Clarity OpenRAILS™ Documentation

CLARITY

CLARITY OPENRAILS AI OVERVIEW

The Clarity OpenRAILs AI platform represents a new era in enterprise AI adoption by making powerful, agentic AI accessible, secure, and customizable for businesses of all sizes. Many organizations hesitate to implement AI due to perceived complexity, high costs, or fears of workforce disruption. Clarity OpenRAILs addresses these challenges by offering an affordable, easy to deploy, enterprise-grade AI model designed for rapid integration with existing business processes. It empowers companies to automate operations, enhance decision-making, and create tailored AI solutions—ranging from chatbots and content generation to analytics and workflow automation—without the traditional barriers of cost and expertise.

At its core, OpenRAILs operates on a **modular, agentic architecture** built around intelligent AI agents that plan, execute, coordinate, and learn from tasks autonomously. The platform provides for the creation and customization of numerous agents. This allows for increased performance, security, and configurability. This structure allows OpenRAILs to grow, adapt, and scale with a business—improving continuously while maintaining security and operational efficiency. Its Retrieval-Augmented Generation (RAG) and integration capabilities ensure responses, and analytics are based on real, organization-specific data, not just general model training.

The platform's design supports a wide variety of use cases, including **AI-powered customer service**, **marketing automation**, **personalized search and recommendations**, **data-driven business intelligence, and hundreds of other business cases**. Businesses adopting OpenRAILs report increased revenue, reduced operational costs, faster response times, and higher customer satisfaction. By enabling executives to deploy and manage AI projects through an intuitive interface—complete with fine-tuning (EVAL), de-identification governance, and scalable hosting—Clarity OpenRAILs makes enterprise AI both approachable and sustainable.

In essence, a **robust, agentic AI platform like Clarity OpenRAILs** transforms AI from a technical challenge into a strategic advantage. It allows companies to start small, scale intelligently, and retain full control over their data and model performance. With built-in automation, compliance, and adaptability, OpenRAILs empowers organizations to stay ahead of competitors, reduce inefficiencies, and harness the full potential of AI innovation across every department.

Documentation Purpose

Clarity OpenRAILs, as an AI platform, is highly customizable and can easily be tailored to fit many businesses' needs, both small and large. The AI industry and available AI models are constantly changing, so an important aspect of the platform's design, is the ability to be constantly upgraded and expanded to benefit from and use updates and enhancements to improvements in AI models and technologies.

With the industry changing daily, there is no way to fully document all the different AI models, capabilities, and customizations available within the platform. However, this document will cover most of the out-of-the-box capabilities and basic customizations of the platform and try to clearly articulate other features and capabilities that are either optional or require additional development time or cost to configure or customize.

Glossary

AGENTS – An AI Agent is a system or program that can perceive its environment, make decisions, and take actions autonomously to achieve specific goals. It uses data inputs (like text, images, or sensor data) to analyze situations, plan responses, and learn from outcomes over time. In practical terms, AI agents can handle tasks such as scheduling, customer support, or system monitoring—acting as digital "assistants" that think and act on behalf of users or systems.

Clarity's OpenRAILs Agentic Architecture uses agents to assign specialty tasks, scale performance and silo access to data. Like a bank, you've got "teller" agents, "loan" agents, and maybe a "supervisor" agent that has vault access. The same design methodology holds for a properly designed AI project infrastructure, separating and assigning agents for different roles, intent, and data access.

AGENTIC ARCHITECTURE – Agentic architecture refers to the design framework that enables AI agents to operate autonomously, make decisions, and collaborate with other agents or systems. It defines how perception, reasoning, planning, and action components interact within an AI system. This architecture allows agents to adapt to changing environments, coordinate tasks, and pursue goals without constant human input—forming the foundation for multi-agent systems and complex autonomous workflows.

Within OpenRAILs, the agentic architecture uses agents to silo specialty functionality, as well as data access. Agents can be trained for specific tasks, so an intake chat agent make take in a request, hand it off to a planning or supervisor agent, which may then hand off the request to a specialized agent for generating a report, or analyzing data, or reviewing documentation, etc.

AI MODEL – An AI model is a mathematical system trained on data to recognize patterns, make predictions, or generate new content. It learns relationships within the data during training and then applies that knowledge to new inputs. Examples include models that can classify images, translate languages, or power chatbots—essentially serving as the "brain" behind intelligent systems.

AI models are commonly referred to as LLMs (large language models). This term refers to the different models that can be used by the OpenRAILs platform. These include, but are not limited to, Open AI, Qwen, Deepseek, Llama, Mistral, etc. OpenRAILs can use many different models, including open-

source models with no ongoing costs associated with your installation. This gives you flexibility of LLM specialty, as well as data ownership, hosting, security, and controlling costs.

API ENDPOINTS – API endpoints are specific URLs or access points within an Application Programming Interface (API) where software systems send requests to exchange data. Each endpoint represents a defined function or resource—such as retrieving user information or submitting a payment. They allow different applications to communicate securely and efficiently, forming the backbone of modern system integrations and automation.

While OpenRAILs includes the Admin UI to create and manage chatbots, if you want an external application to access OpenRAILs for AI functionality, then you access the OpenRAILs API Endpoints, which expose secure access to functionality that OpenRAILs can perform.

API KEYS – API keys are unique codes used to authenticate and authorize access to an API. They act like digital ID cards, verifying that a user or application has permission to make requests or retrieve data from a system. By controlling and tracking usage, API keys help maintain security, prevent abuse, and monitor how external apps interact with a service. OpenRAILs allows you to generate a unique API key for each entity that you want to give access to the platform.

CHATBOT - An AI chatbot is a conversational program that uses artificial intelligence to simulate human-like dialogue through text or voice. It understands user input, interprets intent, and generates relevant responses using natural language processing (NLP). Chatbots are commonly used for customer support, sales, and information retrieval—offering instant, automated interactions that improve efficiency and user experience.

OpenRAILs provides an Admin UI that allows you to create, customize and deploy any number of Chatbots, each providing possibly different capabilities, and retrieving data from different data sources.

CHATBOT TEMPLATE - A chatbot template is a reusable blueprint that defines a bot's structure—intents, conversation flows, prompts, fallback logic, colors, icons, and more — so you can launch new bots quickly. It standardizes best practices (greetings, error handling, handoff to human) while allowing customization of content, branding, and data connections. Teams use templates to ensure consistency across use cases (e.g., support, FAQs, lead capture) and to reduce development time and errors.

OpenRAILs ships with some pre-defined Chatbot templates that you can either use as-is or customize yourself. The UI also allows you to create a new or customize an existing template, then save it as a new chatbot template for you to use later.

CLASSIFICATION - Data classification is the process of labeling documents or data assets based on their sensitivity or security level. In an AI platform, this determines who can view, edit, or share specific information by assigning access levels to user roles. By categorizing data (e.g., public, internal, confidential), organizations can enforce security policies, protect sensitive information, and ensure compliance with privacy and governance standards.

CLEANSING / DATA CLEANSING - Data cleansing is the process of preparing and securing data by detecting, correcting, or removing sensitive, inaccurate, or inconsistent information. In an AI platform, this often includes de-identifying protected data—such as replacing Social Security numbers, account IDs, or patient names with tokens or placeholders. This ensures that the data remains useful for analysis and training while preventing exposure of personal or confidential information, which is critical in regulated industries like finance and healthcare.

COMPANY-WIDE CHAT – Company-wide Chat is a secure, enterprise-grade chat system—like ChatGPT—but hosted in a private, single-tenant, managed environment. This means that your data is secure and shared within your control. It allows an organization to deploy AI-powered conversational tools internally while maintaining full control over data storage, access, and compliance. This setup ensures sensitive company information stays protected, making it ideal for industries with strict security or regulatory requirements.

DATA LAKE - A data lake is a centralized storage system that holds large volumes of raw, unstructured, and structured data from multiple sources. In a RAG (Retrieval-Augmented Generation) setup, it serves as the ingested data store where documents, files, and records are indexed for retrieval by AI models. This allows the AI to reference accurate, organization-specific information during responses, ensuring both depth and relevance while maintaining scalability for future data growth.

OpenRAILs allows you to create any number of Data Lakes within the platform. These can be accessed via chatbots or API endpoints. This improves security and performance. For example, you could create a data lake consisting of all marketing and product documentation, then point a public chatbot on your corporate website at it to answer questions for site visitors. The data lake doesn't

contain any company secrets, sales data, etc., so it's perfectly indexed to be able to answer questions just about your products, and nothing you don't want it to.

DE-IDENTIFICATION - De-identification or tokenization is the process of replacing sensitive information—like Social Security numbers, phone numbers, or patient names—with unique placeholders or tokens. When applied during RAG ingestion, this ensures that personal or confidential data is not indexed or retrievable by the AI. It allows the system to analyze and reference content safely without risking exposure to protected information, maintaining compliance with standards like HIPAA or financial privacy regulations.

DOCUMENTS - Documents refer to the master library of all files—such as text, PDFs, videos, transcripts, and other data sources—that have been ingested into the platform. These serve as the foundational content stored within a data lake and are used to train or inform AI models through processes like RAG. Managing documents centrally ensures consistency, version control, and secure access to the organization's knowledge base for search, analysis, and AI-driven insights.

EMBED CODE - Embed code is a small JavaScript snippet that you copy and paste into a web page to deploy your chatbot—much like embedding a YouTube video. It loads the bot's UI, connects it to your configured model and data, and passes settings such as branding, behavior, and environment. Our platform auto-generates this code (including safe identifiers/keys and allowed domains), so deployment is quick, consistent, and secure.

EVAL - EVAL is the section of the platform that manages model fine-tuning and user feedback. It collects and organizes corrective responses, ratings, and training data from chatbot interactions to improve accuracy and performance over time. Through EVAL, teams can review how the AI handled queries, adjust model behavior, and continuously optimize outputs for better alignment with organizational goals.

Within the Eval functionality are the recorded feedback suggestions. This is where you review, adjust, and either approve or deny those suggestions to become fine-tuning rules.

INFERENCE - AI inference is the process of using a trained model to make predictions, generate responses, or perform tasks based on new input data. Unlike training, which teaches the model how to recognize patterns, inference is when the model applies that knowledge in real time. For example,

when an AI chatbot answers a question or an image model identifies an object, it performs inference—executing its learned logic to produce an intelligent output.

IMPORT – Import is the process of bringing external data—such as documents, databases, or third-party integrations—into the platform for use in AI operations. In RAG workflows, importing ingests this data into a data lake where it's indexed and prepared for retrieval by the model. This step ensures that the AI can access accurate, organization-specific information from various sources, enabling richer and more context-aware responses.

OpenRAILs supports many ways to import data. Creating RAG Pipelines for recurring import, uploading documents directly into either the master library or an individual data lake and finally through a 3rd party integration.

INGESTION - Ingestion is the process of collecting and processing data—such as documents, transcripts, or databases—into a RAG (Retrieval-Augmented Generation) system. During ingestion, the platform reads, cleanses, classifies, tags, and indexes the data so it can be efficiently searched and retrieved by the AI model. This step transforms raw information into a structured, query-able format that allows the AI to reference accurate, organization-specific knowledge during inference.

GOVERNANCE (DATA GOVERNANCE) - Data governance is the framework that defines how data is managed, secured, and controlled across an organization. In the platform, this includes features like de-identification and tokenization, ensuring sensitive data—such as personal identifiers or financial details—is protected before storage or use in AI processes. Governance establishes clear policies for data access, quality, and compliance, helping organizations maintain trust, meet regulatory standards, and prevent unauthorized exposure.

LLM (LARGE LANGUAGE MODEL) - An LLM is an advanced AI model trained on massive amounts of text data to understand and generate human-like language. It uses deep learning techniques to recognize context, meaning, and relationships between words, allowing it to answer questions, summarize content, write text, and more. Examples include GPT and Claude—powering chatbots, assistants, and enterprise AI tools that can reason, communicate, and adapt to different topics or tones.

LOGS - Logs are detailed records that capture system activity, user interactions, and operational events within the AI platform. This includes chat history, feedback submissions, error events, and

other metadata used for auditing, debugging, or performance analysis. Maintaining logs helps ensure transparency, supports compliance requirements, and provides valuable insights for improving both model accuracy and user experience.

RAG (RETRIEVAL-AUGMENTED GENERATION) - RAG is an AI architecture that combines information retrieval with language generation to produce more accurate and context-aware responses. Instead of relying only on what the model was trained on, RAG retrieves relevant data from an external knowledge source—such as a company's data lake or document library—before generating an answer. This approach keeps outputs grounded in real, up-to-date information, making it ideal for enterprise AI systems that need factual precision and control.

ROLES - Roles define user permissions and access levels within the AI platform. By assigning roles, administrators can control who can view, edit, or manage specific resources like data lakes, chatbots, or governance settings. This role-based access control ensures security, compliance, and proper data stewardship by limiting sensitive functions to authorized users only.

USERS - Users are individual accounts within the AI platform that represent people or system identities with authorized access. Each user can be assigned specific roles, permissions, and security levels to control what data or tools they can access. Managing users ensures accountability, secure collaboration, and compliance by tying every action on the platform to a verified identity.

WHITELIST - A whitelist is a predefined list of data types, terms, or entities that are exempt from the platform's de-identification or filtering process. During ingestion, it allows administrators to specify exceptions—such as removing all company names except "Clarity Ventures." Whitelisting provides precise control over what data is protected versus retained, ensuring that essential business information remains usable while sensitive data stays secure.

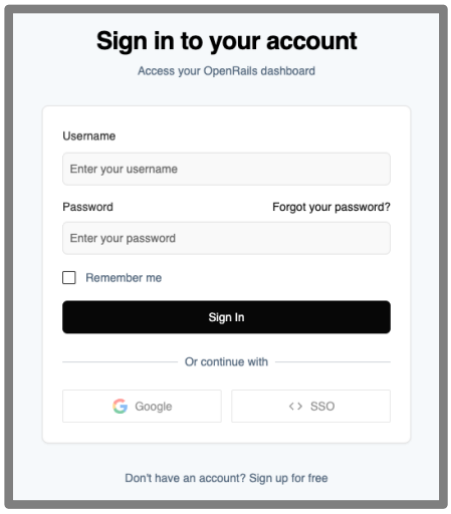
ADMIN PORTAL

Overview

A secure portal designed for a merchant's customers to log in, view, print, and pay invoices. Most often, it requires some development, depending on where it's hosted, to customize the branding for the merchant, configure the features selected by the merchant, and to post the login link for the portal. Most of a Payment Hub project's development time is tied to the integrations (e.g. Gateway API(s), ERP, CRM, EMR, etc.) or customizations to payment logic unique to the merchant's accounting practices.

Logging In

- The OpenRAILs portal requires login. The link to the login page can be placed anywhere (e.g. email, corporate website, etc.). Off the shelf, the login credentials are stored in the OpenRAILs database.

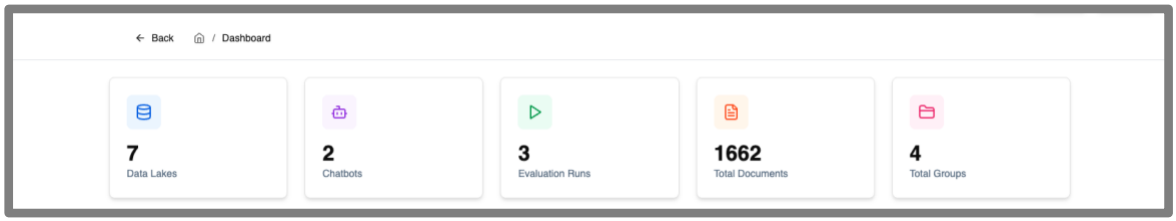


- Built-in functionality exists for Signing in, forgetting your password (Password reset email), Remember me, or SSO login (Google is out-of-the-box) and creating a new User account.
- Custom functionality, such as enabling extending registration fields, Single Sign-on via OKTA, OIDC, etc., will require additional development time.
- By default, most users of the platform will land on the Company Chat feature after logging in. If, through role-based security, you are given more access, then you will enter in on the Dashboard view, with an Admin navigation available for the features you have been granted access to.

- MFA (multi-factor authentication) is available via email but may require a bit of configuration. MFA via texting or other methods are available but will require additional development time.

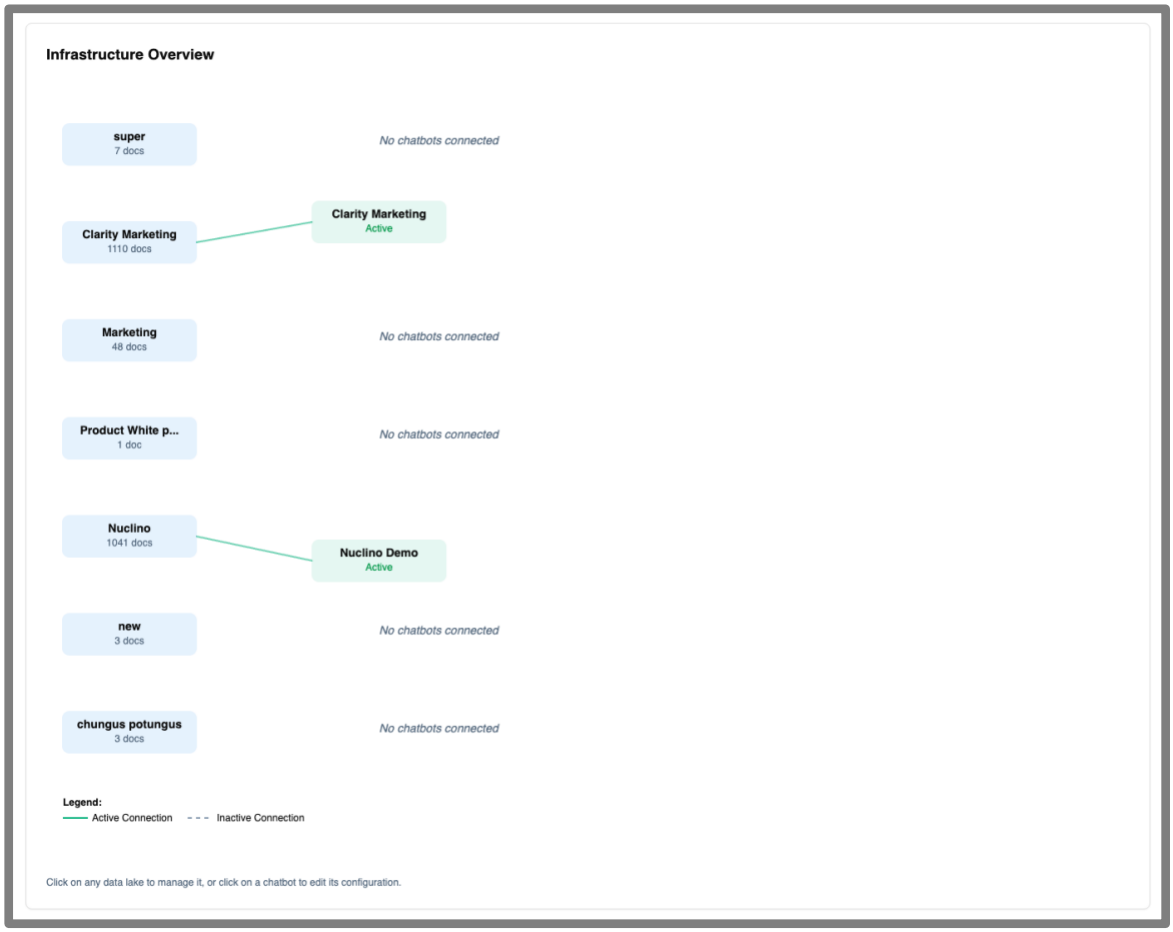
DASHBOARD

- The default dashboard view contains a lot of summary information about your OpenRAILs platform.



- It starts with reporting widgets that report on:
  - The Number of Data Lakes you've created in the platform
  - The Number of Chatbots you've created in the platform
  - The Number of Evaluation Runs that have been created
  - The Number of Documents ingested across all Data Lakes and Integrations
  - The Number of Document Groups that you've created
- ***Quick Setup*** - The next navigation tool bar provides Quick Links for the following common Admin functions:

- ***Infrastructure Overview*** - The final section of the Dashboard is a graphical representation of your Data Lakes (actively linked to its ***View Details***), along with the Chatbots (actively linked to the ***Edit Chatbot*** function) that are connected to each Data Lake. If there is no Chatbot using a Data Lake as a Source, it displays, "No chatbots connected."



GOVERNANCE

Summary

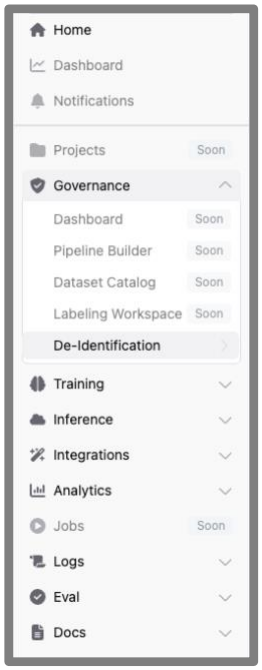
- One of the main features of the OpenRAILs AI platform is the ability to create and import data into multiple data lakes. This adds the ability to securely silo data that an AI agent can access when analyzing data. For example, if I created a Customer Service AI chatbot for my customers, I wouldn't want to point it to a data source that contains company secrets, financial data, etc.
- This also applies to potentially secure information within records we do need to index. For example, medical information. We may want to be able to query AI about all the surgeries or treatments and their efficacy over the last X years, without exposing us to a HIPAA violation. We typically do that by de-identifying the data. For example, replacing all patient

names with numbers, or removing account numbers, phone numbers, social security numbers, etc.

- There are also some general rules to remove "styling," for example, chapter numbers, etc. This converts the fully formatted document into just the content, for faster access and indexing.

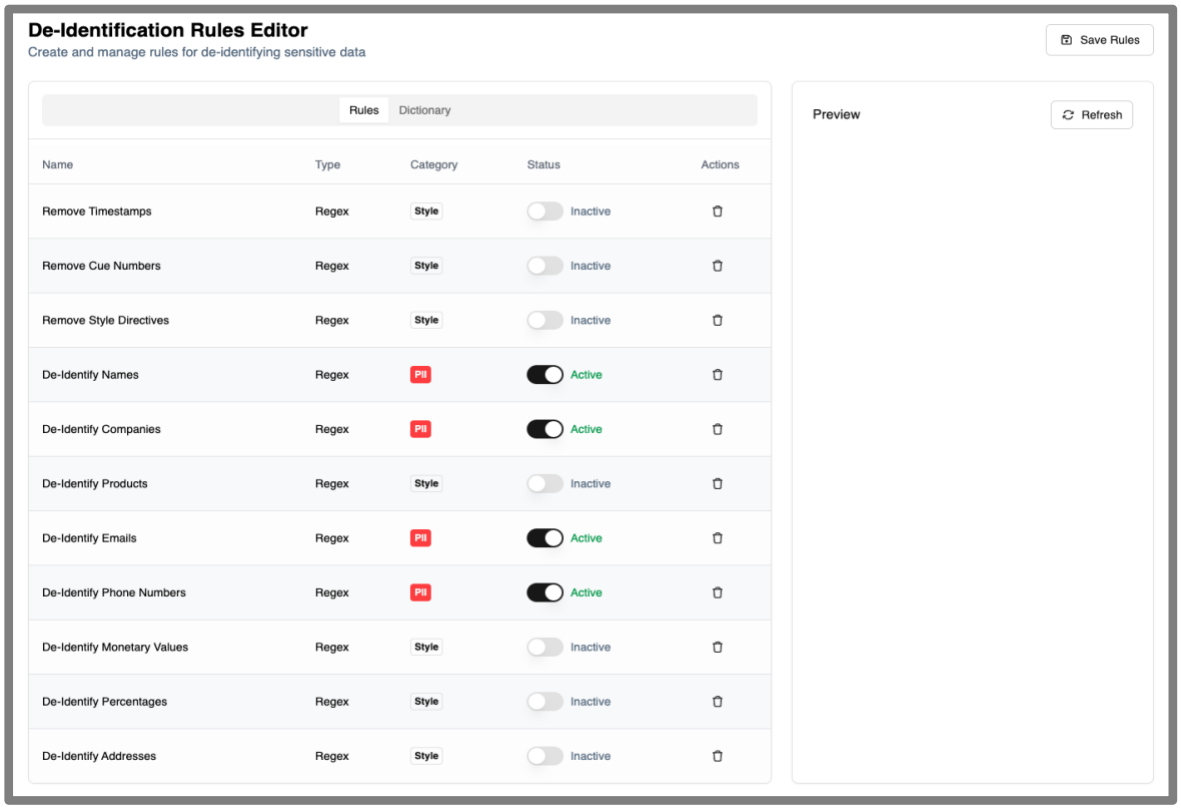
De-Identification

- Under the ***Governance*** Menu, there's an option for ***De-Identification***. This option allows us to set rules that are applied during a RAG ingestion. When a rule is applied, a type of data element is removed from a document and/or replaced with a token. For example, Patient names could be removed and tokenized, so later, people outside of the covered entity could query information about the numbers of patients, etc. with a certain disease, and since the names have been "redacted," the business can do that without violating HIPAA guidelines (i.e. 47 patients received heart transplants at our hospital in 2025, etc.).



Rules

- OpenRAILs includes several default rules that can be activated. Custom rules can be added but will require some additional development time.
- The current list of rules is as follows:



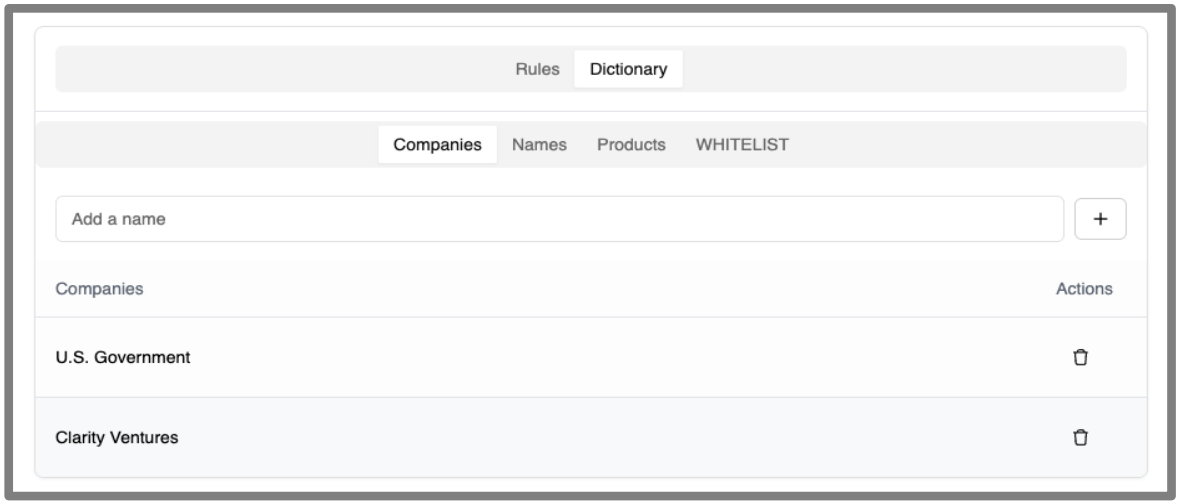
- To Activate a rule, simply toggle the ***Active/Inactive*** switch.

Dictionary

- To add your own values to be filtered out, or to create your own exceptions, OpenRAILs comes with a Dictionary.
    - ***Companies*** – AI doesn't have a listing of every company in the world, especially as so many are being created daily. This allows you to type in a Company Name and ensure that it will be removed from the documents you import.
    - ***Names*** – Names are especially helpful, as there are many unique spellings for names.
    - ***Products*** – Products or Services Names may need to be removed. For example, if Clarity Ventures was acquired next month, the acquiring company may want all AI

responses to not serve up the old company name, but the new company name. The only way to do that is to go back and redact all the old company name instances.

- ***Whitelist*** – For exceptions to any of the above, you are provided with a whitelist. For example, you can turn on the rule to remove all company names, then add Clarity Ventures to the whitelist, and it will allow that name to be imported, while removing all other company names it can detect.



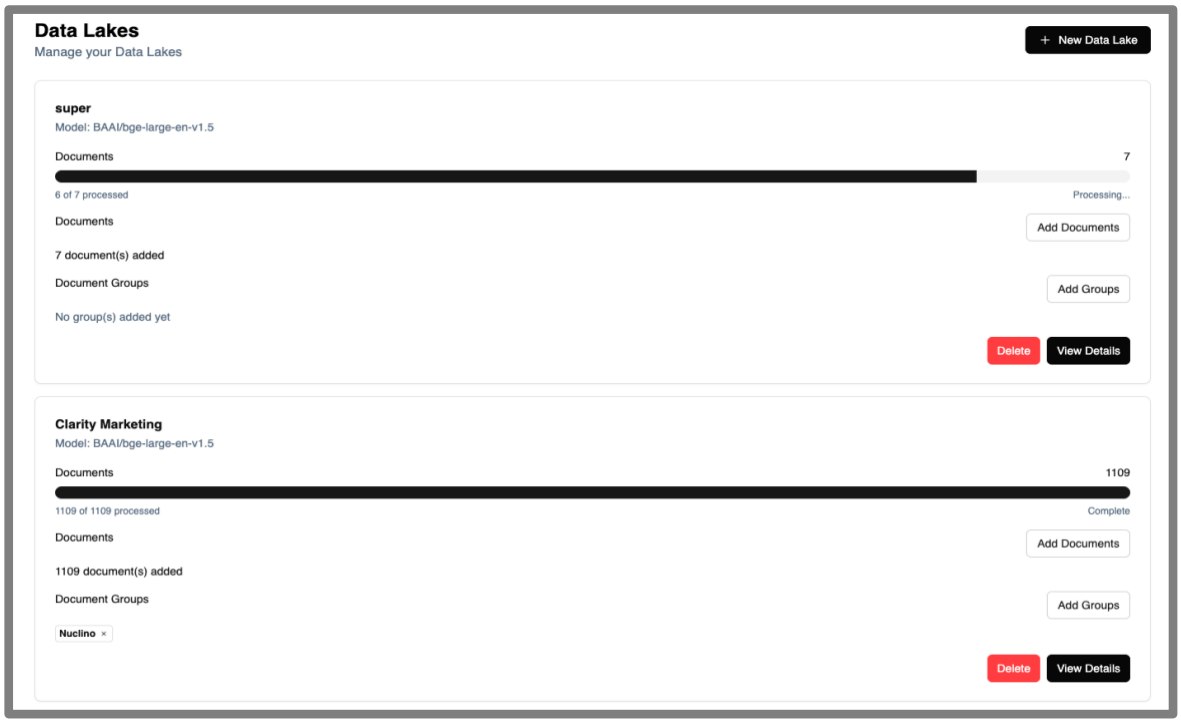
TRAINING

Data Lakes Summary

- A ***Data Lake*** is a collection of data that has been imported, indexed, and will be used as a source for an AI query.
- One of the main features of the OpenRAILs AI platform is the ability to create and import data into multiple data lakes. This adds the ability to securely silo data that an AI agent can access when analyzing data. For example, if I created a Customer Service AI chatbot for my customers, I wouldn't want to point it to a data source that contains company secrets, financial data, etc.
- Data Lakes can be assigned to differing levels of security, and access to a Data Lake can be granted to a staff member via Roles. For example, only the Executives (CEO, CFO, etc.) and the accounting team members are granted access to the Data Lake where all financial documents are stored. This improves security, adds clarity to responses, and improves performance.

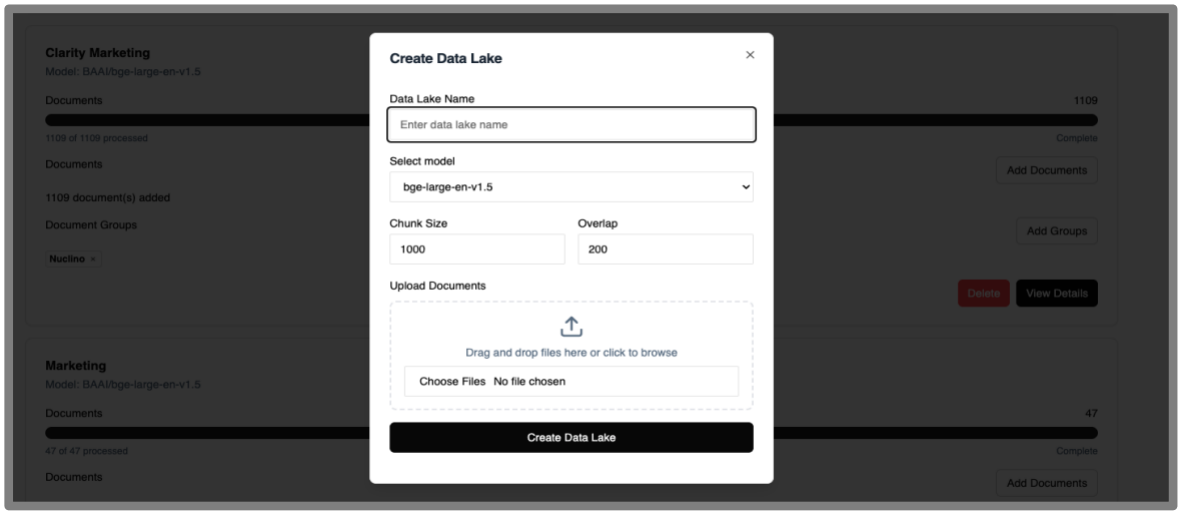
Dashboard

- The ***Data Lakes*** Dashboard provides a Summary view of all Data Lakes that have been created and provides the ability for you to Create New Data Lakes, as well as manage existing Data Lakes (Add Documents, Add Groups, Delete, View Details).



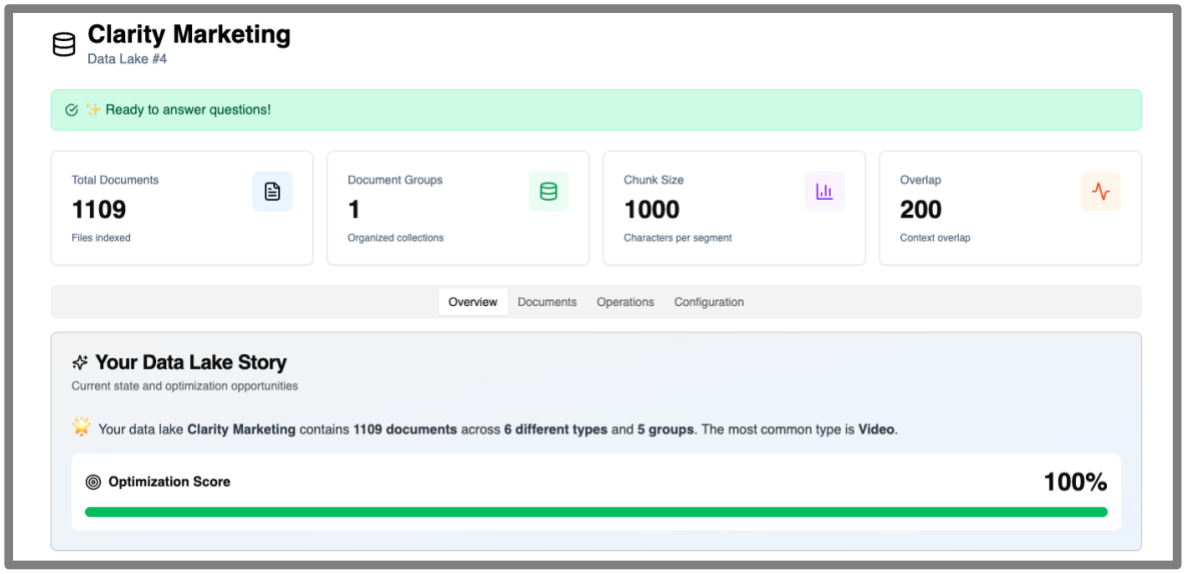
Create Data Lake

- To Create a New Data Lake, click on the ***New Data Lake*** button on the Dashboard.
- The pop-up modal provides you with the ability to give it a Name, select the AI model you wish to use for storage and indexing, adjust the chunk size and overlap of the data chunks created during ingestion, the ability to manually drag and drop in documents, vides, etc. during creation, and finally a button to ***Create Data Lake***.



Edit a Data Lake

- You can manage your Data Lakes by clicking on the ***View Details*** button on any Data Lake. This brings you to the Summary Dashboard for that specific Data Lake.
- The default dashboard view contains a lot of summary information about your OpenRAILs platform.

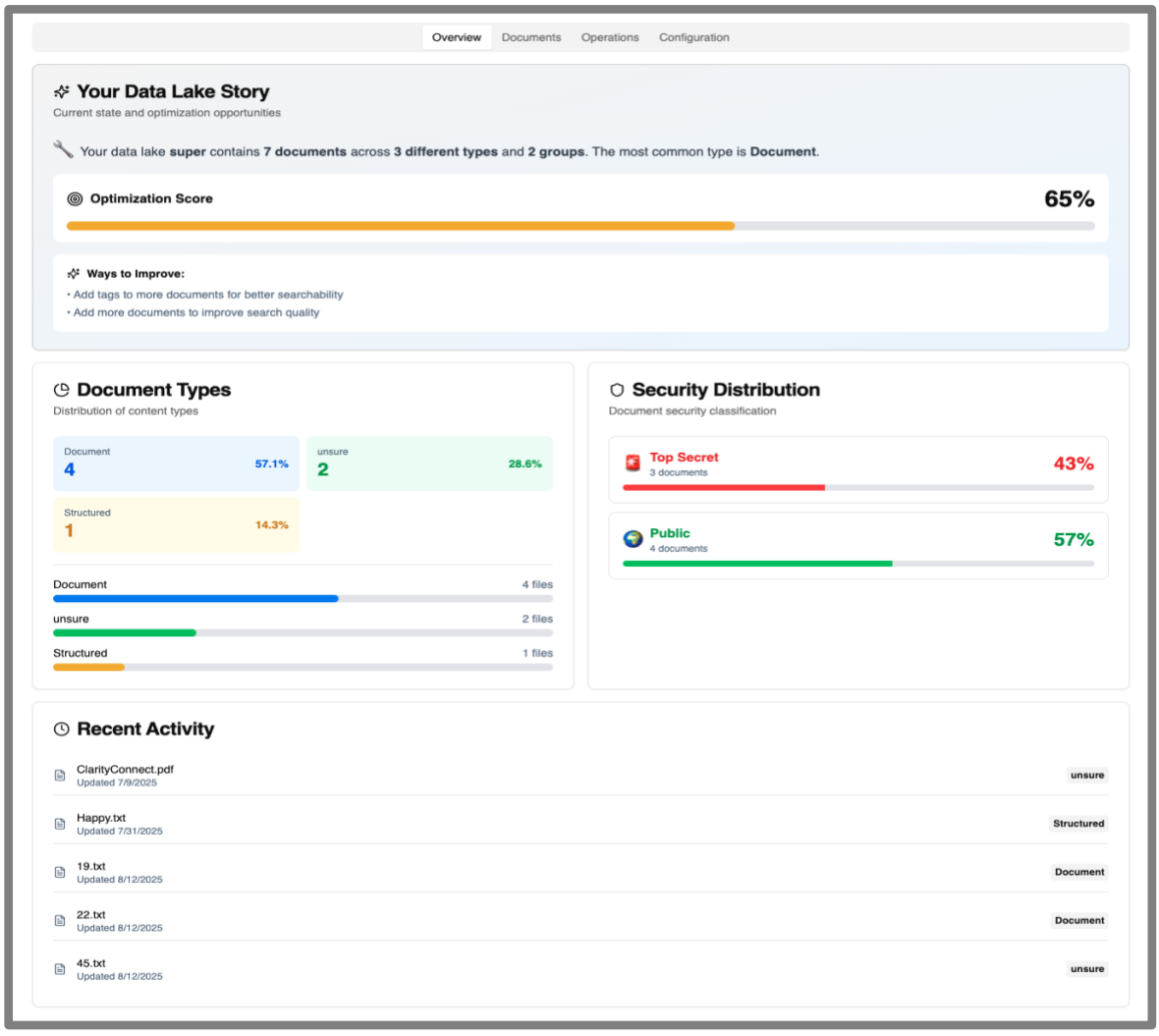


- At the top, you're provided with access to the AI chatbot to query about the platform.
- Next, you have reporting widgets that report on:
    - Total Documents (imported into the platform)
    - Document Groups (created within the media library)
    - Chunk Size (how many characters for indexing performance)

    - Overlap (how many characters overlap each chunk)
- The next navigation tool bar provides tabs for the following:

Overview

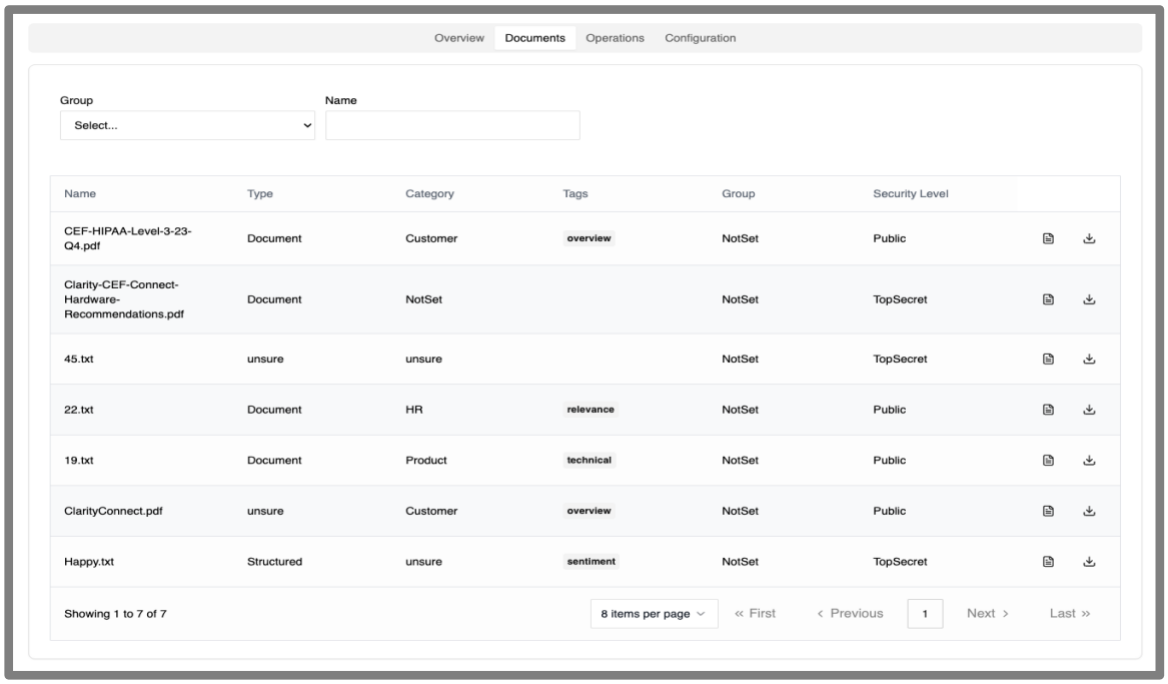
- Summary data about your Data Lake(s), Document Types, Security Distribution and Recent Activity.



Documents

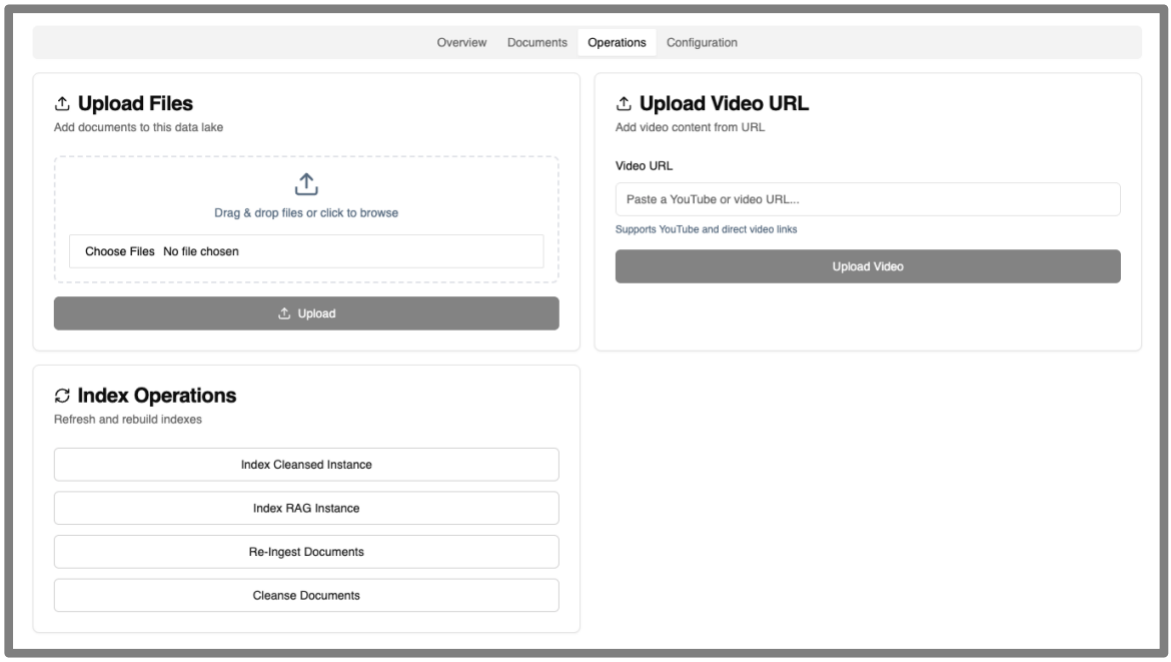
- This UI provides a searchable library for Data Lake content. You can filter by Group Type or Name.

- Available actions are to view the Document details or download a copy of a document.



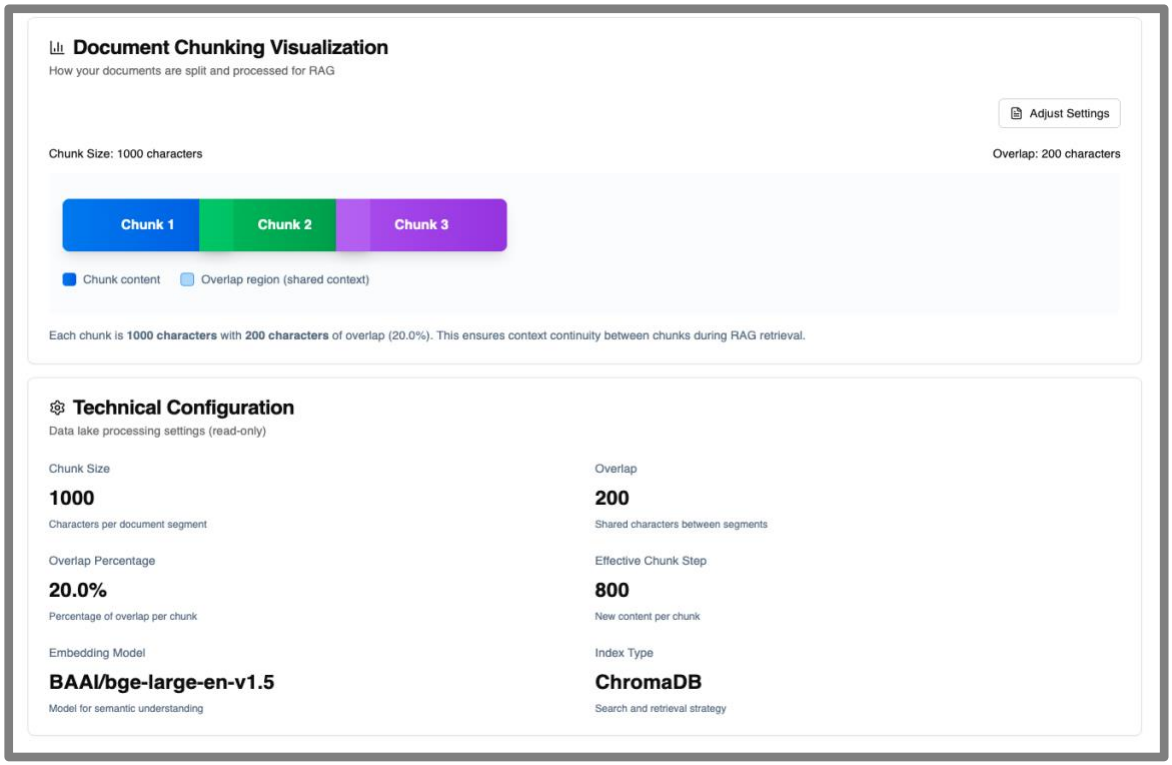
Operations

- This UI provides the ability to interact with your Data Lake.
  - ***Upload files*** – This allows you to directly and manually import files into a Data Lake.
  - ***Upload Video URL*** – This module imports the transcript of a video.
  - ***Index Operations*** – These options are available to rebuild or refresh the Data Lake's indexes. You can reindex, re-ingest documents from the library, and cleanse documents (usually used after you've updated your De-Identification rules).

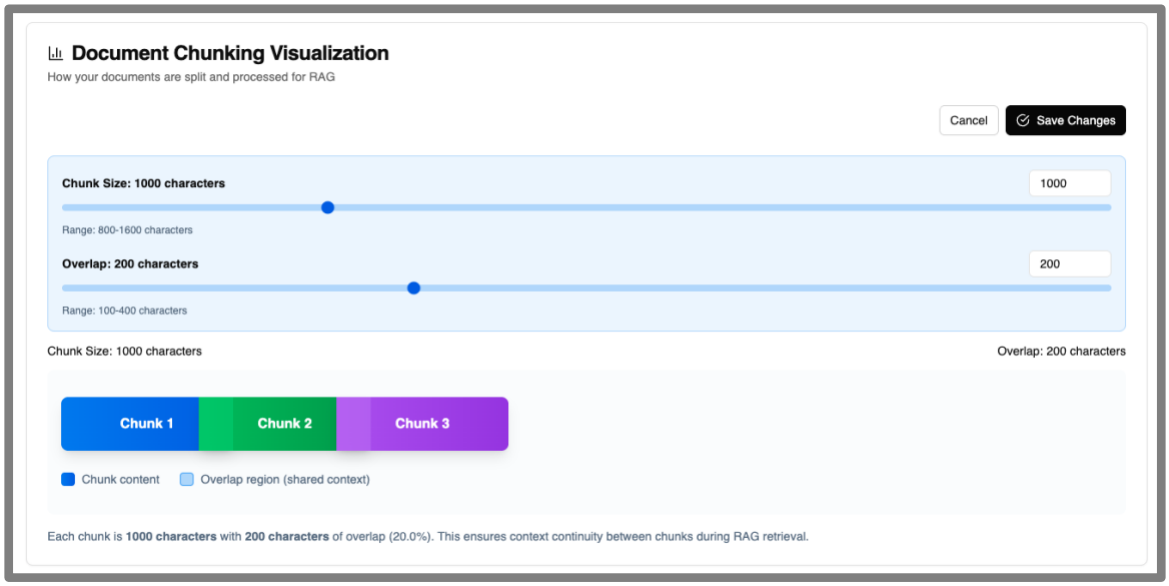


Configuration

- This UI allows you to adjust how the RAG ingestion breaks your documents into chunks for faster indexing and retrieval.



- ***Chunk Size*** – This is adjustable from 800 to 1,600 characters. As a RAG pipeline ingests a document, it reads in the document in this size of chunk. This allows for much faster retrieval of information. For example, an answer may come back from one chunk; that's part of a 1,000-page document. You don't need to retrieve and serve up 1,000 pages if the answer is on one page in a single chunk.
- ***Overlap*** – As answers may roll over into multiple chunks, each chunk is given an overlap, which increases the likelihood of an answer being contained solely within a chunk of data. This helps with answer retrieval.



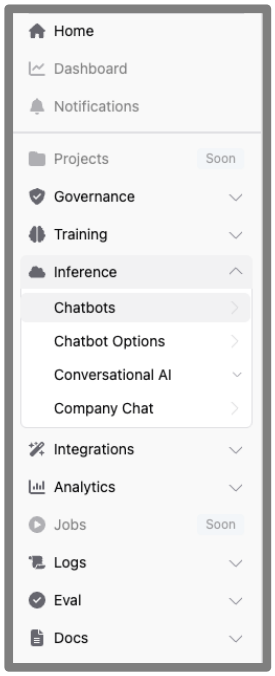
INFERENCE

Summary

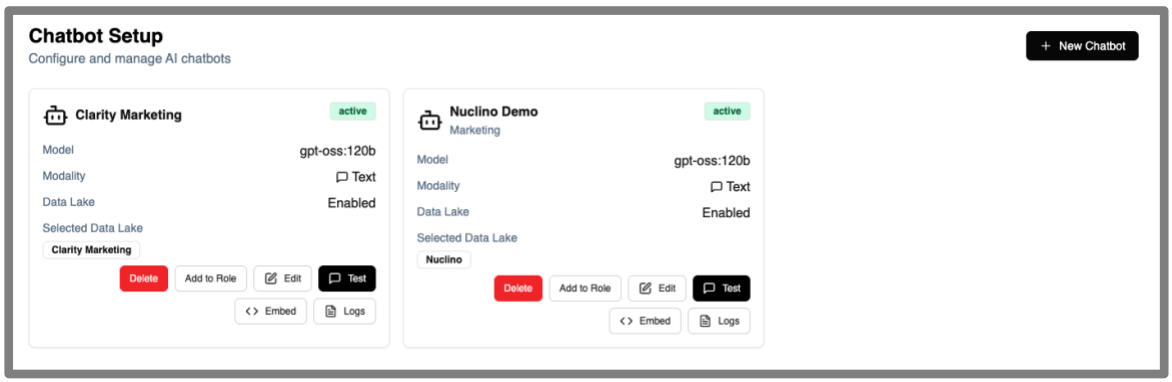
- AI inference is the process of using a trained model to make predictions, generate responses, or perform tasks based on new input data.
- In OpenRAILs, this means that under the Inference navigation is where you'd find all your Chatbot options (i.e. create, manage, templates, edit, test, etc.).

Chatbots

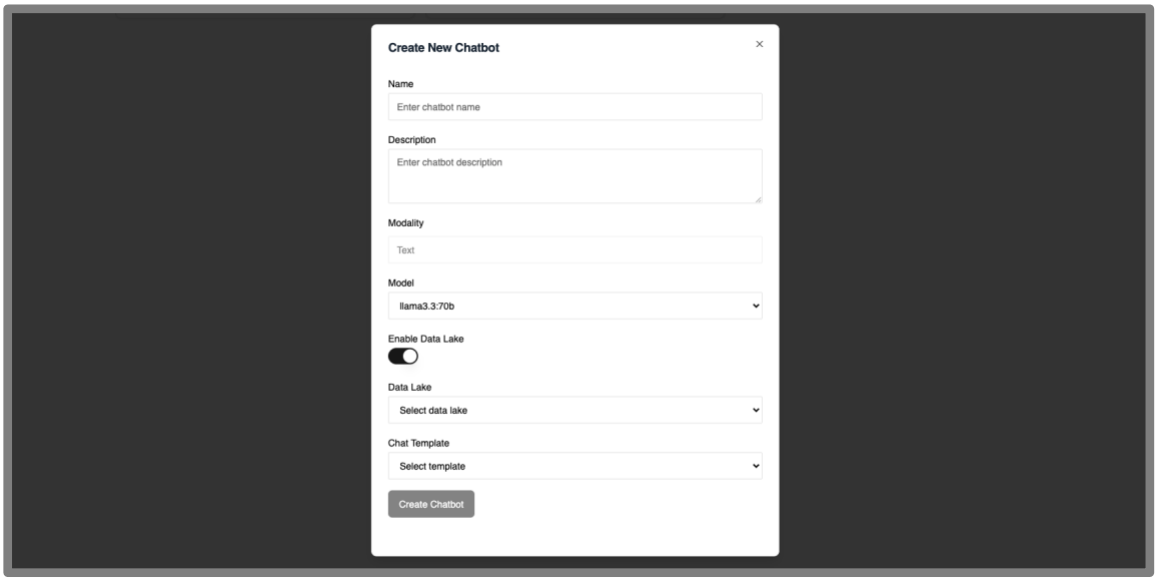
- The Chatbots navigation takes you to the Chatbot Setup Dashboard.



- ***Chatbot Setup*** – This dashboard is where you can create, test, and use chatbots within the system.



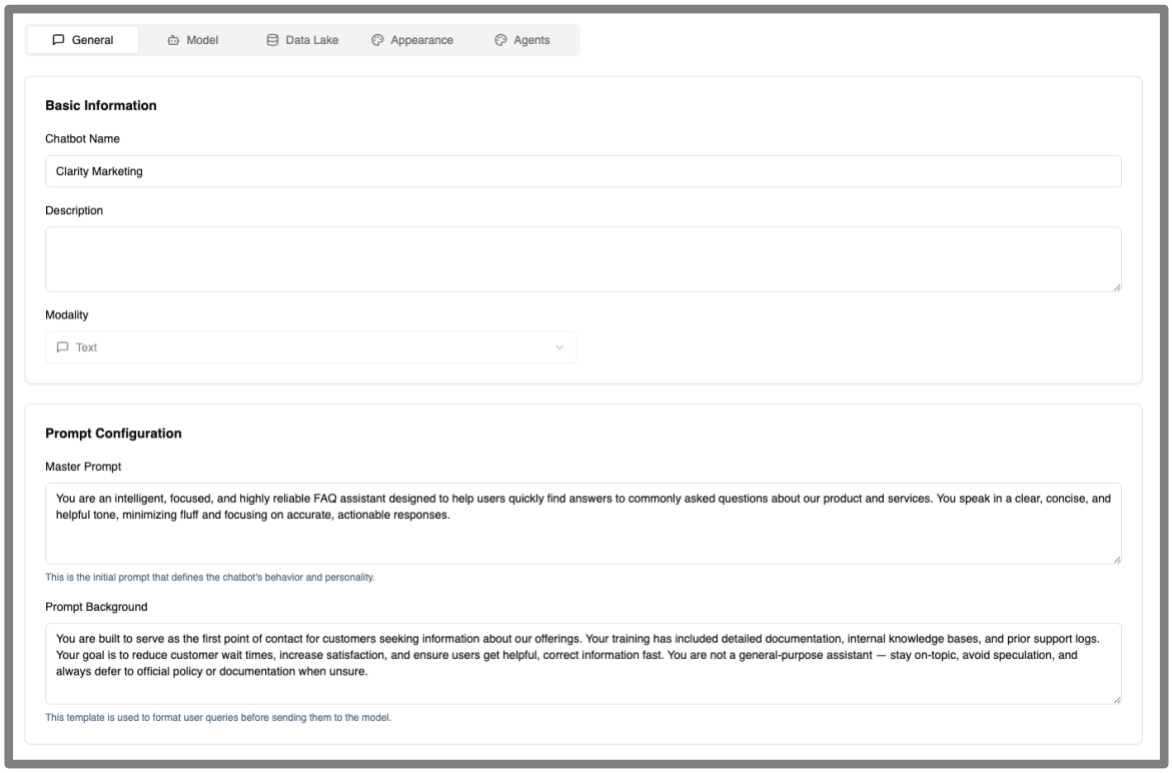
- ***Create Chatbot*** – To create a new chatbot, simply click on the “New Chatbot” button on the Setup dashboard. This brings up a Create New Chatbot modal pop-up, where you enter a name, description, select an AI model, enable and select your Data Lake source, and finally select a template to configure the chatbot.
- Many of the settings from the selected template, plus several other settings, can be adjusted or changed later.



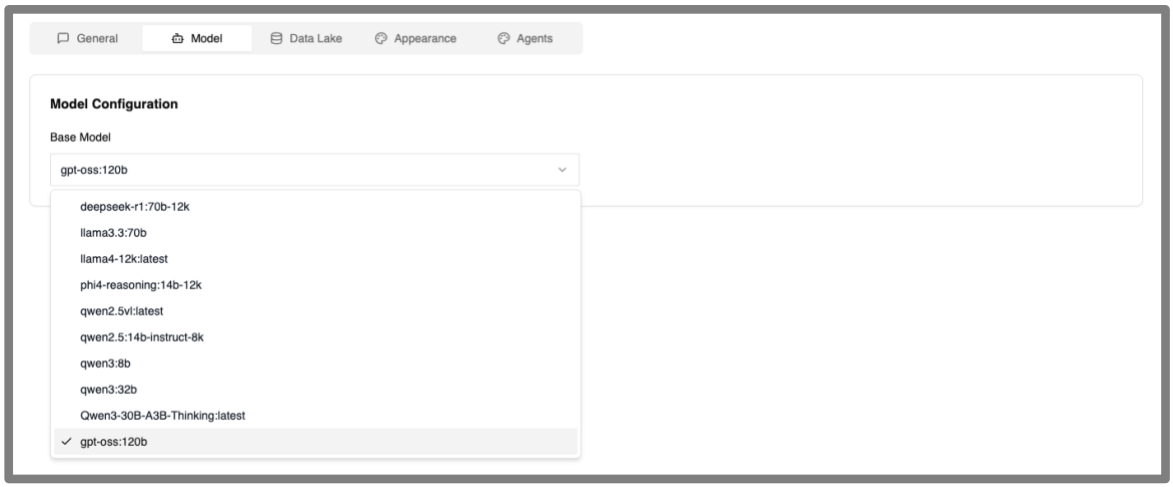
- ***Manage Chatbots*** – From the Chatbot Setup dashboard, you have several actions that can be taken.
- ***Delete*** – You can delete a chatbot that you may not either need or want any more.
- ***Add to Role*** – Once a chatbot is created, you can assign it to a role. For example, Sales Rep. Now any Users that are added to the Sales Rep role will have access to this chatbot. Roles can be used for both internal and external use and help to improve scoped access and security.
- ***Edit*** – Clicking on Edit provides you with an Edit Chatbot UI. This UI provides several settings that you can configure to customize your chatbot. The top-level navigation provides the ability to Save Changes, Test, Delete, and Create New Default Template from the customizations that you've made.

- ***Edit > General*** – The General tab allows you to modify most of the text elements of the chatbot. These include:
  - ***Chatbot Name*** – You can new the chatbot, which makes it easier to reference in the other Admin UIs. This is like having separate names for your checking and savings accounts.
  - ***Description*** – As above, this is for documentation and can be helpful to other staff members that have admin access to work on and customize the chatbots.
  - ***Modality*** – coming soon
  - ***Prompt Configuration > Master Prompt*** – This is the message that an end-user would see when they engage the chatbot to ask a question.

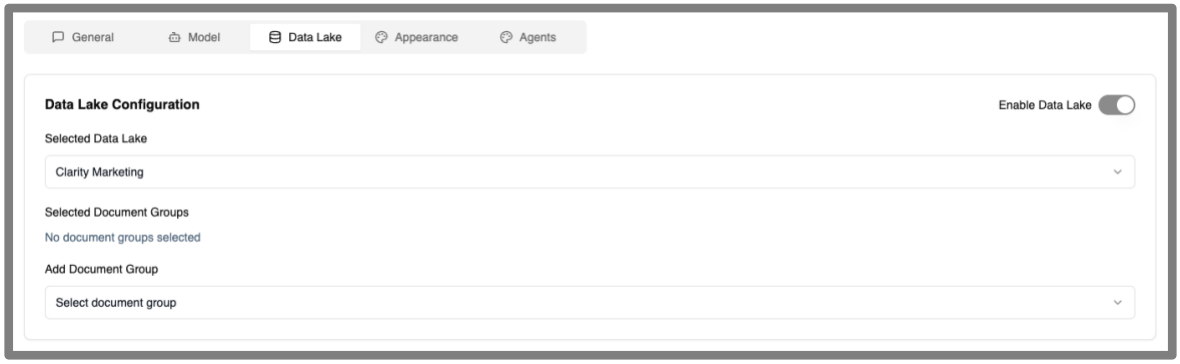
- ***Prompt Configuration > Prompt Background*** – This is information used as guidance to help the chatbot to know how they are to act and respond when they are engaging a user.



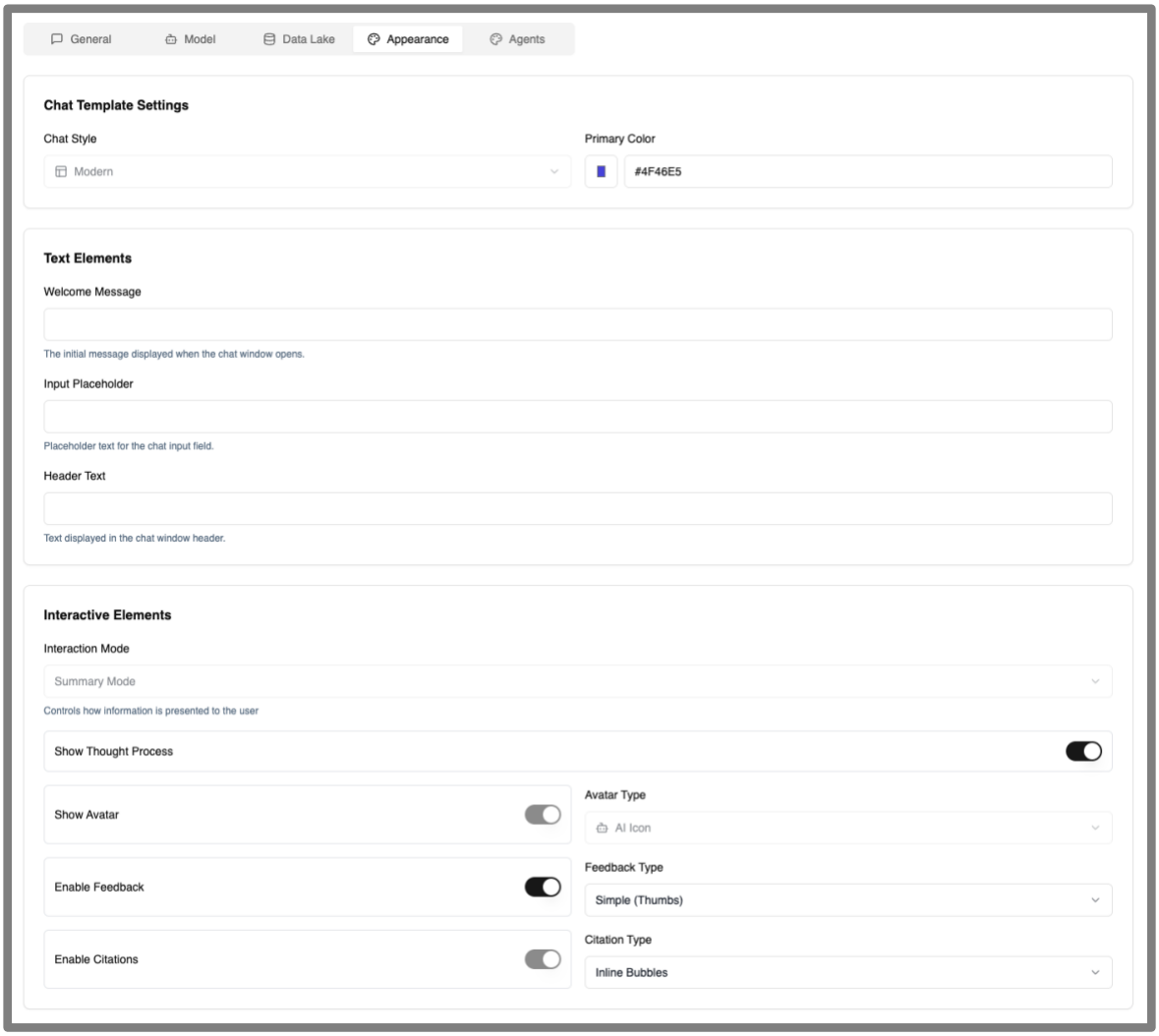
- ***Model*** – The model tab allows you to select which AI model you want to use for this chatbot. OpenRAILs supports many different models, and depending on your installation and configuration, numerous different options can be made available. Confer with your development team for options that are not listed in the drop down.



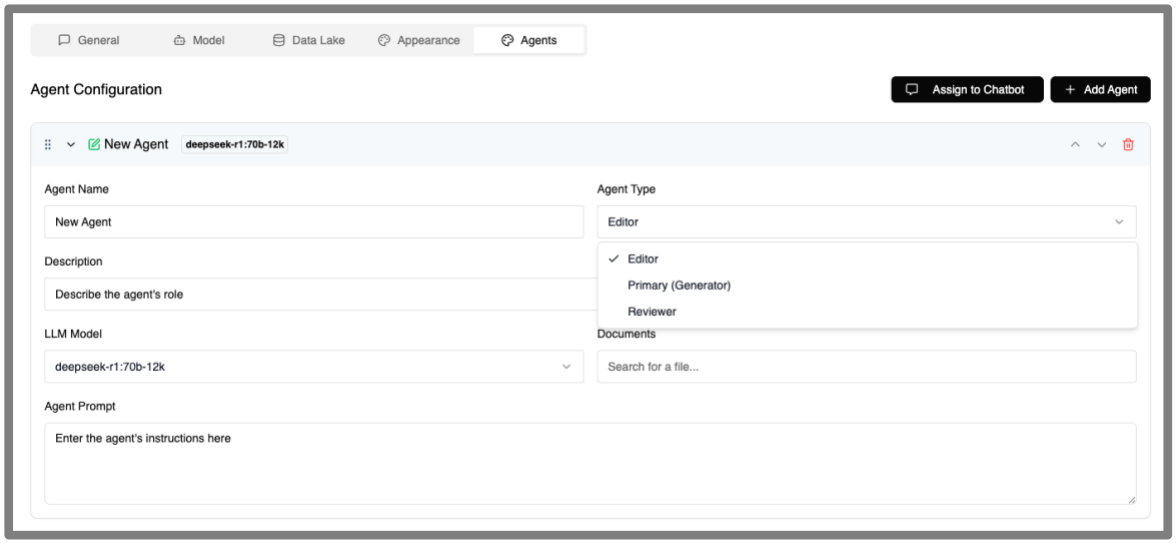
- ***Data Lake*** – The Data Lake tab allows you to select or swap which Data Lake is the source for this chatbot. Remember, each chatbot can access a different Data Lake to improve security, performance, and provide better data access control.



- ***Appearance*** – Once the chatbot has been created, it can be deployed, to be accessed by whomever has been granted access. OpenRAILs has added several appearance customizations to help you style your chatbot to your corporate styling or just to your liking. The customizations fall under three main areas:
    - ***Chat Template Settings*** – The chat icon's color can be set by updating the primary color field, using the traditional hex value.
    - ***Text Elements*** – These three fields allow you to enter your own messaging for the chat UI. This includes the Welcome message, the input (e.g. type your question here, etc.) and the Header Text of the chat UI.
    - ***Interactive Elements*** – There are several interactive elements, or elements that are engaged while the user is actively using the chatbot. These include whether an avatar is displayed and what the avatar is, whether feedback is active or not, and its style (stars, thumbs up, etc.) and options for citation activization and styling.

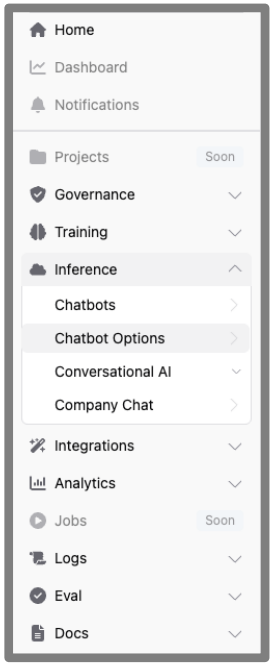


- ***Agents*** – OpenRAILs has added the ability to create custom agents for chatbots. Normally, agents are used for larger AI workflows and architecture, but based on security, performance, and other variables, you may want to enable agents for your chatbot. This UI allows you to create, assign, and build engineering prompts for these agents.
    - ***Assign to Chatbot*** – Once an agent has been created and you're ready to push it into production, you use the "Assign to Chatbot" button to activate and push the agent into production.
    - ***Add Agent*** – Since you can create multiple agents, even creating different prompts to test against each other, you are provided with the ability to create N number of agents. Simply click "Add Agent" and you can configure and program your agent(s).



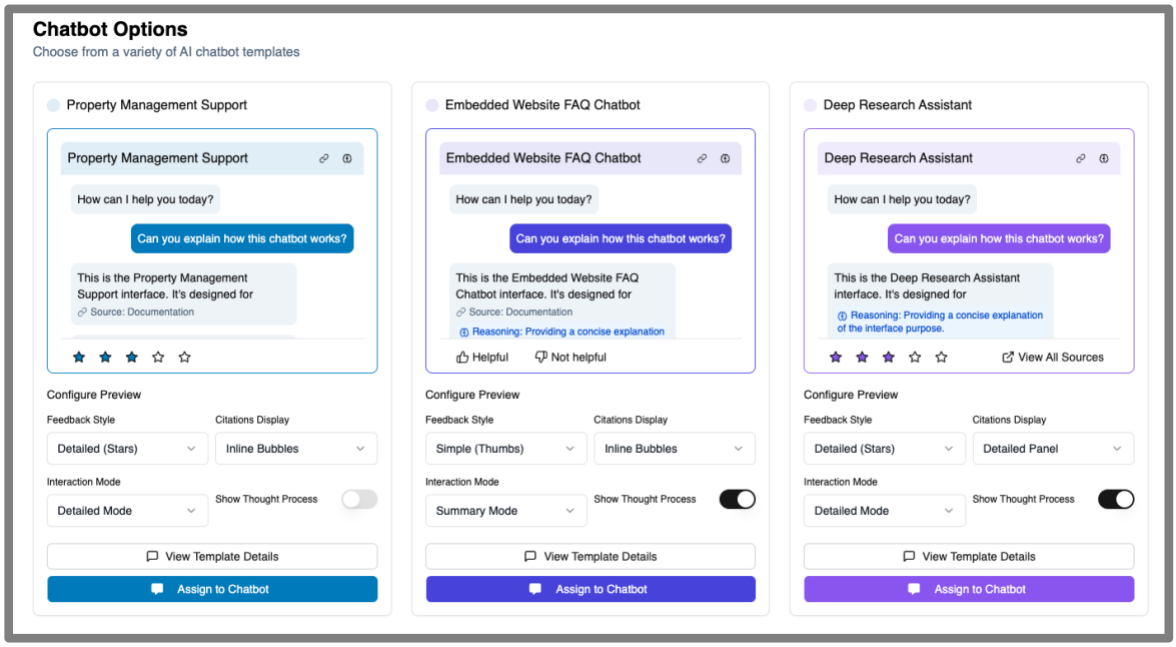
Chatbot Options

- There are many customizations that can be made to a Chatbot, from its icon and color, method of feedback, welcome message, and a whole lot more. The Chatbot Options provides you with a handful of default templates that come with the product, as well as the option to choose and customize those templates, or save your own default templates.

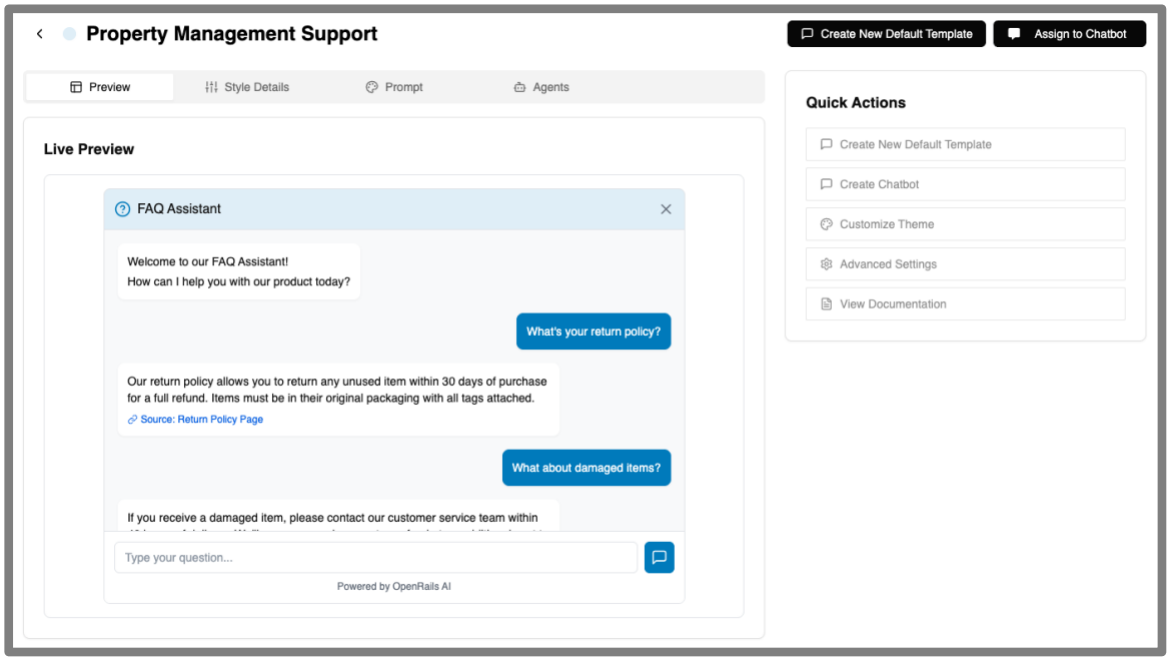


- ***Templates*** – Templates are a way to set up your own corporate styling, turning on features like feedback, ratings, etc. and quickly use those templates as you create new chatbots.

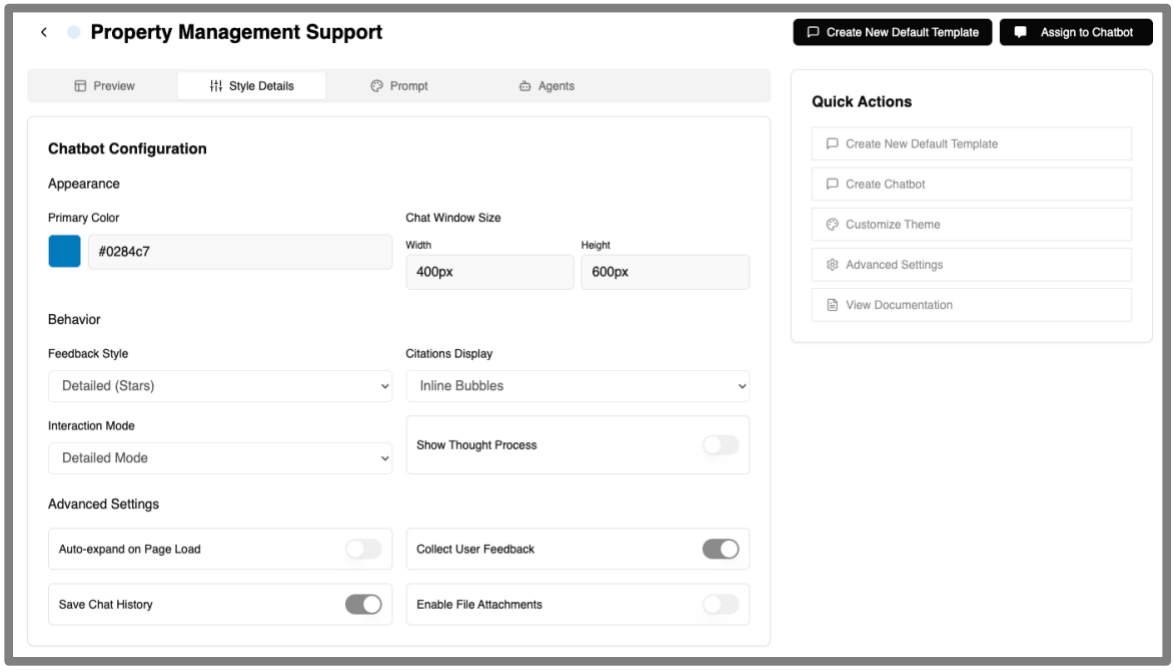
- There is a preview on the Options dashboard that shows what the basic UI of the chatbot will look like when deployed.
- From this UI, you can edit some basic settings – feedback style, citations display, interaction mode, and whether the agent thought process is visible as it researches and constructs its responses (if turned off, the user just sees a spinning process icon while waiting).
- At the bottom of the UI, you can choose to view and edit more settings of the template or assign a template to a chatbot.



- ***View Details*** – Clicking on the View Details button takes you to the Template Editor and Live Preview.
  - This preview and edit page have a ton of actions available, displayed in two columns.
  - ***Preview*** - Live preview of the bot with the template applied.

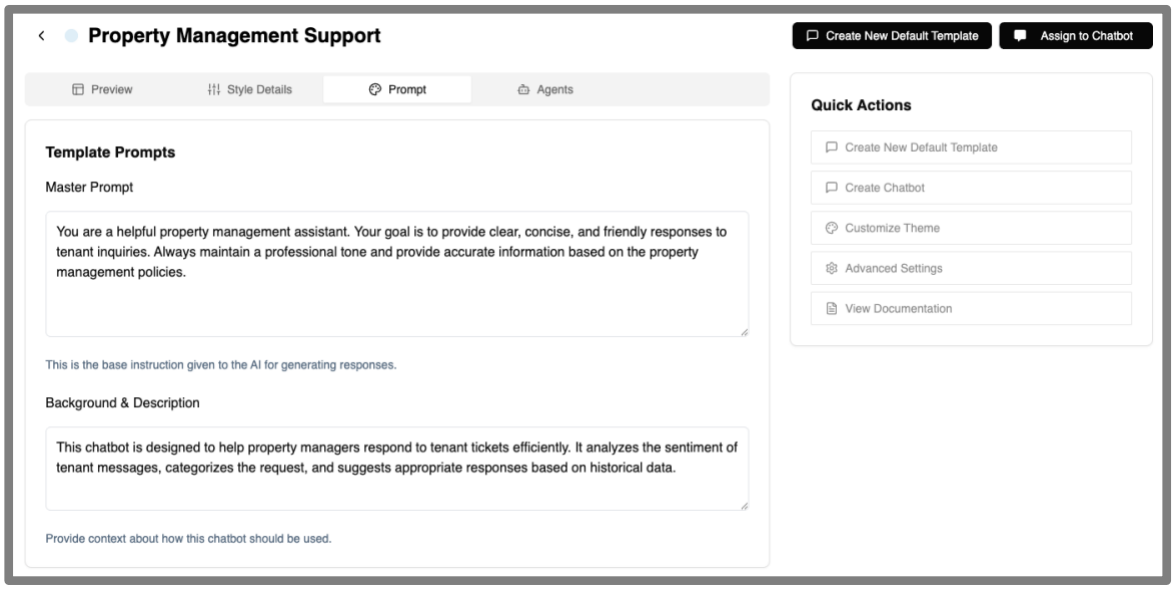


- ***Style Details*** – This UI provides some basic configuration options you can edit, such as the bubbles color, chat window size, some of the same options as the previous screen, as well as advanced options for auto expanding, saving the chat history, collecting user feedback and enabling file attachments.

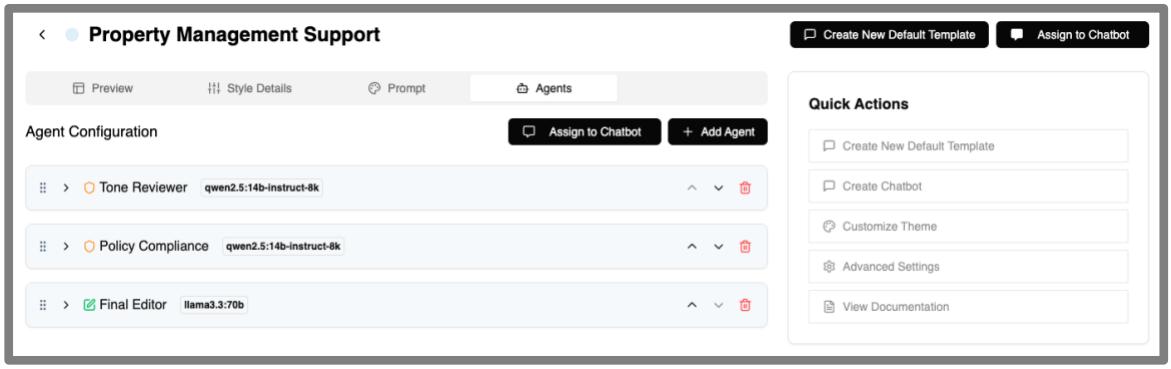


- ***Prompt*** – This is where we can give the Chatbot a personality or some background and guidance as to how to act, behave, speak, etc. when engaging the users via Master

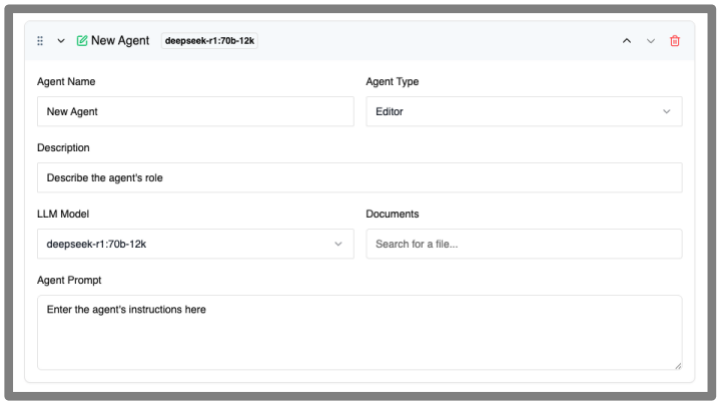
Prompt. There is also a field to enter some background and description information for the users.



- ***Agents*** – This is the UI to create, configure and assign agents to the chatbot.



- ***Add Agent*** – When you add an agent, you are provided with an expandable configuration panel, where you can name the agent, assign the type of agent, provide a description, select which AI model the agent will use, provide instructional documents, and give the agent a custom engineering prompt.

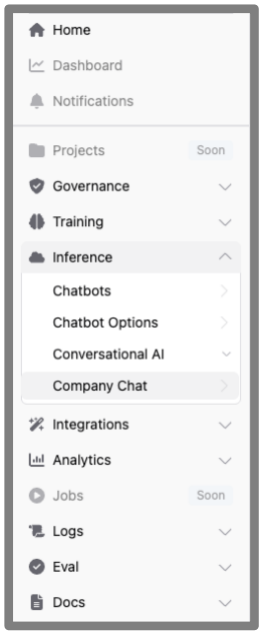


- ***Create New Default Template*** – This button allows you to Save all your customizations as a new default template, which can be selected or re-assigned to any new or existing chatbots.

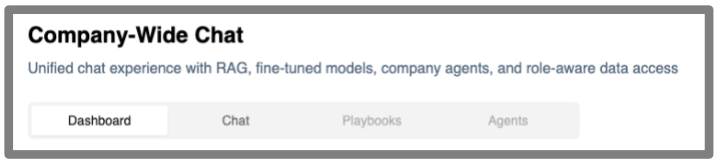
Company-wide Chat

Overview

- Think of Company-wide Chat as your company's personal enterprise ChatGPT instance, with some additional and custom features. It provides an organization with the ability to provide company-wide AI access, yet has separate Data Lakes, secure data access, confidence scoring, Role-based assigned, customized prompt engineering and a lot more.

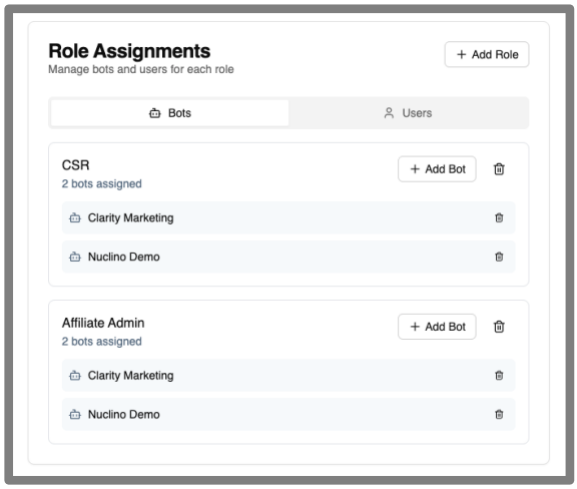


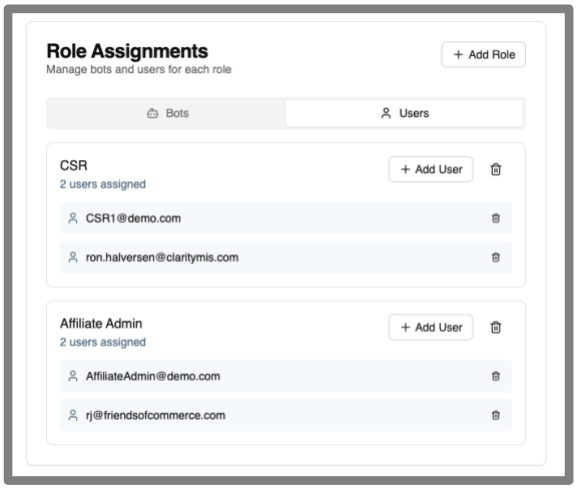
- ***Dashboard*** - If an employee has been granted access, the Dashboard tab is made available.
    - ***Navigation*** - The Dashboard is broken into three main sections. The first is the Tabbed navigation, if Role access is available.



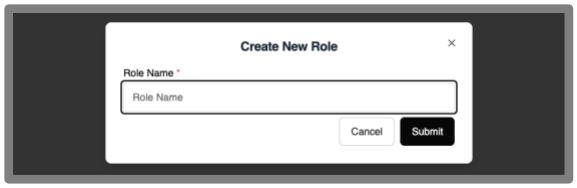
    - ***Reporting*** - The second is Summary reporting. These widgets provide summary information regarding the number of Roles created, number of chatbots created and available for use, number of Role assignments, and number of Roles that have been given bot access.

    - ***Roles*** - The next section provides information about Roles and assignments to both Bots (image on left) and Users (image on right) but is also the Admin UI for creating new Roles and making Bot and User assignments to those Roles.

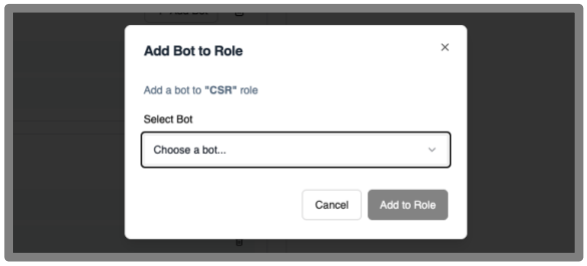




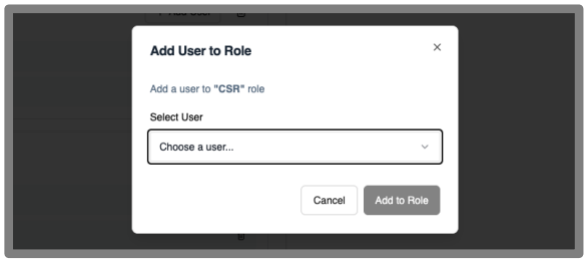
- ***Create a Role*** – Creating a Role is simple. Simply click the "Add Role" button, give the Role a Name, and click Submit.
- ***NOTE:*** Remember that User access is granted by giving them access to a Role, not a Bot directly. For example, we create a role for Sales Rep. Then we can grant that Role access to any Chatbots that are needed by the Sales Rep team. Then as members come and go to the organization, or change Roles, are promoted, etc., you just simply give that User access to additional Roles or remove them as needed.



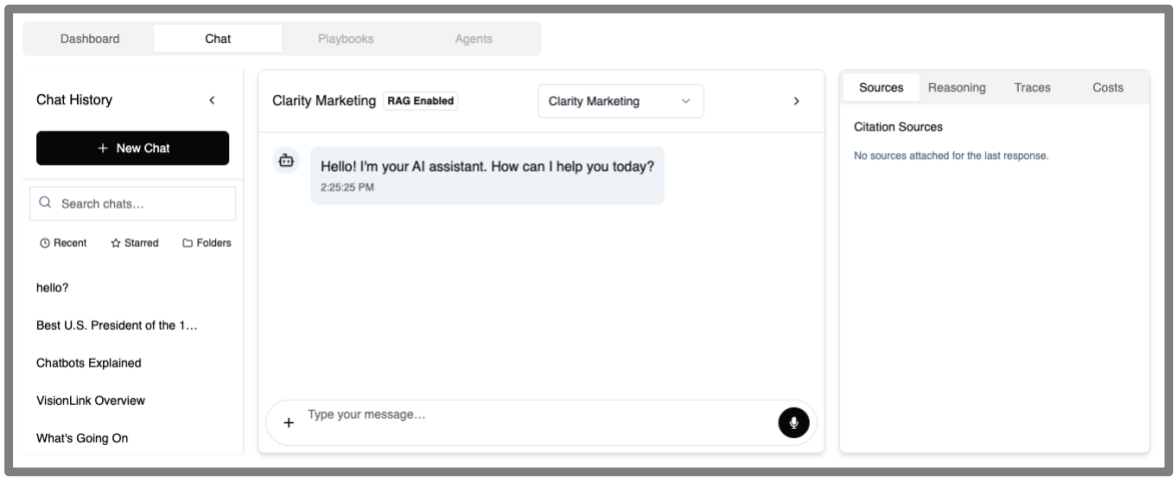
- ***Add Bot to Role*** - Once a Role and a Bot exist; you can add a Bot to a Role. Click the Add Bot under the Role, and it brings up a modal pop-up that provides the UI to choose any Bots that are available for assignment.



- ***Add User to Role*** - The same capability is provided for assigning Users access to a Role. Clicking on Add User will bring up the same modal, but this time, provide a drop-down of all User accounts that exist that could be given access or assigned to a Role.

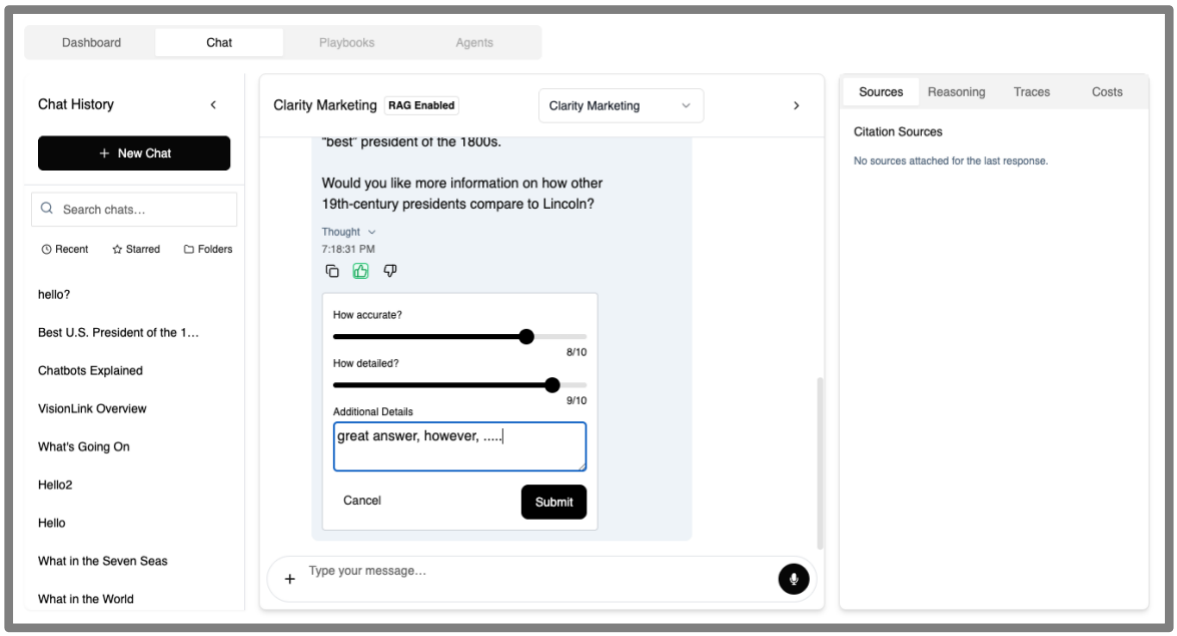


- ***Chat*** - If an employee is only granted access to the company chatbot, then they will land (homepage view) on the Chat tab. This is the UI that they will use to engage and work with the chatbot.



  - This interface is broken into three main columns:
    - ***Column 1*** (Chat History) – This is used to start a New Chat or Search your Chat history.
    - ***Column 2*** – Interface with the Chatbot. This is where you are provided with drop-down access to any Chatbots you've been granted access to, as well as the UI where you type your messages, watch the agent work through your responses, and if enabled, rate and provide feedback for fine tuning the responses.
    - ***Column 3*** (Sources) – As the agent(s) research and construct your response, it may be pulling information from many different documents, videos, etc. It combines all that "related" data, and uses it to both construct its response, and to assign a confidence or relevance score (out of 100) as to how applicable the data found is to answer your query.
- ***Feedback*** – If feedback is enabled, it allows your users to either assign a number of stars (out of 5) or a thumbs up or down. Once given, it pops up a response box, which provides additional

review options for assigning an accuracy rating (out of 10), a “how detailed” rating (out of 19) and an open text box, where the user can propose or correct information about the response. This “corrective” information is then turned into a Suggestion, which is then reviewed, possibly edited and finally either Approved (turns it into a Fine-tuning rule), or Denied (not given to the LLM for correction).



Playbooks – coming soon

Agents – coming soon

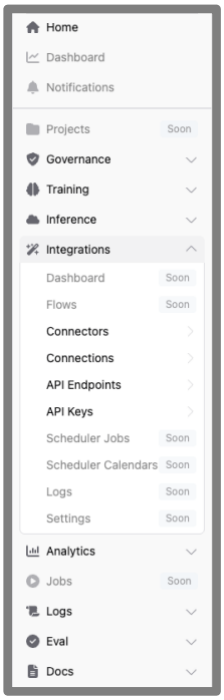
INTEGRATIONS

Summary

Data ingestion into an AI Data Lake typically happens in one of two ways. The first is used manually and by RAG methods, which are typically imported. This can be OCR, or other means, and can even be scheduled. For example, you could set up a RAG pipeline to weekly ingest all new Sales datasheets dumped into the Finance directory. The second method, typically used from application to AI, is via integration. This is mostly done through API endpoints. For example, you could set up an integration that pulls all Sales data from Salesforce, or all Helpdesk data from Zendesk once a day, once an hour, every 15 minutes, etc. These

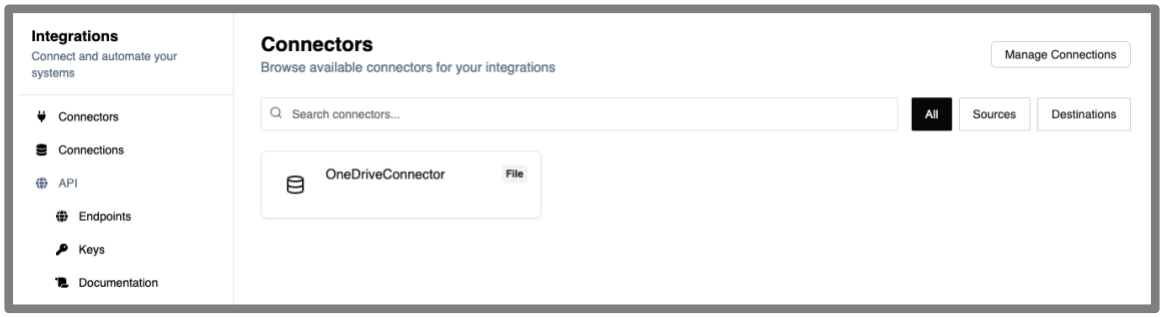
integrations can be bi-directional as well. For example, pull all Sales documents from the RAG pipeline weekly, but push all Sales data analysis into the corporate CRM.

API integrations can be very useful for those applications that communicate mostly via digital means. For example, all transcripts and communications from Teams, Slack, Twilio, One Note, etc.

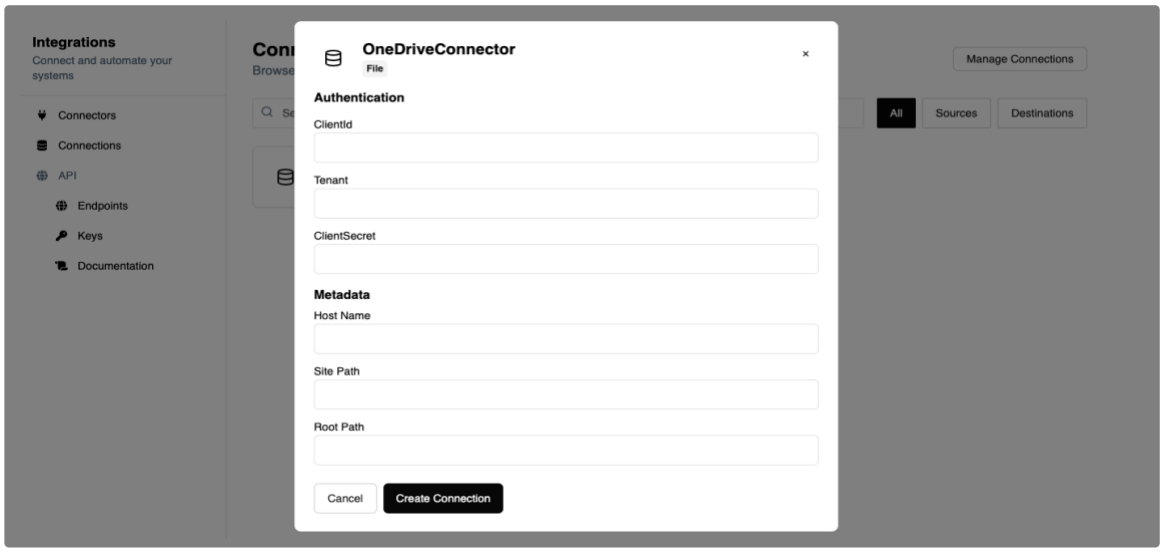


Connectors

- ***Connectors*** are pre-built plugins that allow OpenRAILs to communicate with the API endpoints of other applications. Over the last nearly 20 years, Clarity has developed (for client projects) more than 4,000 integrations. These integrations can provide the seed to port and create new connectors that can be used in OpenRAILs for live integrations to 3rd party applications. As Clarity develops and ports these connectors, they will be placed in the Connectors Marketplace (shown below).

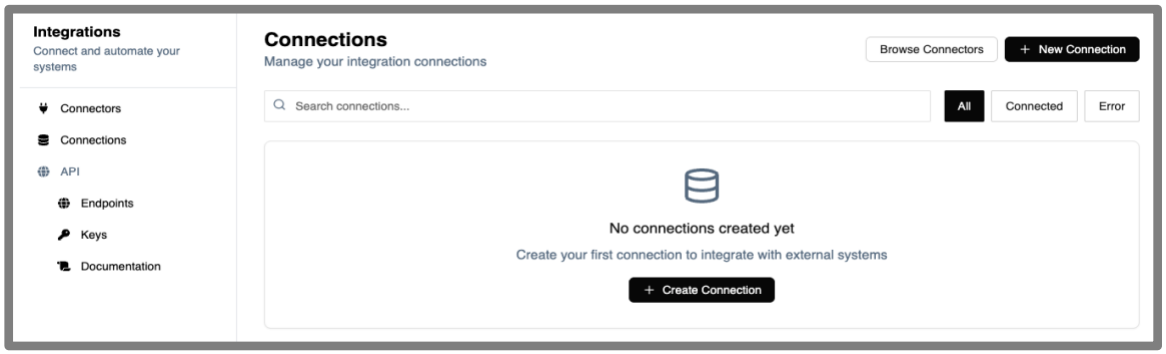


- Once you find a Connector that you want to use, simply click on the listing and it will bring up a configuration modal, where you can set up the connection information.



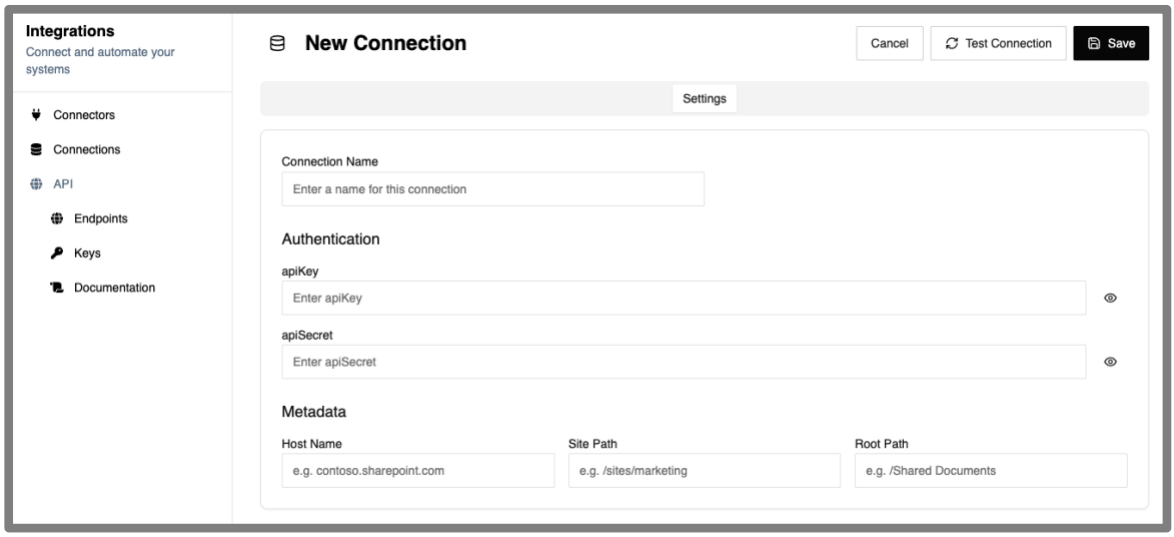
Connections

- Once a Connector has been configured and tested, the integration will be listed in the ***Connections*** screen.



- If an API credential is updated, or an IP address or hostname has changed, you can manage your connection settings here. Simply click on the Connector and it will bring up

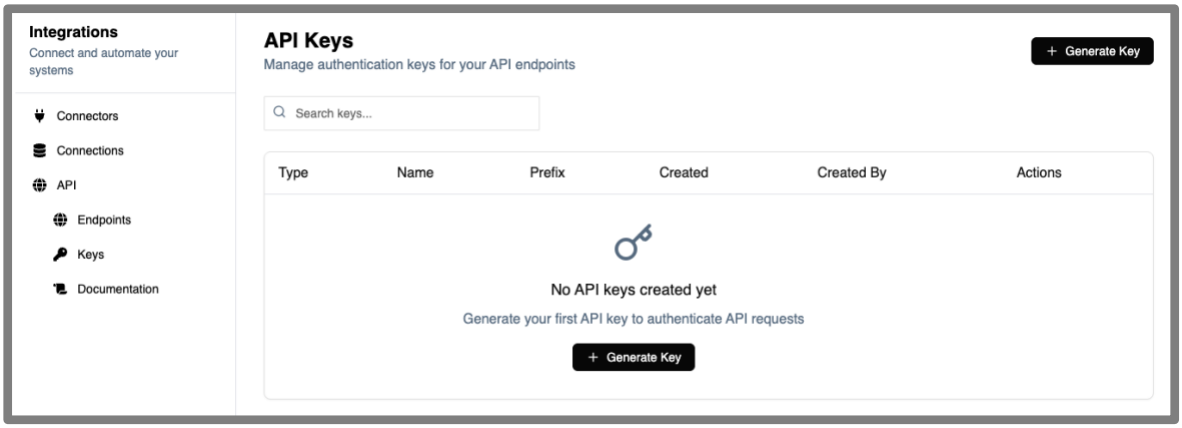
the edit screen (same as New Connection screen) where you can correct the connection information, paste in an API key, change the host access information, etc.



- You can Test your connection credentials here by clicking on the "Test Connection" button at the top of the screen.
- Once the connection has been validated, click the "Save" button to save the information permanently.

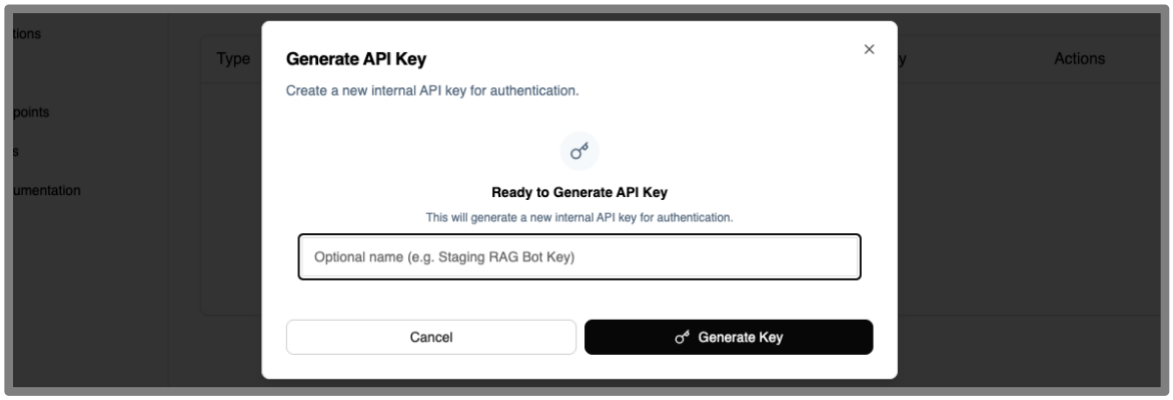
API Keys

- ***API Keys*** are the security tokens that 3rd party applications need to access OpenRAILs API endpoints to push and pull information from your OpenRAILs AI system. Without this "credential," they don't gain access to your system.

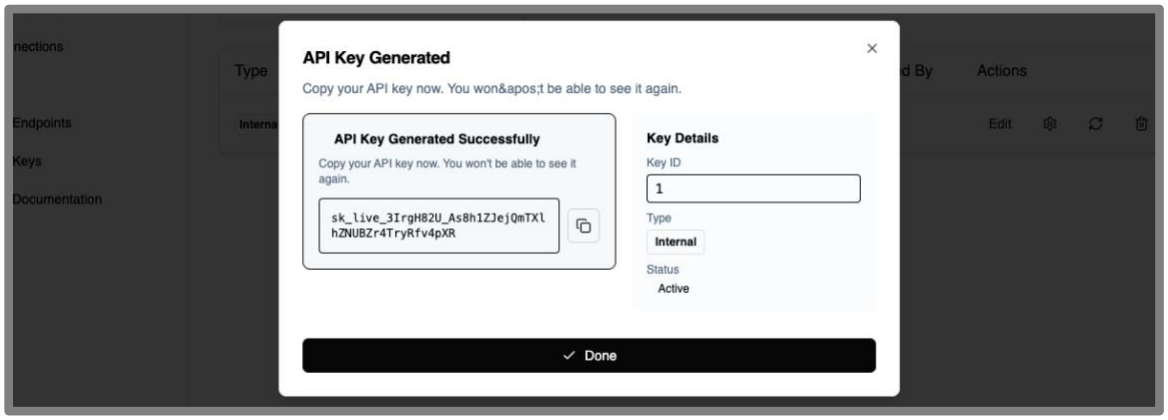


- You can create and manage access to existing endpoints by clicking on the Generate Key button.

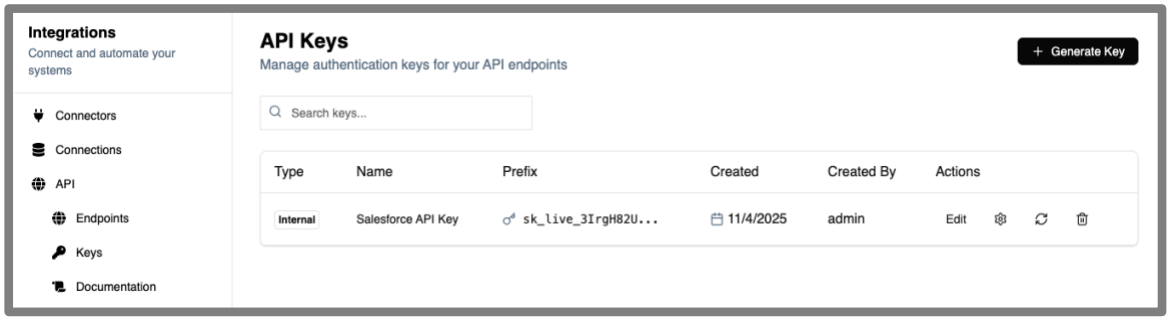
- ***Generate API Key*** – This option allows you to generate a private key that will be used by the 3rd party application that you want to grant access to your OpenRAILs system. To create a key, click the "Generate Key" button, provide a Name, so you can easily find it later, if needed, and click "Generate Key."



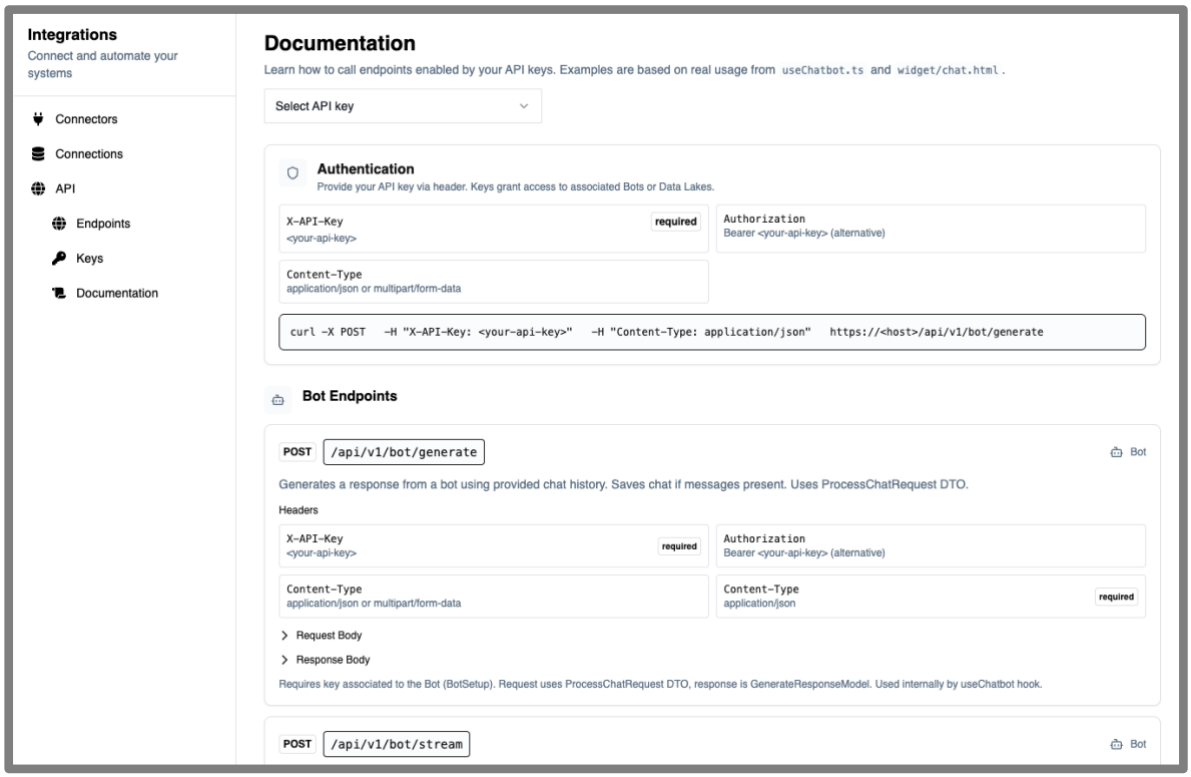
- Once the key has been generated, it will give you an opportunity to copy the key. This is your first opportunity, since it will be hidden going forward, so it can't be stolen or shared again.



- From this point forward, the key will be tokenized in the management view (you'll be able to see just the first few characters, so you can tell the 3rd party team that they should be using the "key starting with ABC," etc.



Documentation

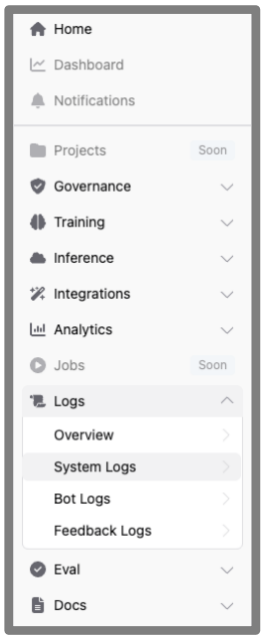
- ***Documentation*** - All available OpenRAILs APIs are documented here. This includes both POST and GET APIs, as well as documentation about access, authorization, connection type, and more.
  - 

LOGS

The logs menu provides access to search and review the queries and responses that have gone through the system. As it generates this information, many different agents perform many different tasks. These actions are recorded in the logs as well. Most actions recorded include the Source of the action, the timestamp of when it occurred, the logged message, and the Level of severity.

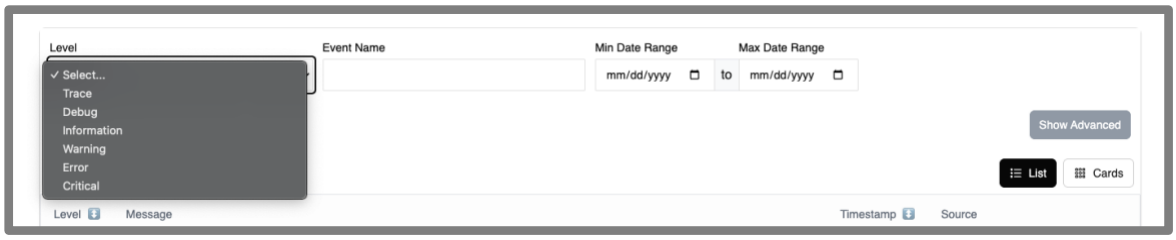
System Logs

- The ***System Logs*** dashboard provides access to all logged information generated by the systems, mainly the agents' actions. You navigate there by clicking on System Logs under the Logs main menu entry.

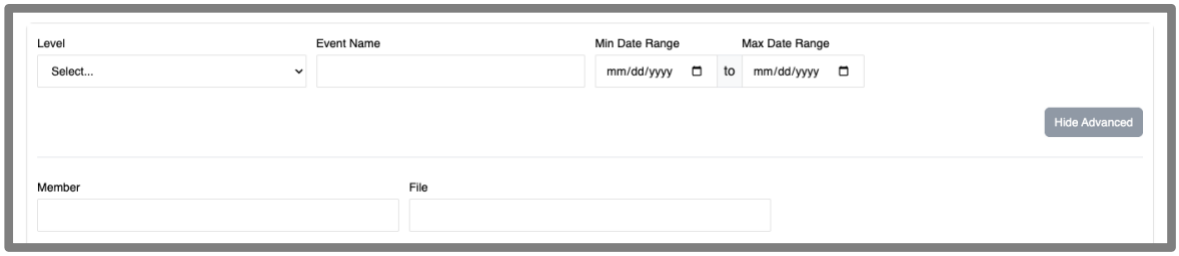


- The first piece of helpful information in the top-right of the dashboard is the display showing the total number of logs stored.
- There are two main ways to look through the logs for specific information. The first is to use the Search feature at the top of the dashboard.

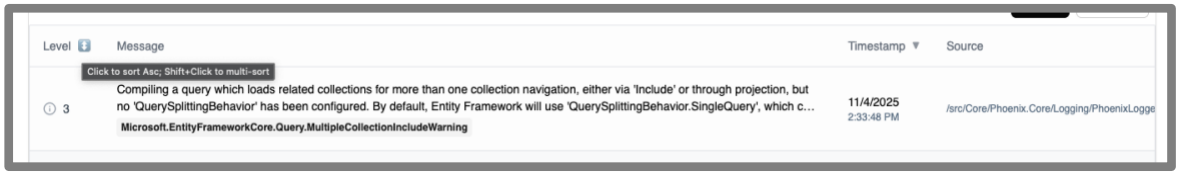
- The second method is to use any of the additional filters that have been provided. This includes filtering by log level (shown below), event name or date range.



- The Advanced filters (toggled on-off) include additional options and can, through custom development, be expanded for new filters that are custom or specific to your business.

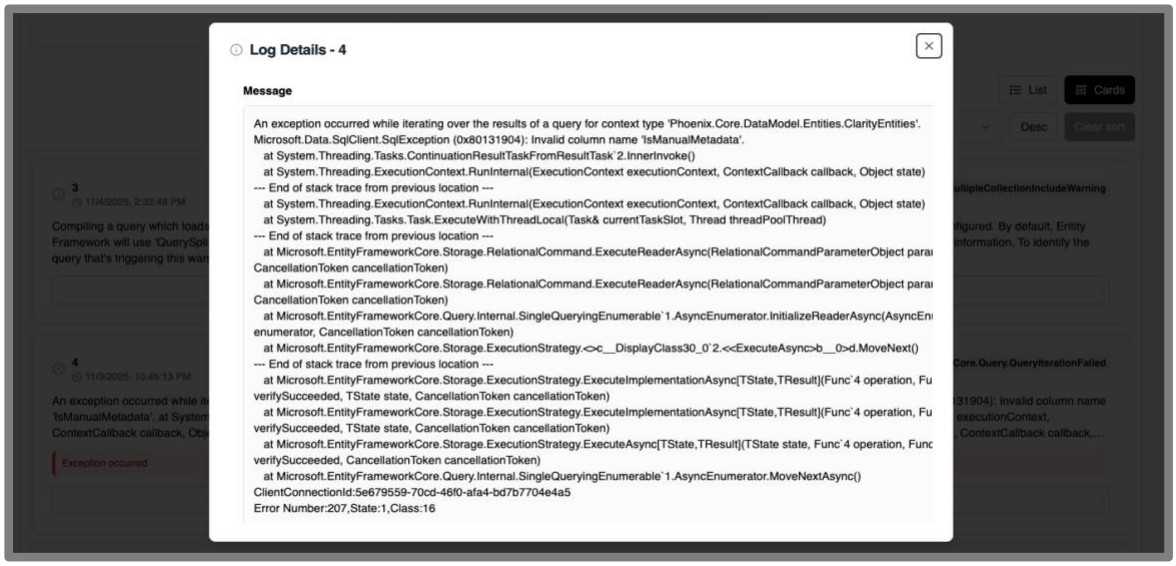


- One helpful feature is grid control. The header of the message table provides the ability, via single click, to Sort the logs by their Level or Timestamp in ascending or descending order, by clicking on the blue double-arrow. This is helpful as you can quickly move the most severe items to the top, or possibly the oldest or most recent entries to the top of the grid.



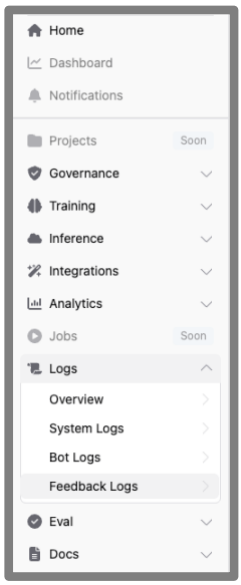
- If you want a different grid view, that provides a better view to quickly see severe items, and a link to drill into more details, then use the Cards grid view. This view (shown below) colorizes the more severe errors and provides a ***View Details*** link allowing you to gain access to all information regarding that log entry. The top-right controls also allow you to select either sorting by timestamp or level, as well as ascending or descending.

- Clicking on ***View Details*** brings up the entirety of the log entry. In the image below, we're viewing the log entry for the Exception shown in the image above.



Feedback Logs

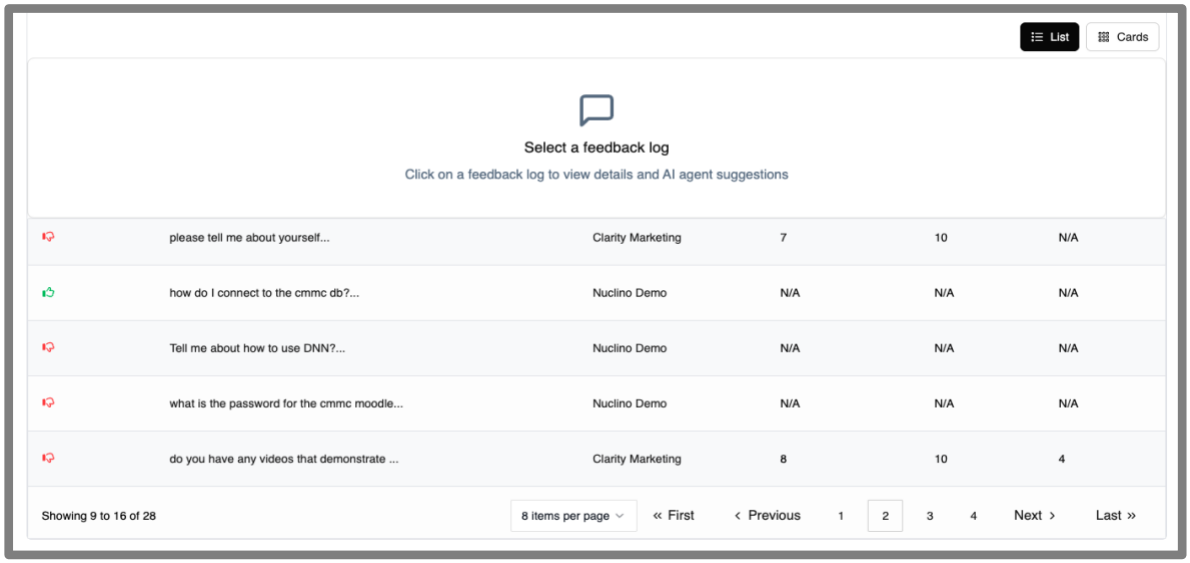
- ***Feedback logs*** are where all the feedback actions taken by chatbot sessions are stored.



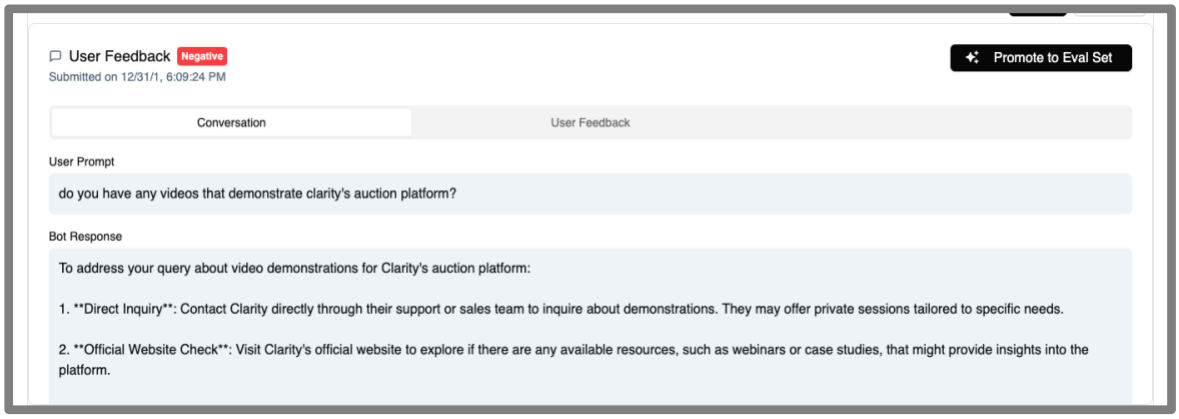
- This dashboard provides your organization with a UI to filter, review, and promote user feedback into fine-tuning rules that improve the accuracy of the bots' responses. The filters allow you to filter either positive or negative reviews and to select a bot, if you're only working on a single bot's responses.

- The ***Show Advanced*** filters give more granular filtering by minimum and maximum Accuracy, Detail or Stars given.

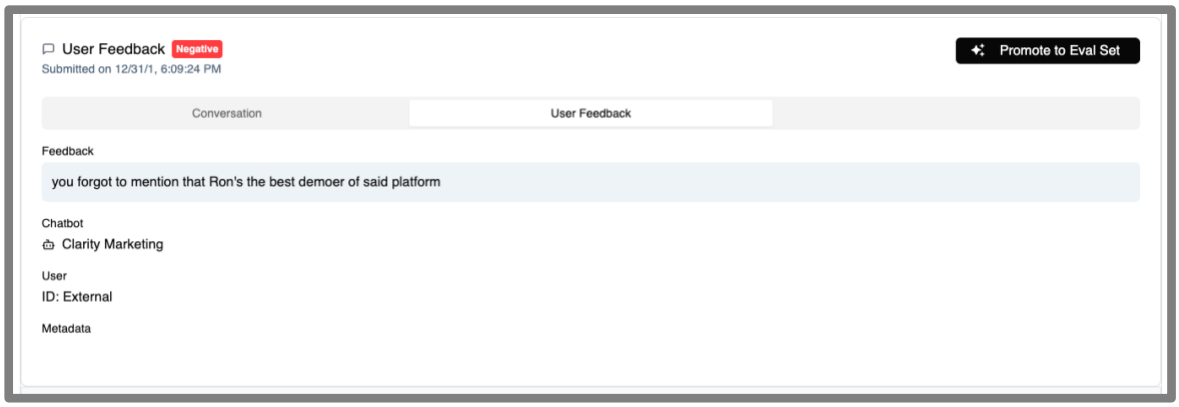
- ***List View*** – The list view shows a quick way to visually see, by color and icon, which responses are positive or negative, a shortened snippet of the users' query, the accuracy and detail score, and if stars were used, the number of stars given. You can click on any entry to drill into the details



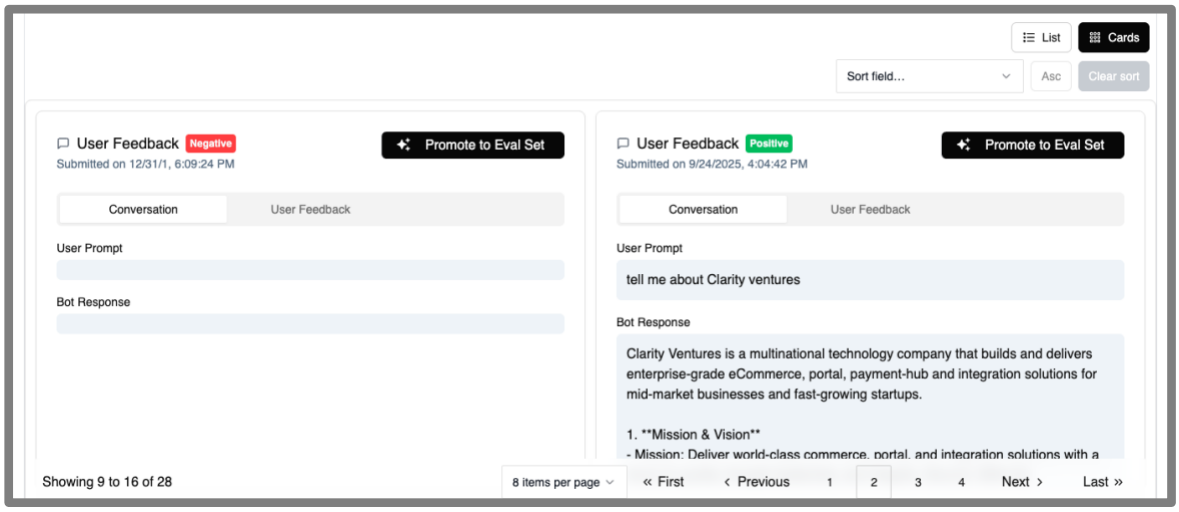
- Clicking on an entry, expands that entry to provide complete information regarding the ***Conversation***.



- You can also click on the ***User Feedback*** tab to view all of the feedback. ***Promote to Eval Set*** is how your team can "approve" this information to become a fine-tuning, corrective rule that will be applied to all future AI responses.



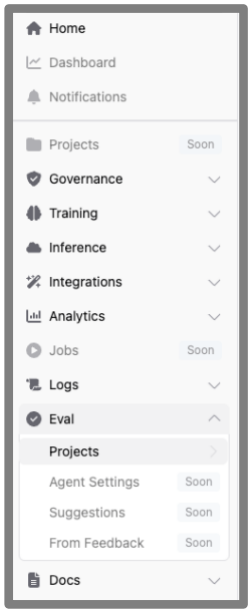
- The ***Cards*** view provides a UI with more visible information, along with controls to scroll and tab so that you don't have to drill into or leave this page to work through the feedback responses.



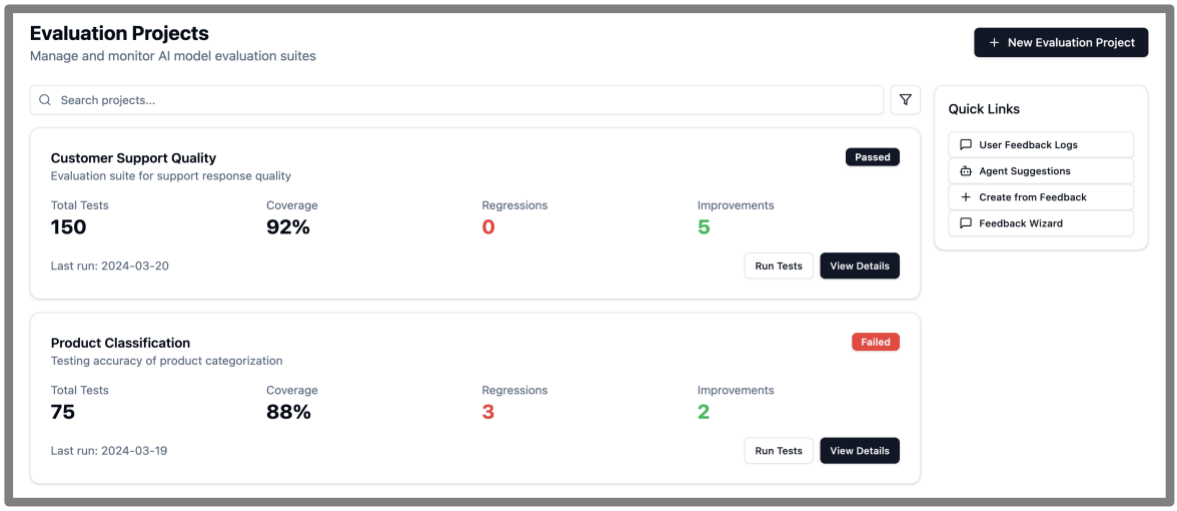
EVAL

Projects

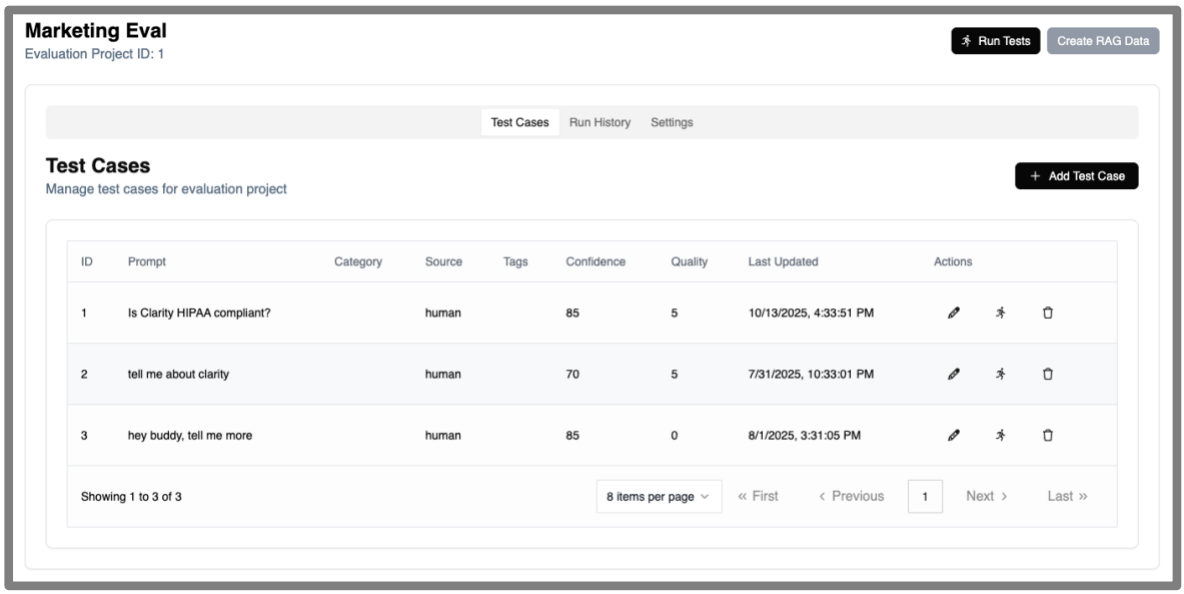
- Evaluation ***Projects*** are set up to run the rules that have been created and approved via all user feedback. These rules can be run many times as new data and feedback is approved. The result is a scoring system to help you measure the accuracy of the data, and to see if your new rules are improving the AI responses, or if there are regressions in the accuracy of the responses. This is normally due to new data being imported that changes or skews the data so that the responses are less accurate than before.



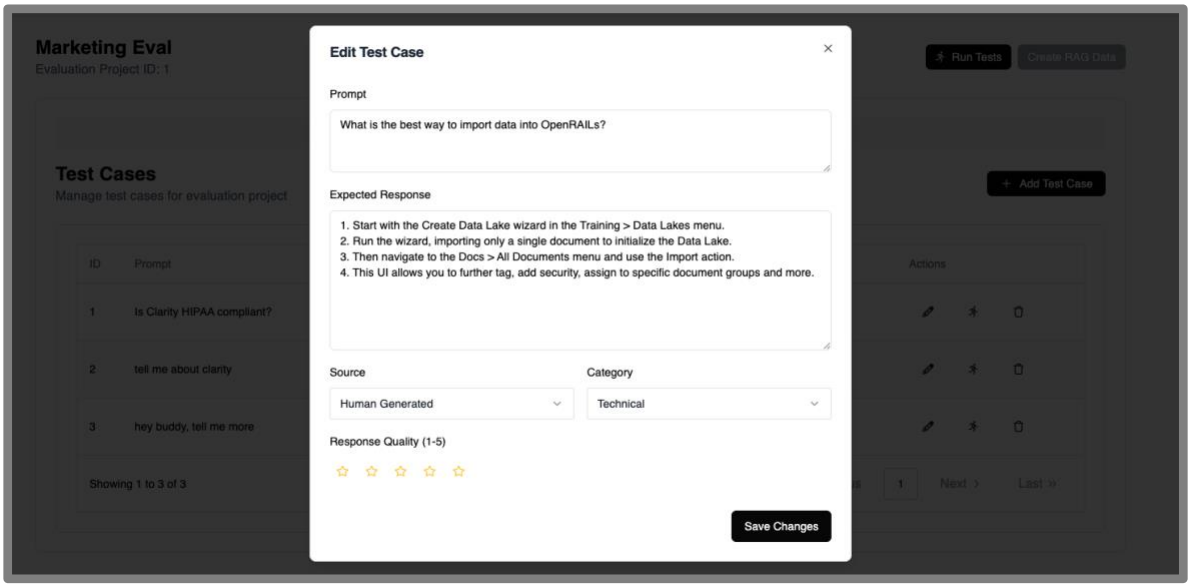
- ***Evaluation Projects*** – The dashboard provides a visual report of all your projects, which we call "Test Cases." Each AI project would normally have their own set of Eval tests to run and correct the responses for that Data Lake or AI project.



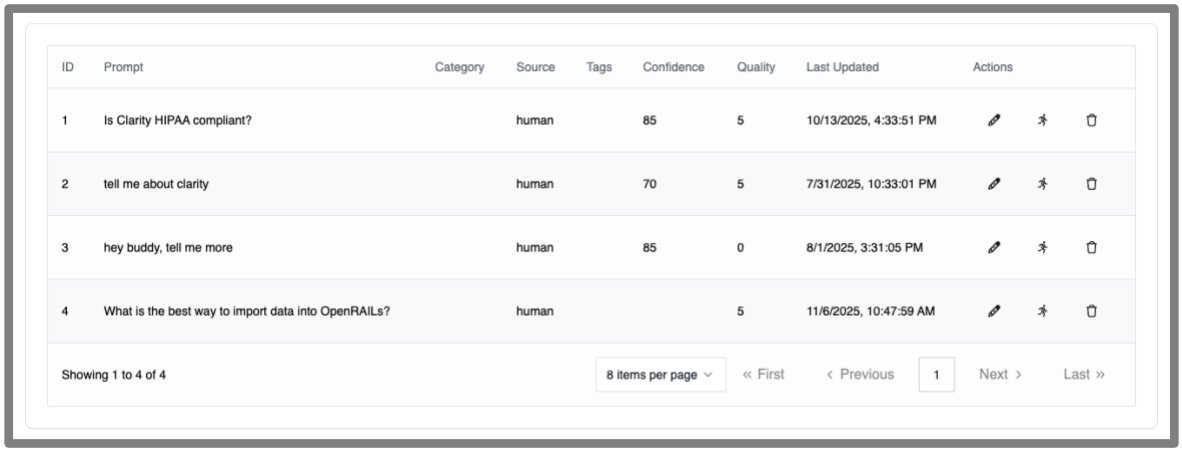
- ***Test Cases*** – If you drill into one of the entries on the dashboard, you will navigate to the Test Case details view, where you can see the individual fine-tuning rules or tests included in this Case.



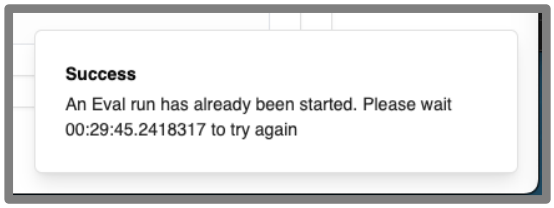
- ***Add Test Case*** – To manually add a corrective case (fine-tuning rule), click on Add Test Case. This modal allows you to enter singular, corrective cases without having to wait for user feedback and review.



- Once you Save the Test Case, it will be added to your Master Test Case and included in the test runs.

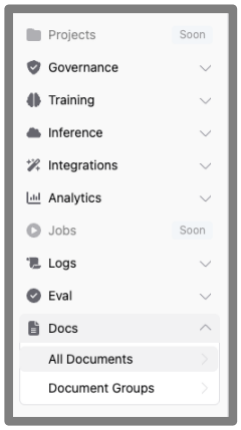


- ***Run Tests*** – These tests run very quickly. Whichever screen you execute from, as soon as the test(s) have run successfully, there will be a pop-up message in the bottom-right of the screen letting you know the outcome of the test run. In the image below, we were too slow to grab the screenshot, so clicking a second time lets me know that the status is Success, but that it's already been started and running in the background, so check back later.



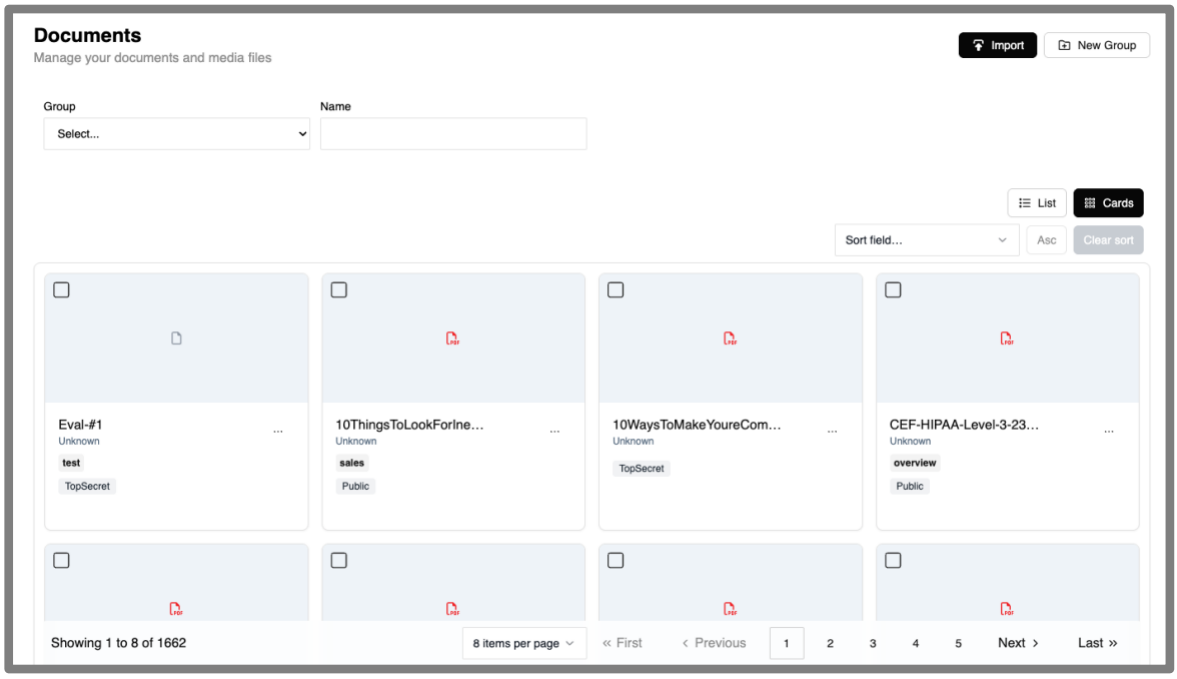
DOCS

OpenRAILs provides a master library of all content ingested through RAG pipelines, integrations or manual imports. The options under the Docs menu provide the tools to manage your document repository.



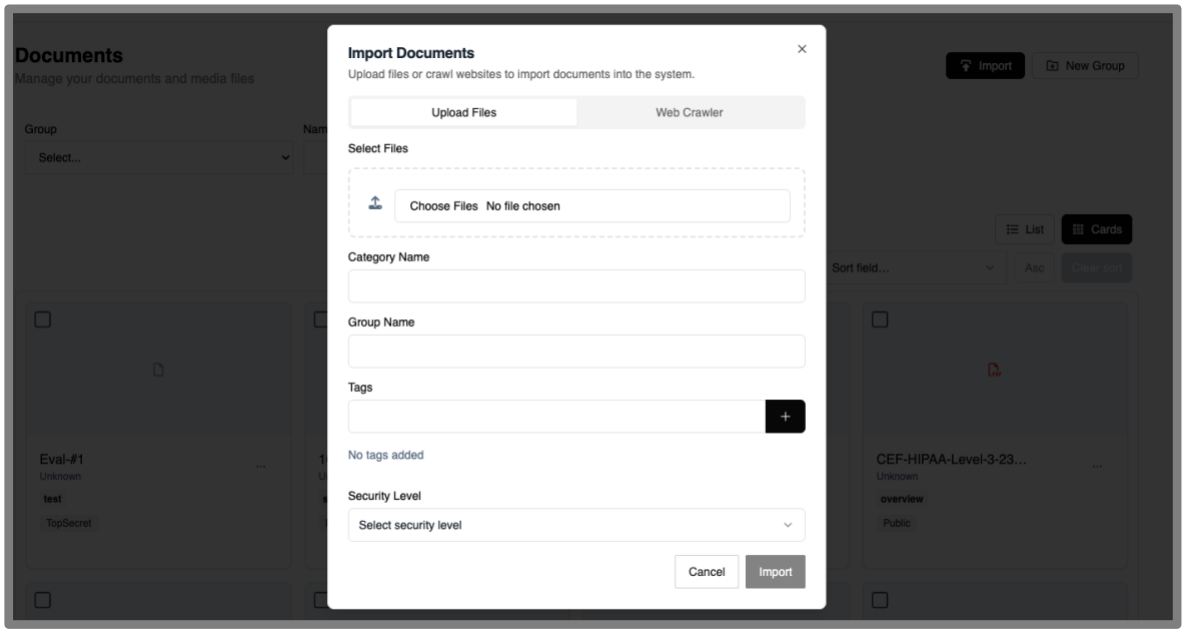
All Documents

- ***All Documents*** links you to the Document Management dashboard.



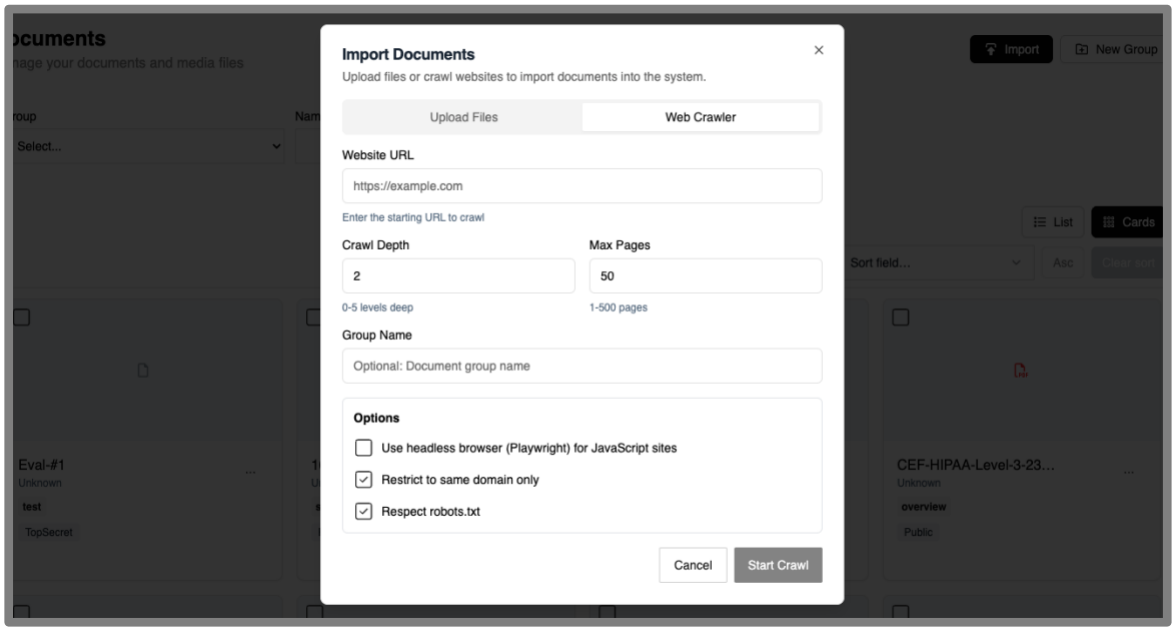
- Top-level navigation – At the top of the dashboard, you're provided with several controls. These include a group selection control, name search, import, create a new group, sorting and grid view options of ***List*** or ***Cards***.

- ***Import*** – This master import action allows you to initially bypass ingesting directly into a Data Lake, directly into the master document repository. Some benefits of this are you may want to use some of the content imported in more than one data lake. For example, all product data should be available both in the corporate website chatbot, as well as the customer support chatbot.

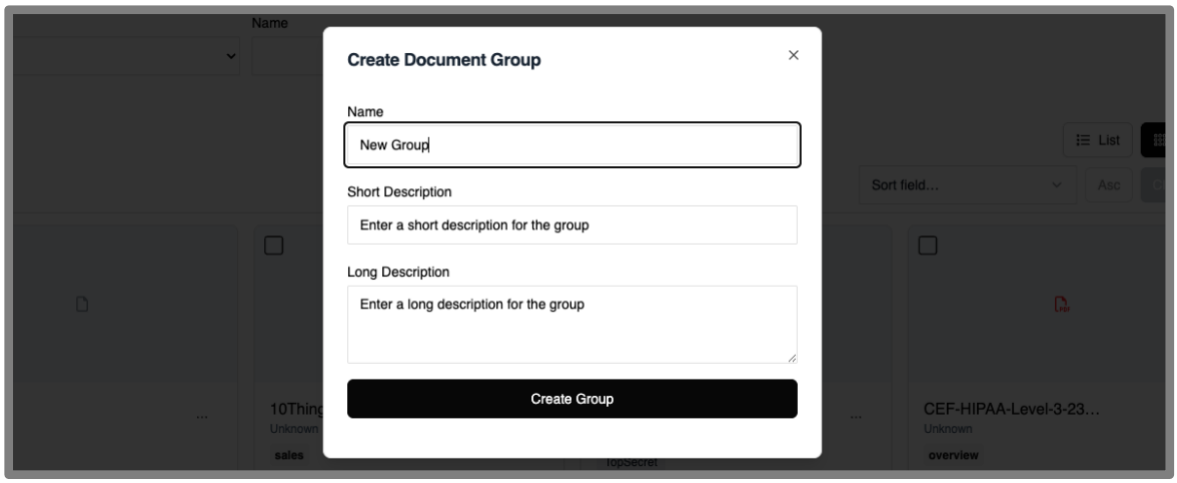


  - The import modal provides you with the ability to drag and drop files for import, assign a category name, group name, custom tags, and a security level. So, per the above example, we may want to create a group for "Product Information" that could be used by both corporate and support chatbots.
  - ***Web Crawler*** – Another option to import both web content and linked documents is to crawl a website. This UI allows you to specify the URL of the site to be crawled, the crawl

depth (how many links do I traverse from any given page crawled?), the maximum number of pages to crawl, the group name to assign the content to, and some more in-depth options regarding the crawler behavior.



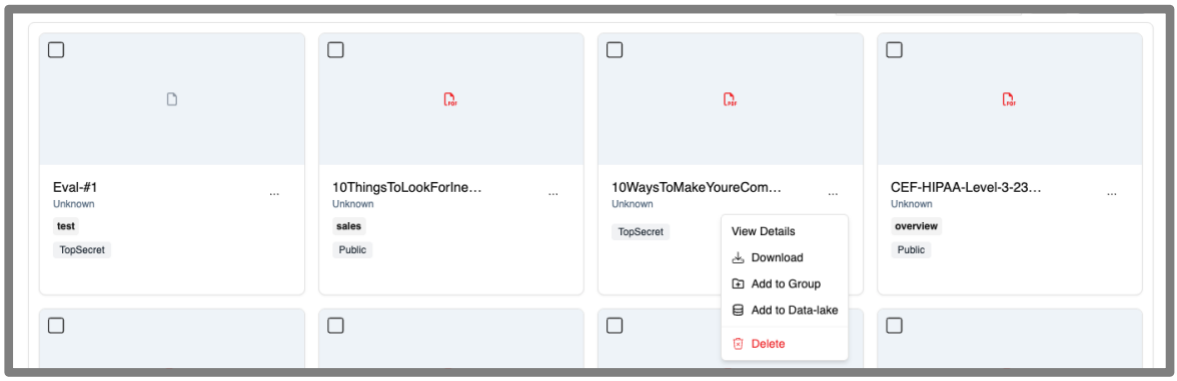
- ***New Group*** – Clicking on New Group launches the Create Document Group modal.



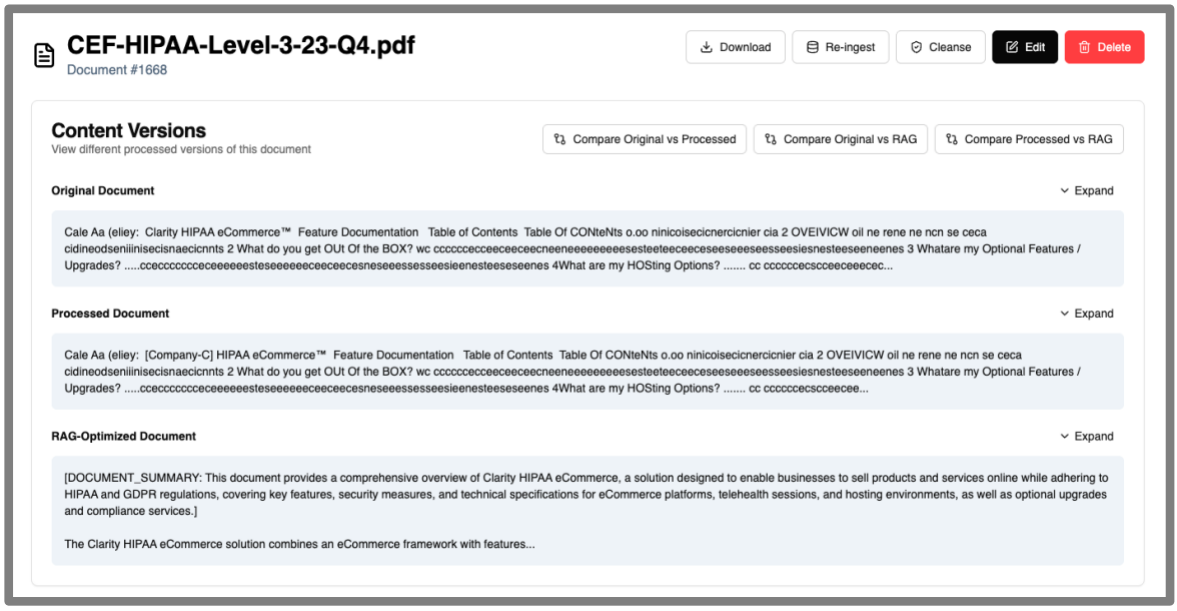
  - A group is just an organizational element that content can be assigned to. In our example above, we wanted to use product-ingested content for both corporate and customer support chatbots. The way to accomplish this is to create a document group for “product information.” Then we import all white papers, videos, web pages, etc. regarding our products and assign them to that group on ingestion. Then we go to each Data Lake and choose the ***Add Group*** action to add the

product information to both Data Lakes, making the information available to both chatbots.

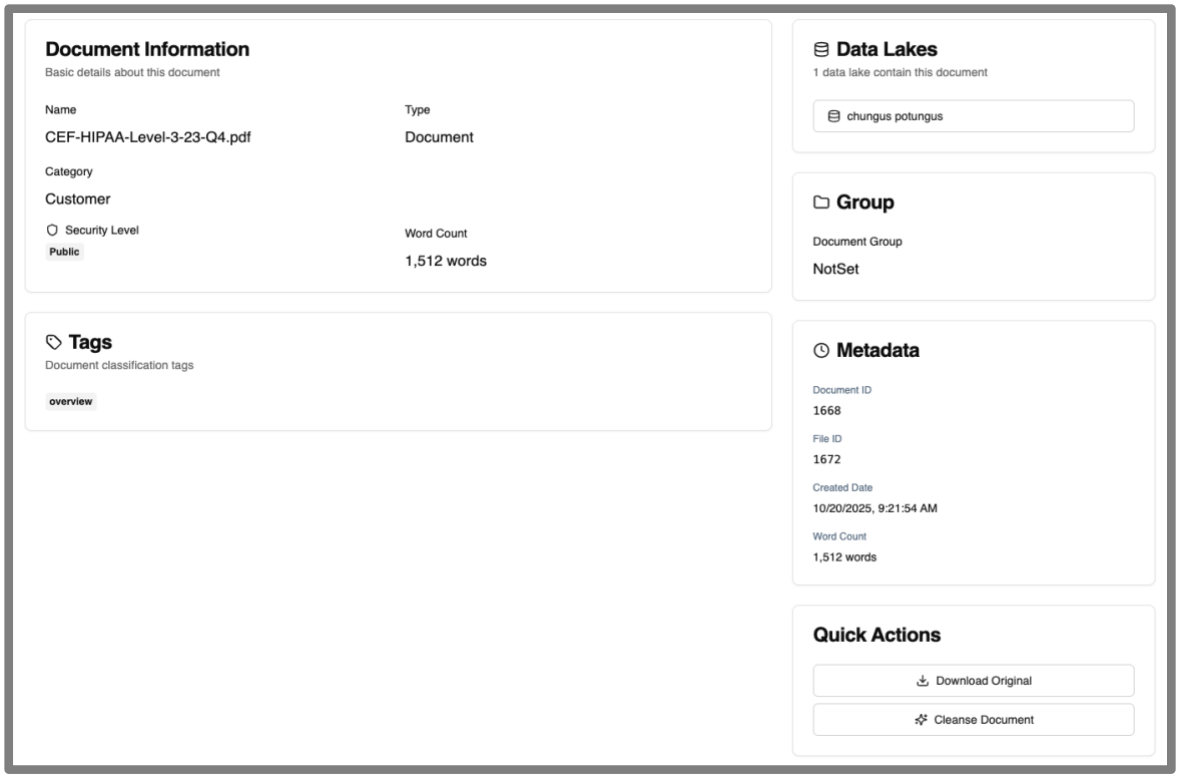
- Document Management – On the Documents dashboard, whether in the List or Cards view, there are actions you can take on each document. Clicking on the ellipses, you are presented with the actions available, which include viewing the details of the document, downloading a copy, adding the document to a group or data lake, and finally deleting the document from the repository.



- Document Details – Clicking on ***View Details*** from the actions takes you into a single document management view. The top half of the dashboard provides basic actions for downloading, re-ingesting, cleansing, editing, and deleting the document.
- ***Content Versions*** – The Content Versions panel allows you to view and compare the three versions of the ingested content, comparing the original document content with the first ingested content (Processed), and the final version, which has been summarized and optimized for AI analysis and retrieval (RAG). You can use the expand and collapse to view all the content in each version or use the buttons to bring up a side-by-side model comparing the two versions selected.

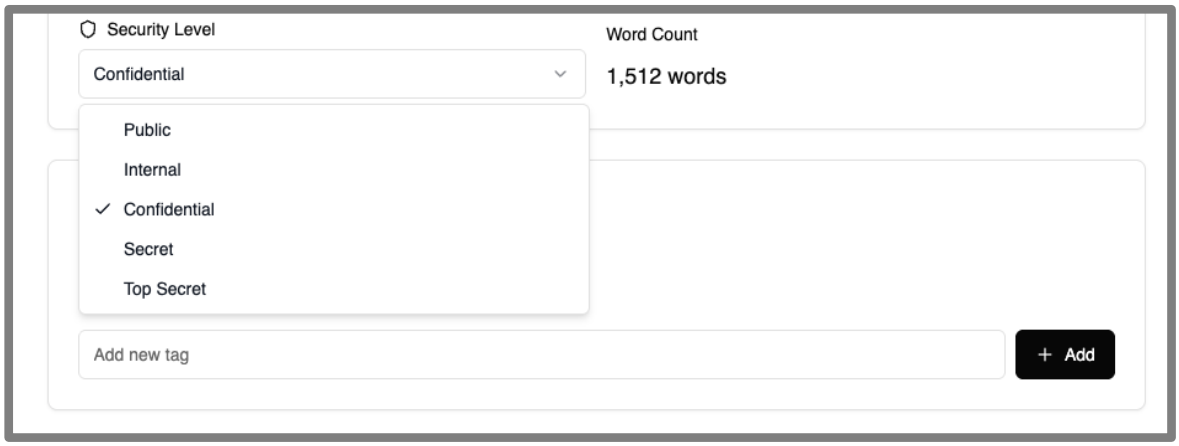


- Bottom-half of Page – The lower section of the dashboard provides meta, security, tagging, group and data lake assignments and other information about the ingested document, as well as some quick action links to download or cleanse the document.



- ***Edit*** Document – Clicking on ***Edit*** enables you to update the security level of the document, add additional document tags, and remove or edit the assigned groups the

document belongs to. Once complete, click the ***Save Changes*** button at the top of the dashboard.

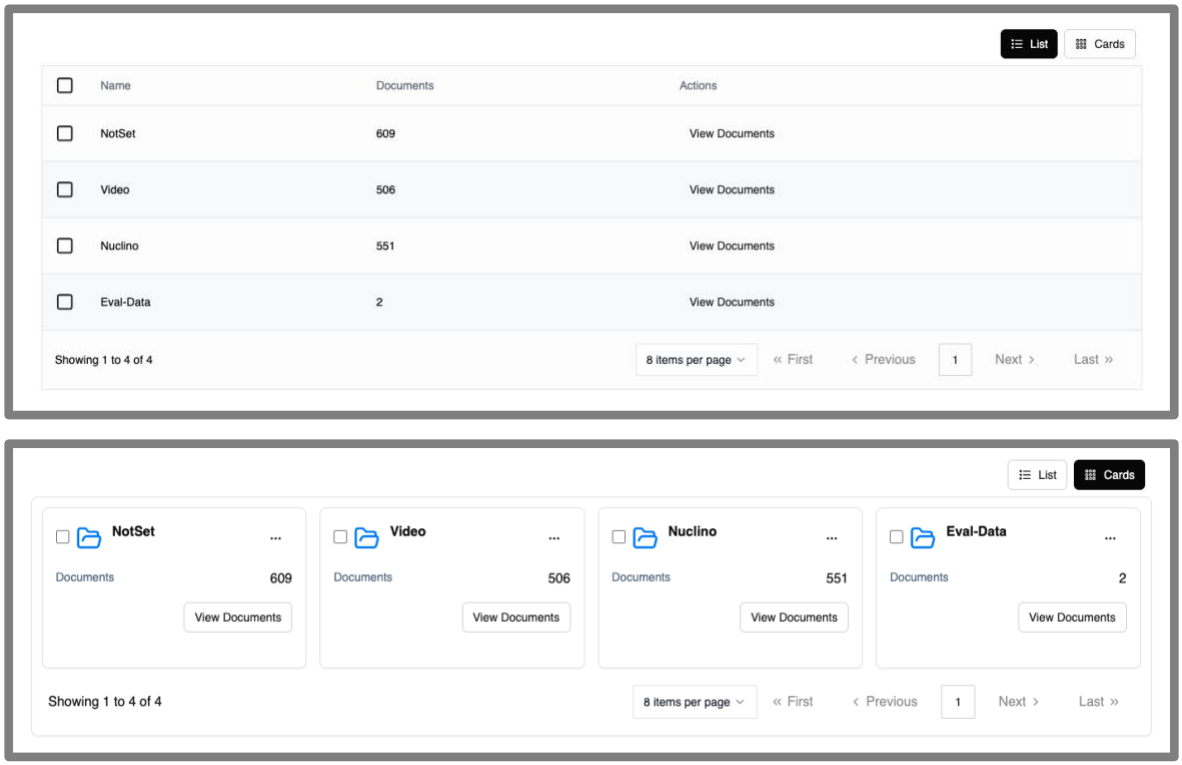


Document Groups

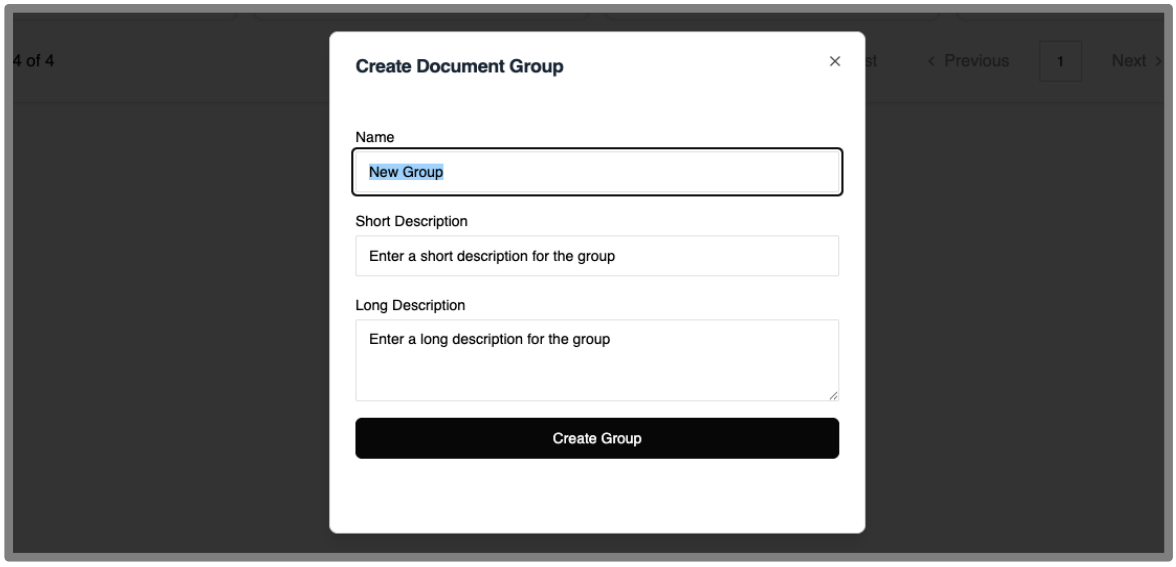
Document Groups behave much like a file manager or Windows explorer. It allows you to create groups (like folders), then view and add content to the group.

- Navigation / Controls – The top-half of the dashboard provides all the controls that are available, such as creating a new group, searching for a group and changing the viewing from ***List*** to ***Cards*** view.

- Grid View – The bottom part of the dashboard is the group listings. Whether in the ***List*** or ***Cards*** view, you are presented with the name of each group, the number of documents in each group, and a link to view the details (***View Documents***) of the group's contents.



- ***New Group*** – To create a group, simply click on the New Group button at the top of the dashboard. A modal will prompt you to give the group a name, with optional short and long descriptions, to help organize your data and communicate with your team about what the intent and purpose of the group should be.

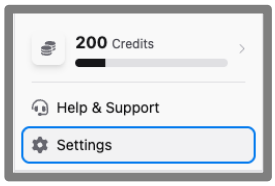


- Once your group has been created, content can be ingested into that group, and the group can be added to a Data Lake for consumption via an AI agent, chatbot, etc.

SETTINGS

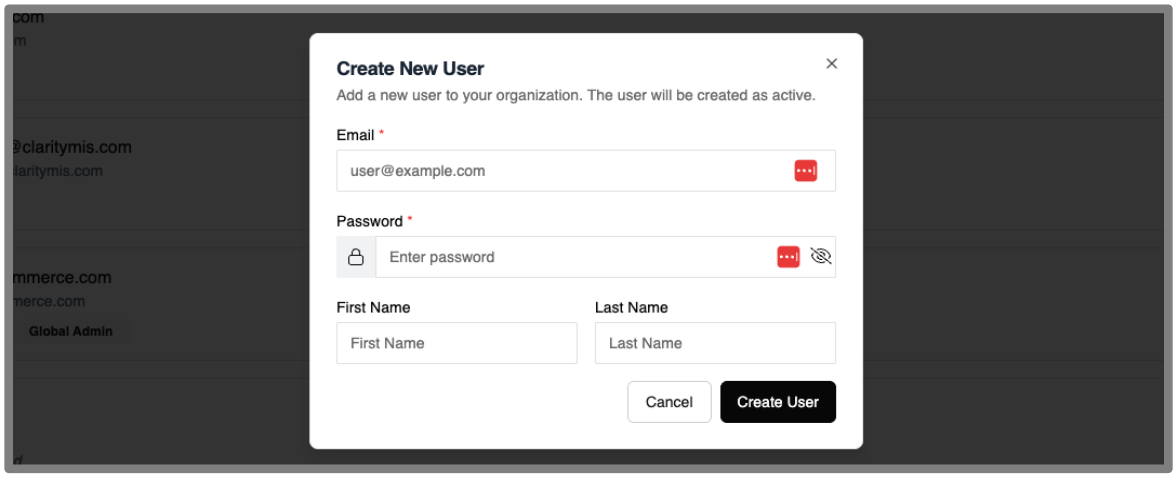
Users

Users are the entity assigned credentials that can log into OpenRAILs and engage with the system, agents, chatbots, etc. via the permissions granted to a user account via Roles.



- ***Users*** tab – The Settings dashboard has two tabs. The first is the Users tab, where we see how many users have been created, create and add new users to the system, activate and deactivate user access (Active toggle), delete user accounts (trash icon), as well as change user passwords (key icon).

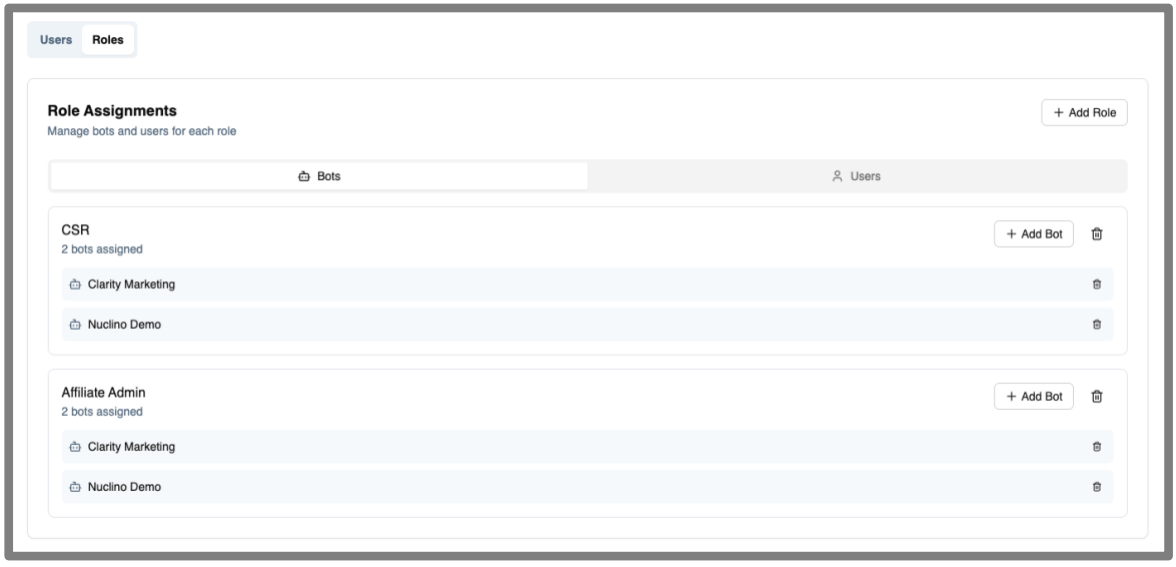
- ***Add User*** – Clicking the Add User button launches the Create New User modal, where you enter their email address, password and first and last name.



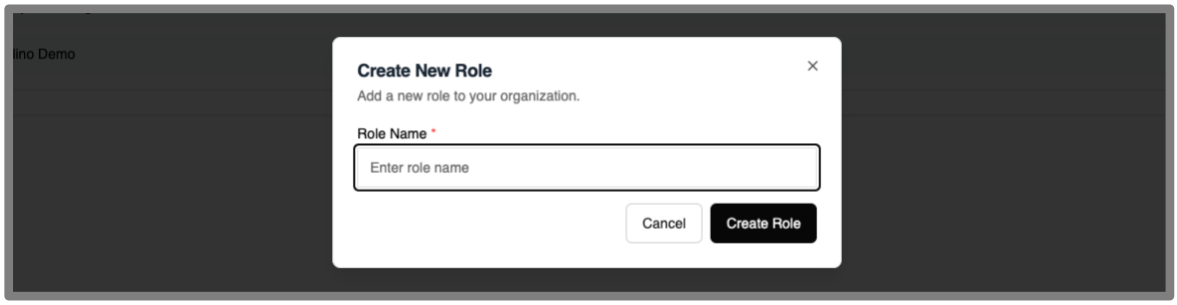
Roles

Roles are the entities or containers that we use to grant Users and Bots access to the system.

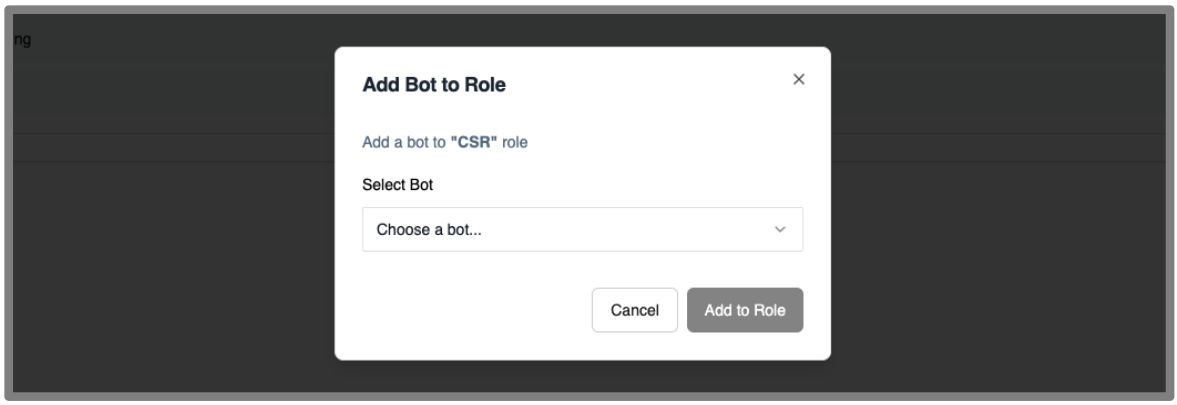
- ***Roles*** tab – The ***Roles*** tab is where we can create new Roles, view created roles, see any bots or user accounts that have been assigned a Role, add bot and user access and finally remove access to and from Roles. For example, if we had three chatbots for our organization (Sales, Support, Executive), and wanted to control access, then when we create a User account for the CEO, we would add that User account to the Executive Role. Then we would grant access for all 3 bots to that role, making them available to any User logging in with Executive "access."



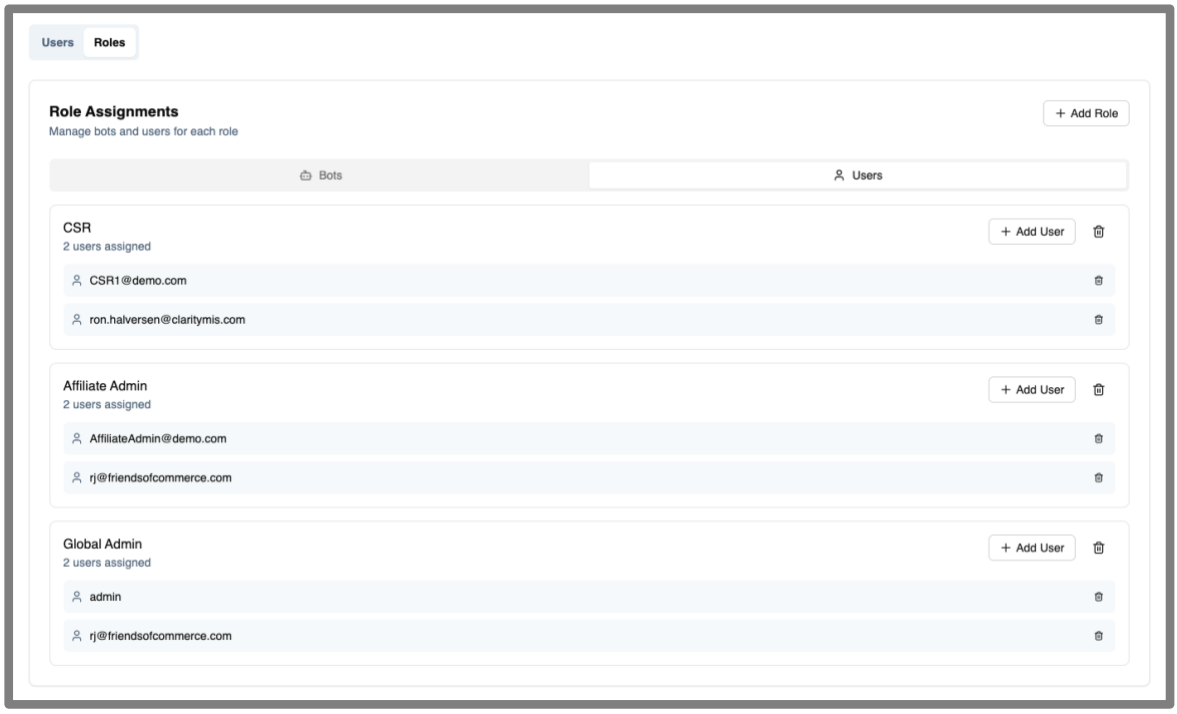
- ***Add Role*** – Creating a role is simple. Click the ***Add Role*** button and give it a name.



- ***Add Bot*** – As with our example above, granting access to a bot is done through Role assignment. Click on ***Add Bot***, which launches the ***Add Bot to Role*** modal pop-up, click the drop-down and add the bot that you want to grant access via this role.



- ***Users*** – The ***Users*** tab allows us to grant User access to a Role. The view provides you with the option to create a new role, view all roles that are created, see the Users that have been granted access to each role, add users to a role, remove users from a role, and delete the role.



- ***Add User*** – Clicking the ***Add User*** button launches the ***Add User to Role*** modal pop-up. Simply click the drop-down to see all available users that can be added to this role and select it. Once a user has been added to a role, that user will no longer show up in the drop-down, since it's not available to add.