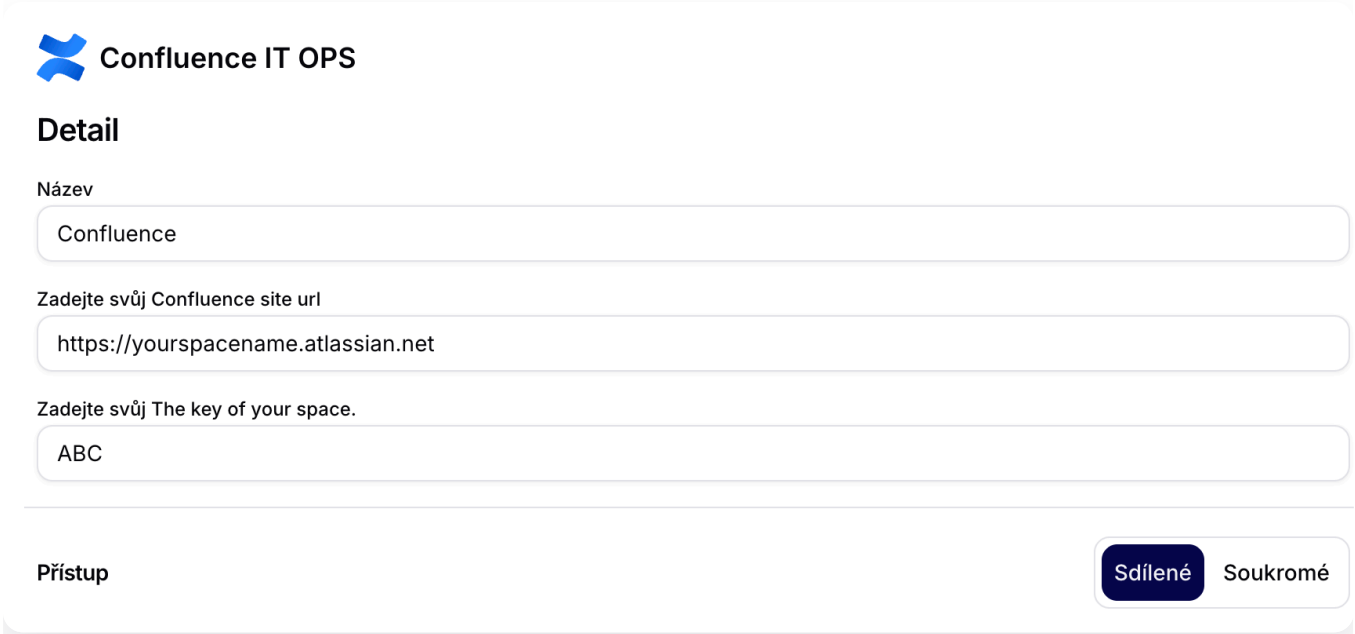


17.6.3 Configuring the Integration

Fill in:

- **Integration Name**
- **Confluence site URL**
- **Space key**
- **Access Mode**

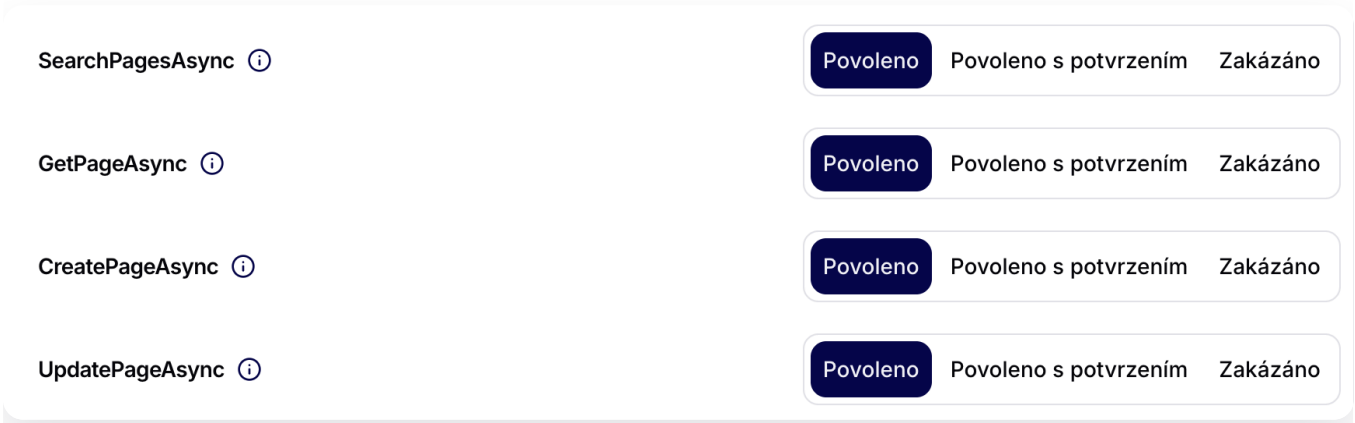
Confirm the creation of the integration.



17.6.4 Setting Permissions

After creating the integration, open **Permission Settings** and set the allowed operations.

Recommendation: set write operations to **Allowed with confirmation**.



17.7 Typical Use Cases

17.7.1 Internal Knowledge Base

- Searching internal documentation
- Answering employee inquiries
- Centralized access to current information

17.7.2 Documentation Automation

- Generating release notes

- Creating meeting minutes
- Updating standard operating procedures (SOP)

17.7.3 Process Integration

- Synchronizing Jira → Confluence
- Automatically creating reports
- Documenting incidents and audit records

17.8 Security Architecture

The integration utilizes:

- authorized access via the Atlassian API,
- restrictions to specific Confluence spaces,
- granular permission management for operations,
- the ability to audit activities.

There is no unauthorized downloading of content or bypassing of Atlassian platform security mechanisms.

17.9 Recommended Operational Configuration

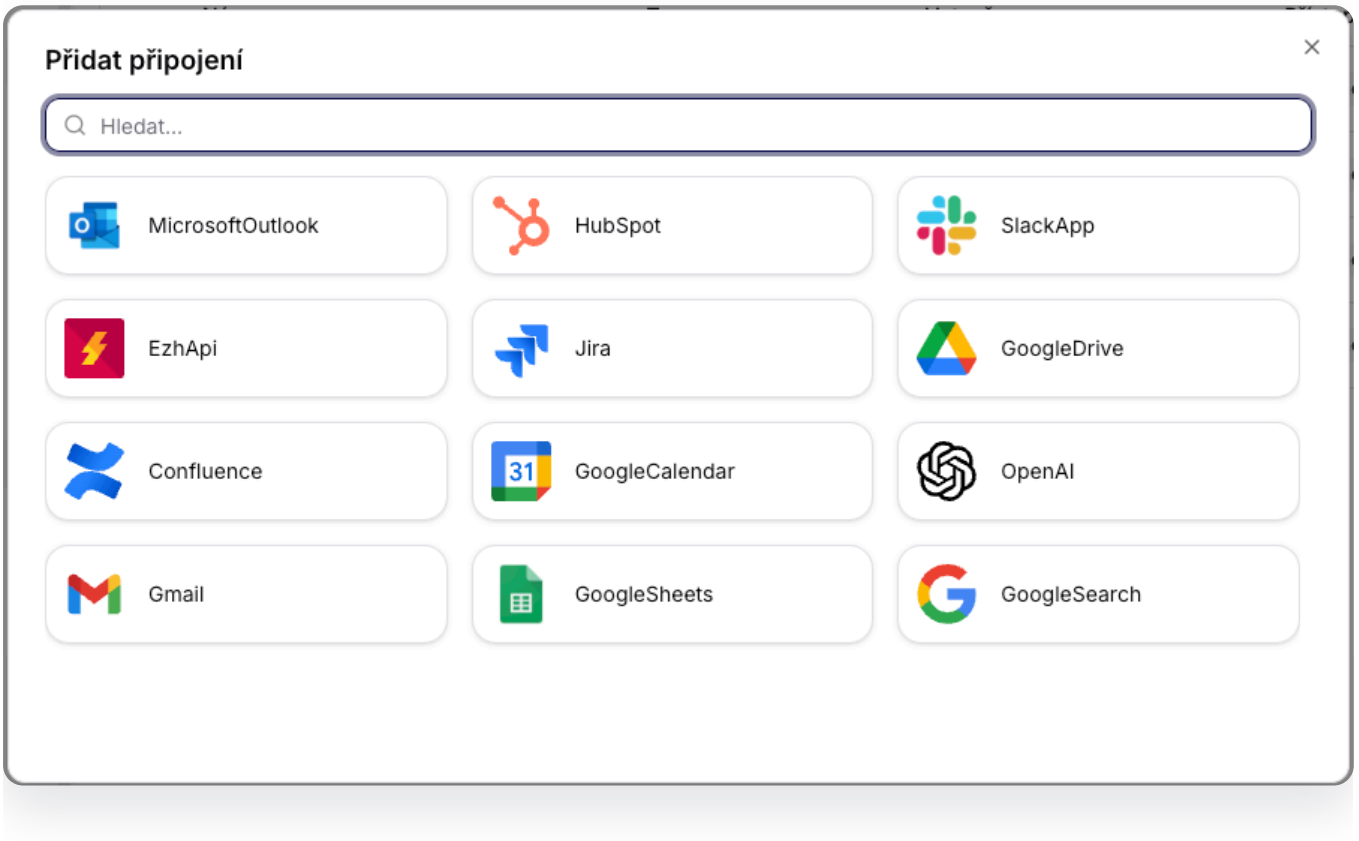
- Allow reading without restrictions.
- Set write operations to **Allowed with confirmation**.
- Use separate spaces for automated documentation.
- Regularly review connection permissions.

18. Gmail

Siesta AI - Gmail allows secure work with a Gmail account via the official Gmail API. The integration provides reading, creating, and sending emails, which can be combined with GPT models and corporate workflows.

18.1 Quick Connection

To create a connection, simply click on **Add Connection**, select **Gmail**, and the page will automatically redirect to Google sign-in. After signing in, the account is linked.



18.2 Available Operations

18.2.1 1. CreateDraft

Creates a new email draft in the Gmail inbox.

Parameter	Type	Required	Description
body	String	Yes	Email text (plain)
subject	String	Yes	Email subject
to	String	Yes	Email address

Use cases: automated email generation by AI models, preparation for approval, templates/proposals.

18.2.2 2. SendEmailAsync

Sends an email directly from the Gmail account.

Parameter	Type	Required	Description
body	String	Yes	Email text
subject	String	Yes	Email subject
to	String	Yes	Recipient address

Use cases: automatic notifications, sales/marketing sequences, follow-ups, AI-generated reports.

18.2.3 3. ListInboxAsync

Retrieves a list of the most recent messages in the inbox.

Parameter	Type	Required	Description
includeSpamTrash	Bool	No	Include spam and trash
maxResults	Int	No	Number of returned messages

Use cases: AI assistants for email, inbox summarization, categorization and routing, detection of priority messages.

18.2.4 4. GetMessageAsync

Retrieves the complete content of a specific message, including metadata.

Parameter	Type	Required	Description
messageId	String	Yes	Message ID in Gmail API

Use cases: content analysis using GPT, data extraction (orders, contacts, SLAs), thread reconstruction, context for automated replies.

18.3 Capabilities Enabled by Gmail Integration

- **AI-enhanced drafting**: generating email drafts from context and corporate data.
- **AI inbox assistant**: automatic replies, labeling, prioritization, summarization of threads.
- **Automation workflows**: follow-ups, escalations, onboarding sequences, client communication.
- **Data extraction**: structured data from emails, conversion to ticketing/CRM/ERP, links to internal systems.

18.4 Requirements for Integration

1. **Google Cloud Project**: Gmail API enabled, OAuth 2.0 Client ID created, authorized redirect URL for Siesta AI.

2. **OAuth 2.0 Authorization**: user grants access; typical scopes:
   - https://www.googleapis.com/auth/gmail.readonly
   - https://www.googleapis.com/auth/gmail.modify
   - https://www.googleapis.com/auth/gmail.send
3. **Secure token storage**: tokens are encrypted, with automatic refresh token rotation.
4. **Google API quotas & rate limits**: minimize calls to GetMessageAsync, batch inbox operations, cache metadata.
5. **Governance & controls**: validation of emails before sending, whitelisting addresses/domains, audit logs, approval workflows.

18.5 Security Considerations

- Emails and metadata are not stored without explicit purpose.
- Access tokens are encrypted and regularly renewed.
- Each operation is auditable; Siesta AI does not send emails without approval or organizational policy.

18.6 How to Connect (OAuth)

1. **Add Connection** -> select `Gmail` (same selection for Gmail/Google Calendar/Google Search).

2. **Sign in with Google** (OAuth login).

3. **Confirm integration name** (internal name in Siesta AI).

4. **Consent to permissions** (scopes according to Gmail integration).

18.7 Conclusion

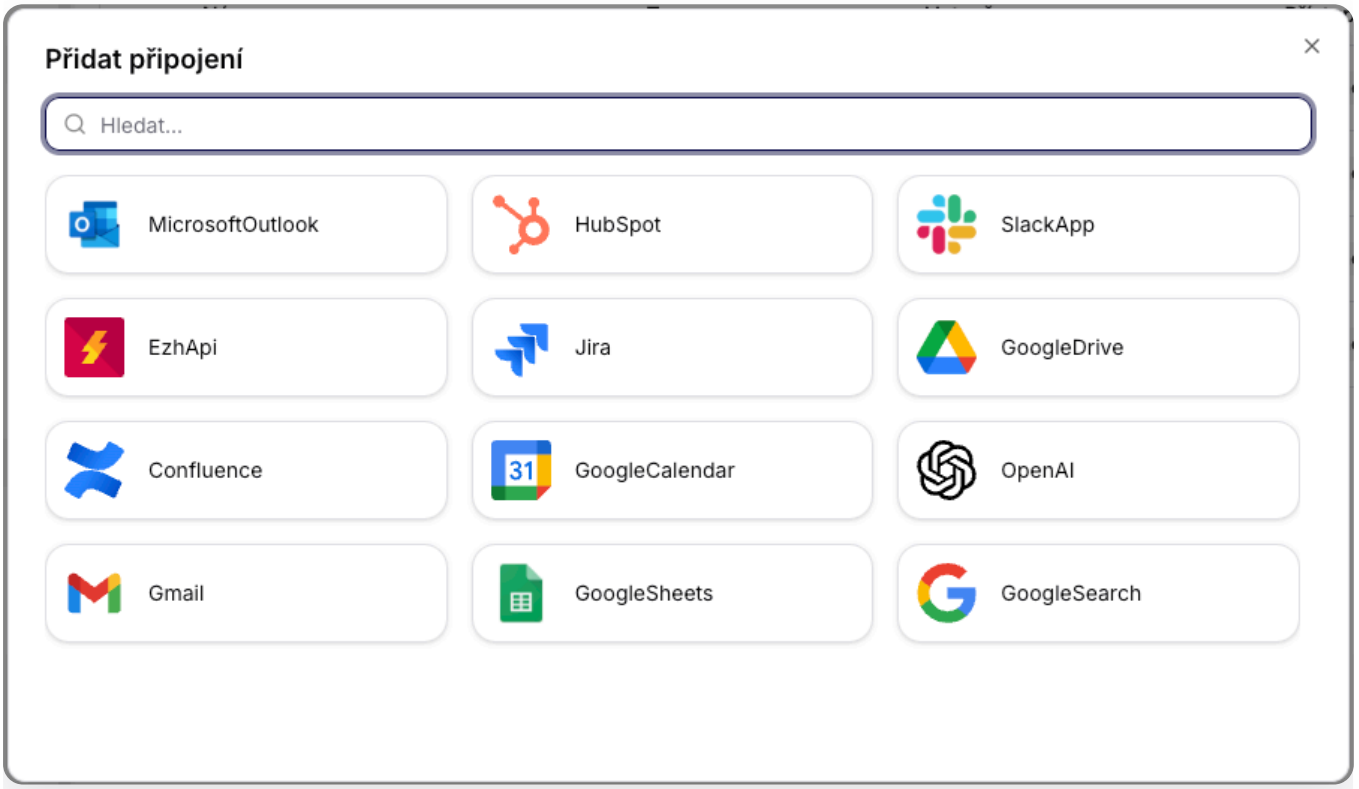
The Gmail integration provides Siesta AI with a reliable and secure way to automate corporate communication. The combination of Gmail API and AI orchestration speeds up email handling, enables analysis of incoming information, and automates routine tasks.

19. Google Calendar

Siesta AI - Google Calendar allows you to create and read events in Google Calendar through the official Google Calendar API. The assistant setup and method of delegated access are the same as for Gmail integration, so the same screens and procedures (OAuth, access assignment, sharing) can be used.

19.1 Quick Connection

To create a connection, simply click on **Add Connection**, select **Google Calendar**, and the page will automatically redirect to Google login. After logging in, the account is linked.



19.2 How to Connect (OAuth, same as Gmail)

1. **Add Connection** -> select `GoogleCalendar`.

2. **Google OAuth Login** (access to Calendar).

3. **Confirm Integration Name** (internal name).

4. **Consent to Permissions** (Calendar scopes analogous to Gmail).

19.3 Connection Overview

- **Connection Name**: GoogleCalendar
- **Type**: Google Calendar API (REST)
- **Authentication**: Google OAuth (delegated user access) - details according to internal configuration (beyond the scope of this document)
- **Scope/operations**: CreateEventAsync, ListEventsAsync

19.4 General Principles

19.4.1 3.1 Time Formats

DateTime parameters use ISO 8601. Recommendation: use explicit time zone (Z for UTC or offset +01:00, +02:00).
Example: `2025-06-21T14:00:00Z`

19.4.2 3.2 Default Calendar

If `calendarId` is not provided, the user's default calendar will be used: `primary`.

19.4.3 3.3 Recurring Events

Listing events supports the `singleEvents` option, which determines whether the recurrence is expanded into individual instances.

19.5 Detailed API Operations

19.5.1 4.1 CreateEventAsync

Description: Creates an event in the user's Google calendar (under their Google/Gmail account).

Parameter	Type	Required	Description
summary	String	Yes	Title / subject of the event
startTime	DateTime	Yes	Start of the event (ISO 8601)
endTime	DateTime	Yes	End of the event (ISO 8601)
description	String	No	Description of the event
location	String	No	Location of the event

sendNotifications	Bool	No	Whether to send notifications to participants/user

Validation Notes

- `endTime` must be strictly after `startTime`.
- Recommendation: use a consistent time zone for both times.

19.5.2 4.2 ListEventsAsync

Description: Returns a list of events from the user's calendar within the specified time range.

Parameter	Type	Required	Description
calendarId	String	No	Calendar ID (default: `primary`)
timeMin	DateTime	No	Start time for listing (inclusive)
timeMax	DateTime	No	End time for listing (exclusive)
maxResults	Int	No	Max number of events (default: 250)
singleEvents	Bool	No	Expand recurrences into instances (default: true)

Recommended Usage

- For stable results, always set `timeMin` and `timeMax`.
- If processing recurring meetings in analytics, keep `singleEvents=true`.

19.6 Security and Governance

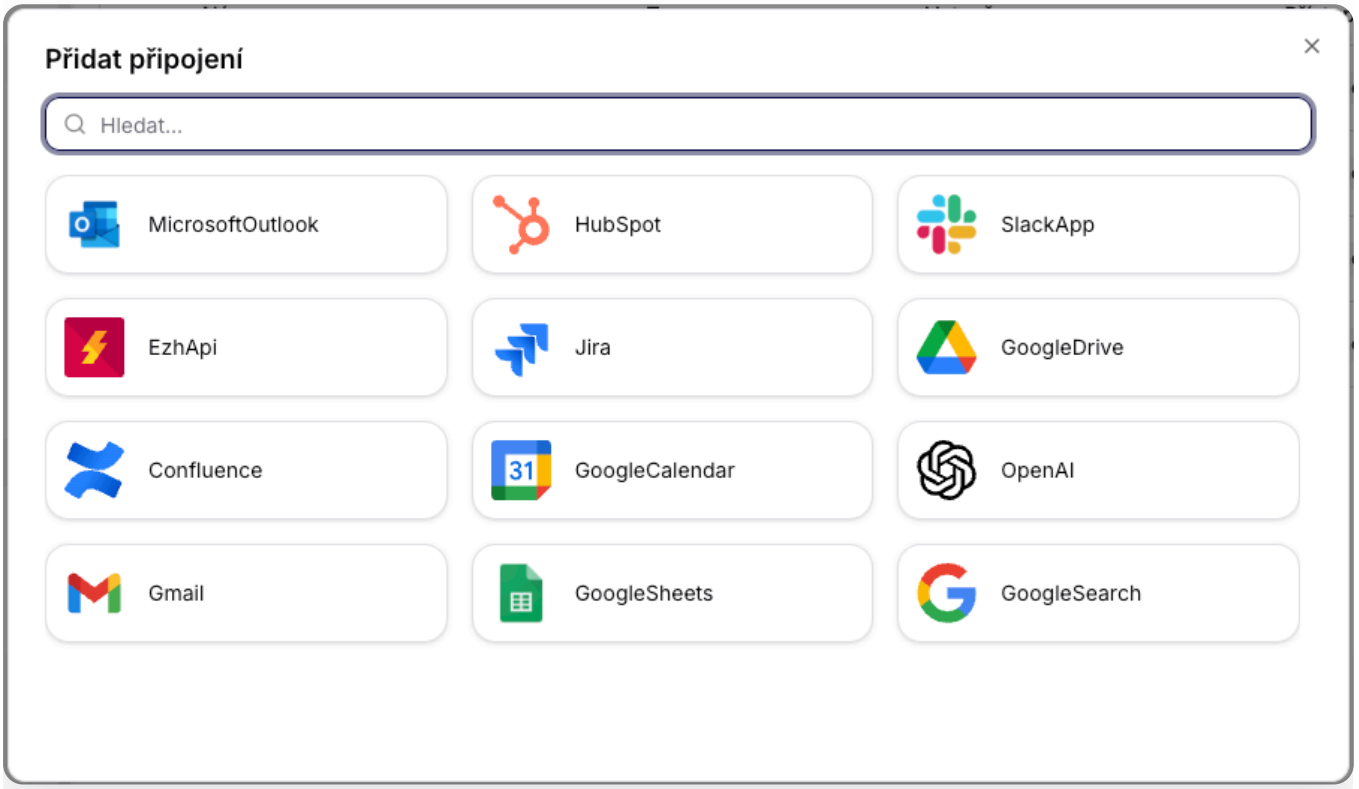
- Operations run in the context of the user (delegated access via OAuth).
- The connection only works with calendar data within the granted permissions.
- Recommendation: audit and log at least `calendarId`, time window (`timeMin` / `timeMax`) for listing, and parameters `summary` / `startTime` / `endTime` for created events.

20. Google Drive

Allows assistants and workflows to view and read files from Google Drive.

20.1 Quick Connection

To create, simply press **Add Connection**, select **Google Drive**, and the page will automatically redirect to Google login. After logging in, the account is linked.



20.2 Settings

1. In **Connections**, choose **Add Connection** and select **Google Drive**.
2. Log in via Google OAuth and grant access to Drive.
3. Set whether the connection should be **Shared** or **Private** and save.

20.3 Usage

- In parameters or workflow steps, use the Connection to load files, folders, or metadata.
- Share only the necessary drives/folders with the connected Google account to limit access scope.

20.4 Tips

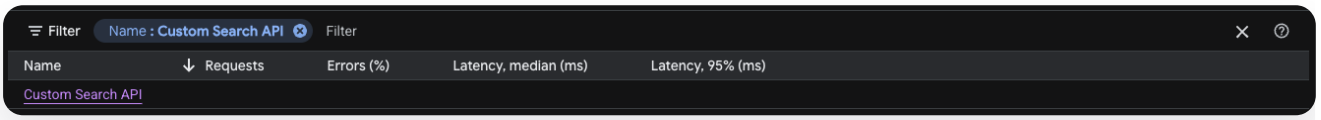
- Upon expiration or change of scope, reauthorize the Connection.
- Enable audit logs to monitor access to files.

21. Google Search API

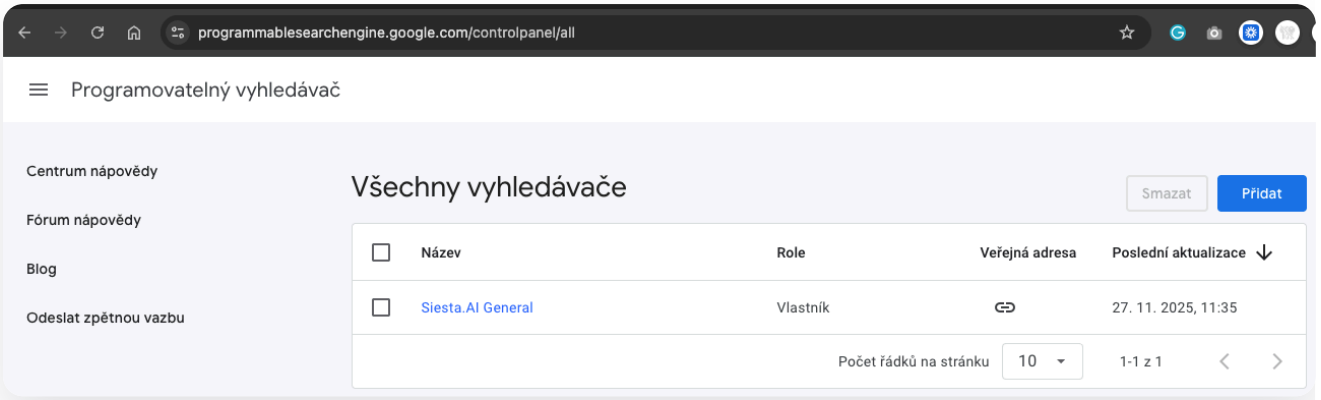
Siesta AI - Google Search allows you to programmatically perform web searches via the Google Custom Search JSON API. The connection is read-only and returns structured JSON with results.

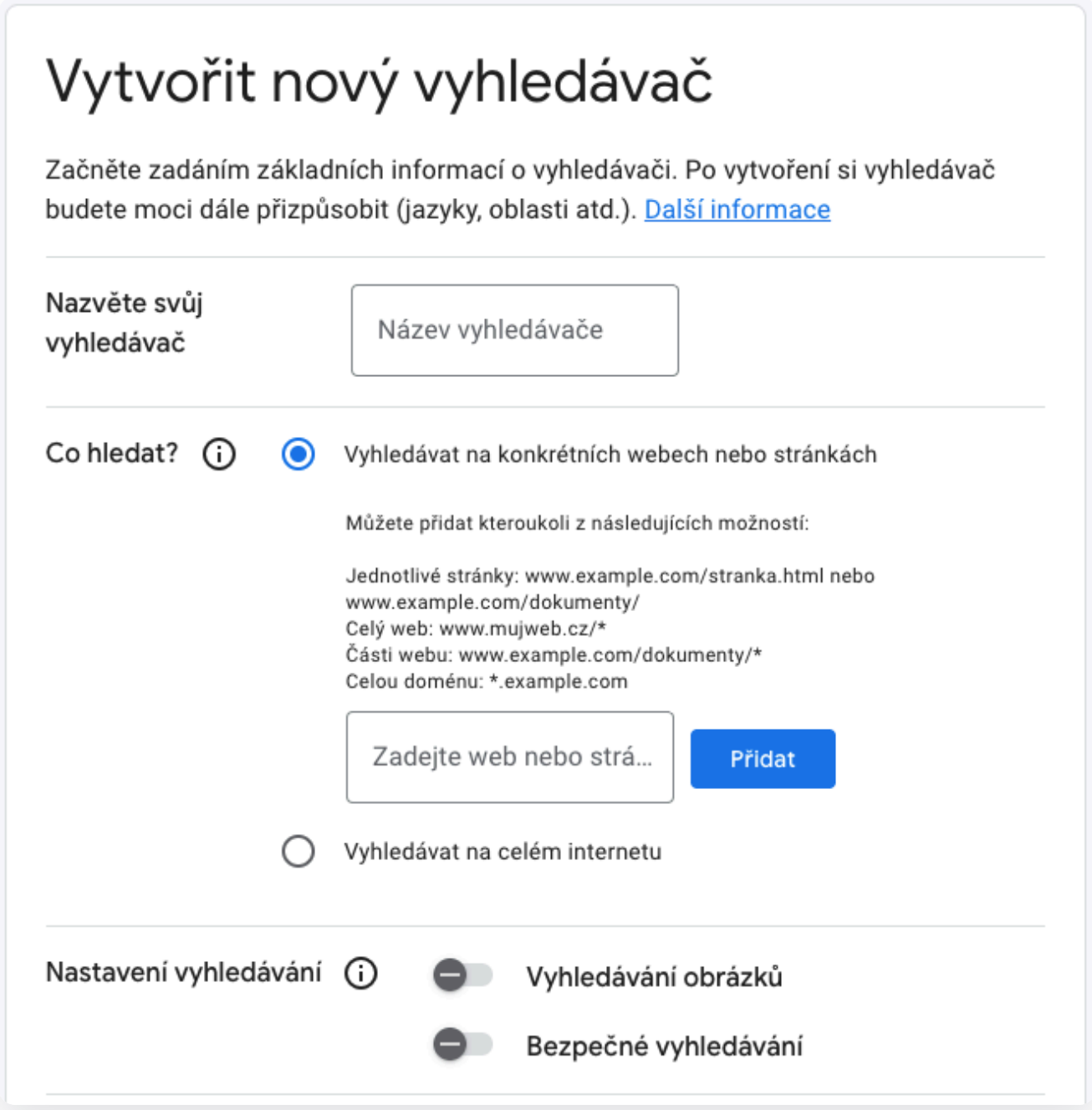
21.1 1. Setting up Google Search API (practical steps)

1. **Project in Google Cloud**: use an existing project or create a new one.
2. **Enable Custom Search API**: search for "Custom Search API" in the API Library and click **Enable**.

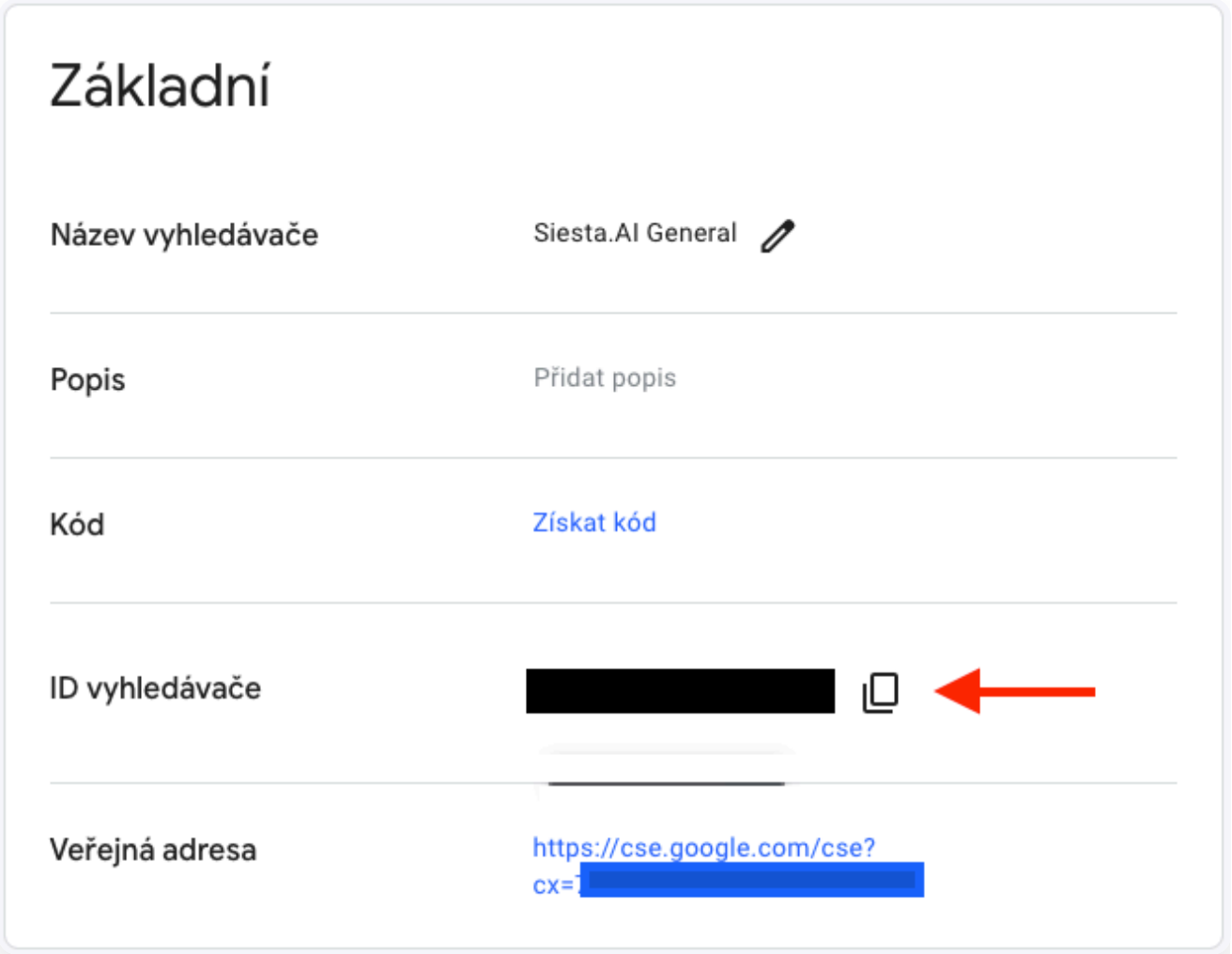


3. **Create Programmable Search Engine**: go to https://programmablesearchengine.google.com/, open the list of search engines, and click **Add**.

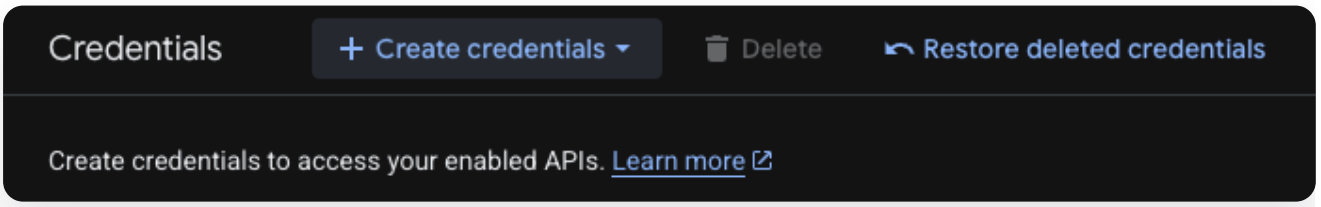

programmablesearchengine.google.com/controlpanel/all
Programovatelný vyhledávač
Centrum nápovědy
Fórum nápovědy
Blog
Odeslat zpětnou vazbu
Všechny vyhledávače
Smazat
Přidat
Název
Role
Veřejná adresa
Poslední aktualizace
Siesta.AI General
Vlastník
27. 11. 2025, 11:35
Počet řádků na stránku
10
1-1 z 1


Vytvořit nový vyhledávač
Začněte zadáním základních informací o vyhledávači. Po vytvoření si vyhledávač budete moci dále přizpůsobit (jazyky, oblasti atd.). Další informace
Nazvěte svůj vyhledávač
Název vyhledávače
Co hledat?
Vyhledávat na konkrétních webech nebo stránkách
Můžete přidat kteroukoli z následujících možností:
Jednotlivé stránky: www.example.com/stranka.html nebo www.example.com/dokumenty/
Celý web: www.mujweb.cz/*
Části webu: www.example.com/dokumenty/*
Celou doménu: *.example.com
Zadejte web nebo strá...
Přidat
Vyhledávat na celém internetu
Nastavení vyhledávání
Vyhledávání obrázků
Bezpečné vyhledávání

4. **Obtain Search Engine ID (`cx`)**: in the search engine details, open the **Basic** section and copy the **Search Engine ID**.



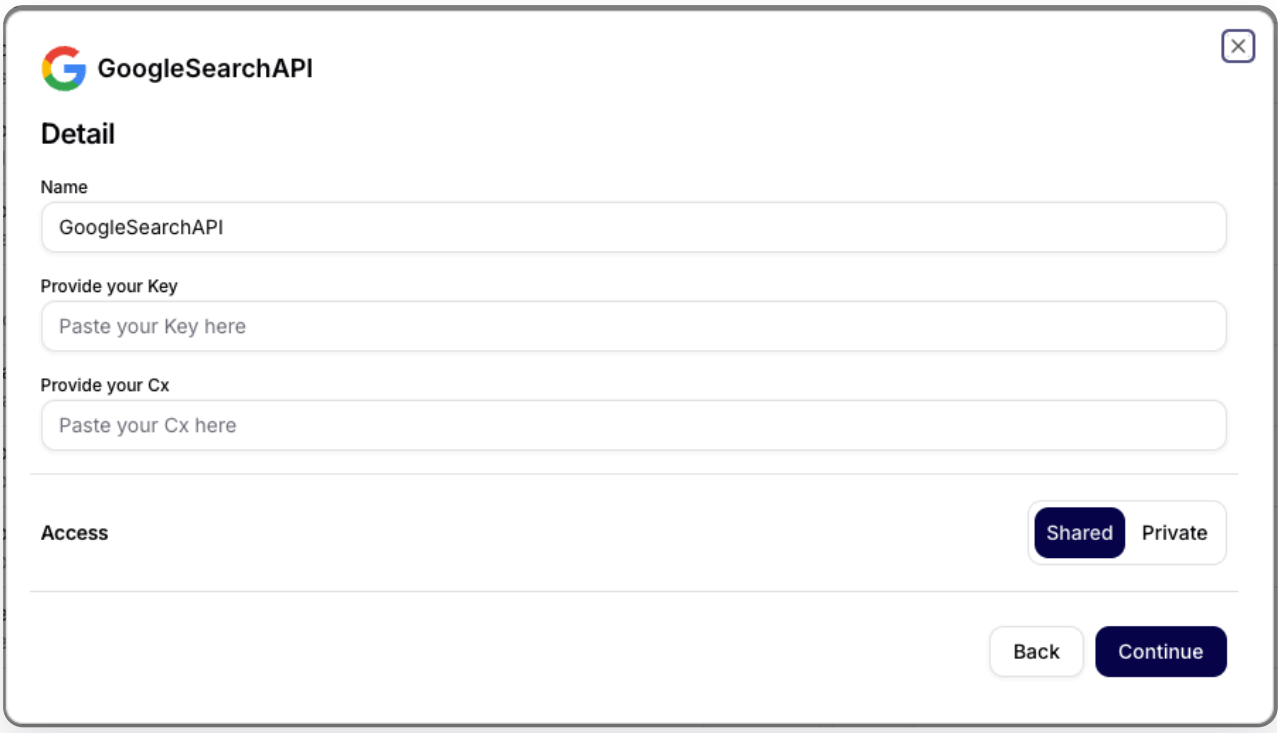
5. **Generate API Key**: in Google Cloud Console -> APIs & Services -> Credentials -> **Create credentials** -> **API key**.



6. **Key restrictions (recommended)**:
   - Application restrictions: as needed (None/Websites/IP).
   - API restrictions: **Restrict key** -> **Custom Search API**.

7. **Settings in Siesta AI**:
   - Connection -> **Add Connection** -> **GoogleSearch**.
   - Fill in `Key` (API Key) and `Cx` (Search Engine ID) and choose **Shared/Private**.

- Save via **Continue**.



21.2 2. Purpose of the document

The goal is to enable programmatic access to web search results via the Google Custom Search JSON API.

21.3 3. Connection Overview

- **Connection Name:** GoogleSearch
- **Type:** REST API - Google Custom Search JSON API
- **Authentication:** API Key (Google Cloud) + Search Engine ID (`cx`) (OAuth not required)
- **Scope:** read/search only
- **Output:** JSON object with search results
- **Note:** There are no write operations; all calls are idempotent.

The Google Custom Search JSON API allows you to programmatically retrieve search results from Google via the Programmable Search Engine, which must be created and configured before use.

21.4 4. General Principles

21.4.1 4.1 Configuration

- **Search Engine ID (`cx`):** identifier for your custom search instance.
- **API Key:** mandatory parameter for authorized calls to Google API.
- **Output:** JSON contains search metadata and result set (title, snippet, URL, pagemap, etc.).

21.4.2 4.2 Query Syntax

- The `query` parameter (alias `q`) specifies the search term.
- Advanced operators such as `site:`, `intitle:` etc. can be used (standard Google query syntax).

21.5 5. API Operations

21.5.1 5.1 Search

Description: Performs a web search via the Google Custom Search JSON API.
HTTP: `GET https://www.googleapis.com/customsearch/v1?key={API_KEY}&cx={SEARCH_ENGINE_ID}&q={query}`

Parameter	Type	Required	Description
query	String	Yes	Search term (e.g., "AI best practices").

Output

- List of results (title, URL, snippet)
- Metadata about the number of results
- Possible additional blocks (`pagemap`)

Behavior and Limits

- Standard response ~10 results per page; additional pages via `start` (outside the scope of Connection).

Typical Errors

- 400 Bad Request - invalid query
- 401 Unauthorized - invalid API Key
- 403 Quota Exceeded - daily quota exceeded

21.6 6. Security and Governance

- Keep the API Key secure; prefer restrictions (domains/IP, limit to Custom Search API).
- Monitor quotas and log for billing control.
- Log at least: `query` string, time of call, number of results, HTTP status.

21.7 7. Operational Recommendations

- Set a rotation policy for the API Key (Rotate key in Google Cloud Console).
- Keep `cx` and API Key in a secure secrets store; update Connection when changing the key.

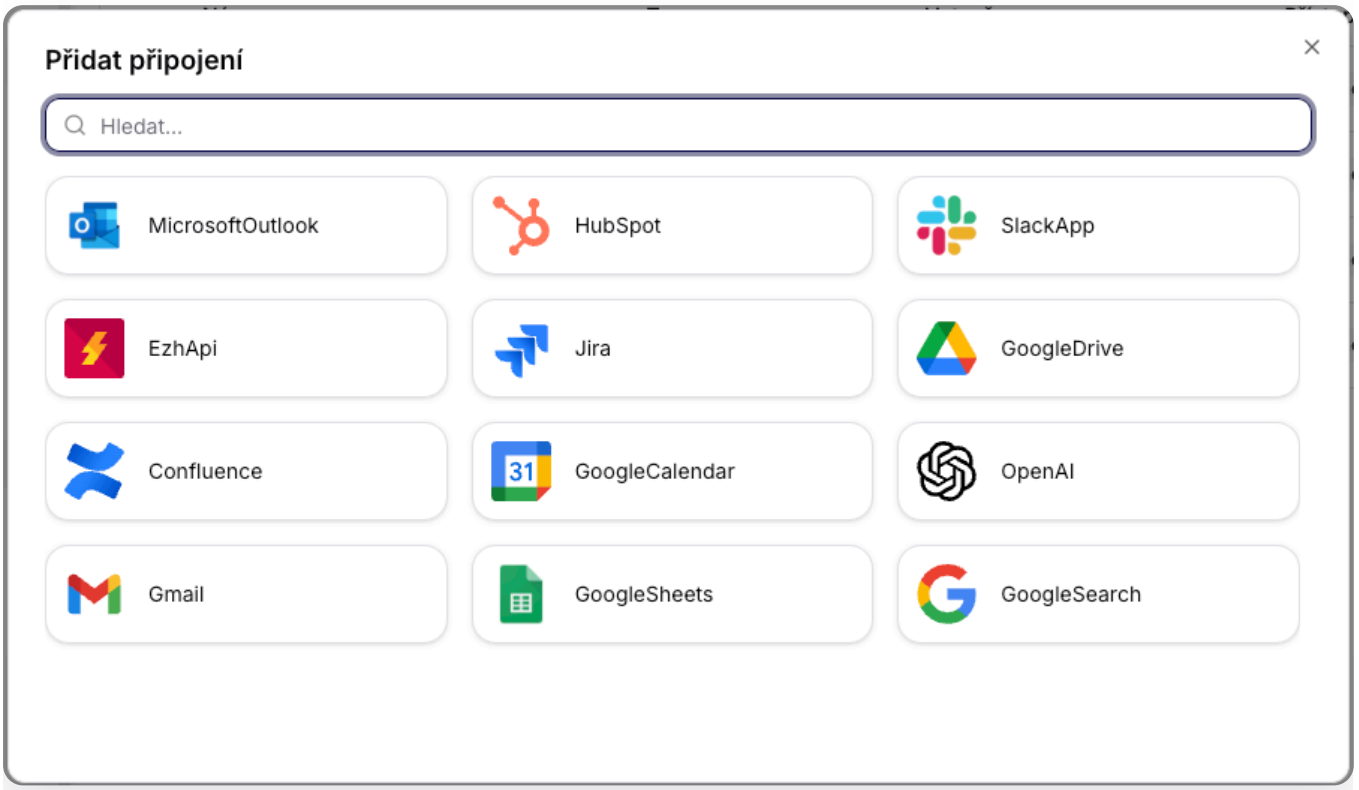
21.8 8. Example Usage

```
GET https://www.googleapis.com/customsearch/v1
  ?key=YOUR_API_KEY
  &cx=YOUR_SEARCH_ENGINE_ID
  &q=cloud+infrastructure+best+practices
```

Shortened JSON:

```
{
  "queries": { "request": [ { "query": "cloud infrastructure best practices" } ] },
  "items": [
    { "title": "...", "link": "...", "snippet": "..." }
  ]
}
```

22. Google Sheets



The connection is established in the same way as with other Google accounts in Siesta AI (OAuth).

22.1 Overview

This document describes the available integrations with Google services:

- GoogleSearch (Custom Search JSON API)
- GoogleSheets (operations on spreadsheets)

The integrations are designed as deterministic, stateless operations suitable for automation, reporting, and data pipelines.

22.2 1. Google Search Integration

22.2.1 Service Name

GoogleSearch

22.2.2 Operations

Search

Description

Performs a search using the Google Custom Search JSON API.

Input Parameters

Parameter	Type	Required	Description
query	String	Yes	The search query passed to the Google API.

Behavior

- Returns results according to the configuration of the Custom Search Engine (CSE).
- Respects the limits and quotas of the Google API.
- Does not perform any subsequent interpretation or filtering of results.

Typical Use Cases

- Market analysis
- Brand monitoring
- Automated research
- Data enrichment

22.3 2. Google Sheets Integration

22.3.1 Service Name

GoogleSheets

22.3.2 Description

The integration allows for the creation, searching, and updating of Google Spreadsheets. It is used as a lightweight data storage or export target for automated processes.

22.3.3 2.1 CreateSheetAsync

Description

Creates a new Google Spreadsheet with specified columns.

Input Parameters

Parameter	Type	Required	Description
name	String	Yes	The name of the Spreadsheet.
columnNames	String	Yes	A comma-separated list of column names.

Behavior

- If a spreadsheet with the given name does not exist, it is created.
- Columns are initialized in the first row.

Typical Use Cases

- Report initialization
- Preparing data structure for subsequent writing

22.3.4 2.2 GetSheetAsync

Description

Searches for an existing Google Spreadsheet by name.

Input Parameters

Parameter	Type	Required	Description
name	String	Yes	The name of the Spreadsheet.

Behavior

- Returns metadata of the spreadsheet.
- If the spreadsheet does not exist, the operation fails.

22.3.5 2.3 UpdateSheetAsync

Description

Replaces the content of an existing Spreadsheet with data in CSV format.

Input Parameters

Parameter	Type	Required	Description
name	String	Yes	The name of the Spreadsheet.
csvContent	String	Yes	CSV data to write. If the field contains a comma, it must be in quotes.

Behavior

- Completely replaces existing content.
- Does not perform data type validation.
- Responsibility for the correct CSV format lies with the caller.

Typical Use Cases

- Data export
- Report synchronization
- Automatic rewriting of pipeline outputs

22.4 Security Notes

- The integration runs exclusively through the official Google API.
- No direct login credentials are exposed.
- Access rights are managed at the level of the Google account or service account.

22.5 Design Decisions

- Stateless operations without cache and shared state.
- Explicit names: the spreadsheet is identified by name, not ID.
- Fail-fast approach: inconsistent data leads to an error.

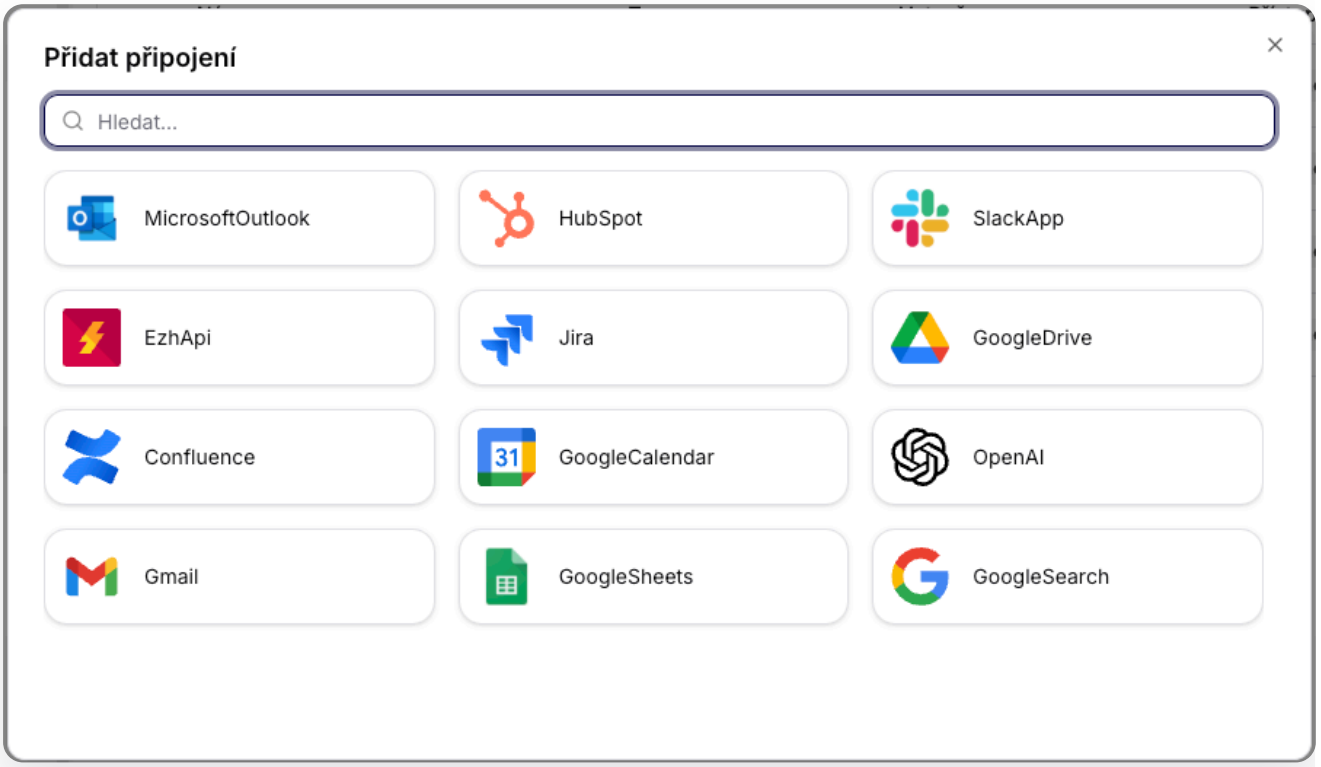
22.6 Summary

- GoogleSearch provides deterministic searching through the Google Custom Search JSON API.
- GoogleSheets covers the creation, retrieval, and updating of spreadsheets.
- Data format and permissions are fully the responsibility of the calling system.

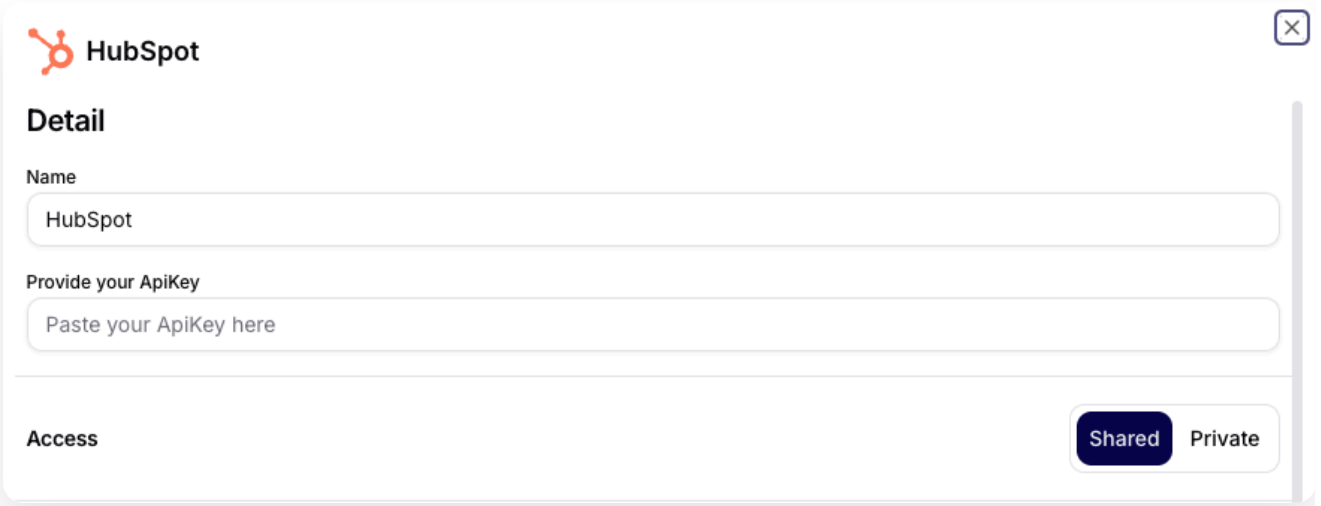
23. HubSpot

23.1 Connecting HubSpot with Siesta AI

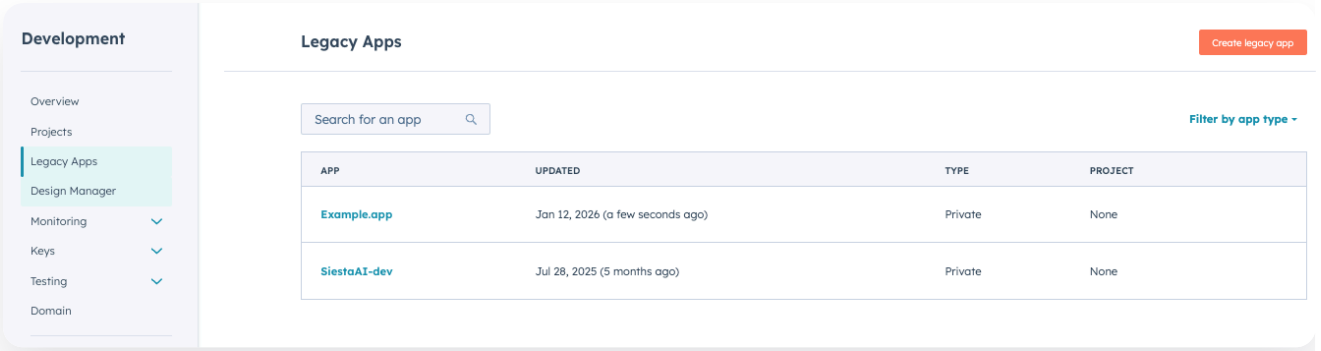
1. In **Connections**, select **Add Connection** and choose **HubSpot**.



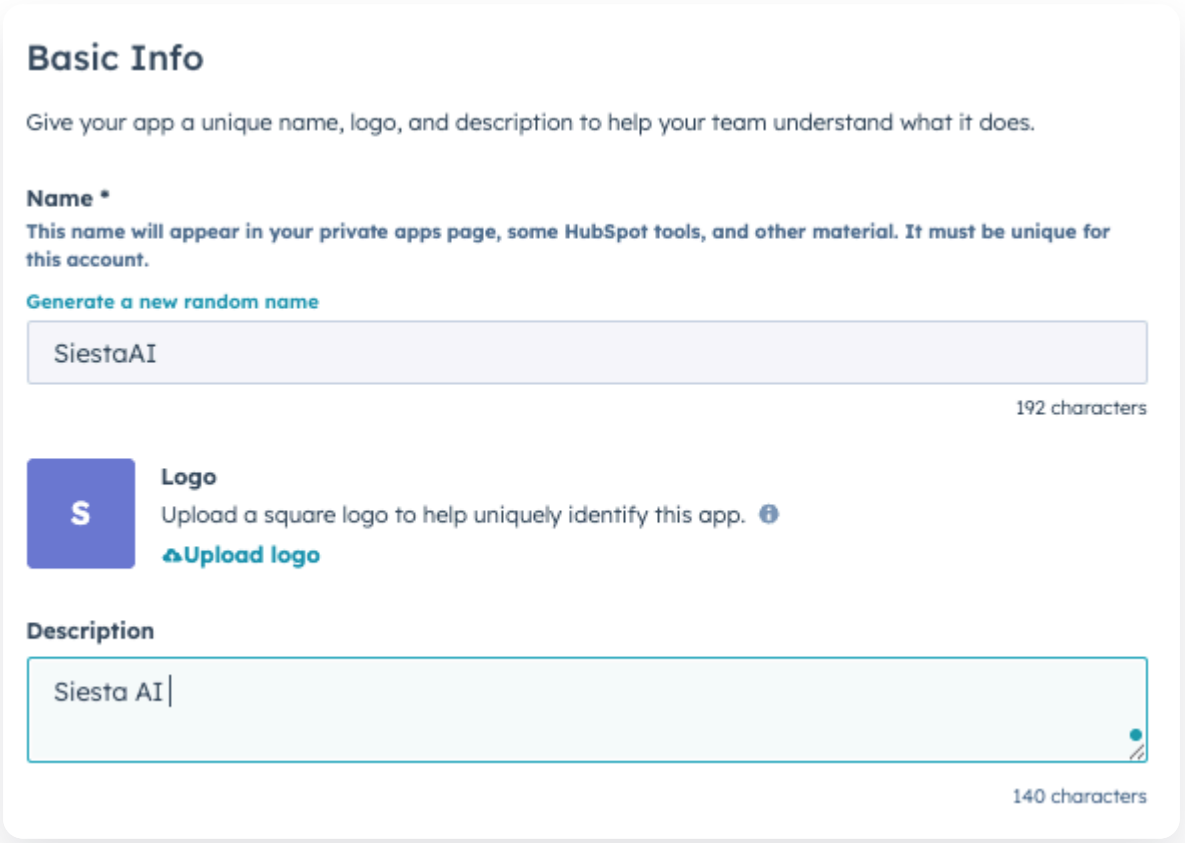
2. Enter the **Private App token** (API key) and set it to **Shared** or **Private**.



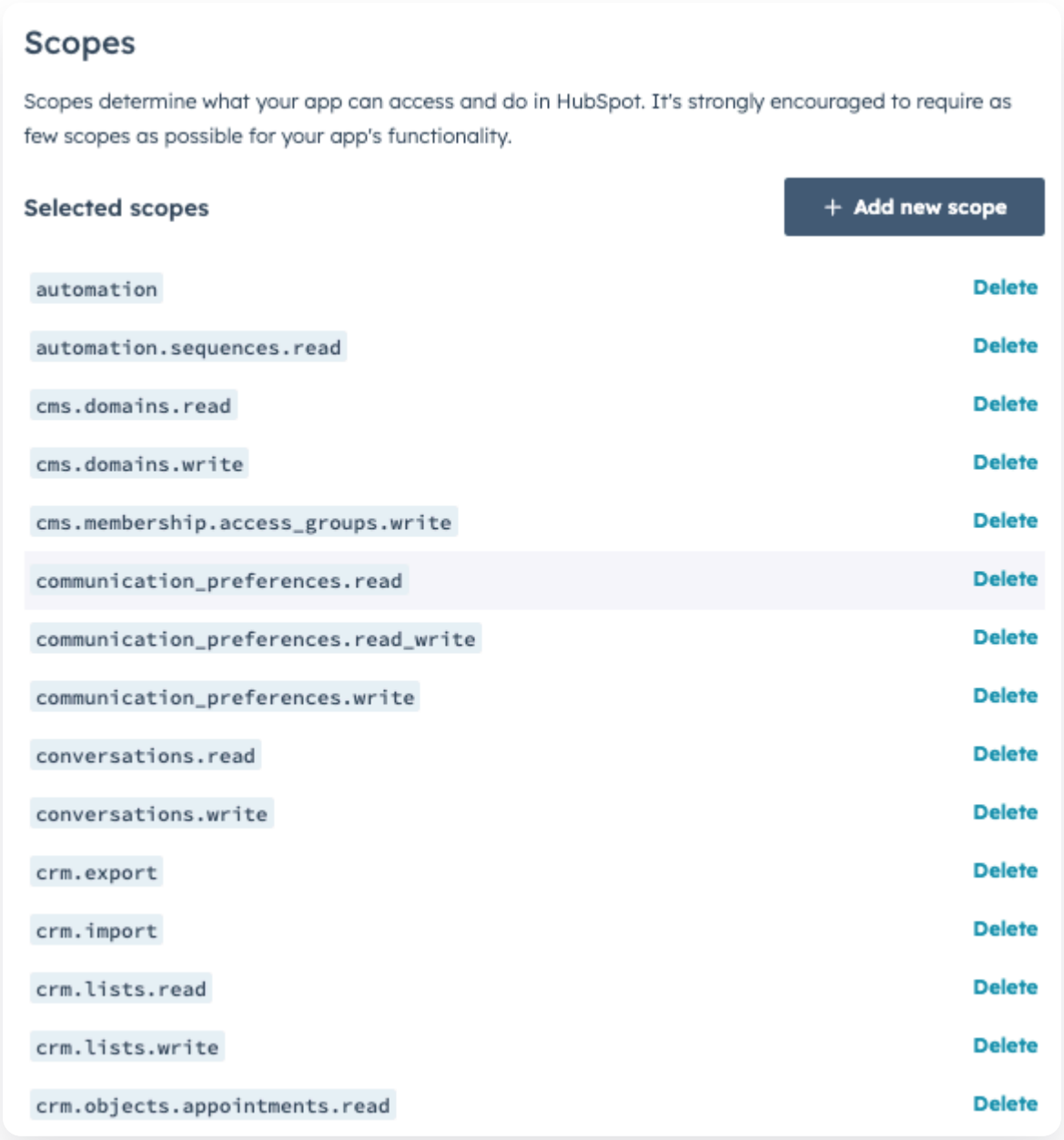
3. In HubSpot, open **Settings -> Integrations -> Private Apps** and create a new app.



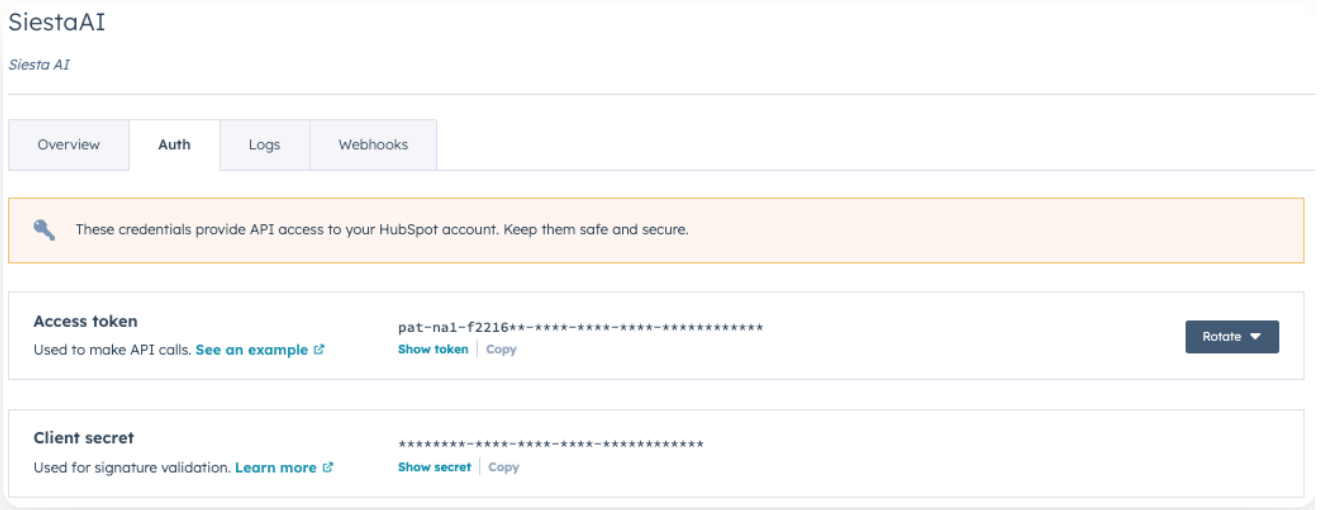
4. Fill in the basic information for the app.



5. Set the necessary scopes for CRM objects.



6. In the **Auth** tab, copy the **Access token** and use it in Siesta AI.



23.2 Overview

This Connection provides a standardized interface for working with HubSpot CRM. It allows for the creation and reading of basic CRM entities: Companies, Contacts, Deals, and Pipelines.

The integration is designed to be stateless, deterministic, and fail-fast, suitable for:

- CRM automation
- Lead synchronization
- Sales and revenue pipeline orchestration
- Auditable enterprise workflows

23.3 Authentication and Security

- The connection communicates exclusively through the official HubSpot API.
- Authentication is handled at the HubSpot account level (OAuth / Private App token).
- No sensitive data is logged or cached.
- All operations run with permissions assigned in HubSpot.

Access rights (scopes) directly affect the availability of operations. Incorrect scopes result in failure.

23.4 Supported Entities

- Company
- Contact
- Deal
- Pipeline

Each operation works with internal HubSpot IDs, not names.

23.5 1. Company Operations

23.5.1 1.1 CreateCompany

Description

Creates a new company in HubSpot CRM.

Input Parameters

Parameter	Type	Required	Description
name	String	Yes	The name of the company.
domain	String	Yes	The company's domain (must be unique).

Behavior

- If the domain already exists, the operation will fail.
- Does not perform fuzzy matching or domain normalization.

Typical Use Cases

- Onboarding new customers
- Synchronizing companies from external systems

23.6 2. Contact Operations

23.6.1 2.1 CreateContact

Description

Creates a new contact in HubSpot CRM.

Input Parameters

Parameter	Type	Required	Description
email	String	Yes	The contact's email (must be unique).
firstName	String	Yes	First name.
lastName	String	Yes	Last name.

Behavior

- Email serves as a unique identifier.
- Duplicate email = hard error.

23.6.2 2.2 GetContactByEmail

Description

Returns a contact based on the email address.

Input Parameters

Parameter	Type	Required
email	String	Yes

23.6.3 2.3 GetContactById

Description

Returns a contact by its unique HubSpot ID.

Input Parameters

Parameter	Type	Required
contactId	Int64	Yes

23.6.4 2.4 GetAllContacts

Description

Returns a list of contacts in HubSpot CRM.

Input Parameters

Parameter	Type	Required	Description
limit	Int	No	Maximum number of returned records.

23.7 3. Deal Operations

23.7.1 3.1 CreateDeal

Description

Creates a new deal and associates it with a specific contact.

Input Parameters

Parameter	Type	Required	Description
dealName	String	Yes	The name of the deal.
contactId	String	Yes	Existing Contact ID.
pipelineId	Int64	Yes	Pipeline ID (not name).

stageId	Int64	Yes	Stage ID (not name).
amount	Int	No	The value of the deal.

Behavior

- Both pipeline and stage must exist.
- Does not perform automatic mapping of names to IDs.
- Incorrect relationship = fail.

23.7.2 3.2 GetDealById

Description
Returns deal details by ID.

Input Parameters

Parameter	Type	Required
dealId	Int64	Yes

23.7.3 3.3 GetAllDeals

Description
Returns a list of deals.

Input Parameters

Parameter	Type	Required
limit	Int	No

23.8 4. Pipeline Operations

23.8.1 4.1 ListAllPipelines

Description
Returns all pipelines including their stages and corresponding IDs.

Input Parameters
None.

Note This step is mandatory if you do not want to create deals blindly.

23.9 5. Search Operations

23.9.1 5.1 SearchCompanies

Description

Searches for companies by name.

Input Parameters

Parameter	Type	Required	Description
name	String	No	The name of the company (without domain).

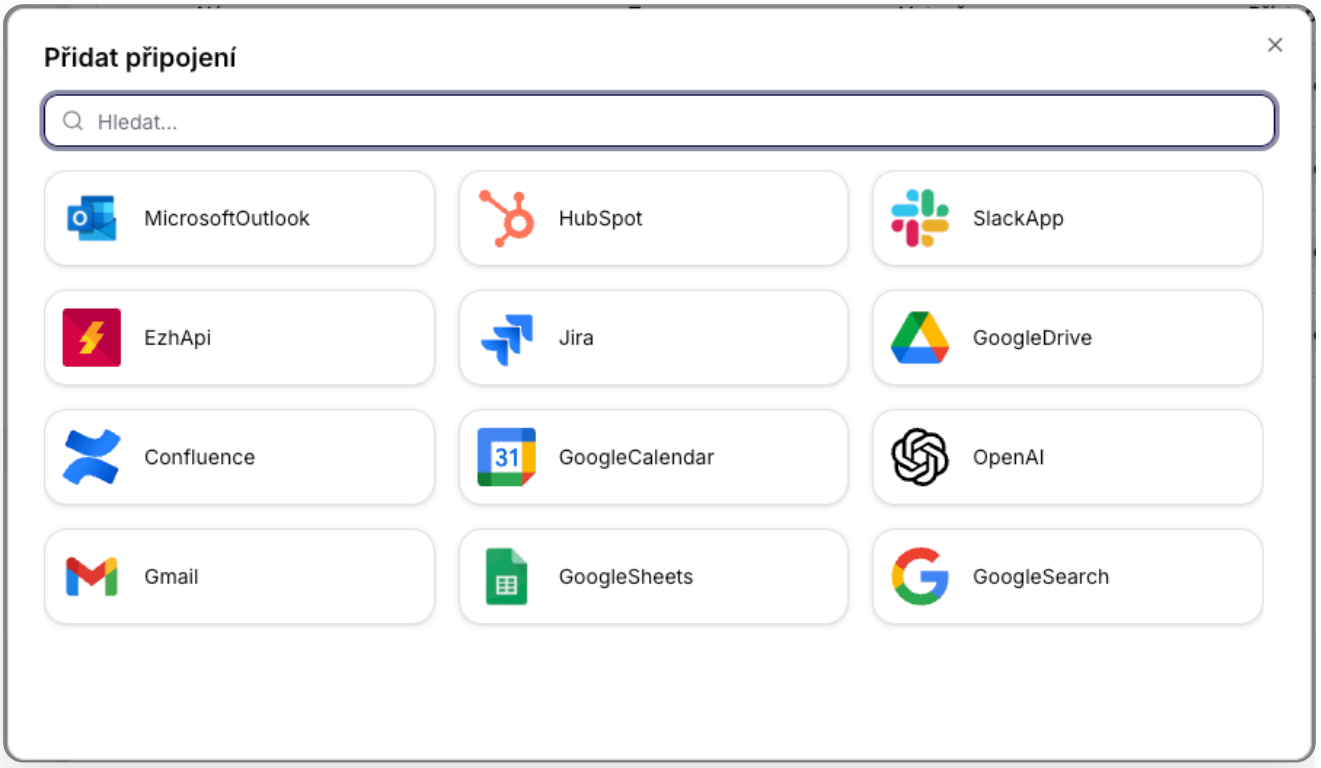
Limitations

- Do not use domains, URLs, or emails.
- The search is textual, without fuzzy matching.

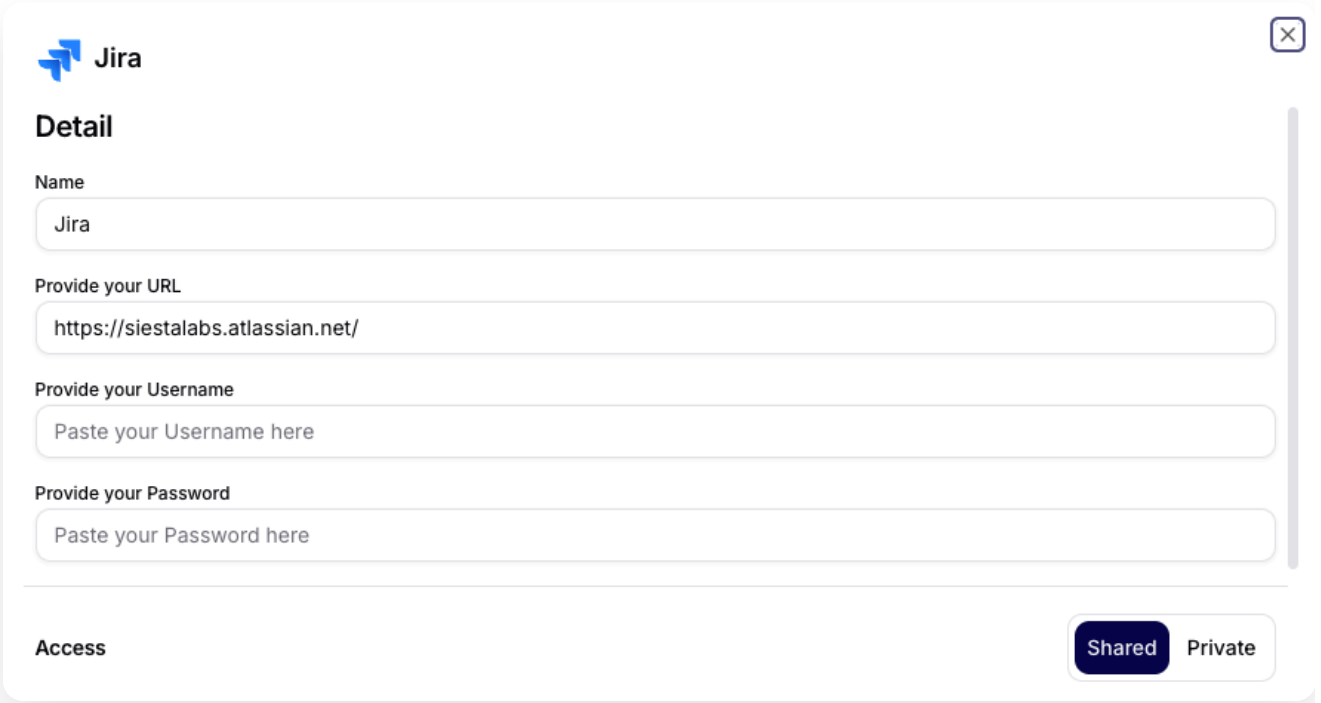
24. Jira

24.1 Connecting Jira with Siesta AI

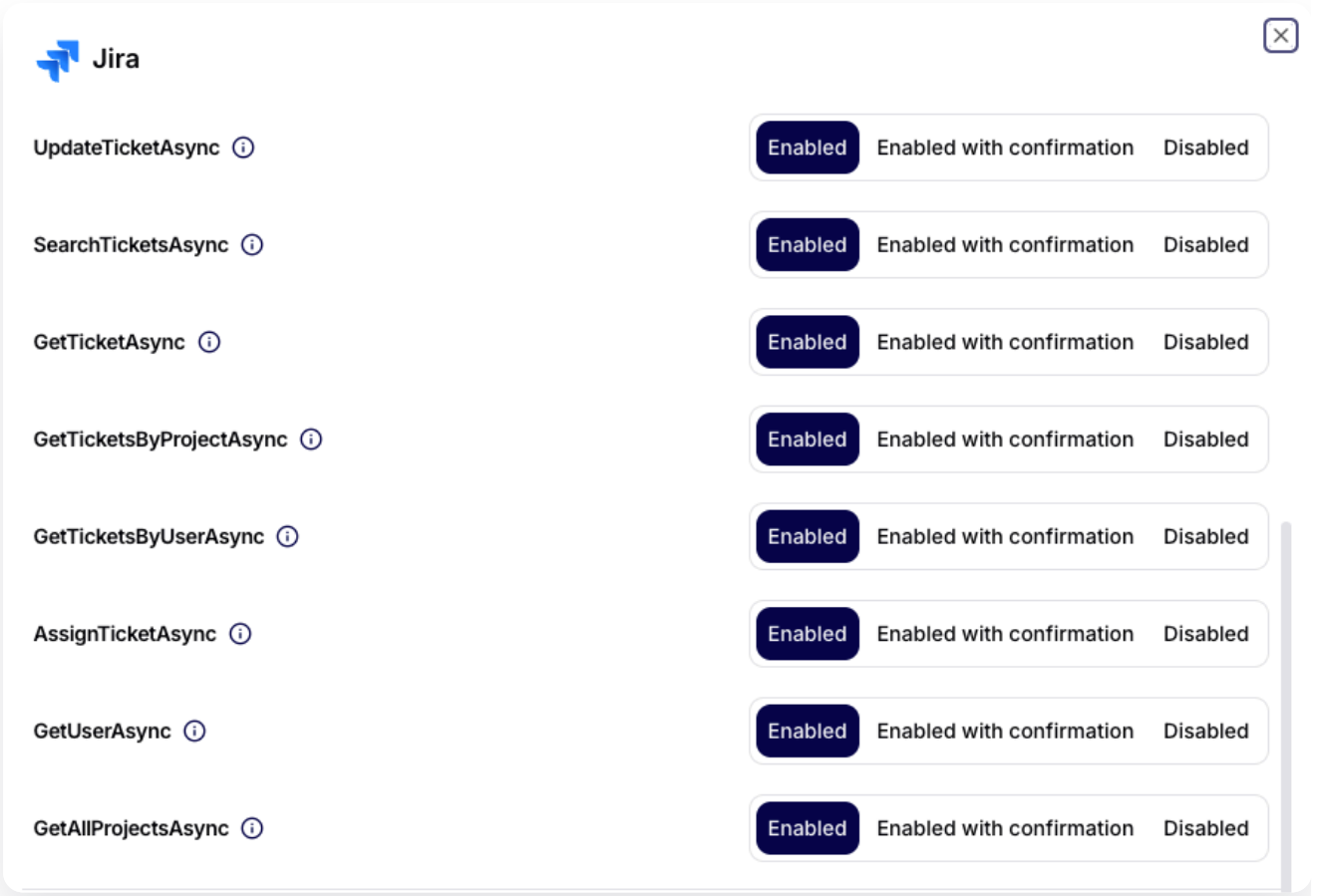
1. In **Connections**, select **Add Connection** and choose **Jira**.



2. Fill in **URL**, **Username**, and **Password** (API token) and set **Shared** or **Private**.



3. Optionally modify which operations are allowed.



24.2 Overview

This Connection provides an API for working with Jira (Atlassian). It allows for creating, reading, searching, updating, and assigning Jira issues (tickets) across projects.

Designed for:

- Incident and ops automation
- Engineering workflows orchestration
- Synchronization of external systems (CRM, monitoring, AI agents)
- Auditable ticket-based processes

24.3 Authentication and Security

- The connection uses the official Jira REST API.
- Authentication is done via an Atlassian account (OAuth / API token).
- The user is identified by the Atlassian Account ID, not by email.
- Permissions are managed directly at the Jira instance level.

If a user does not have permission to see an issue, the Connection will not see it either.

24.4 Basic Terms

- **IssueKey**: Ticket ID (e.g., `PROJ-123`).
- **ProjectKey**: Key of the Jira project (e.g., `PROJ`).
- **AccountId**: Unique identifier of the user in the Atlassian ecosystem.
- **JQL**: Jira Query Language.

24.5 1. Ticket Creation

24.5.1 1.1 CreateTicketAsync

Description

Creates a new Jira issue in the specified project.

Input Parameters

Parameter	Type	Required	Description
projectKey	String	Yes	Key of the Jira project.
issueType	String	Yes	Type of issue (Task, Bug, Story, ...).
summary	String	Yes	Short title of the issue.
description	String	No	Detailed description.
assigneeId	String	No	Atlassian Account ID of the user.

Behavior

- IssueType must exist in the project.
- No fallback or type mapping is performed.
- Incorrect combination = fail.

24.6 2. Ticket Assignment

24.6.1 2.1 AssignTicketAsync

Description

Assigns an existing issue to a specific user.

Input Parameters

Parameter	Type	Required
issueKey	String	Yes
assigneeAccountId	String	Yes

Note Jira ignores emails. Account ID is the only reliable identifier.

24.7 3. Ticket Retrieval

24.7.1 3.1 GetTicketAsync

Description

Returns the details of an issue by `issueKey`.

Input Parameters

Parameter	Type	Required
issueKey	String	Yes

24.7.2 3.2 GetTicketsByProjectAsync

Description

Returns issues belonging to a specific project.

Input Parameters

Parameter	Type	Required
projectKey	String	Yes
maxResults	Int	No

24.7.3 3.3 GetTicketsByUserAsync

Description

Returns issues assigned to a specific user.

Input Parameters

Parameter	Type	Required
assigneeEmail	String	Yes
maxResults	Int	No

24.8 4. Ticket Update

24.8.1 4.1 UpdateTicketAsync

Description

Updates the summary and/or description of an existing issue.

Input Parameters

Parameter	Type	Required
issueKey	String	Yes
summary	String	No
description	String	No

Behavior

- Only provided fields are updated.
- No validation of workflow status is performed.

24.9 5. Search and Query

24.9.1 5.1 SearchTicketsAsync

Description

Searches for issues using a JQL query.

Input Parameters

Parameter	Type	Required
jql	String	Yes
maxResults	Int	No

Example JQL

```
project = PROJ AND status = "To Do"
```

Incorrect JQL returns an immediate error.

24.10 6. Project and User Operations

24.10.1 6.1 GetAllProjectsAsync

Description

Returns a list of projects available to the current user.

Input Parameters

Parameter	Type	Required
maxResults	Int	No

24.10.2 6.2 GetUserAsync

Description

Returns information about a user by Account ID.

Input Parameters

Parameter	Type	Required
accountId	String	Yes

24.11 Design Principles

- Account ID > email (GDPR and Atlassian reality).
- Explicit inputs without assumptions.
- Fail-fast behavior on erroneous requests.
- Respect for Jira workflow rules.

24.12 Summary

The Jira Connection provides direct, secure, and auditable access to Jira issues and projects. It is suitable for automated ticketing, incident agents, engineering productivity tooling, and enterprise workflow integration.

25. Microsoft Outlook (Coming Soon)

We will add the Microsoft Outlook integration to the product documentation later. Here will be the final description of the connection with Siesta AI, authentication options, and an overview of supported operations.

26. Salesforce (Coming Soon)

We will add the Salesforce integration to the product documentation later.
Here will be the final description of the connection with Siesta AI, authentication options, and an overview of supported operations.

27. Slack

We will add the Slack integration to the product documentation later.
Here will be the final description of the connection with Siesta AI, authentication options, and an overview of supported operations.

Připravujeme

28. Profile

The Profile section is used to manage personal information, linked accounts, and access security. The form is divided into two parts: basic information and security.

28.1 Account Information

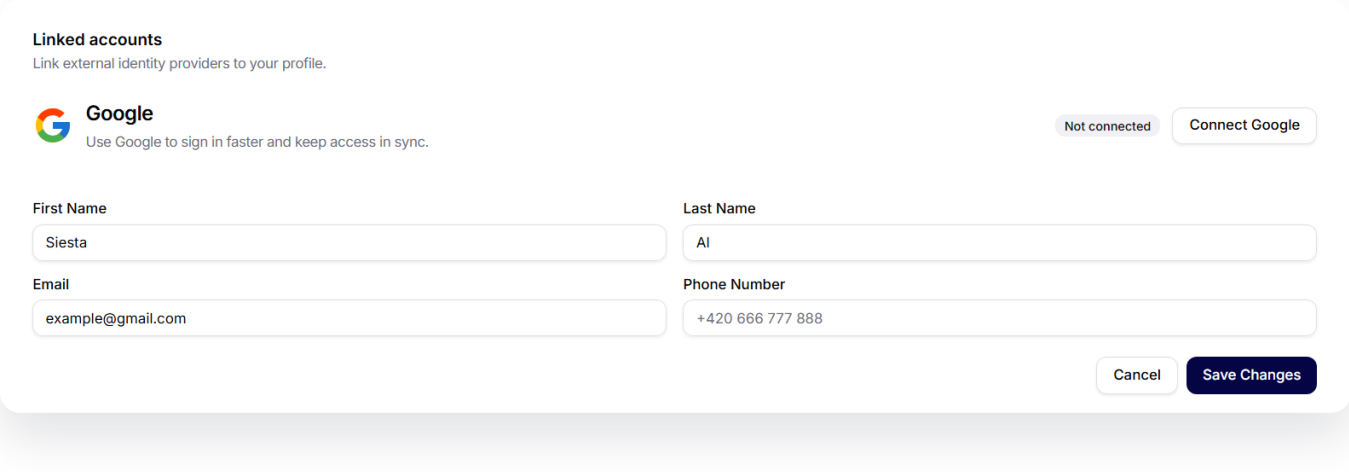
In this section, you can edit basic profile information:

- **First Name** and **Last Name**
- **Email** (login address)
- **Phone Number**

Save changes by clicking the **Save Changes** button. If you want to discard the edits, use **Cancel**.

28.1.1 Linked Accounts

At the top of the page is the **Linked Accounts** section. Here you can see which external logins are connected to the account (e.g., Google). The status is marked with a **Connected** label, and the account can be disconnected by clicking the **Disconnect** button.



28.2 Change Password and Deactivate Account

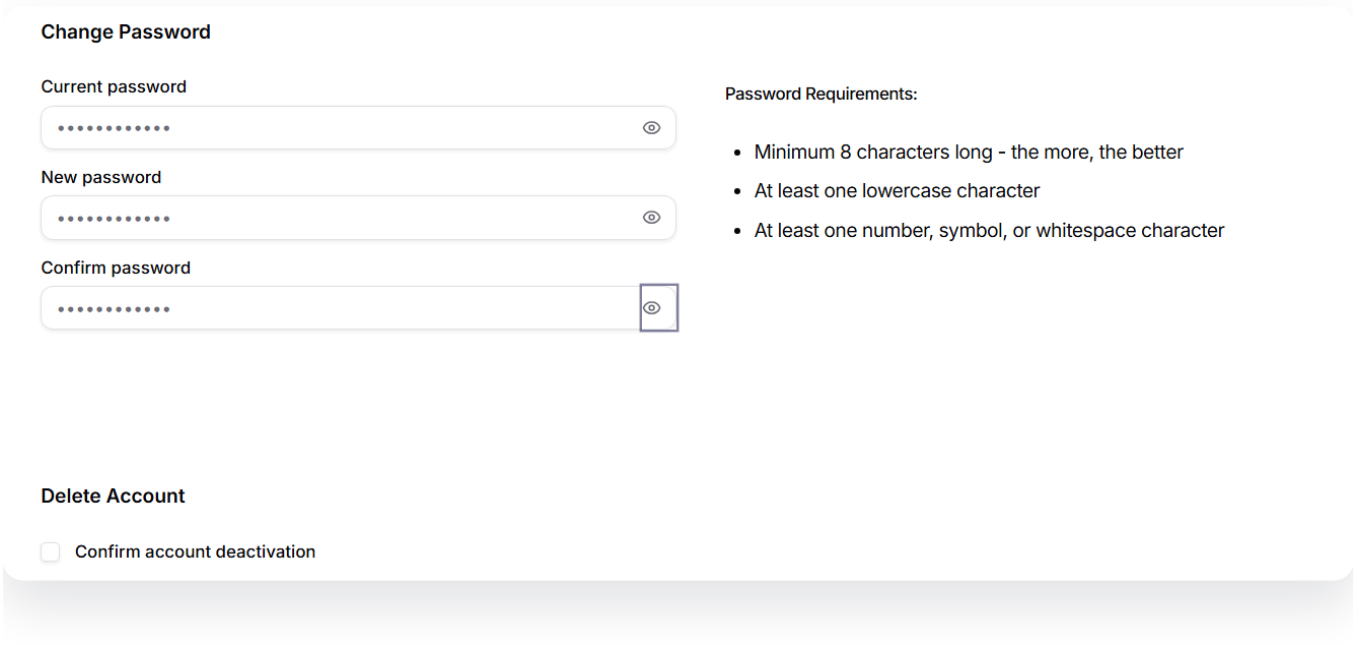
The **Change Password** section requires:

- **Current Password**
- **New Password**
- **Confirm Password**

On the right is an overview of the password requirements that need to be met (e.g., at least 8 characters, at least one lowercase letter, and at least one number/symbol/space).

28.2.1 Delete Account

To deactivate the account, you must check **Confirm Account Deactivation**.



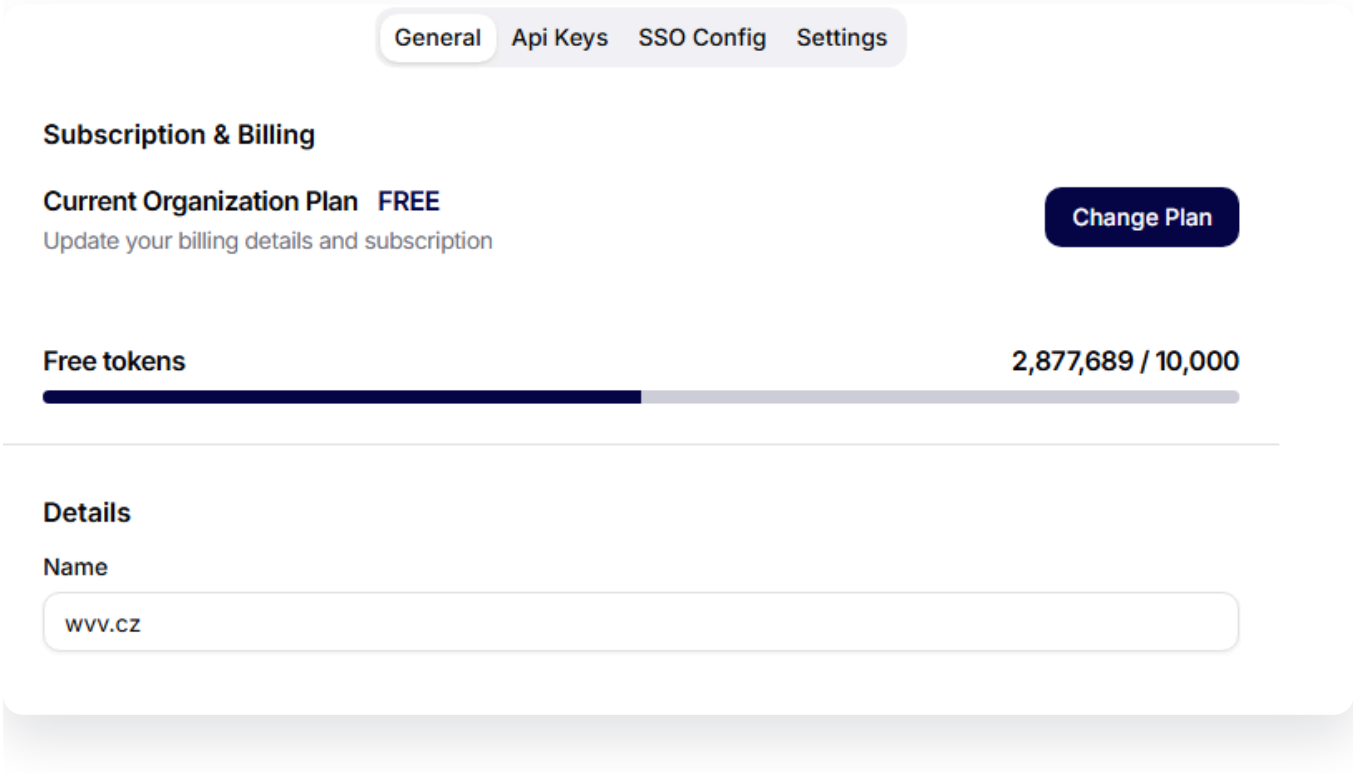
29. Organization

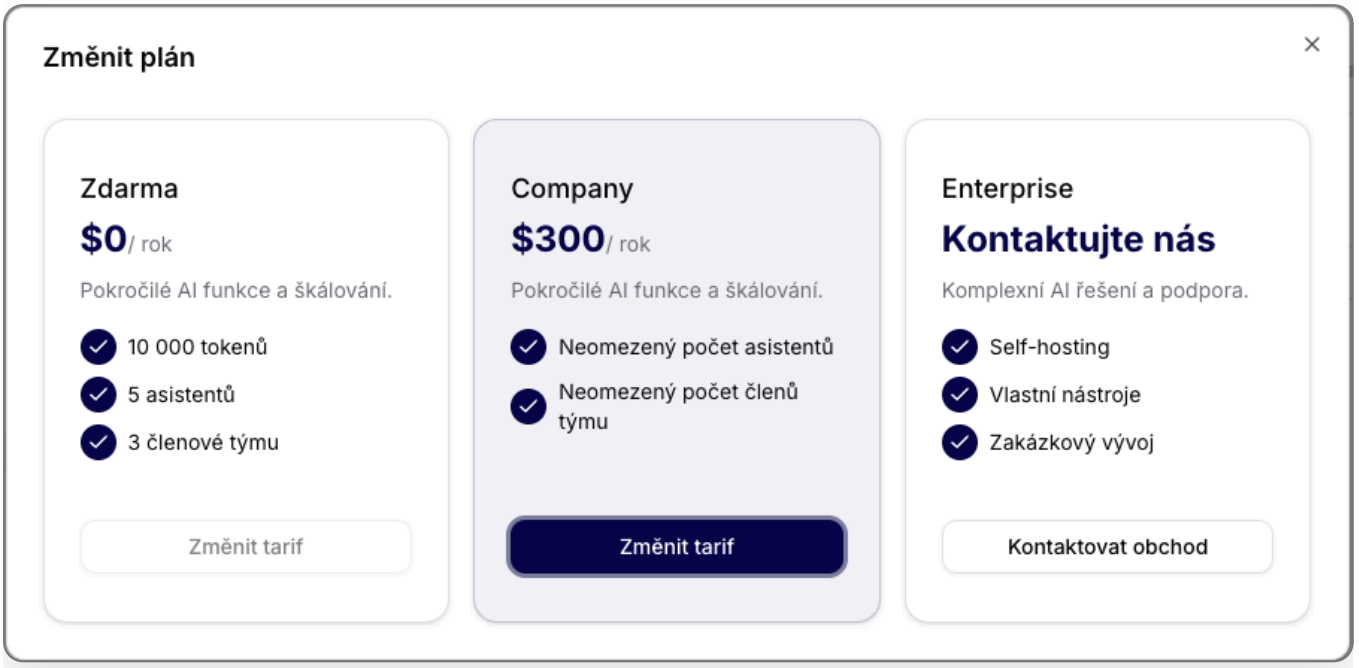
The Organization section is used to manage basic information about the organization, subscription, API access, and billing. Administrators can manage company information, set the platform usage plan, and generate API keys for integration with third-party systems.

29.1 Subscription and Billing

Users manage the organization profile here, see the status of token usage, and have the option to change the subscription type. This page links the basic identity of the company with the licensing and operational settings of the entire platform.

In the **Subscription and Billing** section, you can see the current organization plan, the option to **Change Plan**, and an overview of **Available Tokens**.



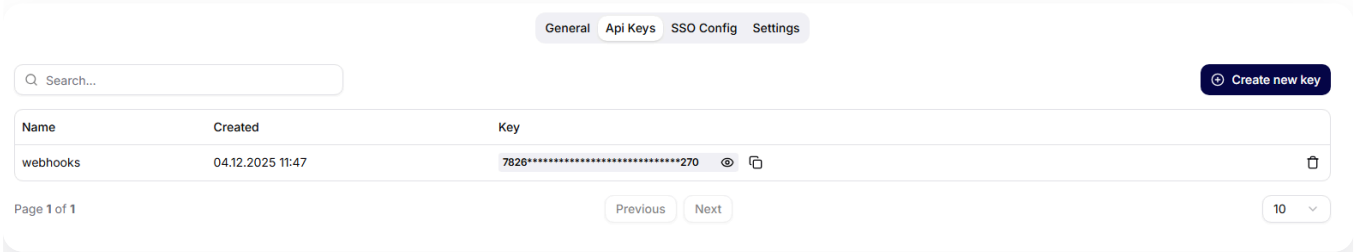


The **Change Plan** dialog offers a comparison of available plans and simplifies the selection of an appropriate package based on usage scope.

29.2 API Keys

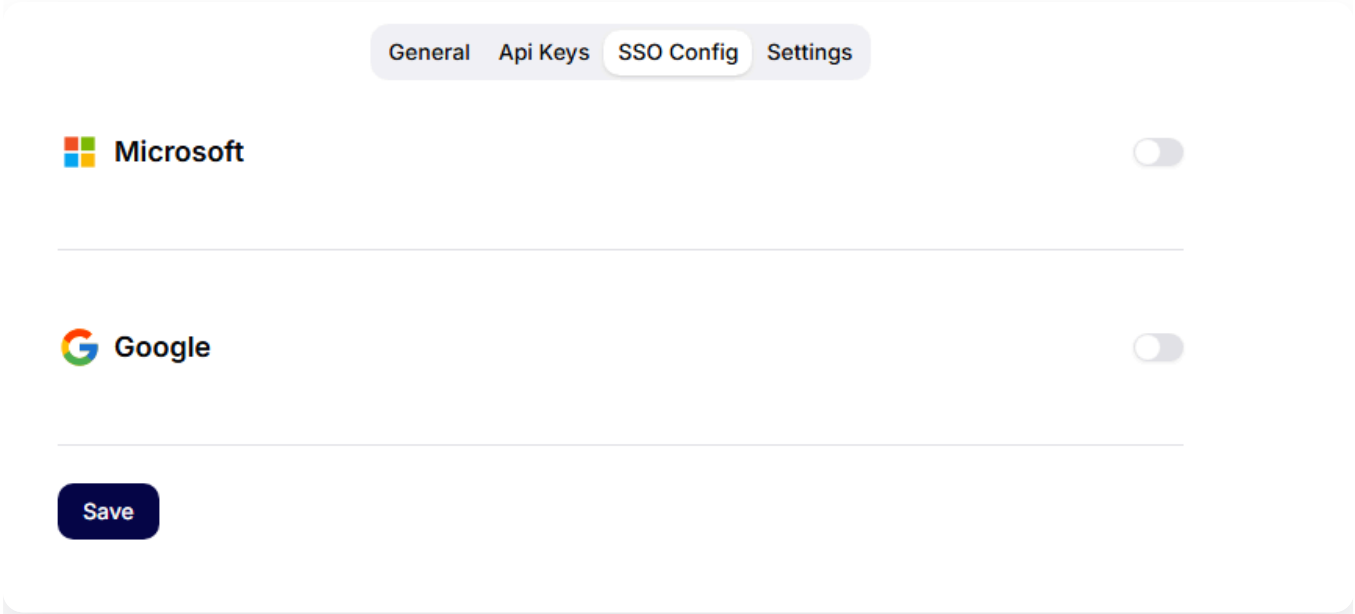
The **API Keys** tab is used for creating and managing keys for integrations.

- The **Create New Key** button opens a dialog for naming the key.
- In the list, you can see **Name**, **Created**, and the masked value of the key.
- The eye icon allows you to view the key, while the copy icon quickly copies the value.
- The trash icon is used to delete the key.



29.3 SSO Settings

In **SSO Settings**, you can enable login via **Microsoft** and **Google** providers. After changing the status of the switches, save the settings by clicking the **Save** button.



29.4 Organization Settings

In this section, you can specify the default AI for transcription and enable uploads. The settings apply to the entire organization and determine what functionalities are available across the platform.

30. Users

This section is for managing users within the Siesta AI platform. Administrators can create new users and assign roles here.

This section is intended for user management within the Siesta AI platform. It is primarily designed for administrators, who can create new users, assign them roles, and set access levels to individual data sources and assistants.

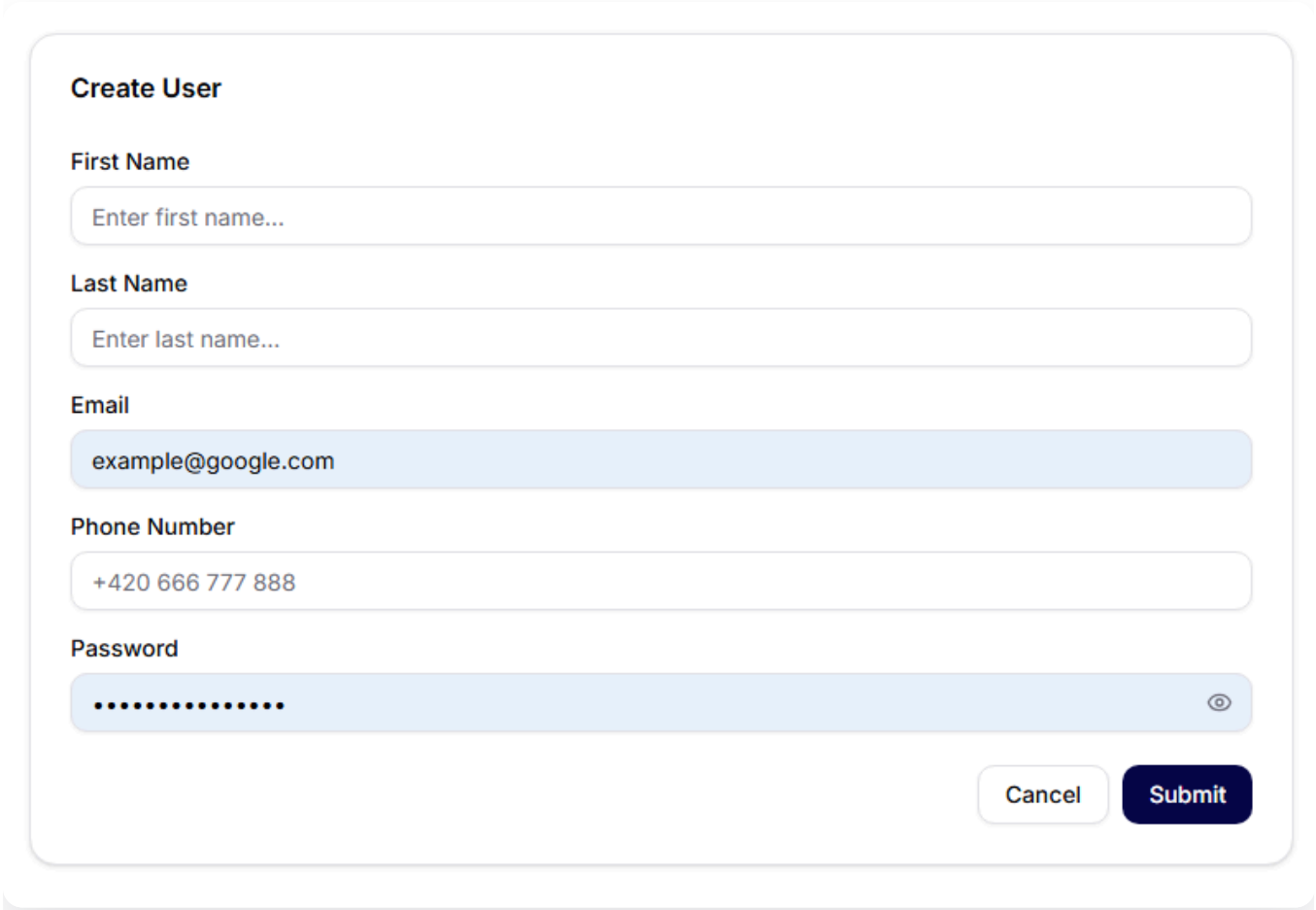
Immediately upon entering the section, you will see a list of all users, including their names, emails, and assigned roles. In the right column, you can view the details of a specific user using the eye icon.

30.1 Creating a New User

After clicking on **Add User**, a form will open where you fill in:

- **First Name**
- **Last Name**
- **Email**
- **Phone Number**
- **Password**

You can confirm by clicking the **Submit** button, or close the dialog by selecting **Cancel**.



30.2 User Roles

In the role assignment dialog, the following options are available:

- **Owner**
- **Administrator**
- **User**


Assign Role
Role
Admin
Owner
Admin
User

31. Teams

31.1 Overview

The Teams tab is used to manage user teams within the organization in Siesta AI. Teams allow for logical grouping of users and managing their access to AI assistants and other application features.

Each team:

- has its own name and description,
- contains specific users,
- determines which assistants team members have access to (assignment of assistants is in preparation).

31.2 Team Overview

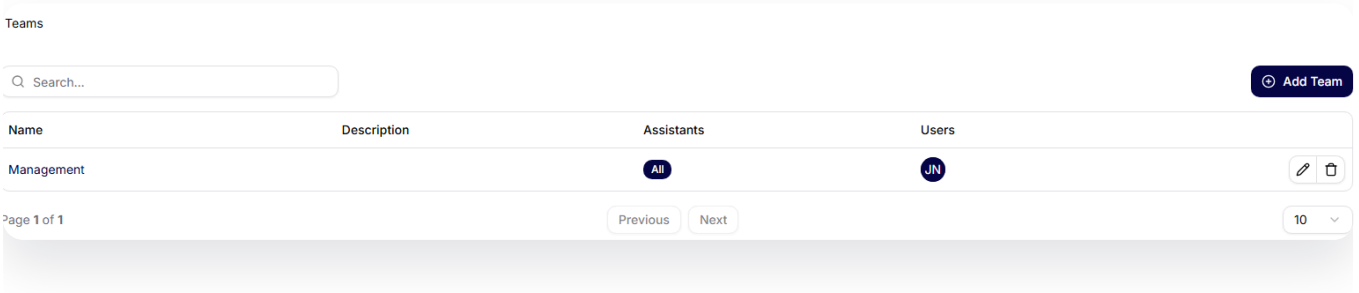
On the main screen of the Teams tab, a list of all created teams is displayed in the form of a table.

Displayed columns:

- **Name** – team name
- **Description** – brief description of the team's purpose
- **Assistants** – indication of which assistants the team has access to (currently only "All")
- **Users** – list or shortcuts of team members
- **Actions** – additional team management options

At the top of the page, the following is available:

- team search,
- **Add Team** button.



31.3 Creating a New Team

Clicking on **Add Team** opens a form for creating a new team.

Form Fields

- **Name** – required field for entering the team name (e.g., Team Fist Alpha).

- **Description** – optional field for a brief description of the team's purpose.
- **Users** – search field for adding users to the team.

Actions

- **Submit** – creates the team and saves its settings.

31.4 Team Detail

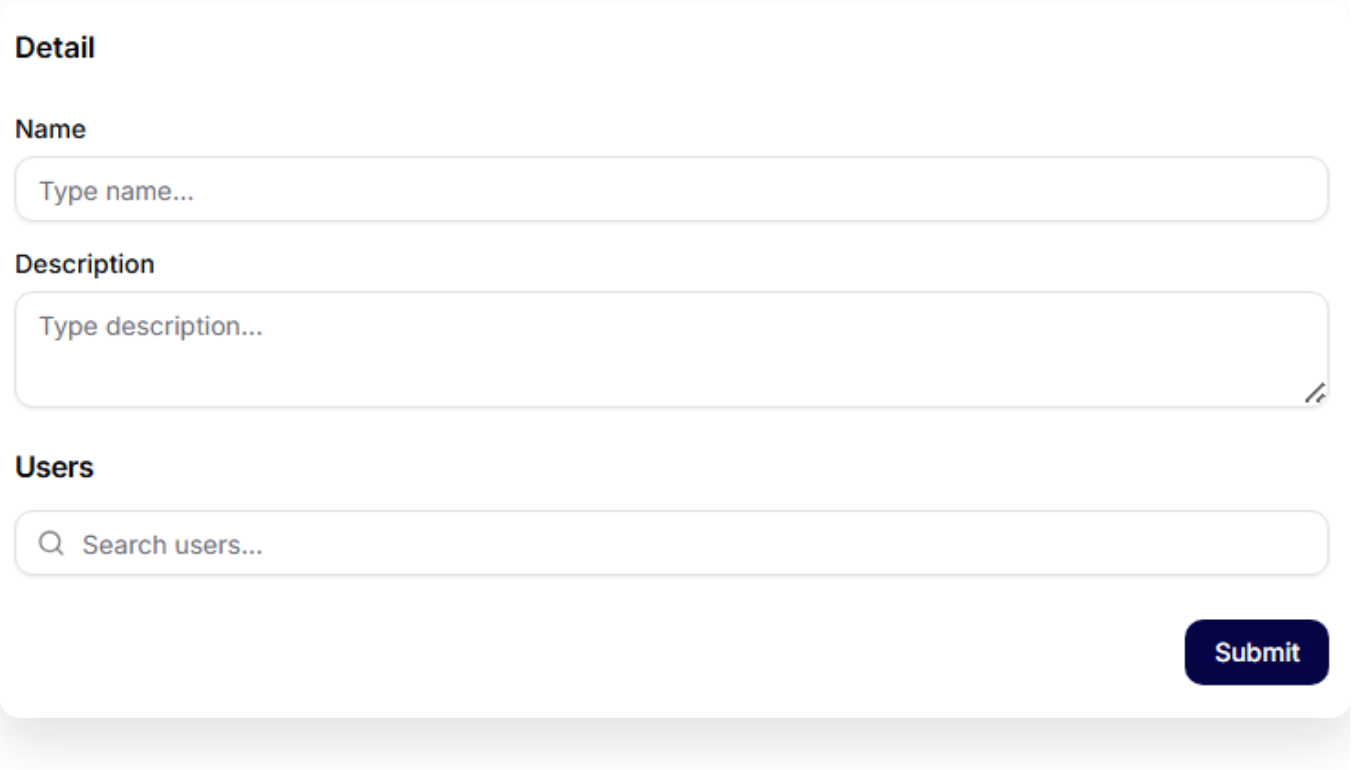
Upon opening a specific team, its detailed page is displayed.

Displayed information:

- team name,
- description,
- list of users who are team members.

From the team detail, it is possible to:

- edit the team name and description,
- add or remove users.



31.5 Access to Assistants

Each team is assigned access to AI assistants either to all assistants or only to selected ones (in preparation).

31.6 Typical Use of Teams

The Teams tab is primarily intended for:

- dividing users by roles or projects,
- managing access to AI assistants,
- simplifying the management of a larger number of users,
- ensuring a clear organizational structure.

31.7 Summary

Teams in Siesta AI provide a fundamental mechanism for organizing users and controlling access to AI functions. Properly configured teams simplify application management and enhance both security and clarity of work.

32. Audit log

The Audit log tab serves to closely monitor all important actions taken within the Siesta AI application. The audit log provides a complete history of changes, which is crucial for traceability, security, auditing, and incident resolution.

Each entry in the audit log corresponds to a specific action performed by a user or the system.

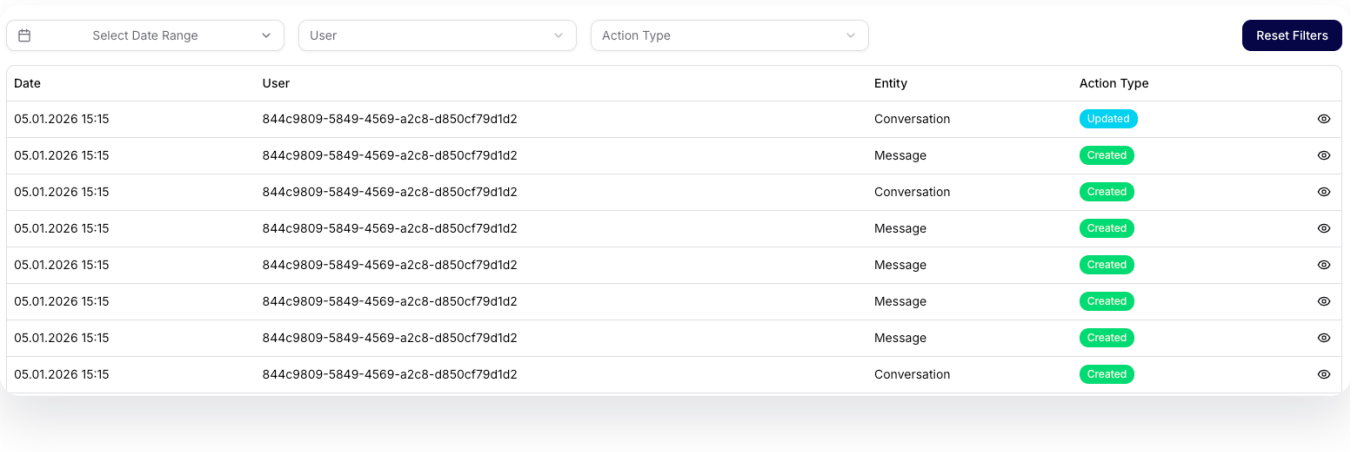
32.1 Overview of Records

The main screen of the audit log displays a list of events in a table.

Displayed columns:

- **Date** – exact date and time of the action performed
- **User** – identification of the user who performed the action
- **Entity** – type of object on which the action was performed (e.g., Conversation, Message, User, Team, DataSource, ApiKey)
- **Action Type** – type of operation performed, e.g., **Created** or **Updated**
- **Detail** – eye icon to open the details of the record

The audit log is paginated to handle very large amounts of records.



32.2 Filtering Records

The audit log allows filtering events to quickly find relevant records.

Available filters:

- **Date Range (Select Date Range)** – limit records to a specific period
- **User (User)** – actions performed by a specific user
- **Action Type (Action Type)** – filtering by type of operation (e.g., only Created or Updated)
- **Reset Filters** – clear all filters

Filters can be combined for more precise results.

32.3 Detail of Audit Record

Clicking on the detail of a specific record displays detailed information about the action.

Details include:

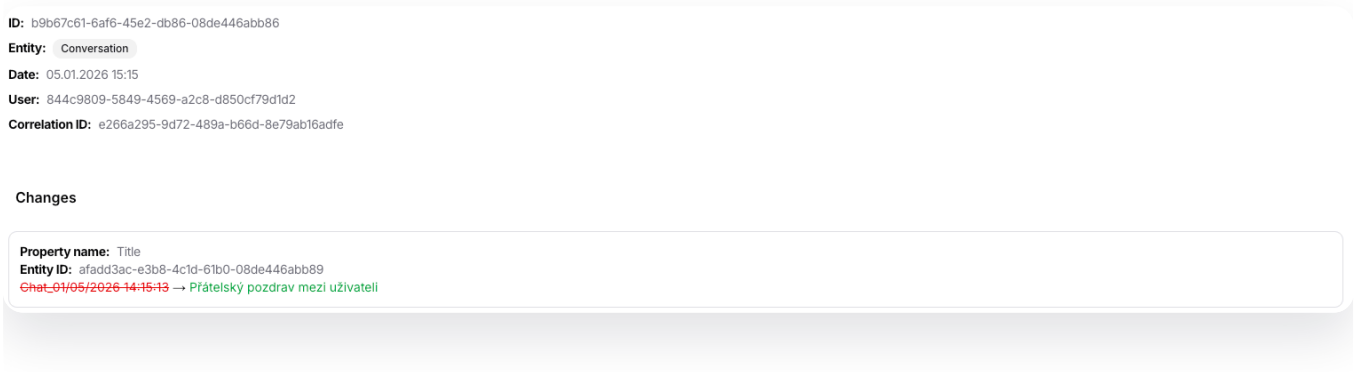
- **Record ID** – unique identifier of the audit event
- **Entity** – type of object affected by the change
- **Date** – time the action was performed
- **User** – identity of the user who performed the action
- **Correlation ID** – identifier that allows tracking related actions across the system

32.3.1 Changes

Displays specific changes that were made, such as:

- the name of the property that changed,
- the original value,
- the new value after the change.

This section allows for precise tracking of what changed and how.



32.4 Typical Uses of the Audit Log

The audit log is primarily intended for:

- security and compliance purposes,
- tracing the history of changes,
- analyzing user behavior,
- resolving incidents and errors,
- internal and external audits.

32.5 Summary

The audit log in Siesta AI provides a transparent and detailed overview of all important actions in the system. Thanks to filters and record details, it enables quick identification of when, by whom, and how a specific change was made.

33. Webhooks

The Webhooks tab is used to manage webhooks that allow Siesta AI to connect with external systems and applications. A webhook provides a unique URL address to which an external service can send HTTP requests, thereby triggering or influencing the behavior of the system.

Each webhook:

- has its own name,
- is tied to a specific API key,
- can be active or inactive,
- has a unique URL address.

Webhooks are often used as triggers for Workflows that can be initiated from external systems.

33.1 Overview of Webhooks

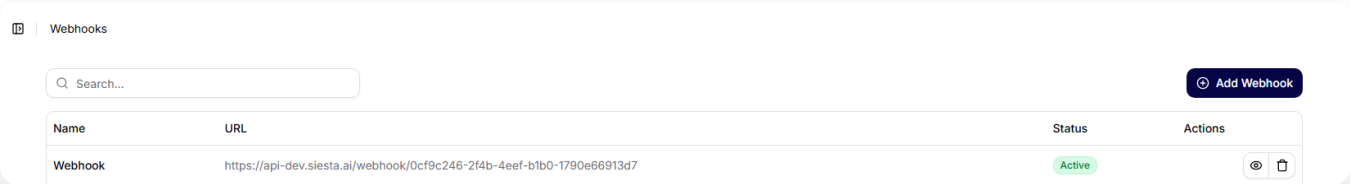
On the main screen of the Webhooks tab, a list of all created webhooks is displayed in a table.

Displayed columns:

- **Name** – the name of the webhook entered by the user
- **URL** – automatically generated URL address of the webhook
- **Status** – current status of the webhook (Active / Inactive)
- **Actions** – additional management options for the webhook (e.g., edit)

At the top of the page, the following is available:

- webhook search,
- **Add Webhook** button.



33.2 Creating a New Webhook

Clicking on **Add Webhook** opens a dialog for creating a new webhook.

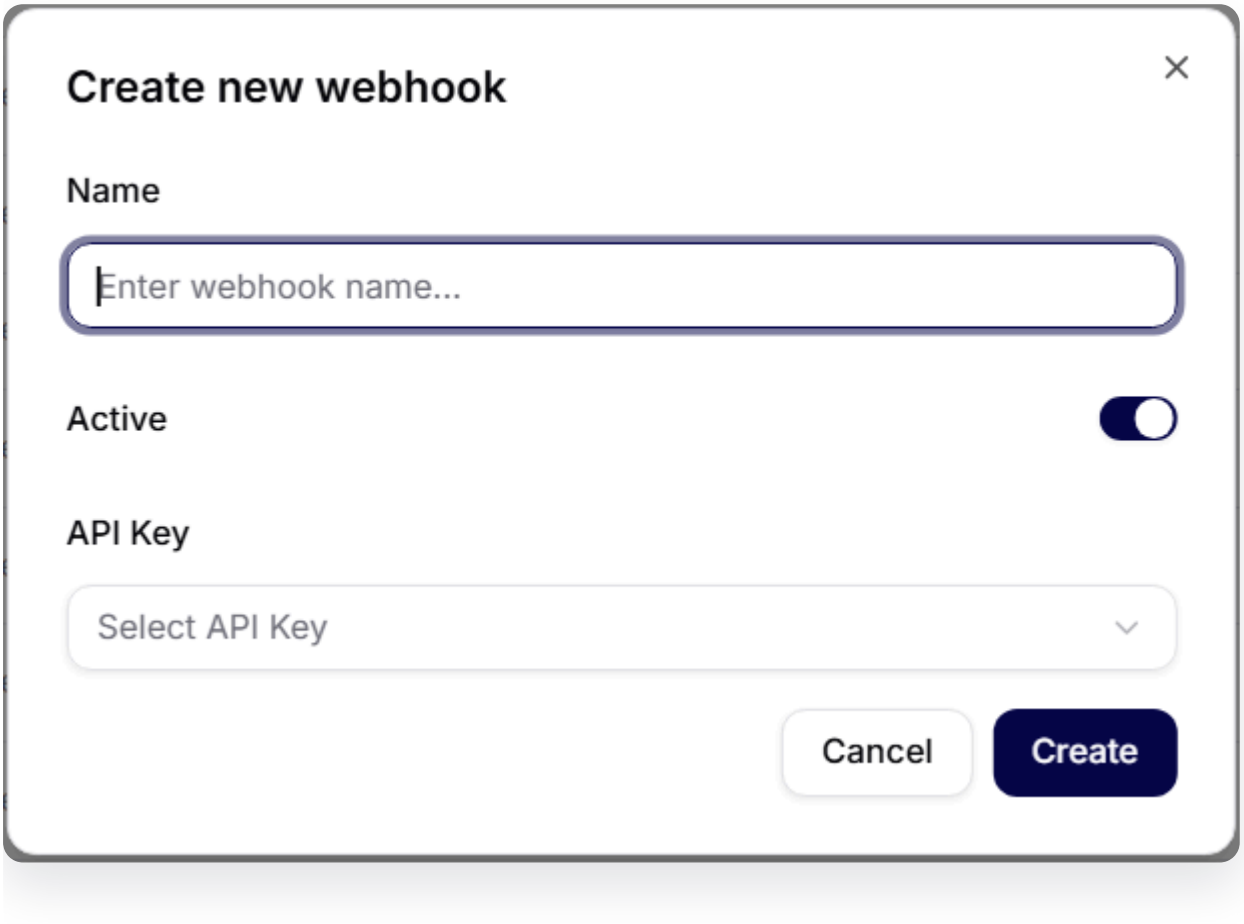
Form Fields

- **Name** – a required field for entering the name of the webhook (e.g., Webhook)

- **Active** – a switch that allows the webhook to be activated or left inactive upon creation
- **API Key** – selection of the API key that will authorize the webhook (e.g., API key for my python script)

Actions

- **Cancel** – closes the dialog without creating the webhook
- **Create** – creates a new webhook and generates its URL



33.3 Webhook Details

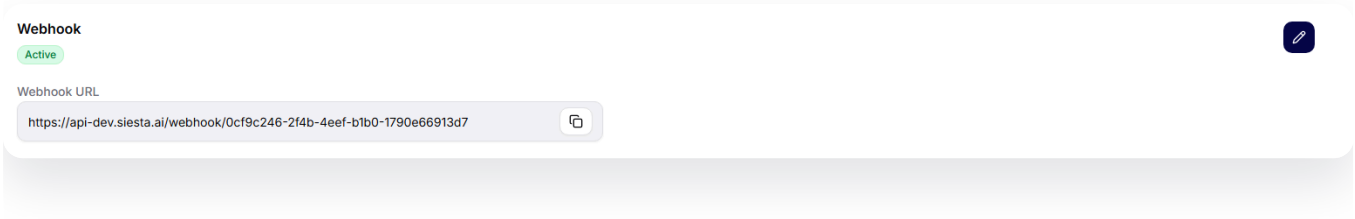
After opening a specific webhook, its detailed page is displayed.

Displayed information:

- name of the webhook,
- status (active / inactive),
- URL of the webhook – a unique address that can be copied with one click.

From the webhook details, it is possible to:

- modify the webhook settings,
- change its active status,
- use the webhook URL in external applications or in Workflows.



33.4 Webhook Status

The status of the webhook determines whether the webhook is ready to receive requests:

- **Active** – the webhook is turned on and available
- **Inactive** – the webhook is turned off and requests are not processed

The status is visible both in the overview of webhooks and in the details of the webhook.

33.5 Typical Uses of Webhooks

Webhooks are primarily used for:

- integrating Siesta AI with external applications,
- triggering automated processes,
- connecting custom scripts (e.g., Python),
- transferring data between systems in real time.

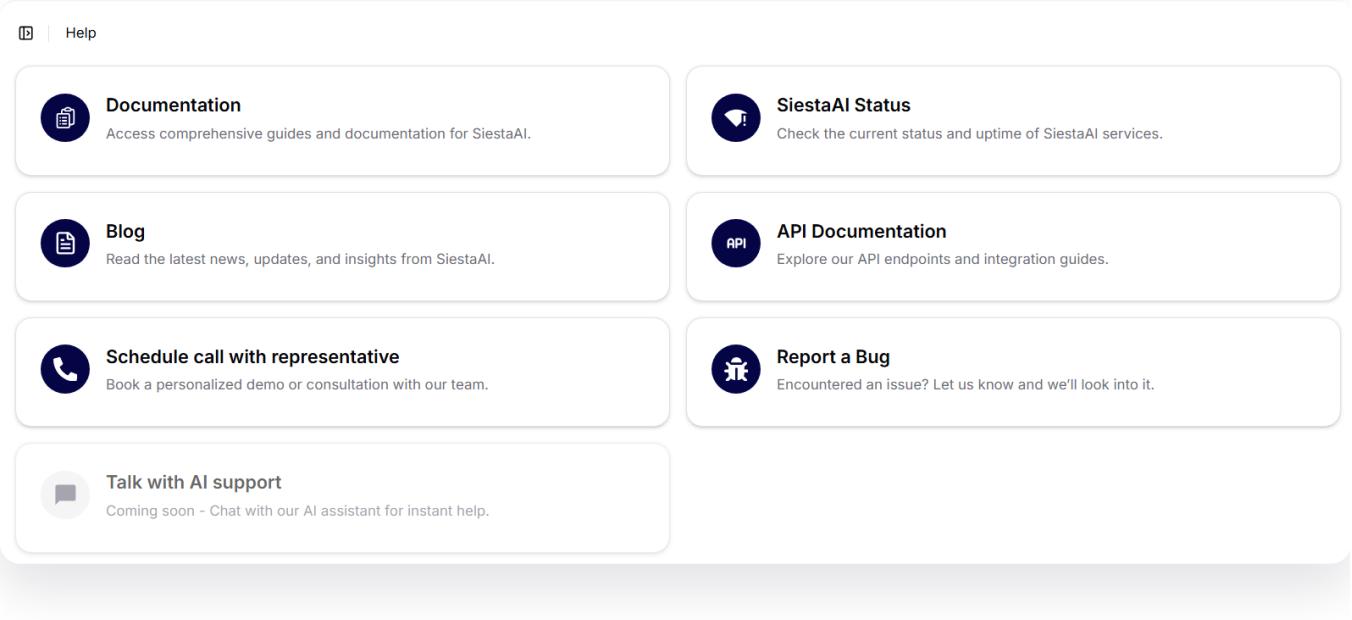
33.6 Summary

Webhooks in Siesta AI provide a simple and secure way to connect the platform with external systems. Thanks to clear management, linkage to API keys, and the ability to activate or deactivate, webhooks can be easily monitored and managed. If you use a webhook as a trigger, we recommend linking its use to Workflows.

34. Help

The Help tab serves as a central hub for support and information within the Siesta AI application. Here, users will find direct access to documentation, API guides, system status information, a bug reporting form, and contact options for the Siesta AI team.

The help center acts as a central portal for all user support and documentation. It contains links to the official user documentation as well as detailed reference materials for the API, a collection of blog posts, and the current service status. Additionally, it provides quick access to a live chat interface for contacting the support team and the option to schedule calls or video conferences with specialists. All resources are grouped together so that users have one place for self-study and immediate resolution of inquiries.



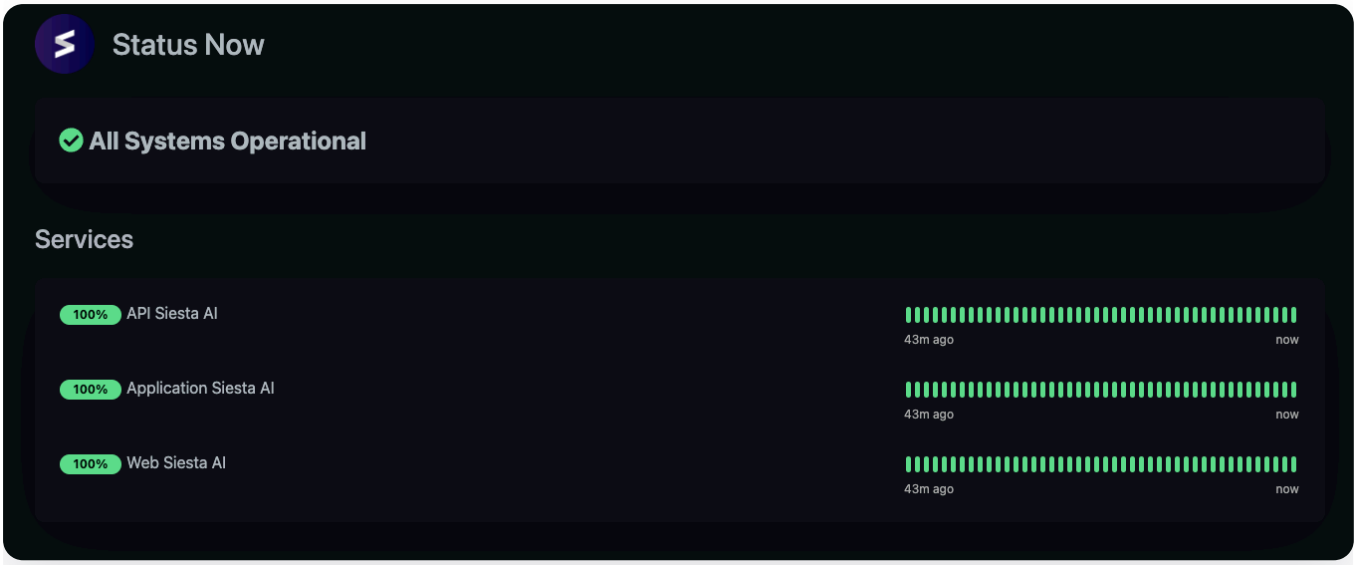
34.1 Available Items

34.1.1 Documentation

Provides access to comprehensive guides and official documentation for the Siesta AI platform. Upon clicking, users are redirected to the documentation portal, where they can find user manuals, detailed feature descriptions, and recommended practices for working with the system.

34.1.2 SiestaAI Status

Used to check the current status and availability of Siesta AI services. The link leads to a public status page where the overall system status, availability of individual services (API, application, web), and their operational history are displayed.



34.1.3 Blog

Contains the latest news, product updates, and announcements regarding the development of the Siesta AI platform. Upon opening, users are redirected to the official Siesta AI blog.

34.1.4 API Documentation

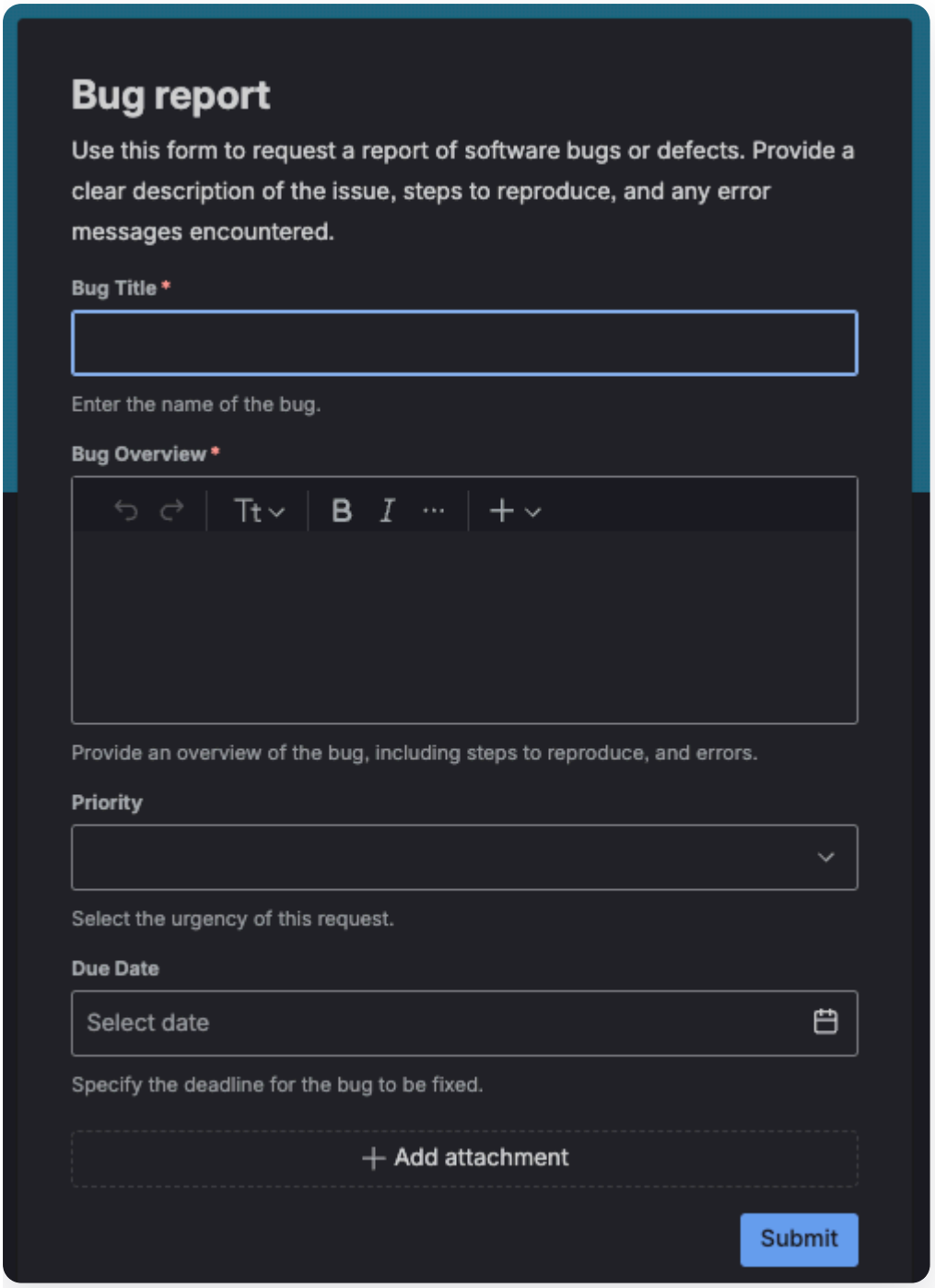
Intended for developers and technical users who integrate Siesta AI using the API. The link leads to the API documentation with an overview of endpoints, authentication descriptions, and integration guides.

34.1.5 Schedule a Call with a Representative

Allows users to book a demo or consultation with the Siesta AI team. Upon clicking, a booking page opens for arranging a call with a sales or technical representative.

34.1.6 Report a Bug

Used to report bugs, technical issues, or unexpected application behavior. The link opens a bug report form where users can enter the bug title, detailed problem description, steps to reproduce, priority, deadline, and attach files.



34.1.7 Talk to AI Support

An upcoming feature that will allow direct communication with an AI assistant for immediate support in the future. It is not yet available in the current version of the application.

34.2 Typical Use

The Help tab is primarily used for quick access to information, verifying service availability, resolving technical issues, working with the API, and contacting support or sales teams.

34.3 Summary

The Help section in Siesta AI functions as a central point for support and orientation within the system. It allows users to quickly access the right source of information without the need to leave the application.

35. Conclusion

This product specification serves as an introductory presentation of the Siesta AI platform and its core functionalities. Its aim is to help clients and partners gain a more detailed understanding of what the platform offers, how it works, and how it can be utilized in various business scenarios.

The described features form the foundation of a system that can be further expanded according to the specific needs of the client, whether it involves integrations, security requirements, or advanced modules.

If you are interested in expanding functionality, technical consultation, or collaboration, please do not hesitate to contact us at **info@siesta.ai** or at the phone number **+420 777 273 391**.

36. Release Notes

Here you will find an overview of all releases and their main changes. Each release has its own detailed entry that serves as a development history. New notes are continuously added to keep the overview complete and up to date. This serves as a central place for tracking progress and key improvements. If you are looking for details, open a specific entry and review the changes by version.

37. Release 1.1.12

Release Date: 3.12.2025
Release Type: Minor Release

37.1 Summary

This release brings significant improvements to chat, new features for assistants, a completely new way to share conversations, and fixes in the areas of login, user permissions, and bug reporting.

37.2 New Features

37.2.1 Login

- Linking profile with Google account: Users who originally registered using email and password can now additionally link their account with a Google account for faster login and easier identity management.

37.2.2 Chat

- Sharing via link: Users can now share conversations through a secure link.
- Public sharing: Anyone with the link can view the conversation, even without an account in Siesta AI.
- Internal sharing: Access is limited to logged-in users from the same organization.
- Copy Message: Users can copy any message in the conversation with a single click.
- Stop button in chat: A Stop button has been added to the chat, which immediately ends the ongoing action or response generation.
- Improved start of a new conversation: The unnecessary popup has been removed, the new chat window opens immediately, and the cursor is automatically set to the input field.
- Searching for assistants when starting a new conversation: When creating a new conversation, it is now possible to directly search for an assistant.

37.2.3 Assistants

- Automatic creation of "General Assistant": Every newly created organization now automatically receives a default assistant.

37.3 Improvements

37.3.1 Chat

- Enhanced feedback system: The modified and clarified rating of individual responses allows for faster and more convenient feedback. This also includes improvements to the Evolution section, which now displays the assistant's development history and learning process more clearly.

37.3.2 Localization

- Missing or inconsistent translations have been fixed in various places within the application.

37.3.3 Help

- New unified bug reporting form: A central form for reporting bugs has been created, available here: https://siestalabs.atlassian.net/jira/software/c/form/1fe30bb7-3755-4f34-95ae-5d93f716546b. The form is also accessible in the Help section to make reporting faster and clearer.

38. Release 1.2.0

Release Date: January 26, 2025
Release Type: Minor Release

38.1 Summary

This release brings improvements focused on faster and clearer work with the platform, better management of assistants and connections, and a significant expansion of chat capabilities and handling of recordings.

38.2 Enhancements

38.2.1 User Interface and UX

- A clearer user interface and navigation across the platform.
- Improved consistency and readability of the user interface.

38.2.2 Users and Permissions

- Simplified management of users and their roles.
- Fixes related to permissions and visibility across the system.

38.2.3 Chat and File Handling

- Inserting files into chat using Ctrl+V.
- Enhanced feedback handling in chat — feedback (thumbs up / down) can be removed or changed by repeated clicking, without the need to delete the entire conversation.
- Centralized feedback overview — all feedback related to conversations is now available in the assistant detail in a separate section.

38.2.4 Deletion and Error States

- Improved logic for deleting assistants and connections.
- More precise and understandable error messages during login.

38.2.5 Public Chat

- The Public Chat Widget is now fully configurable: appearance adjustments, chat behavior settings, file handling, enabling or disabling selected features directly from administration.

38.2.6 Connections

- Improved management of connections: expanded with filters (All / LLM Models / Tools), clearer sharing, better handling of types and visibility.

38.2.7 Assistants

- Enhanced overview of assistants with key information (icon, name, model, number of conversations, feedback status, status).
- Ability to configure the behavior and reasoning of AI assistants.
- Better management of access and sharing of assistants (organizational, shared, private).
- Clearer visibility of assistants according to team and user permissions.
- Added a new chat model GPT-5.2 available in assistant configuration.

38.3 New Features

38.3.1 Workflows (beta)

- Workflows are available in beta version.
- Functionality is available upon consultation with the development team.

38.3.2 Webhooks

- New webhook functionality: simple editing, ability to call external URLs, sending payloads, ability to trigger workflows.

38.3.3 Recordings

- New Recordings section with an overview table and detailed view of recordings.
- Ability to record and manage recordings including support for uploads via API.
- Automatic AI transcription of recordings with an overview of status and resulting transcript.
- Setting the default AI connection for transcriptions at the organizational level.

39. Release 1.2.1

Release Date: 5.2.2025
Release Type: Patch Release

39.1 Summary

This patch focuses on technical fixes and stabilization of the backend without impacting existing functionality for users.

39.2 Improvements

39.2.1 General Enhancements

- Stabilization of the backend and technical fixes without changes in user behavior.

39.2.2 Azure AI Foundry

- Integration with Azure AI Foundry is now fully functional.
- Assistants and workflows can run on the Azure AI Foundry backend without restrictions.

39.2.3 Private Assistants

- Access to assistants is controlled through **Access**: **Organization**, **Shared**, and **Private**.
- **Private** = the assistant is visible and available only to the author; ideal for testing, prototypes, and personal tools.
- **Shared** = the assistant is made available to selected users or teams; allows targeted sharing without opening it to the entire organization.
- **Organization** = the assistant is available to everyone in the organization; suitable for production assistants and standardized use cases.

39.2.4 Default Recording Settings in Organization

- In the **Organization**, a default AI for transcription can be set and recording can be enabled/disabled.
- This setting applies across the entire platform and unifies the behavior of working with recordings for the whole organization.

40. Troubleshooting

40.1 Support Hub

Where to go for help with Siesta AI.

Support Portal	Error Codes
Support portal for contacting support and the right context.	Error codes for identifying the issue and necessary details.

40.2 Support portal

When you need help, create a request and attach:

- A screenshot of the KPI tile or graph.
- Time data and the assistant's name.
- Expected vs. actual result.

You can find details in the Help section.

40.3 Error codes

If an error code appears, save the entire message and context:

- The exact code and error text.
- Where the error occurred (page, action).
- Whether the issue persists after verifying access to data sources.

41. User Manual

In this section, you will find detailed manuals for the individual components of Siesta AI. We mainly focus on guides on how to connect services and data sources and how to use them in real user scenarios on the platform.

Practical procedures are written step by step, similar to the example of connecting Gmail to Siesta AI. We will continuously add more guides for Connecting and other parts of the platform.

Generated: February 09, 2026