



概述

Casibase 是一个由 ChatGPT 驱动的开源 领域知识 数据库、即时通讯和论坛软件。

You need to enable JavaScript to run this app.

Casibase 特点

1. 采用 Golang 开发的前后端分离架构，Casibase 支持高并发，提供基于 Web 的管

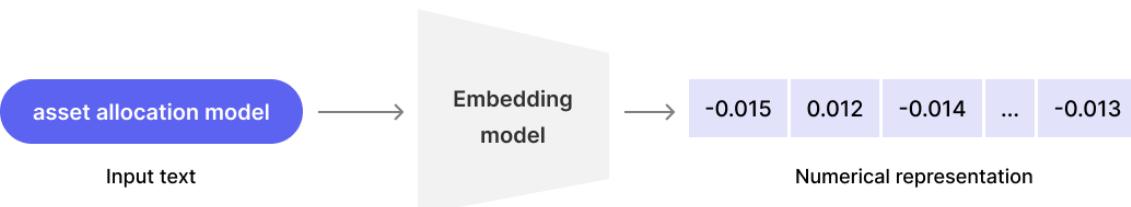
理界面，并支持多语言（中文、英文）。

2. Casibase 支持第三方应用登录，如 GitHub、Google、QQ、微信等，并支持通过插件扩展第三方登录。
3. 基于嵌入和提示工程进行知识管理，Casibase 支持自定义嵌入方法和语言模型。
4. Casibase 通过数据库同步支持与现有系统集成，使用户可以平滑过渡到 Casibase。
5. Casibase 支持主流数据库：MySQL、PostgreSQL、SQL Server 等，并支持通过插件扩展新的数据库。
6. Casibase 是一个强大的资产管理工具，可以通过 RDP、VNC 和 SSH 协议轻松连接资产，并高效处理机器的远程连接。
7. Casibase 的安全日志审计功能允许你轻松跟踪和监控远程连接，详细记录连接开始时间、持续时间和其他相关详情，同时还能捕获和分析 Casdoor 操作的 API 日志，增强安全性和操作透明度。
8. Casibase 支持数据库管理。Casibase 支持数据库管理。Casibase 的数据库管理功能允许你轻松连接、管理和组织数据库，同时控制访问权限，简化数据库资源的用户管理和授权。
9. Casibase is an open-source container cloud platform based on Docker and Kubernetes. It is suitable for individuals or organizations to build their own dedicated container cloud environment. Based on the Casbin permission management engine, Casibase implements fine-grained access control policies. Users can easily create, orchestrate, and manage container applications on Casibase. The project focuses on optimizing Casibase's application orchestration, service governance, and platform visualization core functions, improving platform usability and manageability, making it a leading lightweight container cloud platform.

工作原理

步骤 0（预备知识）

Casibase 的知识检索过程基于嵌入和提示工程，因此强烈建议您简要了解嵌入如何工作。嵌入的 [简介](#)。



步骤 1（导入知识）

要开始使用 Casibase，用户需要按照以下步骤导入知识并创建特定领域的知识数据库：

- 配置存储：**在 Casibase 仪表板中，用户首先应该配置存储设置。这涉及指定用于存储知识相关文件（如文档、图像或任何其他相关数据）的存储系统。用户可以根据其偏好和需求从多种存储选项中选择。这涉及指定用于存储知识相关文件（例如文档、图像或其它相关数据）的存储系统。用户可以根据他们的偏好和需求从多种存储选项中进行选择。
- 上传文件到存储：**一旦设置好存储，用户就可以将包含特定领域知识的文件上传到配置好的存储系统中。这些文件可以是各种格式，如文本文档、图像或结构化数据文件（如 CSV 或 JSON）。这些文件可以采用各种格式，例如文本文档、图像，或 CSV、JSON 等结构化数据文件。
- 选择知识生成的嵌入方法：**文件上传后，用户可以选择用于生成知识和相应向量的

嵌入方法。嵌入是文本或视觉内容的数值表示，有助于高效的相似度搜索和数据分析。嵌入是文本或视觉内容的数字表示，有助于高效的相似性搜索和数据分析。

💡 提示

知识是如何嵌入的？

- 对于文本数据：用户可以选择各种嵌入方法，如 Word2Vec、GloVe 或 BERT，将文本知识转换为有意义的向量。
- 对于视觉数据：如果上传的文件包含图像或视觉内容，用户可以选择基于 CNN 的特征提取等图像嵌入技术来创建代表性向量。
- 更多方法即将推出...

通过遵循这些步骤，用户可以用相关信息和相应的嵌入来填充他们的领域知识数据库，这些将用于在 Casibase 中进行有效的搜索、聚类和知识检索。嵌入过程使系统能够理解不同知识片段之间的上下文和关系，实现更高效和有见地的知识管理和探索。嵌入过程使系统能够理解不同知识之间的上下文和关系，从而实现更高效和富有洞察力的知识管理与探索。

步骤 2（检索知识）

在导入你的 领域知识 后，Casibase 将其转换为 向量 并将这些向量存储在 向量数据库 中。这种向量表示启用了强大的功能，如 相似度搜索 和 相关信息的高效检索。你可以基于上下文或内容快速找到相关数据，实现高级查询并在你的领域知识中发现有价值的见解。这种向量表示使得诸如 相似性搜索 和 高效检索相关信息 之类的强大功能成为可能。您可以根据上下文或内容迅速找到相关数据，从而实现高级查询并在您的领域知识中发现有价值的洞见。

步骤 3（构建提示）

Casibase 对存储的知识向量执行相似度搜索，以找到与用户查询最接近的匹配。使用搜索结果，它为 **语言模型** 创建一个 **提示模板** 来构建特定问题。这确保了基于 Casibase 中的领域知识提供准确和上下文相关的响应。利用搜索结果，它创建了一个 **提示模板** 来构建针对 **语言模型** 的具体问题。这确保了准确且符合上下文的响应，基于 Casibase 的领域知识提供全面答案。

步骤 4（实现目标）

在此阶段，通过使用 Casibase，您已成功获取所需的知识。在这个阶段，使用 Casibase，你已经成功获取了所需的知识。通过创新地将领域知识转换为向量并结合 ChatGPT 等强大的语言模型，Casibase 为你的查询提供准确和相关的响应。这使你能够高效地访问和利用存储在 Casibase 中的特定领域信息，轻松满足你的知识需求。这使您能高效访问和利用存储在 Casibase 中的特定领域信息，从容满足您的知识需求。

步骤 5（可选的微调）

如果你发现结果不完全令人满意，你可以通过以下方式尝试获得更好的结果：

- 调整语言模型参数
- 提出多个问题
- 优化原始文件

通过利用这些微调选项，你可以提高在 Casibase 中的知识管理效率，确保系统更好地与你的目标保持一致，并提供更准确和有见地的信息。



提示

其他优化结果的方法（可能需要源代码更改）：

- 更新 `嵌入` 结果：通过调整领域知识的嵌入结果来改进知识表示。
- 修改 `提示` 模板：通过自定义提示，你可以从语言模型获得更精确的响应。
- 探索不同的 `语言模型`：尝试不同的模型，找到最适合你的响应生成需求的模型。

在线演示

只读站点（任何修改操作都会失败）

- 聊天机器人 (<https://ai.casibase.com>)
- 管理界面 (<https://ai-admin.casibase.com>)

可写站点（原始数据每 5 分钟恢复一次）

- 聊天机器人 (<https://demo.casibase.com>)
- 管理界面 (<https://demo-admin.casibase.com>)

全局管理员登录：

- 用户名：`admin`
- 密码：`123`

架构

Casibase 包含 2 个部分：

名称	描述	语言	源代码
前端	Casibase 应用程序的用户界面	JavaScript + React	https://github.com/casibase/casibase/tree/master/web
后端	Casibase 的服务器端逻辑和 API	Golang + Beego + MySQL	https://github.com/casibase/casibase



支持的模型

语言模型

模型	子类型	链接
OpenAI	gpt-4-32k-0613, gpt-4-32k-0314, gpt-4-32k, gpt-4-0613, gpt-4-0314, gpt-4, gpt-3.5-turbo-0613, gpt-3.5-turbo-0301, gpt-3.5-turbo-16k, gpt-3.5-turbo-16k-0613, gpt-3.5-turbo, text-davinci-003, text- davinci-002, text-curie-001, text- babbage-001, text-ada-001, text- davinci-001, davinci-instruct-beta, davinci, curie-instruct-beta, curie, ada, babbage	OpenAI
Hugging Face	meta-llama/Llama-2-7b, tiiuae/falcon-180B, bigscience/bloom, gpt2, baichuan-inc/ Baichuan2-13B-Chat, THUDM/chatglm2-6b	Hugging Face
Claude	claude-2, claude-v1, claude-v1-100k, claude- instant-v1, claude-instant-v1-100k, claude-v1.3, claude-v1.3-100k, claude-v1.2, claude-v1.0, claude-instant-v1.1, claude-instant-v1.1-100k, claude-instant-v1.0	Claude
OpenRouter	google/palm-2-codechat-bison, google/ palm-2-chat-bison, openai/gpt-3.5-turbo, openai/gpt-3.5-turbo-16k, openai/gpt-4,	OpenRouter

模型	子类型	链接
	openai/gpt-4-32k, anthropic/claude-2, anthropic/claude-instant-v1, meta-llama/llama-2-13b-chat, meta-llama/llama-2-70b-chat, palm-2-codechat-bison, palm-2-chat-bison, gpt-3.5-turbo, gpt-3.5-turbo-16k, gpt-4, gpt-4-32k, claude-2, claude-instant-v1, llama-2-13b-chat, llama-2-70b-chat	
Ernie	ERNIE-Bot, ERNIE-Bot-turbo, BLOOMZ-7B, Llama-2	Ernie
iFlytek	spark-v1.5, spark-v2.0	iFlytek
ChatGLM	chatglm2-6b	ChatGLM
MiniMax	abab5-chat	MiniMax
本地	custom-model	Local Computer

嵌入模型

模型	子类型	链接
OpenAI	AdaSimilarity, BabbageSimilarity, CurieSimilarity, DavinciSimilarity, AdaSearchDocument, AdaSearchQuery, BabbageSearchDocument, BabbageSearchQuery, CurieSearchDocument, CurieSearchQuery, DavinciSearchDocument, DavinciSearchQuery, AdaCodeSearchCode,	OpenAI

模型	子类型	链接
	AdaCodeSearchText, BabbageCodeSearchCode, BabbageCodeSearchText, AdaEmbeddingV2	
Hugging Face	sentence-transformers/all-MiniLM-L6-v2	Hugging Face
Cohere	embed-english-v2.0, embed-english-light-v2.0, embed-multilingual-v2.0	Cohere
Ernie	默认	Ernie
本地	custom-embedding	Local Computer

核心概念

作为 Casibase 的用户，您应当熟悉至少 4 个核心概念：[提供商](#)、[存储](#)、[聊天](#) 和 [向量](#)。

提供商

提供商是 Casibase 的支柱，提供基本服务并与外部系统集成。提供商类定义如下：

```
type Provider struct {
    Owner      string `xorm:"varchar(100) notnull pk" json:"owner"`
    Name       string `xorm:"varchar(100) notnull pk" json:"name"`
    CreatedTime string `xorm:"varchar(100)" json:"createdTime"`

    DisplayName string `xorm:"varchar(100)" json:"displayName"`
    Category    string `xorm:"varchar(100)" json:"category"`
    Type        string `xorm:"varchar(100)" json:"type"`
    ClientId   string `xorm:"varchar(100)" json:"clientId"`
    ClientSecret string `xorm:"varchar(2000)" json:"clientSecret"`
    ProviderUrl string `xorm:"varchar(200)" json:"providerUrl"`
}
```

提示

Casibase 中主要有两种类型的提供商：

- **存储提供商。** 存储提供商便于在 Casibase 中存储和检索数据。它支持多种存储选项，包括：

- AWS
 - Azure
 - 本地文件系统
- **AI 提供商。** AI 提供商负责处理 Casibase 中与 AI 相关的任务和服务。它支持多种 AI 模型和技术，包括：
 - OpenAI
 - ChatGLM
 - InternLM

向量

Casibase 中的向量代表不同类型数据的数值表示。这些向量使得信息处理和分析更加高效。部分可用的向量类型包括：

- 文本向量
- 图像向量
- (其他向量类型)

向量类定义如下：

```
type Vector struct {
    Owner      string      `xorm:"varchar(100) notnull pk"
                           json:"owner"`
    Name       string      `xorm:"varchar(100) notnull pk"
                           json:"name"`
    CreatedTime string     `xorm:"varchar(100)" json:"createdTime"`
    DisplayName string     `xorm:"varchar(100)" json:"displayName"`
}
```

聊天

聊天是用户与 Casibase 中 AI 模型之间交互沟通的核心。它们由三个基本组成部分构成：

- 问题：用户的输入或查询，用于寻求信息或帮助。
- 查询提示：经过格式化的用户问题版本，为 AI 模型的处理做准备。
- 答案：AI 生成的对用户问题的响应，提供相关信息或解决方案。

聊天类定义如下：

```
type Chat struct {
    Owner      string `xorm:"varchar(100) notnull pk"`
    json:"owner"`
    Name       string `xorm:"varchar(100) notnull pk"`
    json:"name"`
    CreatedTime string `xorm:"varchar(100)" json:"createdTime"`
    UpdatedTime string `xorm:"varchar(100)" json:"updatedTime"`

    DisplayName string `xorm:"varchar(100)" json:"displayName"`
    Category    string `xorm:"varchar(100)" json:"category"`
    Type        string `xorm:"varchar(100)" json:"type"`
    User1       string `xorm:"varchar(100)" json:"user1"`
    User2       string `xorm:"varchar(100)" json:"user2"`
    Users       []string `xorm:"varchar(100)" json:"users"`
    MessageCount int     `json:"messageCount"`
}
```

嵌入

嵌入是将各种类型的数据，如文本和图像，转换成密集向量表示的过程。此步骤对于促

进 Casibase 内高效的数据处理和分析至关重要。

 提示

- 通过嵌入，聊天中的问题和存储中的知识文件将转化为向量，并用于下一步的知识搜索。
- Casibase 默认的嵌入方法由 OpenAI 提供，每分钟最多调用三次。我们建议尽量减少知识文件之间的耦合，以便嵌入和后续处理。

服务器安装

要求

操作系统

支持所有主要操作系统，包括Windows、Linux和macOS。

环境

- Go 1.20+
- Node.js LTS (18)
- Yarn 1.x

信息

Casibase的使用分为两个步骤：

- 步骤1: [部署并运行Casdoor](#)
- 步骤2: 部署并运行Casibase（本文档）

我们强烈建议使用[Yarn 1.x](#)来运行和构建Casdoor和Casibase前端，使用NPM可能会导致UI样式问题，详见：[casdoor#294](#)

小心

对于中国用户，为了成功下载Go依赖包，您需要通过配置GOPROXY环境变量来使用Go代理。我们强烈推荐：<https://goproxy.cn/> 我们强烈推荐：

<https://goproxy.cn/>

数据库

Casibase 使用 XORM 与数据库通信。 Casibase使用XORM与数据库通信。 基于Xorm 驱动支持， Casibase目前支持以下数据库：

- MySQL
- MariaDB
- PostgreSQL
- CockroachDB
- SQL Server
- Oracle
- SQLite 3
- TiDB

guacd

Casibase使用guacamole-server提供远程桌面访问。 如果您想使用此功能， 需要先安装 guacamole-server。 如果您尚未安装guacamole-server，请参考[guacamole-server安装](#)。 如果您想使用此功能， 您需要先安装 guacamole-server。 如果您还未安装 guacamole-server，请参阅[guacamole-server 安装指南](#)。

您也可以使用以下命令在docker中运行guacd：

```
docker run -d --name guacd -p 4822:4822 guacamole/guacd
```

下载

Casibase的源代码托管在GitHub上：<https://github.com/casibase/casibase>。Go后端代码和React前端代码都在同一个仓库中。 Go 后端代码和 React 前端代码都在同一个仓库中。

名称	描述	语言	源代码
前端	Casibase的Web前端UI	JavaScript + React	https://github.com/casibase/casibase/tree/master/web
后端	Casibase的RESTful API后端	Golang + Beego + XORM	https://github.com/casibase/casibase

Casibase支持[Go Modules](#)。要下载代码，您只需通过git克隆代码： 要下载代码，只需通过 git 克隆代码即可：

```
cd path/to/folder  
git clone https://github.com/casibase/casibase
```

配置

配置Casdoor

请参考[Casdoor-SSO](#)部分来配置Casdoor。

记住您在Casdoor配置中的 `clientId`、`clientSecret`、`organization`、`application` 等信息，我们稍后会用到。

配置数据库

Casibase支持mysql、mssql、sqlite3、postgres。Casibase默认使用mysql。
Casibase 默认使用 mysql。

MySQL

Casibase将在名为 `casibase` 的MySQL数据库中存储其用户、节点和主题信息。如果数据库不存在，需要手动创建。数据库连接字符串可以在以下位置指定：

<https://github.com/casibase/casibase/blob/master/conf/app.conf> 如果数据库不存在，则需要手动创建。数据库连接字符串可以在此指定：<https://github.com/casibase/casibase/blob/master/conf/app.conf>

```
driverName = mysql
dataSourceName = root:123456@tcp(localhost:3306)-
dbName = casibase
```

PostgreSQL

由于我们必须在使用xorm打开Postgres时选择一个数据库，因此您应该在运行Casibase之前手动准备一个数据库。

假设您已经准备好了个名为 `casibase` 的数据库，那么您应该这样指定 `app.conf`：

```
driverName = postgres
dataSourceName = "user=postgres password=postgres host=localhost
port=5432 sslmode=disable dbname=casibase"
dbName =
```

① 信息

对于PostgreSQL，确保 `dataSourceName` 有非空的 `dbName`，并像上面的例子一样将独立的 `dbName` 字段留空。

CockroachDB

您也可以使用 `postgres` 驱动程序使用 Cockroachdb。它的配置与 PostgreSQL 相同。其配置与 PostgreSQL 相同。

```
driverName = postgres
dataSourceName = "user=postgres password=postgres host=localhost
port=5432 sslmode=disable dbname=casibase
serial_normalization=virtual_sequence"
dbName =
```

① 信息

对于 CockroachDB，不要忘记像上面的例子一样在 `dataSourceName` 中添加 `serial_normalization=virtual_sequence`。否则，每当服务启动或重启时，您都会收到关于数据库已存在的错误。注意，这必须在数据库创建之前添加。否则，每当服务启动或重启时，您会收到有关数据库已存在的错误信息。请注意，此项必须在数据库创建之前添加。

Sqlite3

您应该这样指定 `app.conf`：

```
driverName = sqlite
dataSourceName = "file:casibase.db?cache=shared"
dbName = casibase
```

自定义配置

Casibase支持自定义配置，您可以修改配置文件 `conf/app.conf` 来更改配置。

```
casdoorEndpoint = <您的Casdoor端点>
clientId = <您的Casdoor应用程序的客户端ID>
clientSecret = <您的Casdoor应用程序的客户端密钥>
casdoorOrganization = <您的Casdoor组织名称>
casdoorApplication = <您的Casdoor应用程序名称>
```

运行

目前有两种启动方法，您可以根据自己的情况选择其中一种。



小心

Casibase需要Casdoor提供访问控制和一些后端服务，所以在运行Casibase之前，您必须确保Casdoor正常运行。

如何安装和运行Casdoor：

- [Casdoor安装](#)

开发模式

后端

Casibase的Go后端默认在端口14000上运行。您可以使用以下命令启动Go后端： 您可以使用以下命令启动 Go 后端：

```
go run main.go
```

服务器成功运行后，我们可以启动前端部分。

前端

Casibase的前端是一个非常经典的[Create-React-App \(CRA\)](#)项目。它默认在端口 [13001](#) 上运行。使用以下命令运行前端： 默认在端口 [13001](#) 运行。 使用以下命令运行前端：

```
cd web  
yarn install  
yarn start
```

生产模式

后端

将Casibase Go后端代码构建成可执行文件并启动。

对于Linux:

```
go build  
.casibase
```

对于Windows:

```
go build  
casibase.exe
```

前端

将Casibase前端代码构建成静态资源（.html、.js、.css文件）：

```
cd web  
yarn install  
yarn build
```

Nginx



提示

如果您使用nginx作为反向代理，需要在nginx配置文件中添加以下配置：

```
location / {  
    *** 您的配置 ***  
    proxy_set_header Upgrade $http_upgrade;  
    proxy_set_header Connection "upgrade";  
}
```

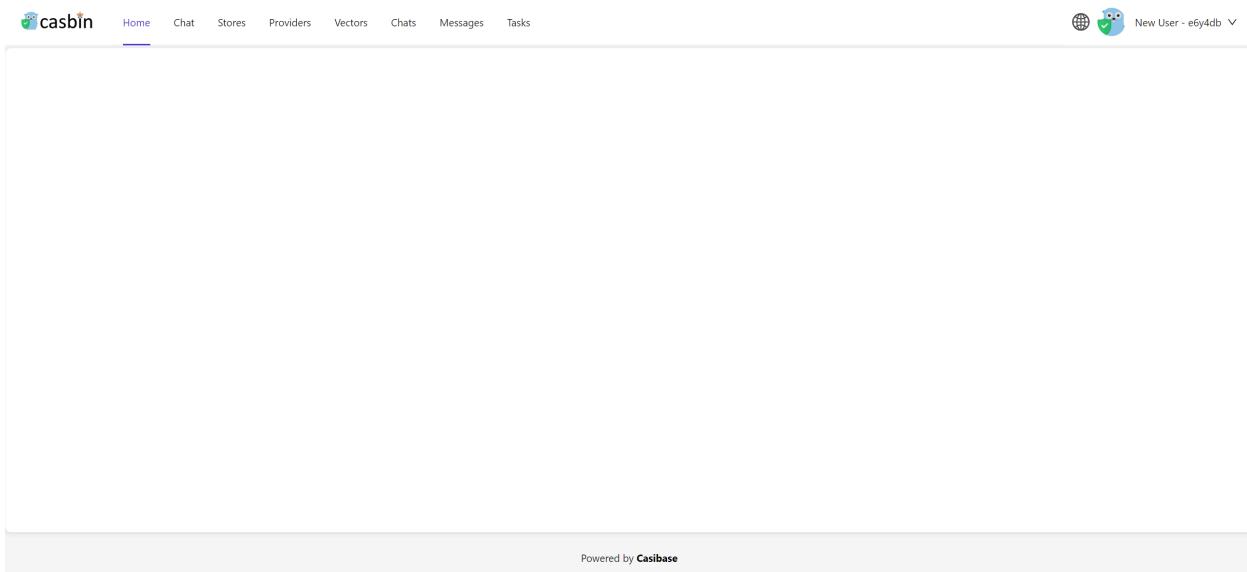
因为Casibase使用websocket与guacd通信。

预览

在浏览器中访问：<http://localhost:13001>。 在浏览器中访问：<http://localhost:13001>。 使用您刚刚在Casdoor中注册的用户账户登录Casibase仪表板：



然后您将进入Casibase的主页：



💡 提示

要使用其他端口，请编辑 `conf/app.conf` 并修改 `httpport`，然后重启Go后端。

(可选) 使用Docker尝试

要求

硬件

如果您想自己构建 Docker 镜像, 请确保您的机器至少有 **2GB** 的内存。Casibase 的前端是一个 React 的 NPM 项目。构建前端需要至少 **2GB** 的内存。内存少于 **2GB** 可能导致前端构建失败。

如果您只需要运行预构建的镜像, 请确保您的机器至少有**100MB**内存。

操作系统

支持所有操作系统 (Linux、Windows和macOS) 。

Docker

您可以在Linux中使用Docker (docker-engine版本 ≥ 17.05) 或在Windows和macOS中使用Docker Desktop。

- [Docker](#)

无论操作系统如何, 用户都必须确保他们拥有 docker-engine 版本 ≥ 17.05 。无论使用什么操作系统, 用户都必须确保有docker-engine版本 ≥ 17.05 。这是因为我们在 docker-compose.yml 中使用了多阶段构建功能, 该功能在17.05及以上版本中得到支持。更多信息, 请参见<https://docs.docker.com/develop/develop-images/multistage-build/>。更多信息, 请参见 <https://docs.docker.com/develop/develop-images/multistage-build/>。

如果您使用 docker-compose，请确保您有 docker-compose 版本 ≥ 2.2 。对于 Linux 用户，请注意 docker-compose 需要与 docker-engine 分开安装。对于 Linux 用户，请注意 docker-compose 需要与 docker-engine 分开安装。

获取镜像

我们提供了两个 DockerHub 镜像：

名称	描述	建议
casibase-all-in-one	镜像中包含 Casibase 和 MySQL 数据库	此镜像已包含一个测试数据库，仅用于测试目的
casibase	镜像中仅包含 Casibase	此镜像可以连接到您自己的数据库并用于生产环境

1. [casbin/casibase-all-in-one](#): 此镜像包含 casibase 可执行文件、MySQL 数据库和所有必要的配置。它是为想要快速尝试 Casibase 的新用户设计的。使用此镜像，您可以仅用一两个命令就立即启动 Casibase，无需任何复杂配置。但是，请注意，我们不建议在生产环境中使用此镜像。它旨在为想要快速体验 Casibase 的新用户而设计。使用此镜像，您可以通过一两条命令立即启动 Casibase，而无需任何复杂配置。但请注意，我们不建议在生产环境中使用此镜像。

选项-1：使用测试数据库

运行容器时将端口 [14000](#) 暴露给主机。如果本地主机上不存在该镜像，将自动拉取。如果本地主机上不存在该镜像，将会自动拉取。

```
docker run -p 14000:14000 casbin/casibase-all-in-one
```

在您的浏览器中访问 <http://localhost:14000>。 在浏览器中访问 <http://localhost:14000>。 使用默认的全局管理员账户登录 Casibase 仪表板: `built-in/admin`

```
admin  
123
```

选项-2：使用 docker-compose 尝试

在 `docker-compose.yml` 文件中创建一个 `conf/app.conf` 目录。 然后，从 Casibase 中复制 `app.conf`。 关于 `app.conf` 的更多详情，您可以查看[通过 Ini 文件](#)。

使用 docker-compose 创建一个独立的数据库：

```
docker-compose up
```

就这样！ ✨

在您的浏览器中访问 <http://localhost:14000>。 使用默认的全局管理帐户登录 Casibas 控制面板：`内置/管理`

```
admin  
123
```

注意：如果你深入到 `docker-compose.yml` 文件，你可能会对我们创建的称为“`RUNNING_IN_DOCKER`”的环境变量感到困惑。当通过 `docker-compose` 创建数据库 `'db'` 时，它在您 PC 的 `localhost` 上可用，但在 `Casibase` 容器的 `localhost` 上不可用。为了防止新用户因修改 `app.conf` 而遇到困难，我们提供了此环境变量，并预先在 `docker-compose.yml` 中进行了赋值。当此环境变量设置为 `true` 时，`localhost` 将被替换为 `host.docker.internal`，以便 `Casibase` 可以访问数据库。

选项-3：直接使用标准镜像



提示

如果不方便将配置文件挂载到容器中，使用环境变量也是一种可能的解决方案。

example

```
docker run \
-e driverName=mysql \
-e dataSourceName='user:password@tcp(x.x.x.x:3306)/*' \
-p 14000:14000 \
casbin/casibase:latest
```

创建 `conf/app.conf`。 创建 `conf/app.conf`。 您可以从 Casibase 的 `conf/app.conf` 复制。 关于 `app.conf` 的更多详情，您可以查看 [通过 Ini 文件](#)。 关于 `app.conf` 的更多详情，您可以查看 [通过 Ini 文件](#)。

然后运行

```
docker run -p 14000:14000 -v /folder/of/app.conf:/conf casbin/casibase:latest
```

总之，只需要将 `app.conf` 挂载到 `/conf/app.conf` 并启动容器即可。

在您的浏览器中访问 <http://localhost:14000>。 在浏览器中访问 <http://localhost:14000>。 使用默认的全局管理员账户登录 Casibase 仪表板：`built-in/admin`

admin

123

(Optional) Try with K8s Helm

Introduction

This guide shows how to deploy Casibase on Kubernetes using Helm for easy and scalable management. Helm simplifies the deployment process and allows for easy configuration management.

Prerequisites

- A running Kubernetes cluster
- Helm v3 installed
- kubectl configured to connect to your cluster
- A MySQL/PostgreSQL database (recommended for production)
- A Casdoor instance for authentication

Configuration

Before installation, you must create an application configuration file. The Helm chart will not work with default values.

Pre step: Create app.conf File

Create an `app.conf` file with your specific settings:

You can view more details about the configuration options in the [Casibase Configuration Documentation](#).

Or check the latest configuration options in [app.conf example](#).

```
appname = casibase
httpport = 14000
runmode = prod
SessionOn = true
copyRequestBody = true

# Database Configuration - REQUIRED
driverName = mysql
dataSourceName = your-username:your-password@tcp(your-db-host:3306)-
dbName = your-database

# Casdoor Authentication - REQUIRED
casdoorEndpoint = https://door.casdoor.com
clientId = your-client-id
clientSecret = your-client-secret
casdoorOrganization = "your-organization"
casdoorApplication = "your-application"
redirectPath = /callback

# Optional Settings
redisEndpoint =
guacamoleEndpoint = 127.0.0.1:4822
isDemoMode = false
disablePreviewMode = false
logPostOnly = true
landingFolder =
cacheDir = "C:/casibase_cache"
appDir = ""
isLocalIpDb = false
audioStorageProvider = ""
providerDbName = ""
socks5Proxy = "127.0.0.1:10808"
publicDomain = ""
adminDomain = ""
enableExtraPages = false
shortcutPageItems = []
usageEndpoints = []
iframeUrl = ""
forceLanguage = ""
defaultLanguage = "en"
```

Optional: Using Secrets for Sensitive Data

For production environments, create a Kubernetes secret with your configuration:

```
# Create secret from app.conf file
kubectl create secret generic casibase-config --from-
file=app.conf=./app.conf
```

Installation Steps

Step 1: Prepare Configuration Files

Ensure you have created both:

- `app.conf` - Application configuration

Step 2: Install with Configuration File



Visit the [Casbin Helm Chart](#) to find the latest version.

Install Casibase by passing the app.conf file directly:

```
# Method 1: Pass app.conf content as appConfig parameter
helm install casibase oci://registry-1.docker.io/casbin/casibase-helm-
chart \
--version v1.549.0 \
--set-file appConfig=./app.conf
```

Alternative Installation with Secret

If using secrets for sensitive data:

```
# Create secret first
kubectl create secret generic casibase-config --from-
file=app.conf=./app.conf

# Install with secret reference (no additional values file needed)
helm install casibase oci://registry-1.docker.io/casbin/casibase-helm-
chart \
--version v1.549.0 \
--set appConfig="" \
--set appConfigFromSecret=casibase-config
```

Step 3: Verify Installation

Check the deployment status:

```
kubectl get pods
kubectl get services
kubectl logs -l app.kubernetes.io/name=casibase
```

Step 4: Access Casibase

Once installed, Casibase will be accessible through the Kubernetes service on port 14000. If you enabled ingress, it will be available at your configured domain.

Configuration Options Reference

The following table shows the main configuration parameters available in the Helm chart:

Parameter	Description	Default Value
<code>replicaCount</code>	Number of Casibase replicas to run	<code>1</code>
<code>image.repository</code>	Docker image repository	<code>casbin</code>
<code>image.name</code>	Docker image name	<code>casibase</code>
<code>image.pullPolicy</code>	Image pull policy	<code>IfNotPresent</code>
<code>image.tag</code>	Image tag (defaults to chart appVersion)	<code>""</code>
<code>appConfig</code>	Application configuration (app.conf content)	See values.yaml
<code>appConfigFromSecret</code>	Mount app.conf from secret instead	<code>""</code>

Parameter	Description	Default Value
<code>service.type</code>	Kubernetes service type	<code>ClusterIP</code>
<code>service.port</code>	Service port	<code>14000</code>
<code>ingress.enabled</code>	Enable ingress	<code>false</code>
<code>ingress.hosts</code>	Ingress hosts configuration	<code>[]</code>
<code>resources</code>	CPU/ Memory resource requests and limits	<code>{}</code>
<code>autoscaling.enabled</code>	Enable horizontal pod autoscaler	<code>false</code>
<code>autoscaling.minReplicas</code>	Minimum number of replicas	<code>1</code>
<code>autoscaling.maxReplicas</code>	Maximum number of replicas	<code>100</code>

Parameter	Description	Default Value
<code>autoscaling.targetCPUUtilizationPercentage</code>	CPU utilization threshold	80
<code>nodeSelector</code>	Node labels for pod assignment	{}
<code>tolerations</code>	Toleration labels for pod assignment	[]
<code>affinity</code>	Affinity settings for pod assignment	{}

Advanced Configuration Options

For production deployments, consider these additional options:

```
# Autoscaling
autoscaling:
  enabled: true
  minReplicas: 2
  maxReplicas: 10
  targetCPUUtilizationPercentage: 70
```

Managing the Deployment

Upgrading Casibase

To upgrade your Casibase deployment to a new version:

```
helm upgrade casibase oci://registry-1.docker.io/casbin/casibase-helm-
chart --version <new-version>
```

To upgrade with custom values:

```
helm upgrade casibase oci://registry-1.docker.io/casbin/casibase-helm-
chart --version <new-version> \
-f custom-values.yaml
```

Checking Deployment Status

Monitor your deployment:

```
# Check pod status
kubectl get pods -l app.kubernetes.io/name=casibase

# Check service status
kubectl get svc -l app.kubernetes.io/name=casibase

# View logs
kubectl logs -l app.kubernetes.io/name=casibase

# Describe deployment
helm status casibase
```

Uninstalling Casibase

To completely remove Casibase from your cluster:

```
helm uninstall casibase
```

Troubleshooting

Common Issues

1. Pod not starting: Check logs with `kubectl logs <pod-name>`
2. Service not accessible: Verify service configuration and ingress setup
3. Database connection issues: Ensure database credentials and connectivity are correct
4. Configuration errors: Validate your `appConfig` syntax
5. Casdoor authentication failures: Verify Casdoor endpoint and credentials
6. Domain/URL issues: Check domain configuration and DNS resolution

Configuration-Related Issues

Problem: Casdoor authentication not working

- Verify `casdoorEndpoint` is accessible from the cluster
- Check `clientId` and `clientSecret` are correct
- Ensure `redirectPath` matches your Casdoor application configuration

Problem: Configuration syntax errors

```
# Validate YAML syntax before deployment
```

Getting Help

- Check pod events: `kubectl describe pod <pod-name>`
- View Helm release info: `helm status casibase`
- Review configuration: `helm get values casibase`

Conclusion

Using Helm to deploy Casibase on Kubernetes provides a robust, scalable solution for managing your knowledge base platform. The chart offers flexible configuration options to suit various deployment scenarios, from development environments to production clusters.

Key benefits of this approach:

- Easy deployment and updates through Helm commands
- Flexible configuration through values files
- Kubernetes-native scaling and management
- Production-ready with proper resource management and health checks

For more advanced configurations and troubleshooting, refer to the [Kubernetes documentation](#) and [Helm documentation](#).

Casibase Public API

Casibase frontend web UI is a [SPA \(Single-Page Application\)](#) developed in React. The React frontend consumes the Casibase RESTful API exposed by the Go backend code. This RESTful API is referred to as the [Casibase Public API](#). In other words, with HTTP calls, you can do everything just like how the Casibase web UI itself does. There's no other limitation. The API can be utilized by the following:

- Casibase's frontend
- Casibase client SDKs (e.g., casibase-java-sdk)
- Any other customized code from the application side

The full reference for the [Casibase Public API](#) can be found on Swagger: <https://ai-admin.casibase.com/swagger>. These Swagger docs are automatically generated using Beego's Bee tool. If you want to generate the Swagger docs by yourself, see: [How to generate the swagger file](#)

How to authenticate with [Casibase Public API](#)

Casibase Public API supports two application-level authentication methods: [Bearer Token](#) and [Basic Auth](#). The [Bearer Token](#) method is recommended as it is more secure.

SDK Authentication Example (Java)

To illustrate how authentication is handled in practice, here is an example from the Casibase Java SDK. The following code shows how the SDK constructs the credential for API requests. This process authenticates the SDK with application-level permissions, effectively acting as an admin.

```
// ...

protected void Service(Config config, AuthTypeEnum authType)
throws Exception {
    this.config = config;
    switch (authType){
        case BASIC:
            this.credential =
Credentials.basic(config.clientId, config.clientSecret);
            break;
        case BEARER:
            String token = config.clientId + ":" +
config.clientSecret;
            this.credential = "Bearer " +
DigestUtils.md5Hex(token);
            break;
        default:
            throw new Exception("Invalid auth type");
    }
}
```

The example above demonstrates how to prepare the credential for both authentication types:

- **BASIC:** It uses a helper (`okhttp3.Credentials`) to perform the standard Base64 encoding for Basic Authentication.
- **BEARER:** It constructs the token by taking the MD5 hash of

`clientId:clientSecret` (using `org.apache.commons.codec.digest.DigestUtils`) and prepending the result with "Bearer".

1. By **Bearer Token** (Recommended)

This method is more secure because it uses a static access token, which is a hashed value of your `clientId` and `clientSecret`.

How to get the access token?

The access token is calculated using the following formula: `md5(clientId + ":" + clientSecret)`

How to authenticate?

The access token must be provided in the `Authorization` header as a Bearer Token.

```
Authorization: Bearer <The access token>
```

2. By **Basic Auth**

This method uses the `clientId` and `clientSecret` directly for authentication. It is considered less secure because the `clientSecret` might be exposed. It is supported for convenience and compatibility purposes.

How to authenticate?

[HTTP Basic Authentication](#): This is the standard way.

```
Authorization: Basic <The Base64 encoding of  
"clientId:clientSecret">
```

If you are not familiar with Base64 encoding, you can use a library for this, as [HTTP Basic Authentication](#) is a widely supported standard.

Where to find the Client ID and Secret?

Both authentication methods require a `clientId` and `clientSecret`. You can find these values for your application in the Casibase configuration file: [conf/app.conf](#).

Container Cloud

Casibase is an open-source Container Cloud Platform built on the foundations of Docker and Kubernetes. It is designed for individuals and organizations to easily build, manage, and operate their own private cloud environments with a focus on simplicity and application-centric management.

The Challenge: Complexity in the Cloud-Native Era

In the world of modern software, Kubernetes has become the standard for running applications. However, its power comes with significant complexity. Deploying even a seemingly simple application, like a WordPress blog, requires orchestrating a multitude of distinct Kubernetes resources:

- Deployments to manage the application pods (the WordPress server itself).
- Services to expose the application to the network.
- PersistentVolumeClaims to request storage for the database and uploads.
- StatefulSets to manage the database pods (like MySQL).
- ConfigMaps and Secrets to handle configuration and sensitive data.

Managing these individual components manually is often called a "resource-centric" approach. This approach presents several challenges:

1. **High Learning Curve:** Users must have a deep understanding of various Kubernetes resources and how they interact.
2. **Operational Burden:** Manually creating, updating, and deleting these

resources is tedious and prone to human error.

3. **Lack of Atomicity:** There is no way to treat the entire "WordPress application" as a single, atomic unit. You cannot simply "install" or "uninstall" it with one action.
4. **Inconsistency:** Ensuring that the application is deployed identically across development, testing, and production environments is difficult.

The Casibase Approach: From Managing Resources to Managing Applications

Casibase fundamentally simplifies this process by shifting the focus from managing individual resources to managing the application as a whole. We believe you should be able to manage your applications without getting lost in the weeds of Kubernetes YAML configurations.

To achieve this, Casibase introduces a powerful, application-centric model built on two core concepts:

1. Templates: The Reusable Blueprint

A Template in Casibase is a complete, reusable blueprint for an application. It encapsulates all the necessary Kubernetes resource manifests required to deploy a service. Think of it as a "package" for a cloud application.

- **What it contains:** A template holds the base YAML configurations for all the components of an application (Deployments, Services, etc.), structured for use with Kustomize.
- **The Goal:** To make the underlying complexity transparent. Once a template for WordPress is created, anyone can use it to deploy WordPress without needing to know the details of its Kubernetes architecture.

2. Applications: The Live Instance

An Application is a live, running instance created from a Template. It represents a specific deployment of that template in your cluster.

- **Customization:** When you create an Application, you select a Template and then provide your own specific configurations, such as the number of replicas, a custom domain name, or a specific database password.
- **How it works:** These custom configurations are treated as "patches" or "overlays." Casibase uses Kustomize to intelligently merge your custom parameters with the base manifests from the template, generating the final, complete configuration.
- **Lifecycle Management:** The Application becomes the single unit you interact with. You can deploy, monitor, update, and delete the entire application with single clicks in the UI.

By adopting this model, Casibase transforms the complex task of cloud-native deployment into a streamlined, intuitive workflow. Instead of wrestling with `kubectl` and YAML files, you can manage the entire lifecycle of your applications through a clean web interface: select a template, fill in a few parameters, and click deploy.

初学者指南

添加存储提供商

了解如何将存储提供商集成到Casibase中

添加AI模型提供商

学习如何添加模型提供商来增强Casibase功能

添加嵌入提供商

探索如何将嵌入提供商集成到Casibase中

添加语音合成提供商

学习如何添加语音合成提供商来增强Casibase功能。

添加语音识别提供商

了解如何添加语音识别以增强Casibase功能。

添加存储

学习如何向您的Casibase知识库系统添加存储

与AI聊天

在您的Casibase知识库系统中实现AI聊天功能

添加存储提供商

本文档是为初学者设计的分步教程。它将指导您完成将存储提供者与Casibase（我们强大的知识库系统）集成的过程。它将引导您完成将存储提供商与 Casibase 我们强大的知识库系统集成的过程。

简介

向Casibase添加存储提供商可以让您高效地管理和存储数据，这是您的知识库系统的重要组成部分。

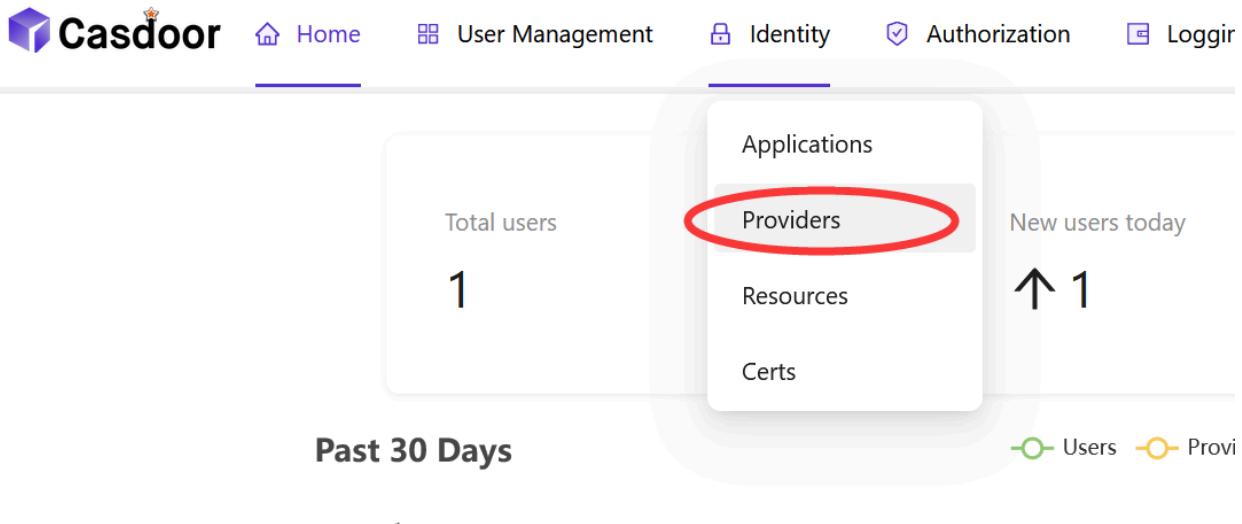
如果您是首次集成存储提供商，不必担心。我们已将整个过程分解为任何人都能遵循的简单步骤。

步骤1：部署Casdoor和Casibase

如果您还没有完成，请参考[部署Casdoor和Casibase](#)教程。

步骤2：添加新的存储提供商

存储提供用于存储数据。存储提供用于存储数据。可以通过点击主页上的 [Identity - Providers](#) 按钮在Casdoor中添加。



步骤2.1：添加存储提供商

点击 **Add** 按钮来添加存储提供商。

The screenshot shows the "Providers" management page. At the top, there are navigation links: Home, User Management, Identity, and Authorization. The "Identity" link is highlighted with a blue underline. Below the navigation, there's a table to manage providers:

Name	Organization	Created time	Di
provider_captcha_default	admin (Shared)	2023-09-10 19:31:50	Ca

The "Add" button in the top-left corner of the table header is circled in red.

步骤2.2：填写存储提供商信息

填写存储提供者信息并点击 **Save & Exit** 按钮。

New Provider [Save](#) [Save & Exit](#) [Cancel](#)

Name [?](#) : provider_storage_1

Display name [?](#) : Provider_storage_1

Organization [?](#) :

Category [?](#) : Storage

Type [?](#) : aws AWS S3

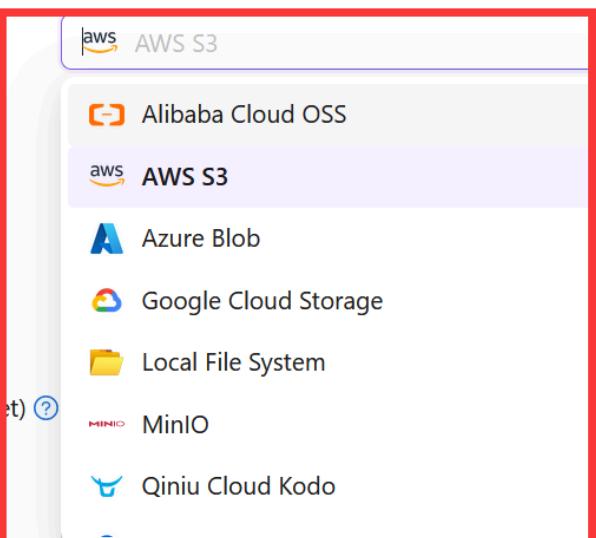
Client ID [?](#) :

Client secret [?](#) :

Endpoint [?](#) :

Endpoint (Intranet) [?](#) :

Bucket [?](#) :



Casdoor支持多种存储提供商，包括：

- [AWS S3](#)
- [Azure Blob](#)
- [Google Cloud Storage](#)
- [MinIO](#)

- [七牛云 Kodo](#)
- [阿里云OSS ...](#)

示例

添加阿里云OSS存储提供者



小心

- Client ID: 您的阿里云OSS账户的AccessKey ID。
- Client Secret: 您的阿里云OSS账户的AccessKey Secret。

*****是您的阿里云OSS账户信息的占位符。

Category [?](#) : Storage

Type [?](#) : Alibaba Cloud OSS

Client ID [?](#) : LTA***NLf

Client secret [?](#) : Vo6***pi8

Endpoint [?](#) : oss-cn-beijing.aliyuncs.com

Endpoint (Intranet) [?](#) :

Bucket [?](#) : xx-bucket-0

Path prefix [?](#) :

Domain [?](#) : https://xx-bucket-0.oss-cn-beijing.aliyuncs.com

Provider URL [?](#) : https://github.com/organizations/xxx/settings/applications/1234567

[Save](#) [Save & Exit](#) [Cancel](#)

步骤2.3：查看存储提供商

添加存储提供商后，您可以查看存储提供商信息。

Name	Organization	Created time	Display name	Category	Type	Client ID	Provider URL	Action
provider_storage_1	admin (Shared)	2023-09-10 21:23:02	Provider_storage_1	Storage	Alibaba Cloud OSS	[REDACTED]	https://github.com/organizations/xxx...	Edit Delete



提示

存储提供商来自Casdoor。 存储提供商来自Casdoor。 您可以在Casdoor中添加存储提供商，然后将其添加到Casibase中。

更多信息请参考步骤2：添加新的存储提供商。

The screenshot shows the Casbin web application interface. At the top, there is a navigation bar with links: Home, Chat, Stores (which is the active tab), Providers, Vectors, Chats, Messages, Tasks, and Help. Below the navigation bar, there is a form titled "Edit Store". The form has several input fields:

- Name: store_v6c22m
- Display name: New Store - v6c22m
- Storage provider: A dropdown menu with an option "Provider_storage_1 (provider_storage_1)" circled in red.
- Model provider: (empty)
- Embedding provider: (empty)
- File tree: (empty)

At the top left of the form, there is a "Save" button.

存储示例

[Home](#)[Chat](#)[Stores](#)[Providers](#)[Vectors](#)[Chats](#)[M](#)[Edit Store](#)[Save](#)

Name:

my_store

Display name:

My_Store

Storage provider:

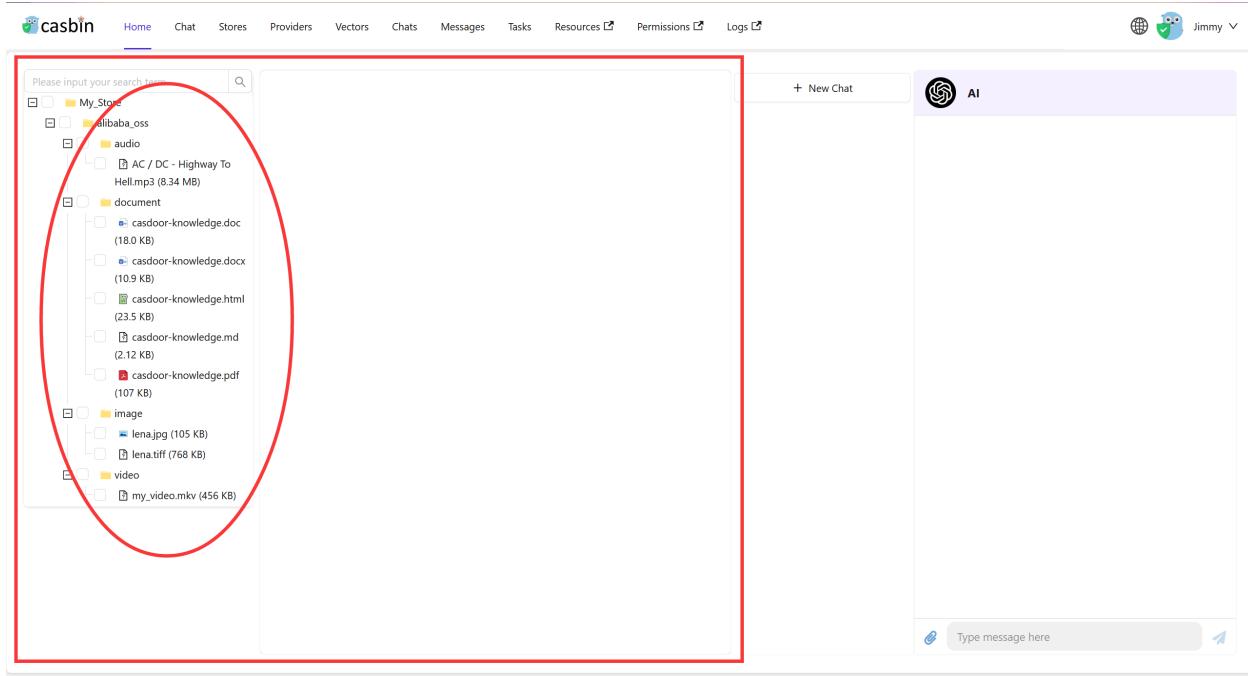
Provider_storage_1 (provider_storage_1)

Model provider:

Embedding provider:

File tree:

保存配置，返回主页，您将看到存储提供商的文件树。



现在您可以在Casibase中管理您的数据了。

在下一章中，我们将学习如何添加AI模型提供商到Casibase。

添加AI模型提供商

本文档是为初学者设计的分步教程。它将指导您完成将模型提供商与Casibase（我们强大的知识库系统）集成的过程。它将指导您完成将模型提供商与Casibase这套强大的知识库系统集成的过程。

简介

向Casibase添加模型提供商可以通过整合机器学习模型和AI功能来增强其功能。模型提供商允许您在知识库系统中分析和处理数据，使其更加智能和高效。模型提供商使您能够在知识库系统内分析和处理数据，从而使其更加智能高效。

如果您刚开始集成模型提供商，不用担心。我们已将该过程分解为任何人都能遵循的简单步骤。

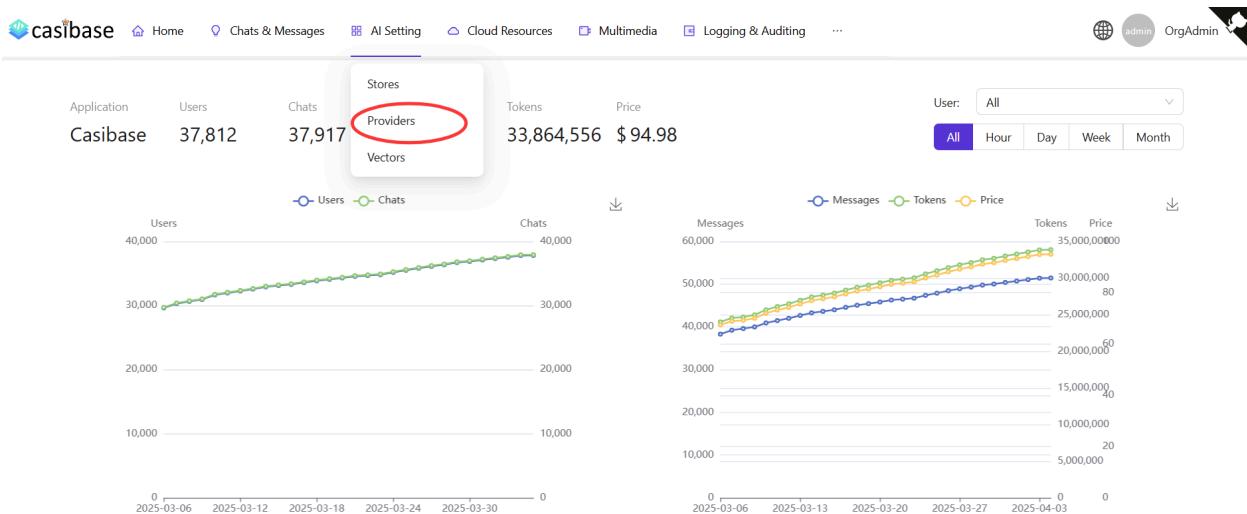
步骤1：部署Casdoor和Casibase

在添加AI模型提供商之前，请确保已部署Casdoor和Casibase。在添加AI模型提供商之前，请确保您已经部署了Casdoor和Casibase。如果您还没有完成，请参考[部署Casdoor和Casibase](#)教程。

步骤2：添加新的模型提供商

模型提供商用于将LLM集成到Casibase中。您可以按照以下步骤添加：您可以按照以下步骤添加它们：

点击主页上的[Providers](#)按钮。



步骤2.1：添加模型提供商

点击 **Add** 按钮来添加模型提供商。

The table lists various model providers:

Name	Display name	Category	Type	Sub type	API key	Secret key	Region
provider_tts_alibabacloud_cosyvoice	Provider TTS AlibabaCloud Cosyvoice	Text-to-Speech	Alibaba Cloud	cosyvoice-v1		***	
provider_blockchain_chainmaker	Provider Blockchain ChainMaker	Blockchain	Tencent ChainMaker (Demo Network)	text-davinci-003	AKIDObZHjqUxnNltkKdIQvAWcS1JK4XidG	***	ap-beijing
provider_model_alibabacloud_deepseek_r1	Provider Model AlibabaCloud DeepSeek-R1	Model	Alibaba Cloud	deepseek-r1		***	
provider_cloud_alibabacloud	Provider Cloud AlibabaCloud	Public Cloud	Aliyun	text-davinci-003	LTAI5tHkUsopAioN6xi2LMg	***	cn-beijing
dall-e-3	dall-e-3	Model	OpenAI	dall-e-3		***	
provider_model_azure_gpt4_1	Provider Model Azure GPT-4	Model	Azure	gpt-4o	deployment-gpt-4o	***	
provider_model_azure_gpt4_2	Provider Model Azure GPT-4	Model	Azure	gpt-4o	deployment-gpt-4o	***	
provider_model_azure_gpt4_3	Provider Model Azure GPT-4	Model	Azure	gpt-4o	deployment-gpt-4o	***	
provider_embedding_openai_v3	Provider Embedding OpenAI V3	Embedding	OpenAI	text-embedding-ada-002		***	
provider_model_openai_gpt4_vision	Provider Model OpenAI GPT4 Vision	Model	OpenAI	gpt-4-vision-preview		***	

步骤2.2：填写模型提供商详情

填写模型提供商详情并点击 **Save & Exit** 按钮。

[Home](#)[Chat](#)[Stores](#)[Providers](#)[Vectors](#)[Chats](#)[Me](#)[Edit Provider](#)[Save](#)

Name:

provider_openai_model

Display name:

OpenAI model

Category:

Model

Type:

OpenAI

Sub type:

text-davinci-003

Secret key:

Provider URL:

<https://platform.openai.com/account/api-keys>[Save](#)

提示

Casibase支持多种模型提供商，包括：

- [Hugging Face](#)
 - meta-llama/Llama-2-7b

- THUDM/chatglm2-6b
- baichuan-inc/Baichuan2-13B-chat
- gpt2
-
- OpenRouter
 - anthropic/claude-2
 - palm-2-chat-bison
 - palm-2-codechat-bison
 - openai/gpt-4
 -
- OpenAI
 - text-davinci-003
 - gpt-3.5-turbo
 - gpt-4
 -

⚠ 小心

- Category: 模型提供商的一级类别。例如, Model和Embedding。例如, Model和Embedding。
- 类型: 模型提供商的二级类别。例如, OpenAI和Hugging Face。
- SecretKey: 您的OpenAI账户的密钥。

示例

添加OpenAI模型提供商

The screenshot shows the 'Edit Provider' form on the casbin platform. A red circle highlights the 'Type' dropdown menu, which is currently set to 'OpenAI'. Other options in the dropdown include 'Hugging Face', 'OpenRouter', and 'Ernie'. The 'Save' button is visible at the top right of the form.

Field	Value
Name:	provider_openai_model
Display name:	OpenAI model
Category	Model
Type:	OpenAI
Sub type:	OpenAI
Secret key:	
Provider URL:	https://platform.openai.com/account/api-keys

⚠ 小心

某些模型不支持流式输出。已知支持流式输出的模型包括：已知支持流式输出的模型包括：

- gpt-3.5-turbo-0613

添加模型提供商后，您可以使用它来分析和处理Casibase中的数据，使用聊天机器人、问答等AI功能。

返回模型提供商列表页面：



The screenshot shows the Casibase interface with the 'Providers' tab selected. A single provider entry is listed:

Name	Display name	Category	Type	Sub type	API key	Secret key	Provider URL	Action
provider_openai_model	OpenAI model	Model	OpenAI	text-davinci-003	***		https://platform.openai.com/account/api-keys	<button>Edit</button> <button>Delete</button>

A red box highlights the entire row for the 'provider_openai_model' entry.

现在您已经添加了模型提供商，您可以使用它来分析和处理Casibase中的数据，使用聊天机器人、问答等AI功能。

在下一章中，我们将学习如何向Casibase添加嵌入提供商。

添加嵌入提供商

本文档是为初学者设计的分步教程。它将指导您完成将嵌入提供商与 Casibase（我们强大的知识库系统）集成的过程。它将引导您完成将嵌入提供商与 Casibase，我们强大的知识库系统集成的过程。

简介

嵌入是一种将单词和文档表示为向量的技术。嵌入提供商允许您在知识库系统中分析和处理数据，从而使系统更加智能高效。

有关嵌入的更多信息，请参考我们之前文档的[核心概念](#)部分。

在Casibase中，您可以按照以下步骤添加嵌入提供商：

步骤1：部署Casdoor和Casibase

在您添加嵌入模型提供商之前，请确保已部署 Casdoor 和 Casibase。在添加嵌入模型提供商之前，请确保您已经部署了 Casdoor 和 Casibase。如果您还没有完成，请参考[部署 Casdoor 和 Casibase 教程](#)。

步骤2：添加新的嵌入提供商

嵌入提供商用于将嵌入功能集成到 Casibase 中。您可以按照以下步骤添加它们：

点击主页上的 [Providers](#) 按钮。



步骤2.1：添加嵌入提供商

点击 **Add** 按钮来添加嵌入提供商。

The table lists the following providers:

Name	Display name	Category	Type	Sub type	API key	Secret key	Region
provider tts_alibabacloud_cosyvoice	Provider TTS AlibabaCloud Cosyvoice	Text-to-Speech	Alibaba Cloud	cosyvoice-v1		***	
provider_blockchain_chainmaker	Provider Blockchain ChainMaker	Blockchain	Tencent ChainMaker (Demo Network)	text-davinci-003	AKIDObZHjqUxnNltkKdIQyAWcS1JK4XidG	***	ap-beijing
provider_model_alibabacloud_deepseek_r1	Provider Model AlibabaCloud DeepSeek-R1	Model	Alibaba Cloud	deepseek-r1		***	
provider_cloud_alibabacloud	Provider Cloud AlibabaCloud	Public Cloud	Aliyun	text-davinci-003	LTAI5tHkUsopAioN6xi2LMg	***	cn-beijing
dall-e-3	dall-e-3	Model	OpenAI	dall-e-3		***	
provider_model_azure_gpt4_1	Provider Model Azure GPT-4	Model	Azure	gpt-4o	deployment-gpt-4o	***	
provider_model_azure_gpt4_2	Provider Model Azure GPT-4	Model	Azure	gpt-4o	deployment-gpt-4o	***	
provider_model_azure_gpt4_3	Provider Model Azure GPT-4	Model	Azure	gpt-4o	deployment-gpt-4o	***	
provider_embedding_openai_v3	Provider Embedding OpenAI V3	Embedding	OpenAI	text-embedding-ada-002		***	
provider_model_openai_gpt4_vision	Provider Model OpenAI GPT4 Vision	Model	OpenAI	gpt-4-vision-preview		***	

步骤2.2：填写嵌入提供商详情

填写嵌入提供商详情并点击 **Save & Exit** 按钮。

[Home](#)[Chat](#)[Stores](#)[Providers](#)[Vectors](#)[CI](#)[Edit Provider](#)[Save](#)

Name:

embedding_openai_adasimilarity

Display name:

Embedding_OpenAI_AdaSimilarity

Category:

Embedding

Type:

OpenAI

Sub type:

AdaSimilarity

Secret key:

Provider URL:

<https://platform.openai.com/account/api-keys>[Save](#)

提示

与模型提供商部分相同， Casibase支持多种嵌入提供商，包括：

- [OpenAI](#)

- AdaSimilarity
- DavinciSimilarity
- AdaEmbedding2
-
- Hugging Face
 - sentence-transformers/paraphrase-MiniLM-L6-v2
 -

返回提供商列表页面：

Name	Display name	Category	Type	Sub type	API key	Secret key	Provider URL	Action
embedding_openai_adasimilarity	Embedding_OpenAI_AdaSimilarity	Embedding	OpenAI	1		***	https://platform.openai.com/account/api-keys	<button>Edit</button> <button>Delete</button>
model_openai_text_davinci_003	Model OpenAI text-davinci-003	Model	OpenAI	text-davinci-003		***	https://platform.openai.com/account/api-keys	<button>Edit</button> <button>Delete</button>

现在，您可以使用嵌入提供商将文本转换为向量。

添加嵌入提供商后，您可以使用它在Casibase中检索相似文档。有关更多信息，请参考我们之前文档的[核心概念](#)部分。有关更多信息，请参阅我们之前文档中的[核心概念](#)部分。

在下一章中，我们将学习如何将存储提供商、模型提供商和嵌入提供商与Casibase集成。

添加语音合成提供商

本文档是为初学者设计的分步教程。它将指导您完成将嵌入提供商与Casibase（我们强大的知识库系统）集成的过程。它将引导您完成将存储提供商与 Casibase 我们强大的知识库系统集成的过程。

简介

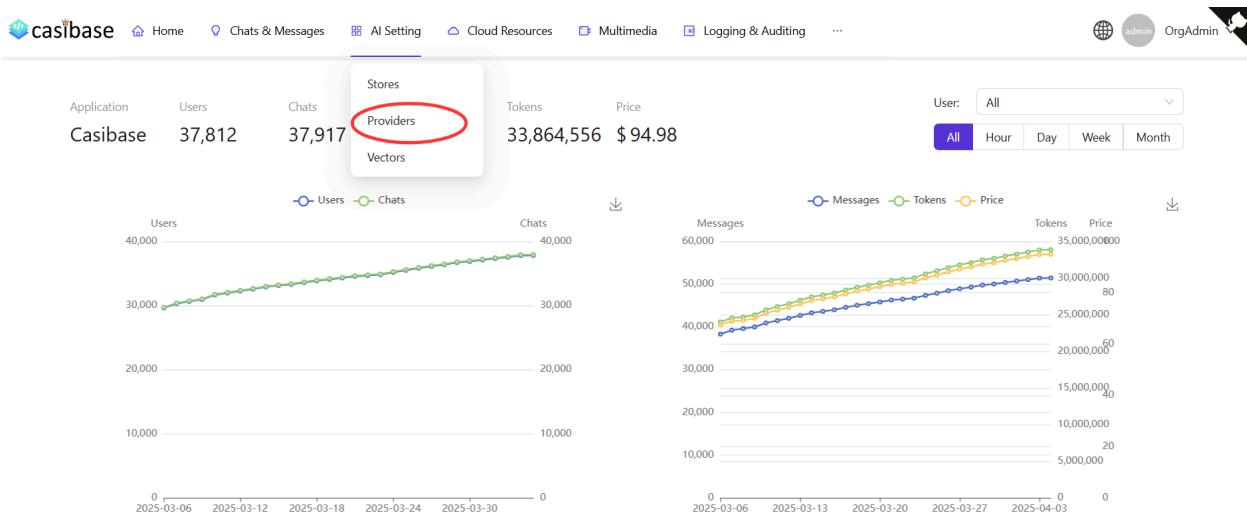
语音合成（TTS）是一项将文本转换为语音输出的技术。 TTS供应商允许您的Casibase 应用程序通过合成语音与用户交流，从而提升用户体验并增强知识库系统的可访问性。

在Casibase中，整合TTS提供商使您的AI应用能够口头回答查询，从而创造更具互动性和吸引力的用户体验。

添加新的语音合成提供商

文本转语音提供者用于将语音合成功能集成到Casibase中。 您可以按照以下步骤添加它们：

点击页面上的 **提供商** 按钮。



语音合成提供商

点击 **Add** 按钮来添加存储提供商。

The table lists existing storage providers and includes an **Add** button to add new ones.

Name	Display name	Category	Type	Sub type	API key	Secret key	Region
provider_tts_alibabacloud_cosyvoice	Provider TTS AlibabaCloud Cosyvoice	Text-to-Speech	Alibaba Cloud	cosyvoice-v1		***	
provider_blockchain_chainmaker	Provider Blockchain ChainMaker	Blockchain	Tencent ChainMaker (Demo Network)	text-davinci-003	AKIDObZHjqUxnNltkKdIQvAWcS1JK4XidG	***	ap-beijing
provider_model_alibabacloud_deepseek_r1	Provider Model AlibabaCloud DeepSeek-R1	Model	Alibaba Cloud	deepseek-r1		***	
provider_cloud_alibabacloud	Provider Cloud AlibabaCloud	Public Cloud	Aliyun	text-davinci-003	LTAI5tHkUsopAioN6xi2LMg	***	cn-beijing
dall-e-3	dall-e-3	Model	OpenAI	dall-e-3		***	
provider_model_azure_gpt4_1	Provider Model Azure GPT-4	Model	Azure	gpt-4o	deployment-gpt-4o	***	
provider_model_azure_gpt4_2	Provider Model Azure GPT-4	Model	Azure	gpt-4o	deployment-gpt-4o	***	
provider_model_azure_gpt4_3	Provider Model Azure GPT-4	Model	Azure	gpt-4o	deployment-gpt-4o	***	
provider_embedding_openai_v3	Provider Embedding OpenAI V3	Embedding	OpenAI	text-embedding-ada-002		***	
provider_model_openai_gpt4_vision	Provider Model OpenAI GPT4 Vision	Model	OpenAI	gpt-4-vision-preview		***	

填写语音合成提供商的详细信息

填写嵌入提供商详情并点击 **保存并退出** 按钮。

casibase

Home Chats & Messages AI Setting Cloud Resources Multimedia Logging & Auditing ...

admin OrgAdmin

Edit Provider

Name: provider_tts_alibabacloud_cosyvoice

Display name: Provider TTS AlibabaCloud Cosyvoice

Category: Text-to-Speech

Type: Alibaba Cloud

Sub type: cosyvoice-v1

Flavor: longxiaochun

Secret key: ***

Provider URL: <https://bailian.console.aliyun.com/?apiKey=1#/api-key>

State: Active

Save Save & Exit

This screenshot shows the 'Edit Provider' page for a Text-to-Speech provider. The provider is named 'provider_tts_alibabacloud_cosyvoice' and has a display name 'Provider TTS AlibabaCloud Cosyvoice'. It is categorized as 'Text-to-Speech' and is of type 'Alibaba Cloud' with sub-type 'cosyvoice-v1' and flavor 'longxiaochun'. The secret key is masked as '***'. The provider URL is set to the official Alibaba Cloud Bailian console API endpoint. The provider is currently active. At the bottom, there are 'Save' and 'Save & Exit' buttons.

💡 提示

Casibase目前支持以下语音合成提供商:

- Alibaba Cloud
 - cosyvoice-v1 (具有多个语音选项)

Testing Your Text-to-Speech Provider

You can test your TTS provider by clicking the `Read it out` button. This will allow you to enter text and hear the synthesized speech output.

The screenshot shows the 'Edit Provider' form for a new provider named 'provider_r7fdnn'. The provider is categorized as 'Text-to-Speech' and is of type 'Alibaba Cloud' with sub-type 'cosyvoice-v1'. A flavor note describes the voice as '龙小淳, 女, 中英双语。龙小淳的嗓音如丝般柔滑, 温暖中流淌着亲切与抚慰, 恰似春风吹过心田。'. The provider URL is set to https://platform.openai.com/account/api-keys. The provider is currently active. A red box highlights the 'Provider test:' input field containing 'Hello, I'm casibase AI.' and the 'Read it out' button.

casibase

Home Chats & Messages AI Setting Cloud Resources Multimedia Logging & Auditing ...

Edit Provider Save Save & Exit

Name: provider_r7fdnn

Display name: New Provider - r7fdnn

Category: Text-to-Speech

Type: Alibaba Cloud

Sub type: cosyvoice-v1

Flavor: 龙小淳, 女, 中英双语。龙小淳的嗓音如丝般柔滑, 温暖中流淌着亲切与抚慰, 恰似春风吹过心田。

Secret key: ***

Provider test: Hello, I'm casibase AI. Read it out

Provider URL: https://platform.openai.com/account/api-keys

State: Active

Save Save & Exit

This testing feature allows you to verify your TTS configuration before implementing it in your applications, ensuring the voice quality and settings meet your requirements.

Alibaba的语音选项

当使用 Alibaba Cloud 的 cosyvoice-v1 时, 您可以从各种语音选项中选择:

- 龙婉
- 龙橙
-

在存储中使用语音合成功能

添加语音合成提供商后, 您可以在商店设置中选择该提供商, 并决定是否启用TTS流式传输。

Edit Store

Name: store-built-in

Display name: Built-in Store

Title:

Avatar:

Storage provider: Built-in Storage Provider (provider-storage-built-in)

Image provider: Storage Aliyun OSS Casibase Casbin (provider_storage_casibase_casbin)

Split provider: Default

Model provider: Provider Model Azure GPT-4 (provider_model_azure_gpt4)

Embedding provider: Provider Embedding OpenAI V3 (provider_embedding_openai_v3)

Text-to-Speech provider: Provider TTS AlibabaCloud Cosyvoice (provider tts_alibabacloud_cosyvoice)

Enable TTS streaming:

Frequencies:

现在，您的商店可以将文本响应转换为语音，从而为用户提供更加互动的体验。

添加语音识别提供商

本文档是为初学者设计的分步教程。它将指导您完成将嵌入提供商与 Casibase（我们强大的知识库系统）集成的过程。它将引导您完成将存储提供商与 Casibase 我们强大的知识库系统集成的过程。

简介

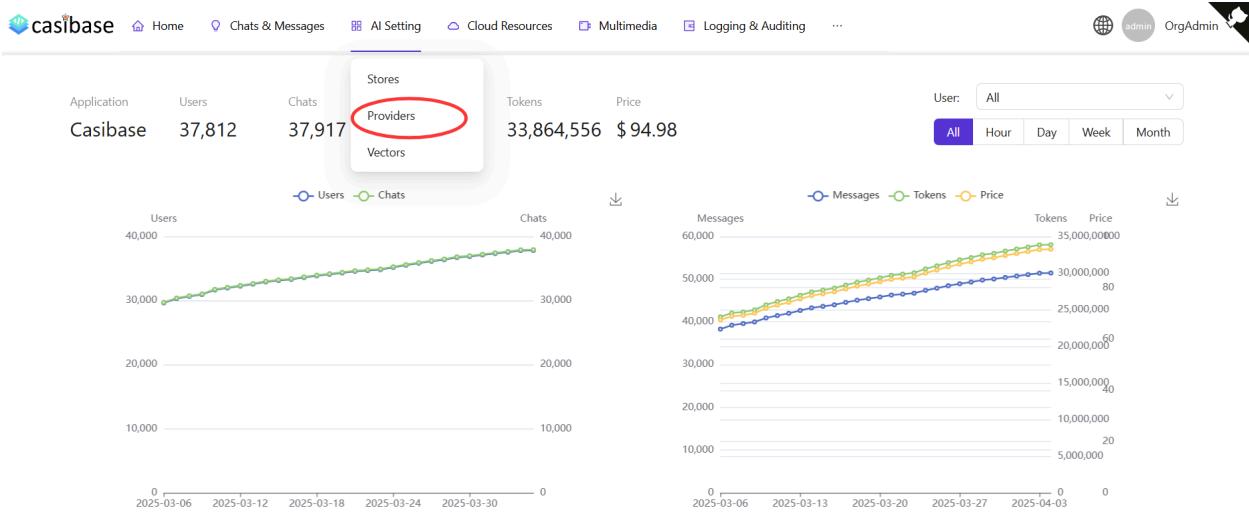
语音识别提供商 (STT) 是一种将口语转换成书面文字的技术。STT 提供商允许您的 Casibase 应用程序理解和处理用户口语输入，增强用户体验和您的知识库系统的访问能力。

在 Casibase 中，整合一个 STT 提供商使您的 AI 应用程序能够接收和处理语音查询，创建更多的互动和自然用户交互。

添加一个新的语音识别提供商

语音识别提供商用将语音识别功能集成到 Casibase 中。您可以按照以下步骤添加它们：

点击页面上的 **Providers** 按钮。



添加语音识别提供商

点击 **添加** 按钮来添加模型提供商。

The table lists the following providers:

Name	Display name	Category	Type	Sub type	API key	Secret key	Region
provider tts_alibabacloud_cosyvoice	Provider TTS AlibabaCloud Cosyvoice	Text-to-Speech	Alibaba Cloud	cosyvoice-v1		***	
provider_blockchain_chainmaker	Provider Blockchain ChainMaker	Blockchain	Tencent ChainMaker (Demo Network)	text-davinci-003	AKIDObZHjqUxnNltkKdIQyAWcS1JK4XidG	***	ap-beijing
provider_model_alibabacloud_deepseek_r1	Provider Model AlibabaCloud DeepSeek-R1	Model	Alibaba Cloud	deepseek-r1		***	
provider_cloud_alibabacloud	Provider Cloud AlibabaCloud	Public Cloud	Aliyun	text-davinci-003	LTAI5tHkUsopAioN6xi2LMg	***	cn-beijing
dall-e-3	dall-e-3	Model	OpenAI	dall-e-3		***	
provider_model_azure_gpt4_1	Provider Model Azure GPT-4	Model	Azure	gpt-4o	deployment-gpt-4o	***	
provider_model_azure_gpt4_2	Provider Model Azure GPT-4	Model	Azure	gpt-4o	deployment-gpt-4o	***	
provider_model_azure_gpt4_3	Provider Model Azure GPT-4	Model	Azure	gpt-4o	deployment-gpt-4o	***	
provider_embedding_openai_v3	Provider Embedding OpenAI V3	Embedding	OpenAI	text-embedding-ada-002		***	
provider_model_openai_gpt4_vision	Provider Model OpenAI GPT4 Vision	Model	OpenAI	gpt-4-vision-preview		***	

填写语音识别提供商详情

填写语音识别提供商详情并点击 **Save & Exit** 按钮。

Screenshot of the Casibase AI Setting interface showing the configuration of a new provider.

The provider details are as follows:

- Name: provider_njowpc
- Display name: New Provider - njowpc
- Category: Speech-to-Text
- Type: Alibaba Cloud
- Sub type: paraformer-realtime-v1
- Secret key: *** (highlighted with a red box)
- Provider URL: <https://platform.openai.com/account/api-keys>
- State: Active

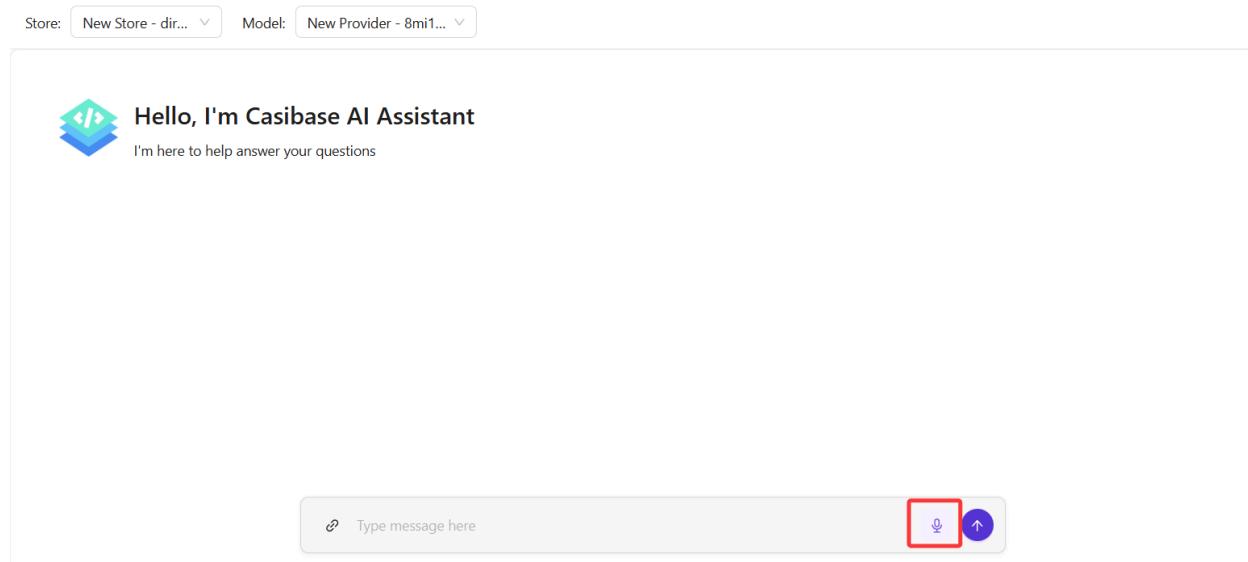
Buttons at the bottom: Save (gray), Save & Exit (purple).

Powered by  casibase

使用语音识别

当您在 Casibase 应用程序中点击语音识别按钮时，将发生以下过程：

1. 浏览器将请求访问您麦克风的权限
2. 一旦授予，系统将开始收听并自动将您的语音转换为文本
3. 在您完成发言后，识别的文本将自动作为消息发送



此功能允许与您的 Casisbase 应用程序进行无操作的互动，使它们更容易访问和使用。

 提示

CasiBase 目前支持以下语音识别提供商：

- [Alibaba Cloud](#)
 - paraformer-realtime-v1

添加存储

我们已经添加了存储提供商、模型提供商和嵌入提供商。现在我们需要配置一个存储来使用这些提供商。现在我们需要配置一个存储来使用这些提供商。

小心

本指南假定您已经部署了Casibase知识库系统。如果您还没有完成，请参考[部署 Casdoor](#)和[Casibase 教程](#)。

此外，本指南假定您已经添加了存储提供商、模型提供商和嵌入提供商。如果您没有，请关注[添加存储提供者](#), [添加一个 AI Model Provider](#), 和 [添加嵌入式提供商](#) 指南。

步骤1：添加新的存储

存储用于将存储、模型和嵌入提供商集成到Casibase中。您可以按照以下步骤添加：您可以按照以下步骤添加它们：

点击主页上的 **Stores** 按钮，然后点击 **Add** 按钮来添加存储。



Home Chat Stores Providers Vectors Chats Message

Stores	Add	Stores	Providers	Vectors	Chats	Message
Name		Display name		Storage provide		
my_store		My_Store		provider_storage		

步骤2：填写存储详情

选择您之前添加的存储提供商、模型提供商和嵌入提供商。

填写存储详情并点击 **Save & Exit** 按钮。

casbin

Home Chat Stores Providers Vectors Chats Messages Tasks Resources P

Edit Store Save

Name:	my_store
Display name:	My_Store
Storage provider:	Provider_storage_1 (provider_storage_1)
Model provider:	Model OpenAI text-davinci-003 (model_openai_text_davinci_003)
Embedding provider:	Embedding_OpenAI_Adasimilarity (embedding_openai_adasimilarity)
File tree:	<pre> └─ My_Store ├─ alibaba_oss │ ├─ audio │ │ └─ AC / DC - Highway To Hell.mp3 (8.34 MB) │ ├─ document │ │ └─ casdoor-knowledge.doc (18.0 KB) │ │ └─ casdoor-knowledge.docx (10.9 KB) │ └─ casdoor-knowledge.html (23.5 KB) └─ image └─ lena.jpg (105 KB) └─ lena.tiff (768 KB) └─ video └─ my_video.mkv (456 KB) </pre>

点击 **Save & Exit** 按钮并返回存储列表页面：

casbin

Home Chat Stores Providers Vectors Chats Messages Tasks Resources Permissions Logs

Jimmy

Stores Add

Name	Display name	Storage provider	Model provider	Embedding provider	Action
my_store	My_Store	provider_storage_1	model_openai_text_davinci_003	embedding_openai_adasimilarity	View Refresh Vectors Edit Delete

< 1 >

现在，您可以使用存储来存储知识库数据、将文本转换为向量，并与聊天机器人对话。

在下一节中，我们将学习如何在Casibase中与聊天机器人对话。

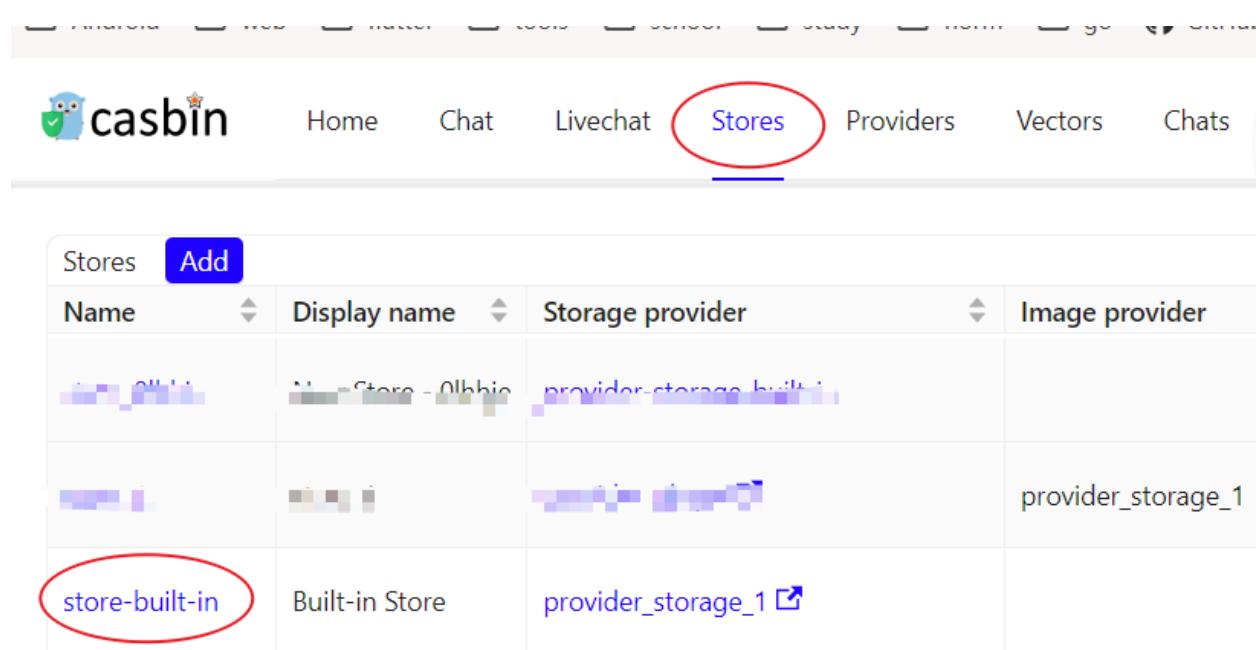
支持多存储

多存储模式为用户在每个不同的存储中提供不同的模型、建议等。

步骤1：启用多存储

首先，您应该在内置存储中启用多存储模式。

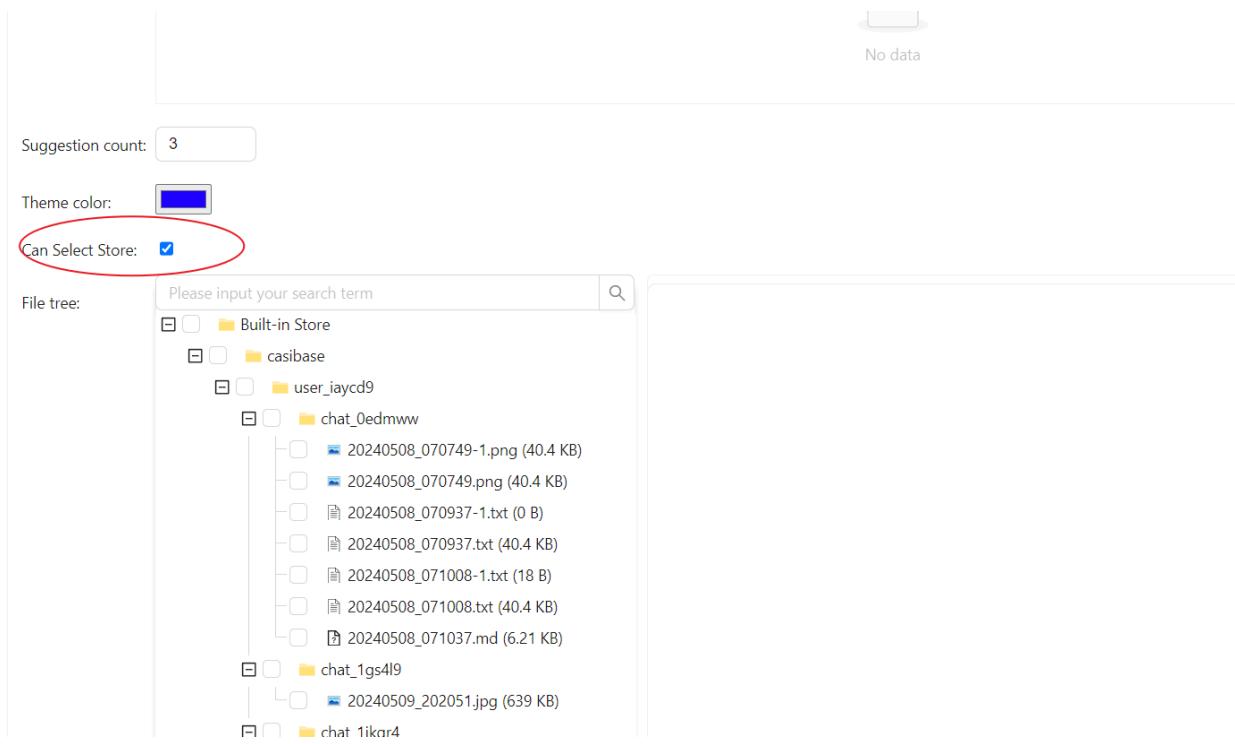
点击主页上的 Stores 按钮，然后点击 store-built-in 按钮进入内置存储。



The screenshot shows the Casibase interface with the 'Stores' tab selected. A red circle highlights the 'store-built-in' entry in the table below. The table has columns: Name, Display name, Storage provider, and Image provider. The 'store-built-in' entry has a blue border around its 'Name' field.

Name	Display name	Storage provider	Image provider
store-built-in	Built-in Store	provider_storage_1	

向下滚动找到 Can Select Store 字段，勾选它。



步骤2：添加可用存储

多存储模式仅提供可用的存储。要使存储可用，您需要配置其存储提供商、模型提供商和嵌入提供商。

步骤3：为对话选择存储

Casibase提供了一种非常方便的选择存储的方法。



Home Chat Livechat Stores Providers Vectors Chats Messages Usages Frameworks

The screenshot shows the casbin Chat interface. At the top, there's a navigation bar with links: Home, Chat (which is highlighted in blue), Livechat, Stores, Providers, Vectors, Chats, Messages, Usages, and Frameworks. Below the navigation bar, there's a purple header bar with the text "New Chat - 7". On the left side, there's a sidebar with a red border containing a button labeled "+ New Chat". Below this button, there's a list of storage options: "store_1" and "store-built-in". To the right of the sidebar, the main chat area shows a message from a user: "You are an expert in your field". Below this message, there's a response from a bot: "Thank you for recognizing my expertise. Whether it's related to my specific area of knowledge or expertise to provide insightful answers and solutions to any problems you may have." At the bottom of the main chat area, there are five small blue icons representing different actions: a square, a circle, a triangle pointing up, a triangle pointing down, and a triangle pointing right.

只需将鼠标悬停在"New Chat"上，然后您就可以从下面出现的列表中选择您想要使用的存储。

如果您点击"New Chat"按钮，系统将为您分配一个默认存储。

与AI聊天

本文档是为初学者设计的分步教程。它将指导您完成将嵌入提供商与Casibase（我们强大的知识库系统）集成的过程。本文档是为初学者设计的分步教程。它将指导您完成在Casibase知识库系统中实现AI聊天功能的过程。

简介

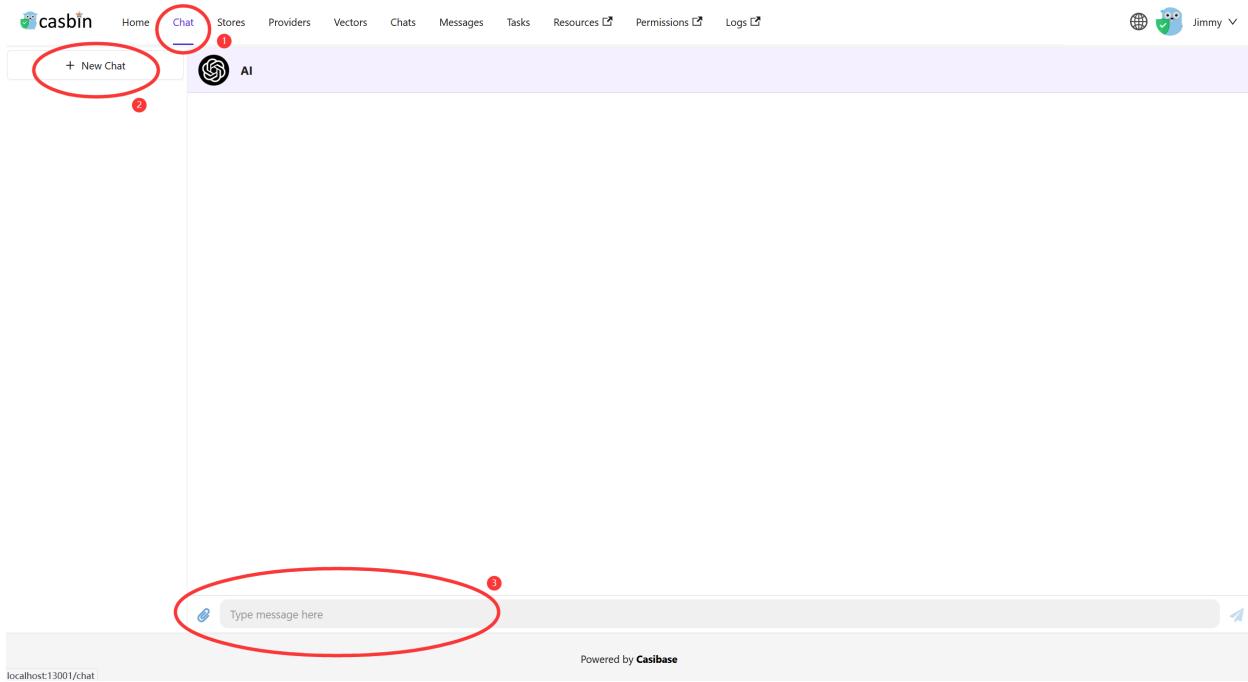
在之前的部分中，我们已经部署了Casdoor和Casibase，将存储提供商、模型提供商和嵌入提供商集成到Casibase中，并添加了一个存储来使用这些提供商。

有关存储的更多信息，请参考我们之前文档的[添加存储](#)部分。

现在，让我们在Casibase中实现AI聊天功能。

步骤1：添加新的聊天

点击主页上的Chats按钮，然后点击New Chat按钮来添加聊天。



步骤2：发送消息

写一条消息并点击 Send 按钮来发送。



步骤3：知识库聊天

此外，您还可以在知识库中与聊天机器人对话。

知识库聊天有一些要求：

- 知识库必须有一个存储。
- 存储必须有一个模型提供商。
- 存储必须有一个嵌入提供商。

- 存储必须有一个存储提供商。
- 存储提供商必须有一个可读的文档（例如markdown文件、docx文件和pdf文件）。

一旦您满足了这些要求，您可以返回到 Stores 页面并点击 Refresh Vectors 按钮来嵌入知识库数据。

Name	Display name	Storage provider	Model provider	Embedding provider	Action
my_store	My_Store	provider_storage_1	model_openai_text_davinci_003	embedding_openai_adasimilarity	<button>View</button> <button>Refresh Vectors</button> <button>Edit</button> <button>Delete</button>

当嵌入正在进行时，按钮将被禁用。

嵌入完成后，您可以点击导航栏中的 Vectors 按钮来查看向量。

结果：

Name	Display name	Store	File	Text	Data	Action
vector_7rss8s	Simplified development	my_store	alibaba_oss/document/casdoor-knowledge.pdf	Simplified development: Casdoor pro...	[{"-0.000106310275,0.02166452,0.02304..."]	<button>Edit</button> <button>Delete</button>
vector_gldg4u	Installation and Deployment: You can	my_store	alibaba_oss/document/casdoor-knowledge.pdf	Installation and Deployment: You ca...	[{"-0.0029990207,0.018568026,-0.00580..."]	<button>Edit</button> <button>Delete</button>
vector_0wrasj	Privilege Control: With Casdoor	my_store	alibaba_oss/document/casdoor-knowledge.pdf	Privilege Control: With Casdoor, de...	[{"0.0054717776,0.017982274,0.0103428..."]	<button>Edit</button> <button>Delete</button>
vector_3tet51	Casdoor Knowledge Points	my_store	alibaba_oss/document/casdoor-knowledge.pdf	Casdoor Knowledge Points Casdoor is...	[{"-0.007692282,0.024387684,0.0001651..."]	<button>Edit</button> <button>Delete</button>

让我们在知识库中与聊天机器人对话。

AI

What's casdoor? Casdoor is an OAuth2 and OIDC based authentication portal designed to help developers easily add user authentication and authorization features to their applications.

与非知识库聊天的结果比较：



AI

What's casdoor?



Casdoor is an online development platform for web and mobile applications that helps companies create and manage an end-to-end development process. It allows businesses to build, deploy, monitor, and manage apps quickly and cost effectively. It is equipped with powerful analytics tools to track the performance and usage of an app, as well as allowing companies to quickly identify and address any issues or problems.

⚠ 小心

嵌入速率与两个因素有关：

- 知识库中的文档：
 - 文档数量：文档越多，嵌入时间越长。
 - 文档大小：文档大小越大，嵌入时间越长。
- 嵌入提供商：
 - API速率限制：API速率限制越高，嵌入速度越快。
 - API并发：API并发越高，嵌入速度越快。

例如，如果您使用[OpenAI API](#)作为嵌入提供商，嵌入速率与[OpenAI API](#)的速率限制和并发有关。

结论

在本指南中，我们学习了如何在Casibase中实现AI聊天功能。

现在，您可以在Casibase中与聊天机器人对话了。尽情享受吧！好好享受！

有关Casibase的更多信息，可以在我门文档的[核心概念](#)部分找到。

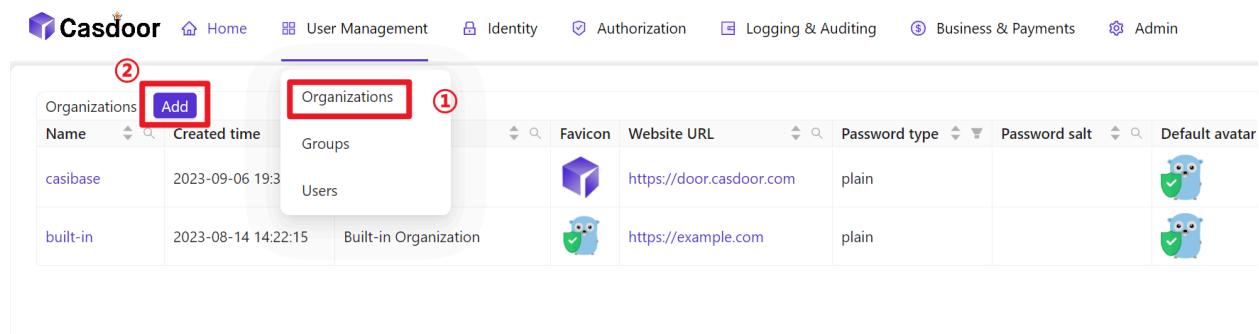
Casdoor-SSO

Casibase 使用 Casdoor 作为其身份认证和单点登录(SSO)提供商。请确保提前部署好 Casdoor。请确保提前部署它。

请参考 [Casdoor 服务器安装指南](#) 来安装和配置 Casdoor。

按照以下步骤为 Casibase 设置 Casdoor:

- 创建组织



Name	Created time	Favicon	Website URL	Password type	Password salt	Default avatar
casibase	2023-09-06 19:33		https://door.casdoor.com	plain		
built-in	2023-08-14 14:22:15		https://example.com	plain		

- 配置组织信息

casibase

Casibase

Save & Exit

- 创建新应用

Name	Created time	Display name	Logo	Organization	Providers
app-casibase	2023-09-06 19:38:54	Casibase		casibase	provider_captcha_default
app-built-in	2023-08-14 14:22:15	Casdoor		built-in	provider_captcha_default

- 配置应用信息（请记住应用名称、ClientID 和 ClientSecret）

Save & Exit (④)

Name (①): app-casibase

Display name (②): Casibase

Logo (③): URL: https://cdn.casbin.org/img/casdoor-logo_1185x256.png

Preview: 

Home (④):

Description (⑤):

Organization (⑥): casibase (②)

Tags (⑦):

Client ID (⑧): 548c8b9c7431d2621db1 (③)

Client secret (⑨): 2bc7640d487fc4dea6f4b77f07f1bf4433e4ad40

Cert (⑩): cert-built-in

- Create a Certificate: In the Casdoor dashboard, choose Cert → Add, keep Algorithm as RS256 (default), enter a name, and click Save.

New Cert (④) Save & Exit (④) Cancel

Organization (①): casbin

Name (②): cert_casbin (④)

Display name (③): New Cert - casbin

Scope (⑤): JWT

Type (⑥): x509

Crypto algorithm (⑦): RS256 (RSA + SHA256)

Bit size (⑧): 4096

Expire in years (⑨): 20

Certificate (⑩): Copy certificate Download certificate (⑪) Private key (⑫) Copy private key (⑬) Download private key (⑭)

```
-----BEGIN CERTIFICATE-----
MIIE2zCCASoGwIBAgDAAjAMA0GCSqGSIb3DQEBCwUAMCoxD2ANBgNVBAotBmNh
c2JpbEUUMBIGAUEAwvY2Vydj94dGhry3iwfhNMjUwNzAyM1TmQ1WhhNNdLw
NzAyMT2mQ1WjAnQ8QwvQyDVQQKewZjYXNlAwfx4fDASgRnVnBAMMC2hcnfReGx0
b2NyMiCjIaNBlqkjhkG9w0IAOFaaOCa9aMILCc9KAqEa01lWzbk2z7b3jy
xpVdb+ikyngf+eprnOE0f7qK2slqMSWAuIjCWSKsMuIjOjGMfpriaHwpekl
WjWjpAmH4IAUPcR6G+ba2TcaBf8Gua5fRYAVWgtewUra4hSP87cb2UNQe8
zdNxDAgePldZXqoWhzLLEGu1zyCj:ch1XawhHvhnhGW02W184XxJslJq
b63UjC8005Cwvpu9sgSdgIWSoi7i4VwQxjujusCR29dx6yrlx143l0hapNkAT
wqaqv8VfgeI27jQtU/MvgznbillG8f06EZOMMNNS4EcNmRRSAUjUp552fIn0
6j8nxzbeBMsG8bZ6T+2/DP0P0tzlaaxLKMhM7ifGEZ739kXMuflsNtNgfEr
XccQzQqybpzl32vx3iVNuJgOnelylFAOrmLEkjsgogm+j2zdc0Bz+diy3h
-----BEGIN RSA PRIVATE KEY-----
MIICjtAfAAKCAgEAgA01lWzbk2z7b3jyxpVdb+ikyngf+eprnOE0f7qK2slq
SMSWAuIjCWSKsMuIjOjGMfpriaHwpeklWjWjpAmH4IAUPcR6G+ba2TcaBf8Gua5f
tYAVWgtewUra4hSP87cb2UNQe8zDUnQce5zdn0XAgcsPldZXqoWhzLLEGu1zyQ
CjIHLXawhHvhnhGW02W184XxJslJq:ch1XawhHvhnhGW02W184XxJslJq
XjijUSCR29dx6yrlx143l0hapNkATwqaqv8VfgeI27jQtU/AvgzbgnlbtG6
f06EZMNNLj4AfNmRRAUjUp552fIn068ndZrbeBMsG8bZ6T+2/DP0P0tzlaaxL
KMcM7ifGEZ739kXMuflsNtNgfErXccQzQqybpzl32vx3iVNuJgOnelylFAOrm
LEkjsgogm+j2zdc0Bz+diy3h
-----END RSA PRIVATE KEY-----
```

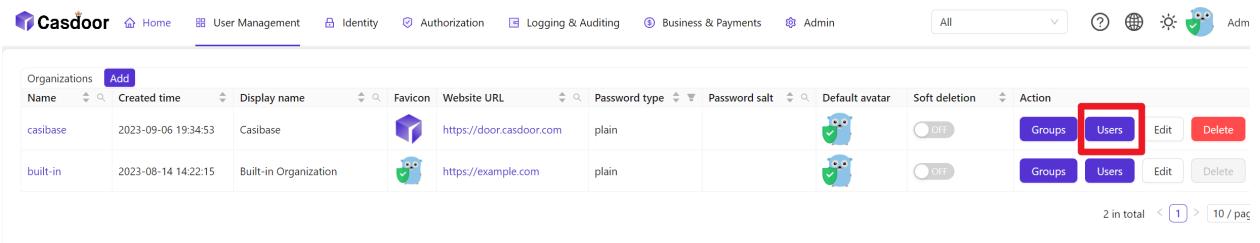
Cert (⑮): cert_casbin

- Bind the Certificate to the Application: Open the Config tab of your newly

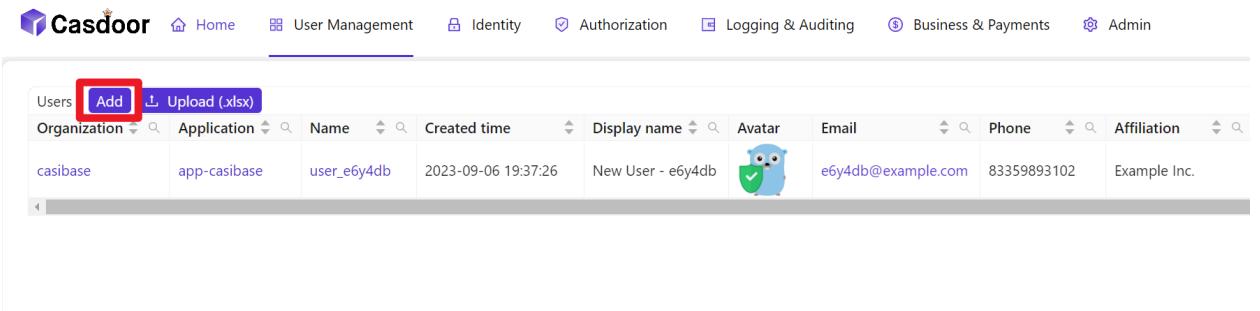
created Application, select the certificate you just created from the Cert dropdown, and click Save.



- 在新创建的组织中添加成员



Name	Created time	Display name	Favicon	Website URL	Password type	Password salt	Default avatar	Soft deletion	Action
casibase	2023-09-06 19:34:53	Casibase		https://door.casdoor.com	plain			<input checked="" type="radio"/>	Groups Users Edit Delete
built-in	2023-08-14 14:22:15	Built-in Organization		https://example.com	plain			<input checked="" type="radio"/>	Groups Users Edit Delete



Organization	Application	Name	Created time	Display name	Avatar	Email	Phone	Affiliation
casibase	app-casibase	user_e6y4db	2023-09-06 19:37:26	New User - e6y4db		e6y4db@example.com	83359893102	Example Inc.

- 配置成员信息（请记住用户名和密码）

Casdoor

Home User Management Identity Authorization Logging & Auditing Business & Payments Admin All

Edit User Save Save & Exit ④

Organization ③: casibase

ID ③: 97a6ce88-be20-4840-b8d4-b2ebb255d0ee

Name ③: user_e6y4db ①

Display name ③: New User - e6y4db

Avatar ③: Preview:

Avatar preview: A blue cartoon bear holding a green shield with a white checkmark.

Upload a photo...

User type ③: normal-user

Password ③: Modify password... ②

Email ③: e6y4db@example.com

Phone ③: +1 83359893102

Homepage ③:

Bio ③:

Tag ③: staff

Language ③:

Gender ③:

Birthday ③:

Education ③:

Score ③: 0

Karma ③: 0

Ranking ③: 1

Signup application ③: app-casibase ③

Groups ③:



>

Developer Guide

Developer Guide



Generating Swagger Files

Generating Swagger Files

Generating Swagger Files

Overview

As we know, the beego framework provides support for generating swagger files to clarify the API via the command line tool called "bee". Casibase is also built based on beego. However, we found that the swagger files generated by bee failed to categorize the APIs with the "@Tag" label. So, we modified the original bee to implement this function.

How to write the comment

Most rules are exactly identical to the original bee comment formats. The only discrepancy is that the API shall be divided into different groups according to the "@Tag" label. Therefore, developers are obliged to ensure that this tag is correctly added. Here is an example:

```
// @Title Login
// @Tag Login API
// @Description login
// @Param oAuthParams     query    string  true      "oAuth
parameters"
// @Param body    body    RequestForm  true      "Login
information"
// @Success 200 {object} controllers.api_controller.Response The
Response object
// @router /login [post]
func (c *ApiController) Login() {
```

APIs with the same "@Tag" labels will be put into the same group.

How to generate the swagger file

0. Write comments for the API in the correct format.
1. Fetch this repository: <https://github.com/casbin/bee>.
2. Build the modified bee. For example, in the root directory of casbin/bee, run the following command:

```
go build -o mybee .
```

3. Copy mybee to the base directory of Casibase.
4. In that directory, run the following command:

```
mybee generate docs
```

5. (Optional) If you want to generate swagger document for specific tags or apis, here are some example commands:

```
mybee generate docs --tags "Adapter API"  
mybee generate docs --tags "Adapter API,Login API"  
mybee generate docs --apis "add-adapter"  
mybee generate docs --apis "add-adapter,delete-adapter"
```

Notably: We only accept a comma  as the separator when multiple tags/apis provided.

Then you will find that the new swagger files are generated.



>

部署

部署



部署 Casdoor 和 Casibase

了解如何部署 Casdoor 和 Casibase。

部署 Casdoor 和 Casibase

简介



提示

什么是 Casdoor?

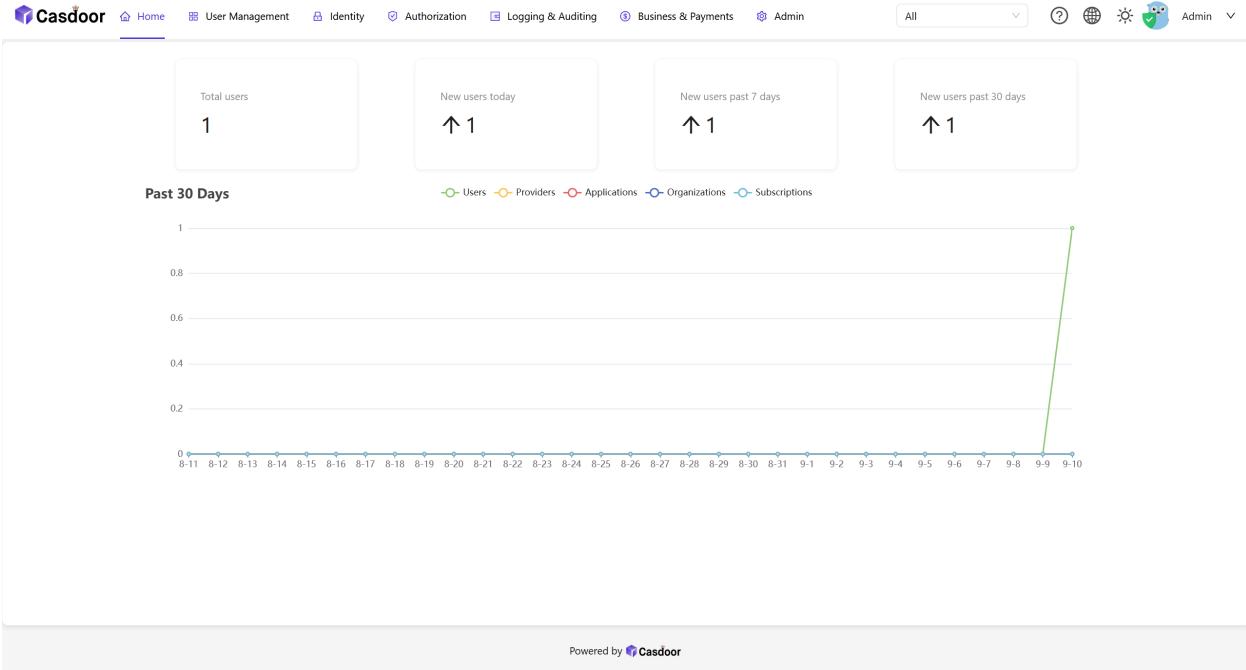
Casdoor 是一个强大的身份认证系统，提供安全可靠的登录体验。它是 Casibase 的前置要求，所以请确保先部署它。由于 Casibase 是前提条件，所以请务必先部署它。

访问 [Casdoor](#) 网站了解更多信息。

步骤 1：部署 Casdoor

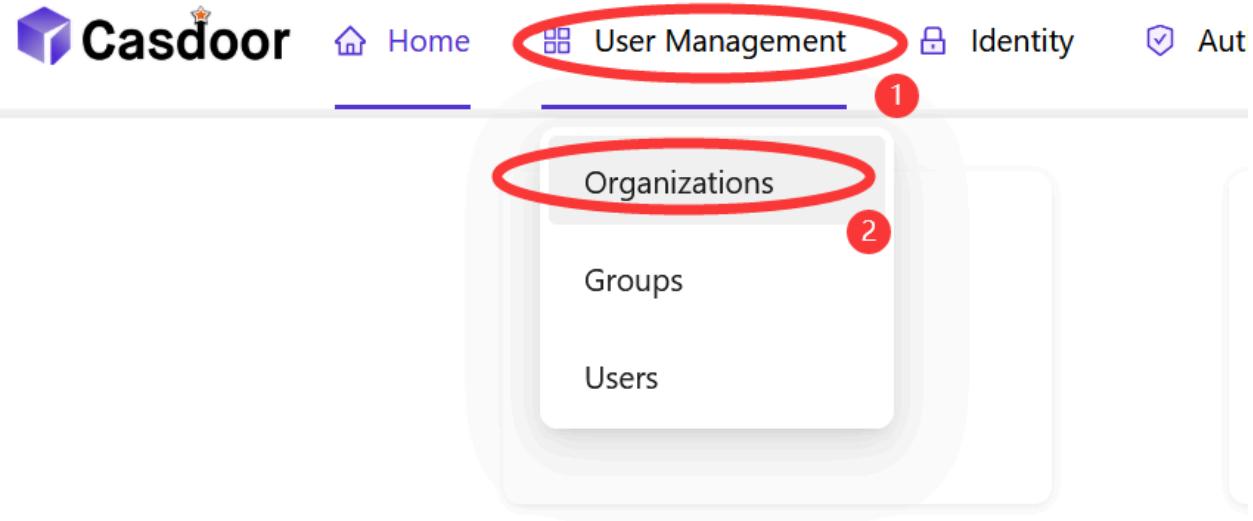
在 [Casdoor 部署指南](#) 中，你可以找到部署 Casdoor 的详细步骤。

部署完成 Casdoor 后，你将会看到如下效果：



步骤 2：在 Casdoor 中创建组织

在 Casdoor 中，你可以创建一个组织来管理你的用户和应用。你可以通过点击主页上的 **用户管理 - 组织** 按钮来创建组织。你可以通过点击主页上 **用户管理 - 组织** 按钮来创建一个组织。



Past 30 Days

步骤 2.1: 添加组织

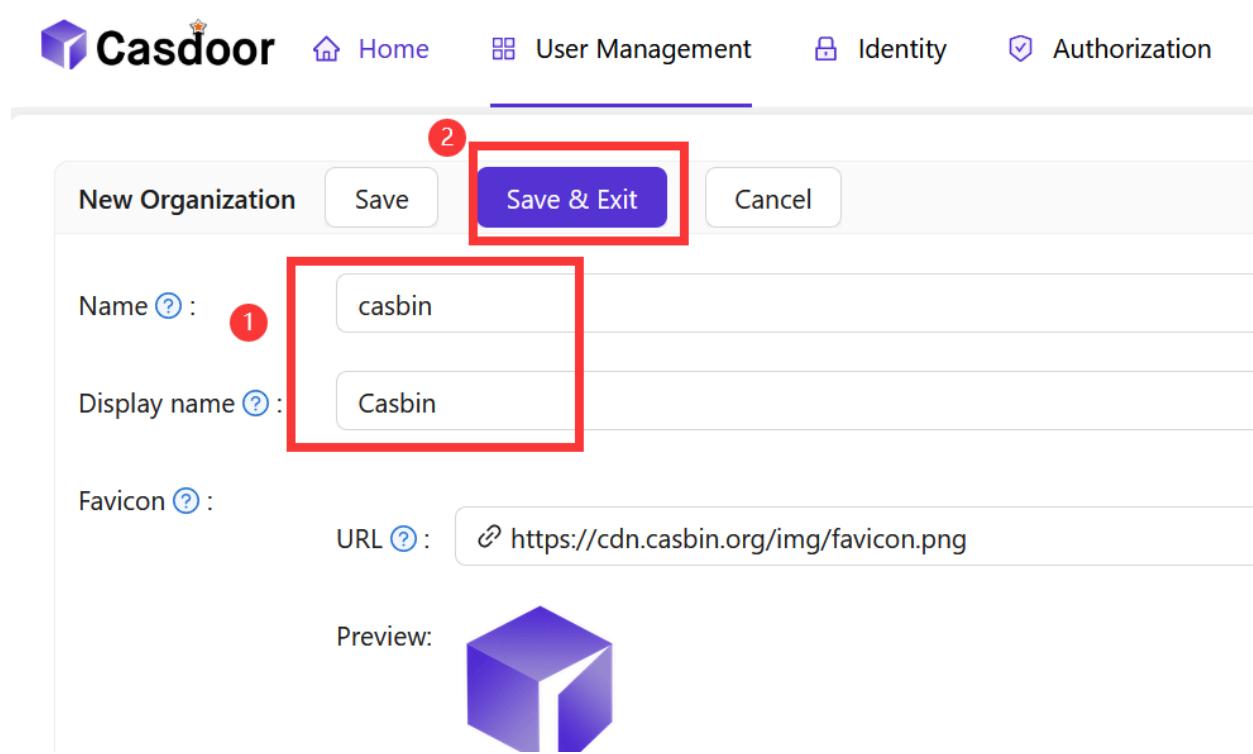
点击 **添加** 按钮来添加一个组织。

The screenshot shows the 'Organizations' management page. At the top, there is a search bar labeled 'Organizations' and a blue 'Add' button, both of which are circled in red. Below the search bar is a table with three columns: 'Name', 'Created time', and 'Display name'. A single row is visible, showing 'built-in' in the Name column, '2023-09-10 19:31:50' in the Created time column, and 'Built-in Organization' in the Display name column.

Name	Created time	Display name
built-in	2023-09-10 19:31:50	Built-in Organization

步骤 2.2：填写组织信息

填写组织信息并点击 **保存并退出** 按钮。



New Organization

Name ②: casbin

Display name ①: Casbin

Save Save & Exit Cancel

Favicon ②:

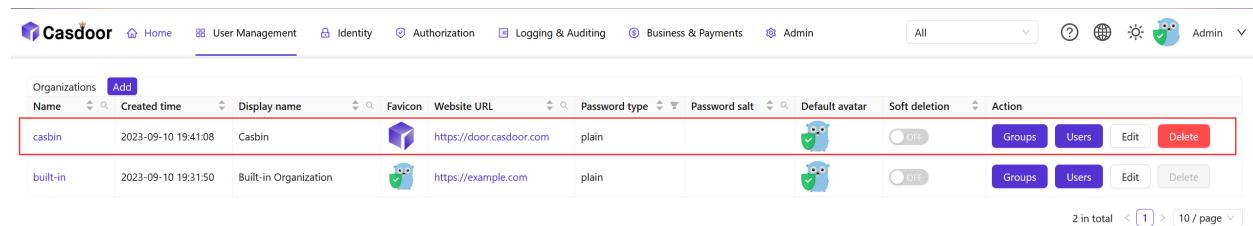
URL ①: <https://cdn.casbin.org/img/favicon.png>

Preview:



步骤 2.3：查看组织

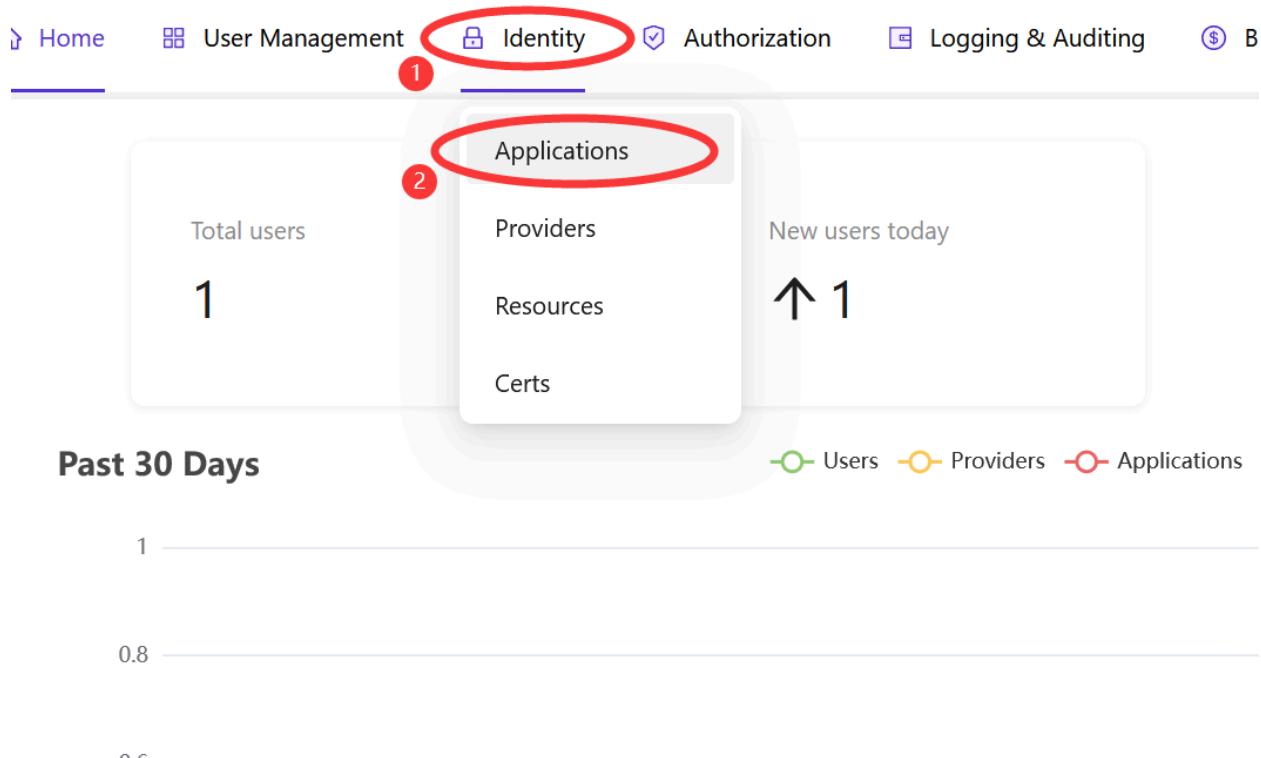
添加组织后，你可以查看组织信息。



Name	Created time	Display name	Favicon	Website URL	Password type	Password salt	Default avatar	Soft deletion	Action
casbin	2023-09-10 19:41:08	Casbin		https://door.casdoor.com	plain			<input checked="" type="radio"/> OFF	<button>Groups</button> <button>Users</button> <button>Edit</button> <button>Delete</button>
built-in	2023-09-10 19:31:50	Built-in Organization		https://example.com	plain			<input checked="" type="radio"/> OFF	<button>Groups</button> <button>Users</button> <button>Edit</button> <button>Delete</button>

步骤 3：在 Casdoor 中创建应用

在 Casdoor 中，你可以创建一个应用程序来管理你的用户和组织。你可以通过点击主页上 **身份 - 应用程序** 按钮来创建一个应用程序。



步骤 3.1：添加应用

点击 **添加** 按钮来添加一个应用。



步骤 3.2：填写应用信息

填写应用信息并点击 **保存并退出** 按钮。

Casdoor Home User Management Identity Authorization Logging & Auditing Business & Payments Ad

New Application Save Save & Exit Cancel

Name ? : app-casibase 5

Display name ? : Casibase 1

Logo ? : URL ? : https://cdn.casbin.org/img/casdoor-logo_1185x256.png

Preview: 

Home ? :

Description ? :

Organization ? : casbin 2

Tags ? :

Client ID ? : 2786e0cbadfb56287a9a 3

Client secret ? : 4f9957d3e679efdb3391eb42b38d274d46fa1232

Cert ? : cert-built-in

Redirect URLs ? :

Redirect URLs Add

Redirect URI 4

<http://localhost:14000/callback>

步骤 3.3：查看应用

添加应用后，你可以查看应用信息。

Casdoor Home User Management Identity Authorization Logging & Auditing Business & Payments Admin

All

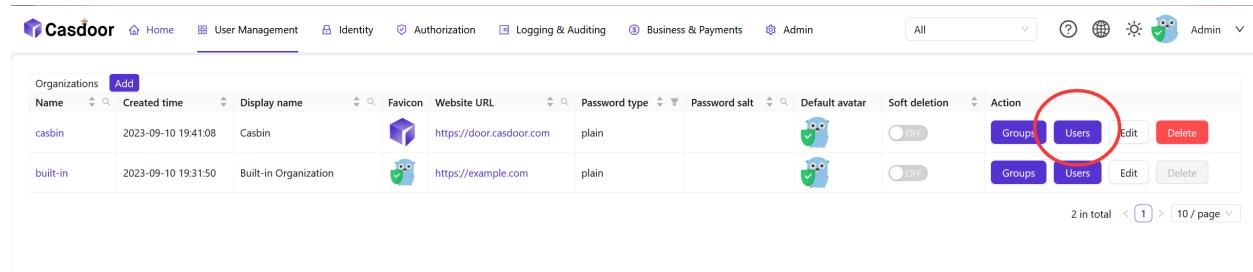
Applications Add

Name	Created time	Display name	Logo	Organization	Providers	Action
app-casibase	2023-09-10 19:44:08	Casibase		casbin	provider_captcha_default	<button>Edit</button> <button>Delete</button>
app-built-in	2023-09-10 19:31:50	Casdoor		built-in	provider_captcha_default	<button>Edit</button> <button>Delete</button>

2 in total < 1 > 10 / page

步骤 4：在 Casdoor 中为 Casibase 创建用户

在 Casdoor 中，你可以创建一个用户来登录 Casibase。你可以通过点击主页上的 **用户管理 - 组织 - 用户** 按钮来创建用户。你可以通过点击主页上 **用户管理 - 组织 - 用户** 按钮来创建一个用户。



The screenshot shows the Casdoor User Management interface. In the top navigation bar, the 'User Management' tab is selected. Below it, the 'Organizations' section lists two entries: 'casbin' and 'built-in'. For each organization, there are columns for Name, Created time, Display name, Favicon, Website URL, Password type, Password salt, Default avatar, Soft deletion, and Action. The 'Action' column contains three buttons: 'Groups', 'Users' (which is highlighted with a red circle), and 'Edit'. At the bottom right of the table, it says '2 in total' and '10 / page'.

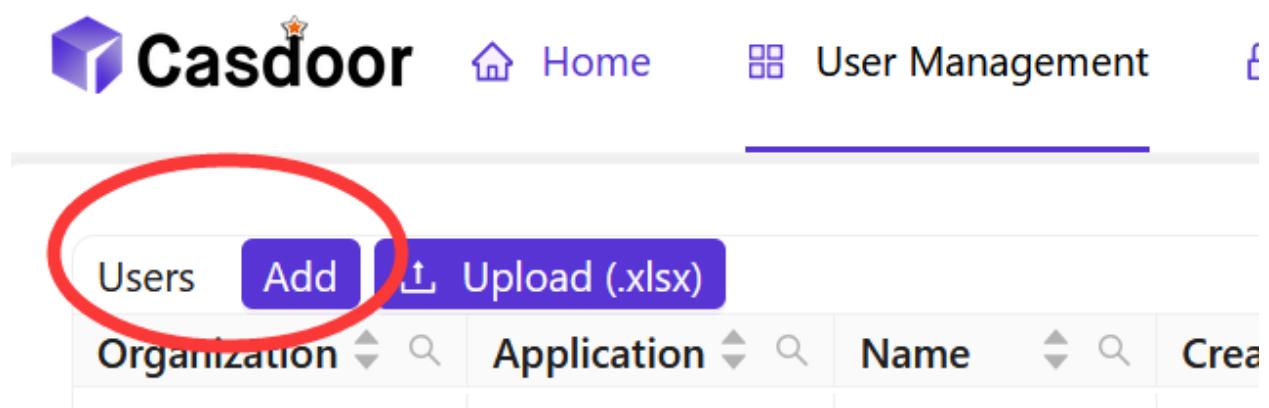


用户是组织中的成员，可以登录该组织中的应用。

访问 [Casdoor](#) 网站了解更多信息。

步骤 4.1：添加用户

点击 **添加** 按钮来添加一个用户。



The screenshot shows the Casdoor User Management interface. The top navigation bar has 'Home' and 'User Management' tabs. Below the navigation, a search bar includes fields for 'Organization', 'Application', 'Name', and 'Create'. A prominent red circle highlights the 'Add' button in the top left of the user list area.

步骤 4.2：填写用户信息

填写用户信息并点击 **保存并退出** 按钮。

New User Save Save & Exit Cancel

Organization ②: casbin 1

ID ②: d5bc730c-312c-406e-ae03-e6580d7590f4

Name ②: jimmy

Display name ②: Jimmy 2

Avatar ②: Preview:

User type ②: normal-user

Password ②: Modify password... 3

Email ②: t414w5@example.com

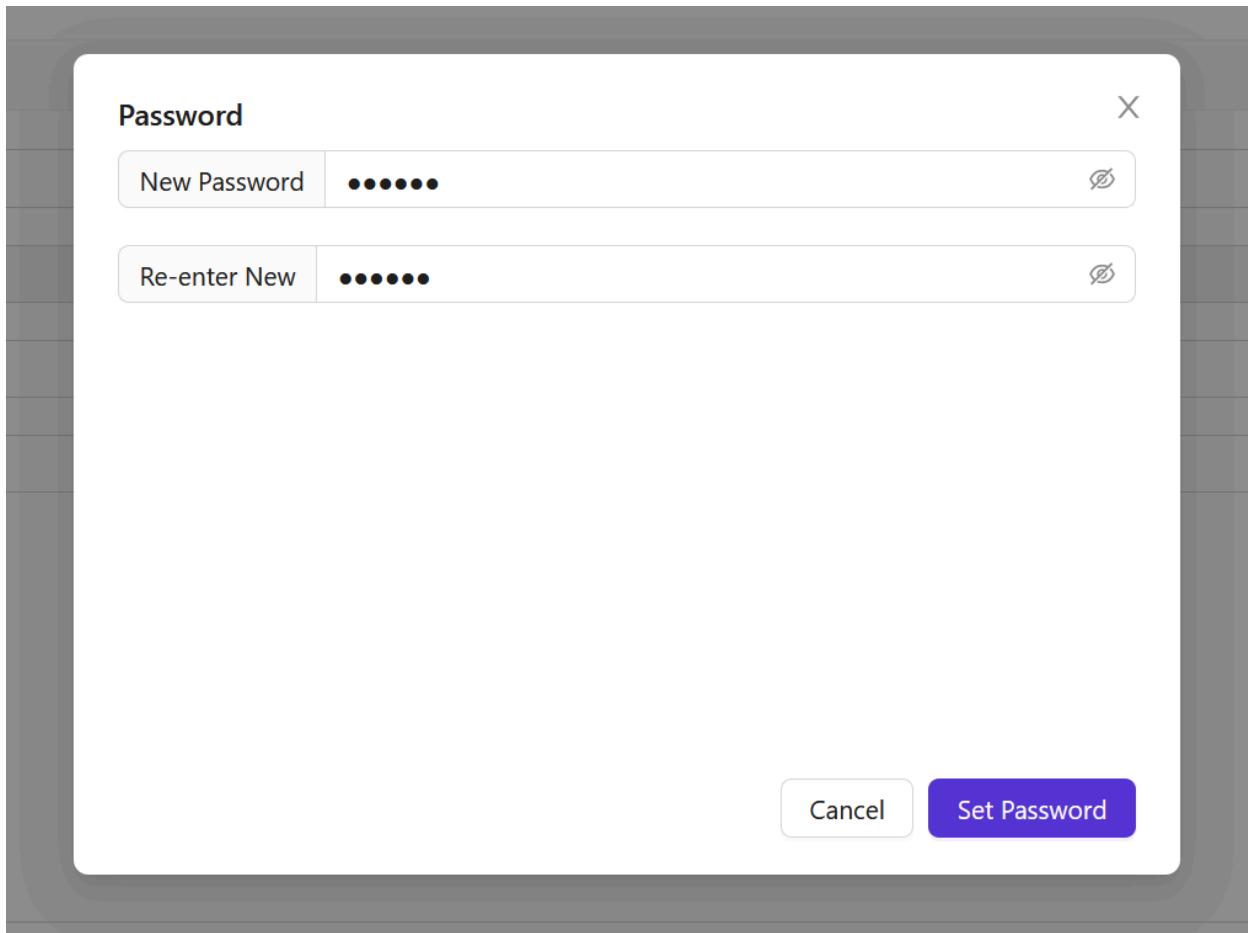
Phone ②: +1 71700415009

Country/Region ②: Please select country/region

Location ②:

- 密码

你可以通过点击 **修改密码** 按钮来设置用户密码。



- 管理员

你可以通过点击 是否为管理员 按钮来设置用户的管理员权限。

Permissions [?](#) :

Groups [?](#) :

3rd-party logins [?](#) :

Is admin [?](#) :



Is forbidden [?](#) :



Is deleted [?](#) :



Multi-factor authentication [?](#) :

Multi-factor methods

步骤 4.3：查看用户

添加用户后，你可以查看用户信息。

The screenshot shows the Casdoor User Management page. At the top, there are navigation links: Home, User Management, Identity, Authorization, Logging & Auditing, Business & Payments, Admin, and a search bar. Below the header is a table with the following data:

Organization	Application	Name	Created time	Display name	Avatar	Email	Phone	Affiliation	Country/Region	Tag	Is ac	Action
casbin	app-built-in	jimmy	2023-09-10 20:51:18	Jimmy		t414w5@example.com	71700415009	Example Inc.			Edit	Delete

At the bottom right of the table, it says "1 in total".

步骤 5：部署 Casibase

与 Casdoor 类似，你可以按照 [Casibase 部署指南](#) 来部署 Casibase。

部署完成 Casibase 后，你将会看到如下效果：



[Home](#) [Stores](#) [Providers](#) [Vectors](#) [Chats](#) [Messages](#) [Tasks](#) [Resources](#) [Permissions](#) [Logs](#)

Jimmy ▾

Powered by **Casibase**



> How to Connect to Casibase

How to Connect to Casibase



Overview

Learn about different ways to connect to and integrate with Casibase.



Casibase SDKs

Learn how to integrate and use Casibase SDKs with your applications.



Using Casibase OpenAI API Compatible Interface

Learn how to connect external chat UIs to Casibase using OpenAI API compatibility.

Overview

Overview

In this section, we will show you how to connect your application to Casibase.

Casibase provides two main methods for integrating with your applications:

- [Casibase SDK](#) - For direct integration with Casibase's API
- [OpenAI API Compatibility](#) - For connecting existing OpenAI-compatible UIs and clients

Casibase SDK

What is Casibase SDK?

Casibase SDK provides a programmatic way to interact with Casibase services. It offers a convenient set of APIs that allow developers to manage tasks, knowledge bases, and other features of Casibase directly from their applications.

We recommend using the Casibase SDK for the following reasons:

1. It provides direct access to Casibase-specific functionality
2. It simplifies authentication and configuration
3. It handles error cases and provides a more developer-friendly experience

Currently, Casibase offers a Java SDK, with more language support planned for the future.

OpenAI API Compatibility

What is OpenAI API Compatibility?

Casibase supports the OpenAI API format, allowing you to connect any OpenAI-compatible chat UI or client application to Casibase. This makes it easy to use popular open-source chat interfaces with Casibase's backend.

We recommend using the OpenAI API compatibility for the following reasons:

1. It allows you to use your preferred chat UI with Casibase
2. It simplifies integration if you're already using OpenAI-compatible tools
3. It provides a standardized way to interact with Casibase's AI capabilities

This approach is particularly useful if you want to quickly integrate Casibase with existing applications that already support the OpenAI API format.

Casibase SDKs

Introduction

Casibase provides SDKs to help developers integrate with Casibase's APIs more easily. The SDKs offer a convenient way to interact with Casibase's services for tasks like managing AI conversations, knowledge bases, and more.

Currently, Casibase offers a Java SDK, with more language support planned for the future.

Backend SDK	Description	SDK code	Example code
Java SDK	For Java backends	casibase-java-sdk	-

How to use Casibase SDK?

1. Backend SDK configuration

When your application starts up, you need to initialize the Casibase SDK config by providing the required parameters.

Take casibase-java-sdk as an example:

```
CasibaseConfig config = new CasibaseConfig(  
    "https://demo-admin.casibase.com", // endpoint
```

All the parameters for initialization are explained as follows:

Parameter	Required	Description
endpoint	Yes	Casibase Server URL, like https://demo-admin.casibase.com or http://localhost:14000
clientId	Yes	Client ID for the Casibase application
clientSecret	Yes	Client secret for the Casibase application
organizationName	Yes	The name for the Casibase organization, e.g., casbin
applicationName	No	The name for the Casibase application, e.g., app-casibase

2. Available Services

Once you have initialized the configuration, you can create and use the available services. Currently, the only available service is `TaskService`.

```
TaskService taskService = new TaskService(config);
```

TaskService

`TaskService` supports basic task operations, such as:

- `getTask(String name)`: Get a single task by task name.
- `getTasks()`: Get all tasks under the `organizationName`.

- `addTask(Task task)`: Add a new task to the database.
- `updateTask(Task task)`: Update an existing task in the database.
- `deleteTask(Task task)`: Delete a task from the database.



Using Casibase OpenAI API Compatible Interface

This document is a step-by-step tutorial designed for beginners. It will guide you through the process of connecting external chat UIs to Casibase using its OpenAI API compatibility feature.

Introduction

Casibase now supports integration with external chat UIs through OpenAI API compatibility. This feature allows you to use popular open-source chat interfaces with Casibase's backend, giving you more flexibility in how you interact with your knowledge base system.

If you're looking to use your preferred chat UI with Casibase, this guide will walk you through the simple setup process.

Step 1: Set Up Casibase with a Model Provider

Before connecting an external UI, ensure you have Casibase properly set up with a model provider. If you haven't done this yet, please refer to the [Add an AI Model Provider](#) tutorial.

Step 2: Get Your OpenAI-compatible API Key

When you create a model provider in Casibase, an API key is automatically generated. This key allows external applications to communicate with Casibase

using the OpenAI API format.

Step 2.1: Access Your API Key

Navigate to the **Providers** section and select your model provider. Only administrators can view and modify API keys.

Name:	provider_prm93r
Display name:	New Provider - prm93r
Category:	Model
Type:	OpenAI
Sub type:	text-davinci-003
Client secret	(?)
Temperature:	1.00
Top P:	1.00
Presence penalty:	0.00
Frequency penalty:	0.00
API key:	sk-UflKsbjYjzBvjeUFjbDpxuKg
Provider URL:	https://platform.openai.com/account/api-keys
State :	Active



If the API key field is empty, Casibase will automatically generate a new key when you save the provider.

Step 3: Configure Your External Chat UI

Once you have your API key, you can configure your external chat UI to connect to Casibase.

Step 3.1: Configure with chatgpt-web

For this example, we'll use [chatgpt-web](#), a popular open-source ChatGPT interface.

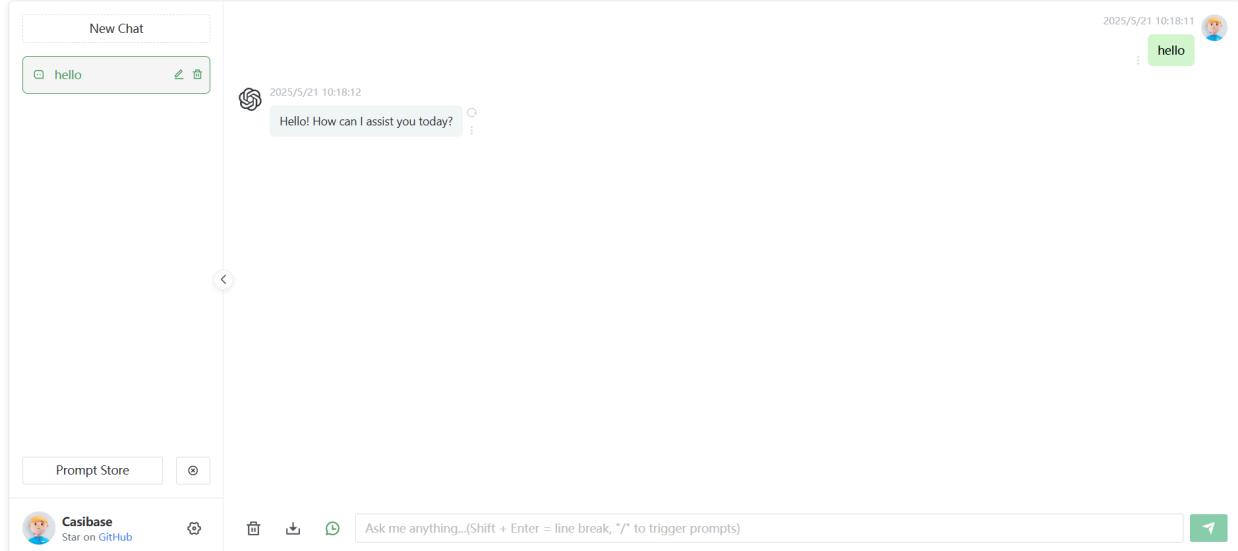
1. Locate the `service/.env` file in your chatgpt-web installation
2. Modify the following variables:
 - `OPENAI_API_KEY`: Set this to your Casibase-generated API key
 - `OPENAI_API_BASE_URL`: Set this to `http://your-casibase-backend:port/api`

```
# Example configuration
OPENAI_API_KEY=sk-UflKsbiYjzBvjeUFJbDpxuKg
OPENAI_API_BASE_URL=http://localhost:14000/api
```

Make sure your Casibase backend is accessible from the machine running your chat UI. Check firewall settings if you encounter connection issues.

Step 4: Test Your Integration

Start your chat UI application and test the connection. You should now be able to interact with Casibase through your preferred interface. If everything is set up correctly, you should see responses from Casibase in your chat UI.



Compatible Chat UIs

Casibase's OpenAI API compatibility has been tested with these popular chat interfaces:

- [chatgpt-web](#)

Other chat UIs that use the standard OpenAI API format should also work with Casibase.



>

提供商

提供商

概述

提供商概述

模型提供商

简介

嵌入提供商

简介

存储提供商

简介

 **文本转语音提供商**

简介

 **语音识别提供商**

简介

 **Blockchain Providers**

2 个项目

 **Private Cloud Providers**

1 个项目

概述

Casibase 是一个开源的 AI 知识库系统，旨在为企业提供高效且灵活的知识管理与对话解决方案。其核心功能之一是 Providers，它允许用户整合多个 AI 模型和存储服务，以提升系统的功能性和性能：Providers 分为三大类：模型提供者、嵌入式提供者和存储提供者，其中模型提供者与嵌入式提供者统称为 AI 提供者，与存储提供者一起分别负责处理 AI 模型和数据存储。

1. 模型提供商

模型提供商是 Casibase 中用于集成和管理 AI 模型的组件。它允许用户将各种预训练的 AI 模型集成到系统中，以实现更智能的知识处理和对话生成。通过模型提供商，用户可以轻松地在不同的 AI 模型之间切换，根据具体需求选择最合适的模型。它允许用户将各种预训练的 AI 模型整合到系统中，从而实现更智能的知识处理和对话生成。借助模型提供者，用户可以轻松切换不同的 AI 模型，根据特定需求选择最合适的模型。

Casibase 支持多种流行的 AI 模型，包括但不限于：

模型提供商类型

- Hugging Face：如 meta-llama/Llama-2-7b、THUDM/chatglm2-6b
- OpenAI：如 gpt-3.5-turbo、gpt-4
- Claude：如 claude-2、claude-instant-v1
- 文心一言：如 ERNIE-Bot、ERNIE-Bot-turbo

2. 嵌入提供商

数据向量化

嵌入式提供者的主要作用是将各种类型的数据（例如，文本、图像等）进行转换 转换为稠密的向量表示。这种转换是 Casibase 中数据处理和分析的关键步骤，使得数据能够以更高效的方式存储、检索和分析。

知识检索

通过将知识库中的数据和用户的查询都转换为向量，嵌入提供商使系统能够基于向量相似度进行快速的知识检索。这大大提高了知识库检索的效率和准确性。这极大地提高了知识库检索的效率和准确性。

灵活的模型支持

嵌入提供商支持多种嵌入模型，用户可以根据需求选择最适合的模型。

3. 存储提供商

我们可以在 Casdoor 中配置存储提供者。并将其用于 Casibase，该组件用于管理 Casibase 的数据存储和检索。它允许用户将数据存储在不同的存储服务中，并通过统一的界面访问这些数据。借助存储提供者，用户可以灵活选择存储服务，以确保数据安全和高效访问。支持两种类型的存储：本地和云端。

4. 语音合成提供商

语音合成提供商是Casibase的一个组成部分，能够将文本答复转换成自然语音。它使

系统能够通过语音合成与用户进行交流，从而增强知识库系统的互动体验。

提供商支持

Currently, Casibase supports Alibaba Cloud's Text-to-Speech service, with various voice options available through the cosyvoice-v1 interface. 该系统设计为可扩展，以便未来集成更多的文本转语音服务提供商。

5. 语音识别提供商

语音识别提供商是Casibase的一个组成部分，可以将语音转化为书面文本。它使该系统能够理解和处理语音查询，加强知识库系统的互动经验。

本地

我们支持将文件上传到本地系统。

云端

我们支持 AWS S3、Azure Blob Storage、MinIO、阿里云 OSS、腾讯云 COS，并且我们正在不断添加更多的云存储服务。

模型提供商

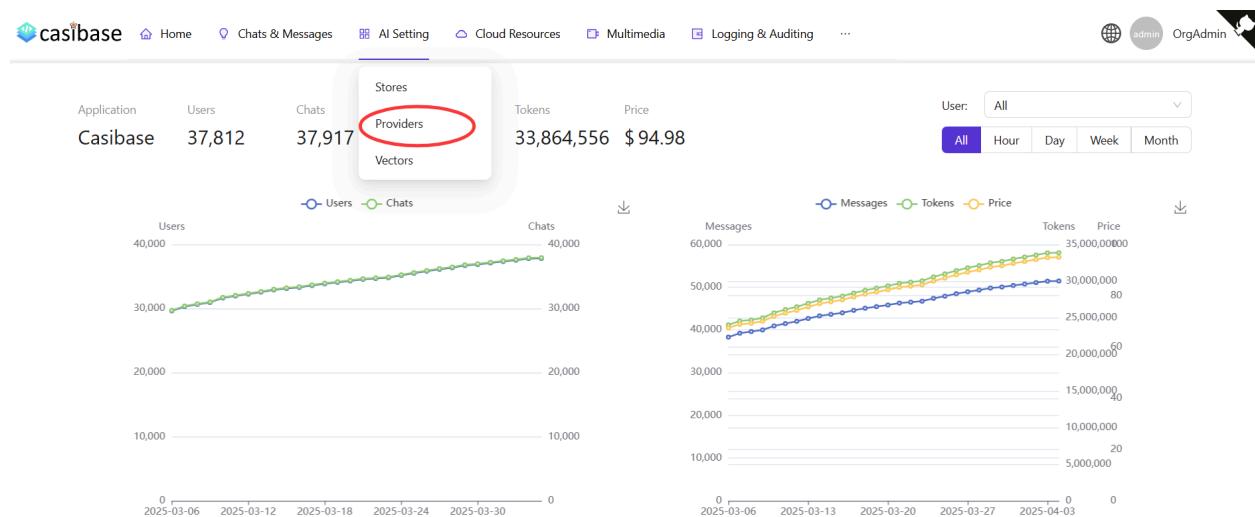
简介

向 Casibase 添加模型提供商可以通过整合机器学习模型和 AI 功能来增强其功能。模型提供商使您能够分析和处理知识库系统中的数据，使其更加智能和高效。模型提供商允许您在知识库系统中分析和处理数据，使其更智能、更高效。

添加模型提供商

模型提供商用于将LLM集成到Casibase中。您可以按照以下步骤添加。您可以按照以下步骤添加它们：

点击主页上的 **提供商** 按钮。



点击 **添加** 按钮来添加模型提供商。

Providers	Add	Add Storage Provider						
Name		Display name	Category	Type	Sub type	API key	Secret key	Region
provider_tts_alibabacloud_cosyvoice		Provider TTS AlibabaCloud Cosyvoice	Text-to-Speech	Alibaba Cloud	cosyvoice-v1		***	
provider_blockchain_chainmaker		Provider Blockchain ChainMaker	Blockchain	Tencent ChainMaker (Demo Network)	text-davinci-003	AKIDObZHjqUrxnNltkKdfQvAWcS1JK4XidG	***	ap-beijing
provider_model_alibabacloud_deepseek_r1		Provider Model AlibabaCloud DeepSeek-R1	Model	Alibaba Cloud	deepseek-r1		***	
provider_cloud_alibabacloud		Provider Cloud AlibabaCloud	Public Cloud	Aliyun	text-davinci-003	LTAI5tHkUsopAioN6xi2LMg	***	cn-beijing
dall-e-3		dall-e-3	Model	OpenAI	dall-e-3		***	
provider_model_azure_gpt4_1		Provider Model Azure GPT-4	Model	Azure	gpt-4o	deployment-gpt-4o	***	
provider_model_azure_gpt4_2		Provider Model Azure GPT-4	Model	Azure	gpt-4o	deployment-gpt-4o	***	
provider_model_azure_gpt4_3		Provider Model Azure GPT-4	Model	Azure	gpt-4o	deployment-gpt-4o	***	
provider_embedding_openai_v3		Provider Embedding OpenAI V3	Embedding	OpenAI	text-embedding-ada-002		***	
provider_model_openai_gpt4_vision		Provider Model OpenAI GPT4 Vision	Model	OpenAI	gpt-4-vision-preview		***	

填写模型提供商详情

填写模型提供商详情并点击 **保存并退出** 按钮。

[Home](#)[Chat](#)[Stores](#)[Providers](#)[Vectors](#)[Chats](#)[Me](#)[Edit Provider](#)[Save](#)

Name:

provider_openai_model

Display name:

OpenAI model

Category:

Model

Type:

OpenAI

Sub type:

text-davinci-003

Secret key:

Provider URL:

<https://platform.openai.com/account/api-keys>[Save](#)

Casibase 支持多种模型提供商，包括：

- [Hugging Face](#)
 - meta-llama/Llama-2-7b

- THUDM/chatglm2-6b
- baichuan-inc/Baichuan2-13B-chat
- gpt2
-
- OpenRouter
 - anthropic/clause-2
 - palm-2-chat-bison
 - palm-2-codechat-bison
 - openai/gpt-4
 -
- OpenAI
 - text-davinci-003
 - gpt-3.5-turbo
 - gpt-4
 -

⚠ 小心

- 类别：模型提供商的一级类别。例如，模型和嵌入。例如，模型和嵌入。
- 类型：模型提供商的二级类别。例如，openAI和Hugging Face。
- 密钥：您的 OpenAI 账户的密钥。

示例

添加 OpenAI 模型提供商

The screenshot shows the 'Edit Provider' form on the casbin platform. A red circle highlights the 'Type' dropdown menu, which is currently set to 'OpenAI'. Other options in the dropdown include 'Hugging Face', 'OpenRouter', and 'Ernie'. The 'Save' button is visible at the top right of the form.

Field	Value
Name:	provider_openai_model
Display name:	OpenAI model
Category	Model
Type:	OpenAI
Sub type:	OpenAI
Secret key:	
Provider URL:	https://platform.openai.com/account/api-keys

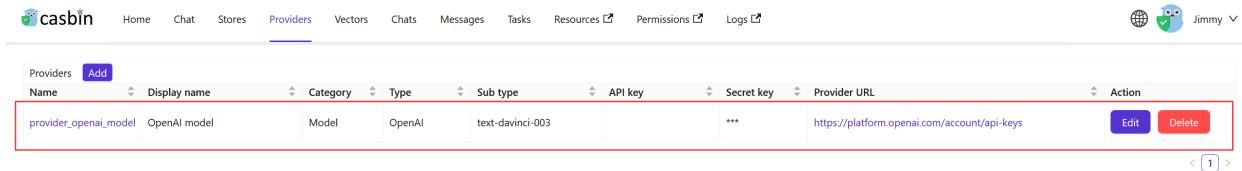
⚠ 小心

某些模型不支持流式输出。已知支持流式输出的模型包括： 已知支持流式输出的模型包括：

- gpt-3.5-turbo-0613

添加模型提供商后，您可以使用它通过聊天机器人、问答等 AI 功能来分析和处理 Casibase 中的数据。

返回模型提供商列表页面：



The screenshot shows the Casibase interface with the 'Providers' tab selected. A single provider entry is listed:

Name	Display name	Category	Type	Sub type	API key	Secret key	Provider URL	Action
provider_openai_model	OpenAI model	Model	OpenAI	text-davinci-003	***		https://platform.openai.com/account/api-keys	<button>Edit</button> <button>Delete</button>

A red box highlights the entire row for the provider entry.

现在您已经添加了模型提供商，您可以使用它通过聊天机器人、问答等 AI 功能来分析和处理 Casibase 中的数据。

嵌入提供商

简介

嵌入是一种用于将单词和文档表示为向量的技术。嵌入提供者使您能够在知识库系统内分析和处理数据，从而使其更加智能高效。

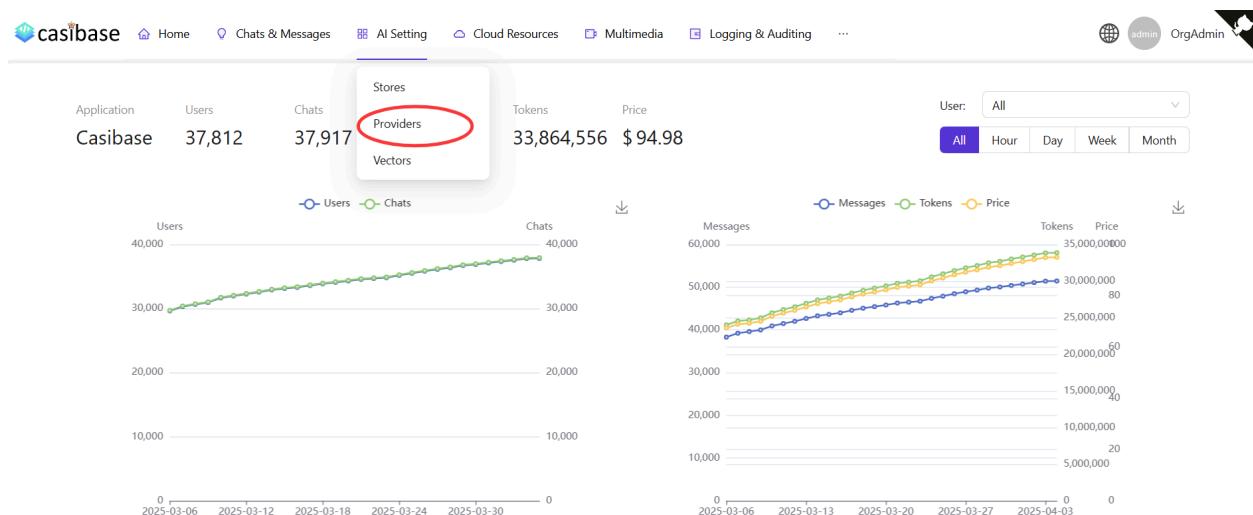
关于嵌入的更多信息，请参考我们之前文档中的[核心概念](#)部分。

在 Casibase 中，您可以按照以下步骤添加嵌入提供商：

添加新的嵌入提供商

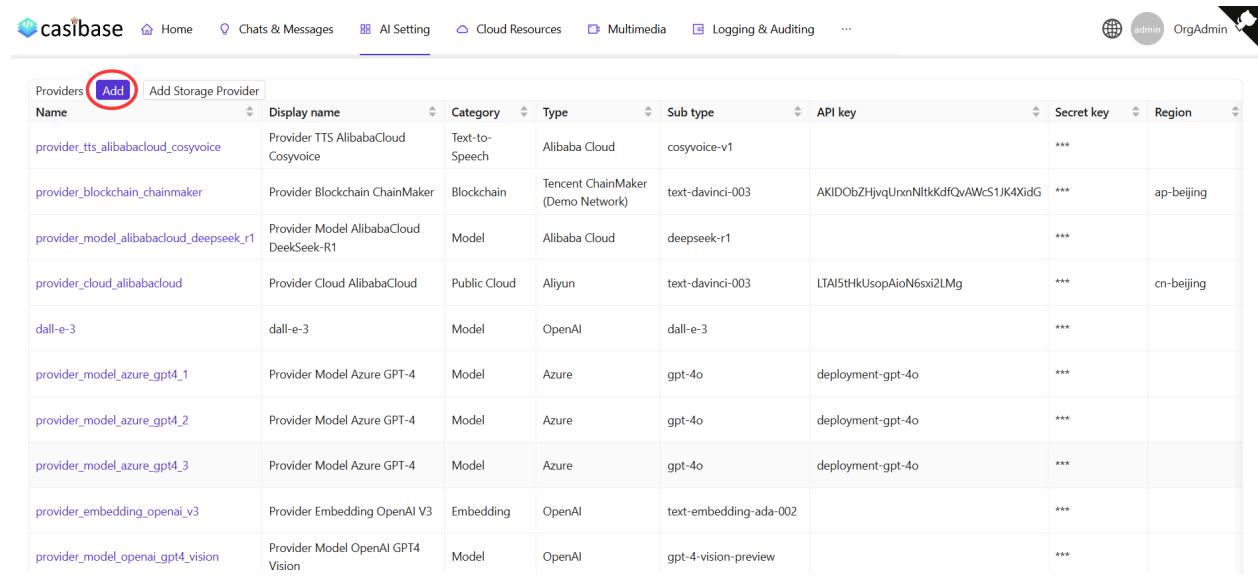
嵌入提供商用于将嵌入功能集成到 Casibase 中。您可以按照以下步骤添加它们：您可以按照以下步骤添加它们：

点击页面上的[提供商](#)按钮。



添加嵌入提供商

点击 **添加** 按钮来添加嵌入提供商。



The screenshot shows the Casibase AI Setting interface. At the top, there are navigation links: Home, Chats & Messages, AI Setting (which is highlighted), Cloud Resources, Multimedia, Logging & Auditing, and more. On the right side, there are user profile icons for 'admin' and 'OrgAdmin'. Below the navigation, a table titled 'Providers' is displayed. The first column has a blue 'Add' button with a red circle around it. The table columns are: Name, Display name, Category, Type, Sub type, API key, Secret key, and Region. The table lists various providers, such as provider_tts_alibabacloud_cosyvoice, provider_blockchain_chainmaker, provider_model_alibabacloud_deepseek_r1, provider_cloud_alibabacloud, dall-e-3, provider_model_azure_gpt4_1, provider_model_azure_gpt4_2, provider_model_azure_gpt4_3, provider_embedding_openai_v3, and provider_model_openai_gpt4_vision.

Providers	Add	Add Storage Provider	Name	Display name	Category	Type	Sub type	API key	Secret key	Region
provider_tts_alibabacloud_cosyvoice			Provider TTS AlibabaCloud Cosyvoice	Provider TTS AlibabaCloud Cosyvoice	Text-to-Speech	Alibaba Cloud	cosyvoice-v1		***	
provider_blockchain_chainmaker			Provider Blockchain ChainMaker	Provider Blockchain ChainMaker	Blockchain	Tencent ChainMaker (Demo Network)	text-davinci-003	AKIDObZHjqUrxnNltkKdfQvAWcS1JK4XidG	***	ap-beijing
provider_model_alibabacloud_deepseek_r1			Provider Model AlibabaCloud DeepSeek-R1	Provider Model AlibabaCloud DeepSeek-R1	Model	Alibaba Cloud	deepseek-r1		***	
provider_cloud_alibabacloud			Provider Cloud AlibabaCloud	Provider Cloud AlibabaCloud	Public Cloud	Aliyun	text-davinci-003	LTAI5tHkUsopAioN6xi2LMg	***	cn-beijing
dall-e-3			dall-e-3	dall-e-3	Model	OpenAI	dall-e-3		***	
provider_model_azure_gpt4_1			Provider Model Azure GPT-4	Provider Model Azure GPT-4	Model	Azure	gpt-4o	deployment-gpt-4o	***	
provider_model_azure_gpt4_2			Provider Model Azure GPT-4	Provider Model Azure GPT-4	Model	Azure	gpt-4o	deployment-gpt-4o	***	
provider_model_azure_gpt4_3			Provider Model Azure GPT-4	Provider Model Azure GPT-4	Model	Azure	gpt-4o	deployment-gpt-4o	***	
provider_embedding_openai_v3			Provider Embedding OpenAI V3	Provider Embedding OpenAI V3	Embedding	OpenAI	text-embedding-ada-002		***	
provider_model_openai_gpt4_vision			Provider Model OpenAI GPT4 Vision	Provider Model OpenAI GPT4 Vision	Model	OpenAI	gpt-4-vision-preview		***	

填写嵌入提供商详情

填写嵌入提供商详情并点击 **保存并退出** 按钮。

[Home](#)[Chat](#)[Stores](#)[Providers](#)[Vectors](#)[CI](#)[Edit Provider](#)[Save](#)

Name:

embedding_openai_adasimilarity

Display name:

Embedding_OpenAI_AdaSimilarity

Category:

Embedding

Type:

OpenAI

Sub type:

AdaSimilarity

Secret key:

Provider URL:

<https://platform.openai.com/account/api-keys>[Save](#)

Casibase 支持多种嵌入提供商，包括：

- [OpenAI](#)

- AdaSimilarity
- DavinciSimilarity
- AdaEmbedding2
-
- Hugging Face
 - sentence-transformers/paraphrase-MiniLM-L6-v2
 -

返回提供商列表页面：

The screenshot shows the Casibin provider list interface. At the top, there are navigation links: Home, Chat, Stores, Providers (which is highlighted in blue), Vectors, Chats, Messages, Tasks, Resources, Permissions, and Logs. On the right, there is a user profile icon for 'Jimmy'.

Name	Display name	Category	Type	Sub type	API key	Secret key	Provider URL	Action
embedding_openai_adasimilarity	Embedding_OpenAI_AdaSimilarity	Embedding	OpenAI	1		***	https://platform.openai.com/account/api-keys	<button>Edit</button> <button>Delete</button>
model_openai_text_davinci_003	Model OpenAI text-davinci-003	Model	OpenAI	text-davinci-003		***	https://platform.openai.com/account/api-keys	<button>Edit</button> <button>Delete</button>

现在，您可以使用嵌入提供商将文本转换为向量。

添加嵌入提供商后，您可以使用它在 Casibase 中检索相似文档。更多信息请参考我们之前文档中的[核心概念](#)部分。有关更多信息，请参阅我们之前文档中的[核心概念](#)部分。

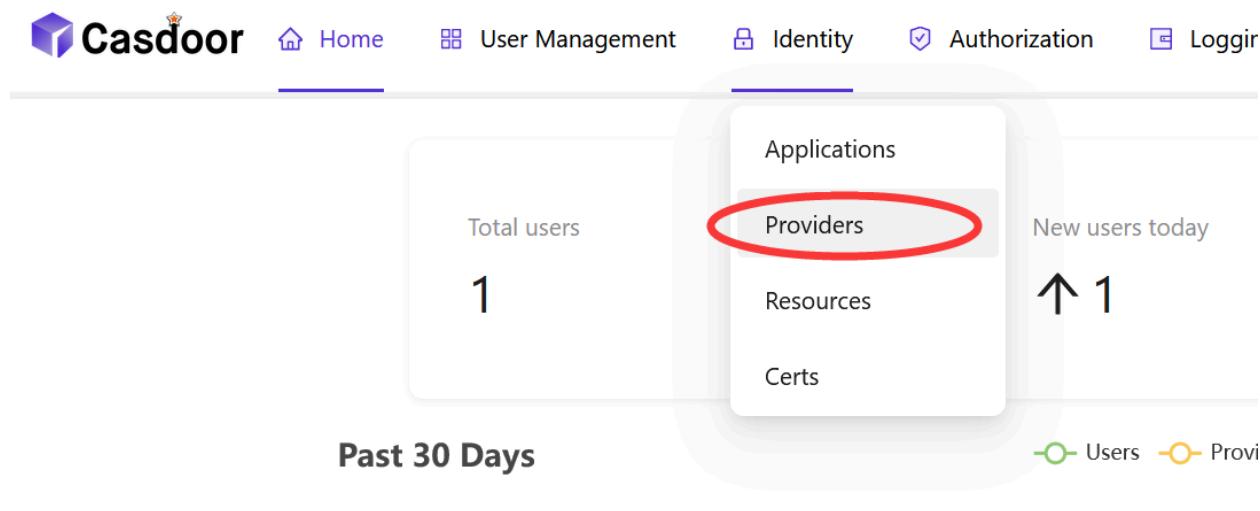
存储提供商

简介

向 Casibase 添加存储提供商可以让您高效地管理和存储数据，这是知识库系统的一个重要组件。

添加新的存储提供商

存储提供商用于存储数据。存储提供商用于存储数据。您可以通过点击主页上的 **身份 - 提供商** 按钮在 Casdoor 中添加它们。



点击 **添加** 按钮来添加存储提供商。



Providers		Add	
Name	Organization	Created time	Di
provider_captcha_default	admin (Shared)	2023-09-10 19:31:50	Ca

填写存储提供商信息

填写存储提供商信息并点击 **保存并退出** 按钮。

New Provider [Save](#) [Save & Exit](#) [Cancel](#)

Name [?](#) : provider_storage_1

Display name [?](#) : Provider_storage_1

Organization [?](#) :

Category [?](#) : Storage

Type [?](#) : aws AWS S3

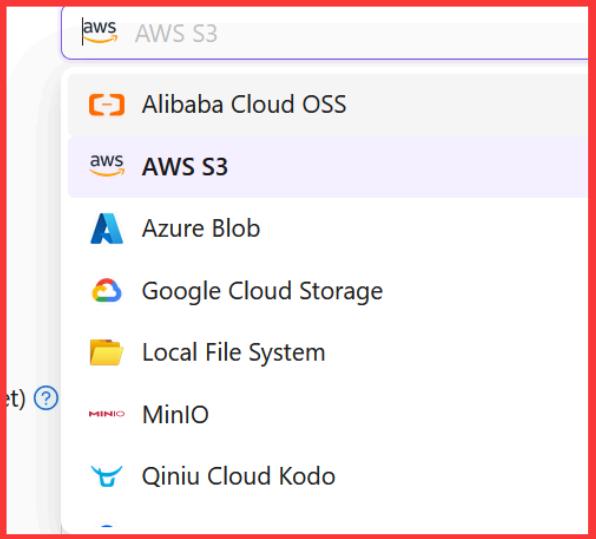
Client ID [?](#) :

Client secret [?](#) :

Endpoint [?](#) :

Endpoint (Intranet) [?](#) :

Bucket [?](#) :



The dropdown menu is styled with a light purple background and white text. Each item has a small icon to its left. The first item, "aws AWS S3", is highlighted with a blue border.

 提示

Casdoor 支持多种存储提供商，包括：

- [AWS S3](#)
- [Azure Blob](#)
- [Google Cloud Storage](#)
- [MinIO](#)

- [七牛云 Kodo](#)
- [阿里云 OSS ...](#)

示例

添加阿里云 OSS 存储提供商



小心

- 客户端 ID: 您的阿里云 OSS 账户的 AccessKey ID。
- 客户端密钥: 您的阿里云 OSS 账户的 AccessKey Secret。

***** 是您的阿里云 OSS 账户信息的占位符。

Category [?](#) : Storage

Type [?](#) : Alibaba Cloud OSS

Client ID [?](#) : LTA***NLf

Client secret [?](#) : Vo6***pi8

Endpoint [?](#) : oss-cn-beijing.aliyuncs.com

Endpoint (Intranet) [?](#) :

Bucket [?](#) : xx-bucket-0

Path prefix [?](#) :

Domain [?](#) : https://xx-bucket-0.oss-cn-beijing.aliyuncs.com

Provider URL [?](#) : https://github.com/organizations/xxx/settings/applications/1234567

[Save](#) [Save & Exit](#) [Cancel](#)

查看存储提供商

添加存储提供商后，您可以查看存储提供商信息。

Name	Organization	Created time	Display name	Category	Type	Client ID	Provider URL	Action
provider_storage_1	admin (Shared)	2023-09-10 21:23:02	Provider_storage_1	Storage	Alibaba Cloud OSS	[REDACTED]	https://github.com/organizations/xx...	Edit Delete

文本转语音提供商

简介

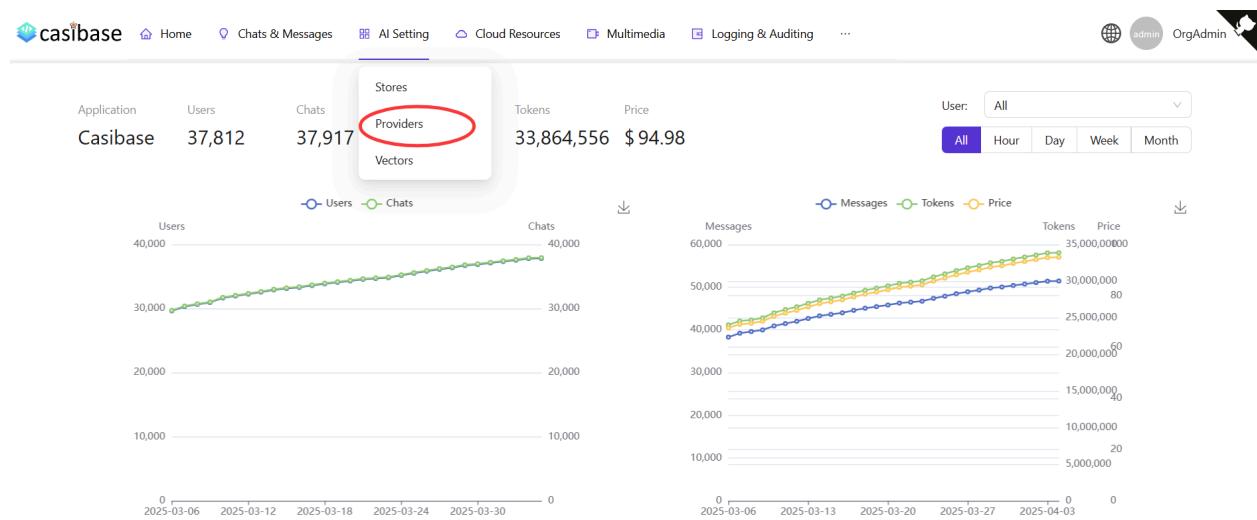
语音合成(TTS)是一种将文本转换成语音输出的技术。TTS 提供商使您的 Casibase 应用程序能够通过合成语音与用户交流，从而提高知识库系统的用户体验和可访问性。

在Casibase中，集成TTS提供商使您的AI应用能够口头响应查询，从而创造出更加互动且引人入胜的用户体验。

添加新的语音合成提供商

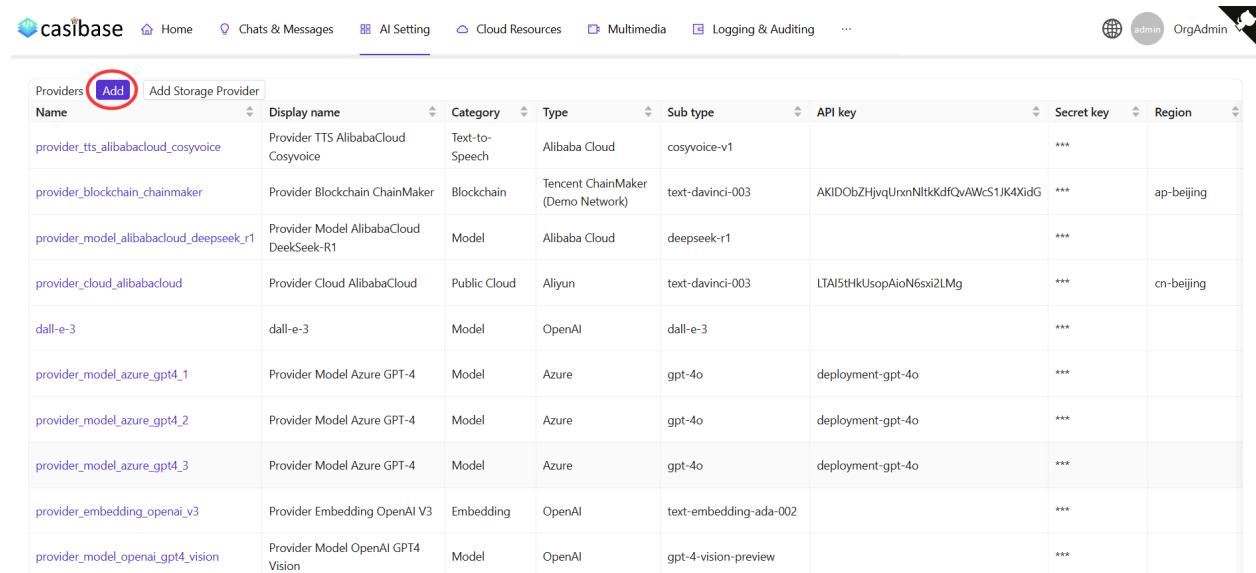
语音合成提供商用于将语音合成功能集成到 Casibase。您可以按照以下步骤添加它们：

点击页面上的 **提供商** 按钮。



添加语音合成提供商

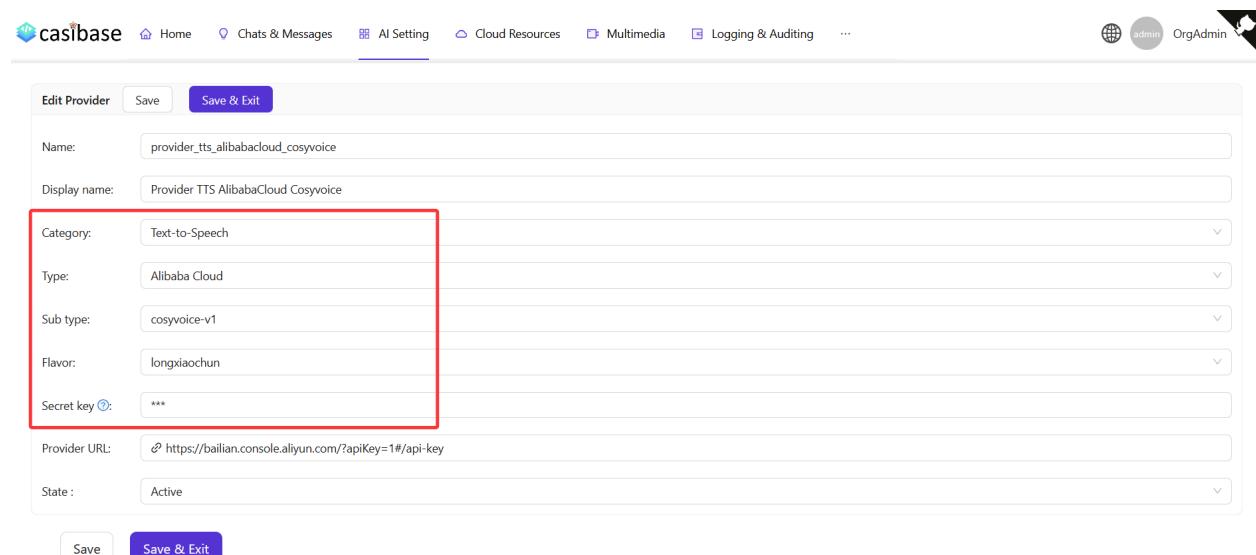
点击 **Add** 按钮来添加存储提供商。



Providers	Add	Add Storage Provider						
Name	Display name	Category	Type	Sub type	API key	Secret key	Region	
provider_tts_alibabacloud_cosyvoice	Provider TTS AlibabaCloud Cosyvoice	Text-to-Speech	Alibaba Cloud	cosyvoice-v1		***		
provider_blockchain_chainmaker	Provider Blockchain ChainMaker	Blockchain	Tencent ChainMaker (Demo Network)	text-davinci-003	AKIDObZHjqUrxnNltkdfQvAWcS1JK4XidG	***	ap-beijing	
provider_model_alibabacloud_deepseek_r1	Provider Model AlibabaCloud DeepSeek-R1	Model	Alibaba Cloud	deepseek-r1		***		
provider_cloud_alibabacloud	Provider Cloud AlibabaCloud	Public Cloud	Aliyun	text-davinci-003	LTAI5tHkUsopAioN6xi2LMg	***	cn-beijing	
dall-e-3	dall-e-3	Model	OpenAI	dall-e-3		***		
provider_model_azure_gpt4_1	Provider Model Azure GPT-4	Model	Azure	gpt-4o	deployment-gpt-4o	***		
provider_model_azure_gpt4_2	Provider Model Azure GPT-4	Model	Azure	gpt-4o	deployment-gpt-4o	***		
provider_model_azure_gpt4_3	Provider Model Azure GPT-4	Model	Azure	gpt-4o	deployment-gpt-4o	***		
provider_embedding_openai_v3	Provider Embedding OpenAI V3	Embedding	OpenAI	text-embedding-ada-002		***		
provider_model_openai_gpt4_vision	Provider Model OpenAI GPT4 Vision	Model	OpenAI	gpt-4-vision-preview		***		

填写语音合成提供商详情

填写嵌入提供商详情并点击 **Save & Exit** 按钮。



Edit Provider Save Save & Exit

Name:	provider_tts_alibabacloud_cosyvoice
Display name:	Provider TTS AlibabaCloud Cosyvoice
Category:	Text-to-Speech
Type:	Alibaba Cloud
Sub type:	cosyvoice-v1
Flavor:	longxiaochun
Secret key:	***
Provider URL:	https://bailian.console.aliyun.com/?apiKey=1#api-key
State:	Active

Save Save & Exit

💡 提示

Casibase 目前支持以下语音合成提供商:

- Alibaba Cloud
 - cosyvoice-v1 (具有多个语音选项)

Testing Your Text-to-Speech Provider

You can test your TTS provider by clicking the `Read it out` button. This will allow you to enter text and hear the synthesized speech output.

The screenshot shows the 'Edit Provider' page in the Casibase interface. The 'Type' field is set to 'Alibaba Cloud' and the 'Sub type' field is set to 'cosyvoice-v1'. In the 'Provider test' section, the input field contains 'Hello, I'm casibase AI.' and the 'Read it out' button is highlighted with a red box. Other fields include 'Name', 'Display name', 'Category', 'Flavor', 'Secret key', 'Provider URL', and 'State'.

Name:	provider_r7fdnn
Display name:	New Provider - r7fdnn
Category:	Text-to-Speech
Type:	Alibaba Cloud
Sub type:	cosyvoice-v1
Flavor:	龙小淳，女，中英双语。龙小淳的嗓音如丝般柔滑，温暖中流淌着亲切与抚慰，恰似春风吹过心田。
Secret key:	***
Provider test:	Hello, I'm casibase AI.
Provider URL:	https://platform.openai.com/account/api-keys
State:	Active

This testing feature allows you to verify your TTS configuration before implementing it in your applications, ensuring the voice quality and settings meet your requirements.

Alibaba 的语音选项

当使用 Alibaba Cloud 的 `cosyvoice-v1` 时，您可以从各种语音选项中选择:

- 龙婉

- 龙橙
-

在商店中使用语音合成功能

添加语音合成提供商后，您可以在您的商店设置中选择此提供商，然后选择是否启用TTS串流。

The screenshot shows the 'Edit Store' configuration page with the following fields:

- Name: store-built-in
- Display name: Built-in Store
- Title: (empty)
- Avatar: (empty)
- Storage provider: Built-in Storage Provider (provider-storage-built-in)
- Image provider: Storage Aliyun OSS Casibase Casbin (provider_storage_casibase_casbin)
- Split provider: Default
- Model provider: Provider Model Azure GPT-4 (provider_model_azure_gpt4)
- Embedding provider: Provider Embedding OpenAI V3 (provider_embedding_openai_v3)
- Text-to-Speech provider: Provider TTS AlibabaCloud Cosyvoice (provider_tts_alibabacloud_cosyvoice)
- Enable TTS streaming:

现在，您的商店可以将文本回复转换为语音，为用户提供更具互动性的体验。

语音识别提供商

简介

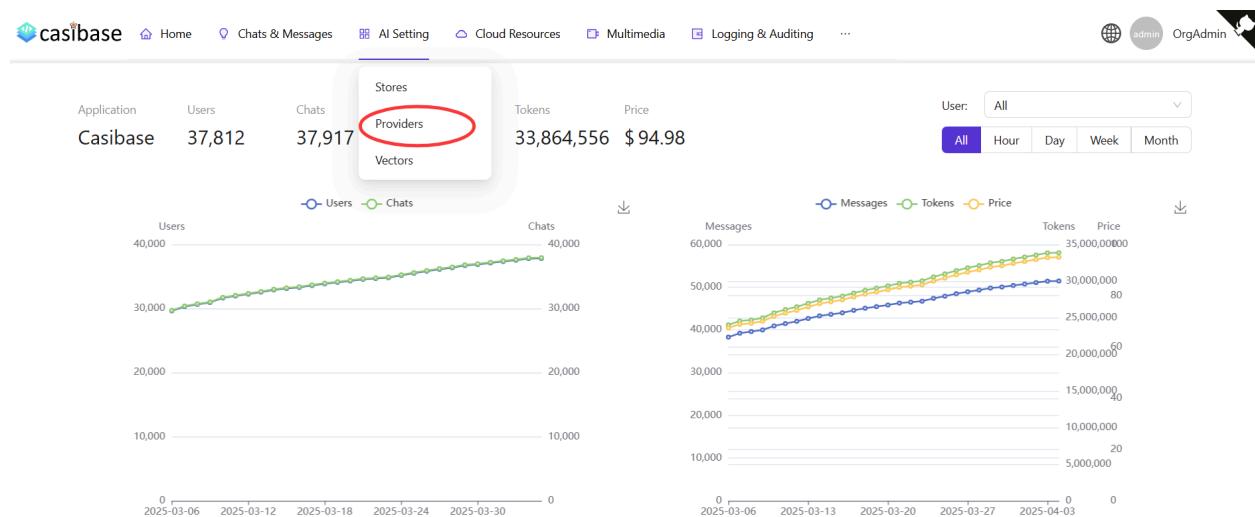
语音识别（STT）是一种将口语转换成书面文字的技术。STT 提供商允许您的 Casibase 应用程序理解和处理用户口语输入，增强用户体验和您的知识库系统的访问能力。

在 Casibase 中，整合一个 STT 提供商使您的 AI 应用程序能够接收和处理语音查询，创建更多的互动和自然用户交互。

添加一个新的语音识别提供商

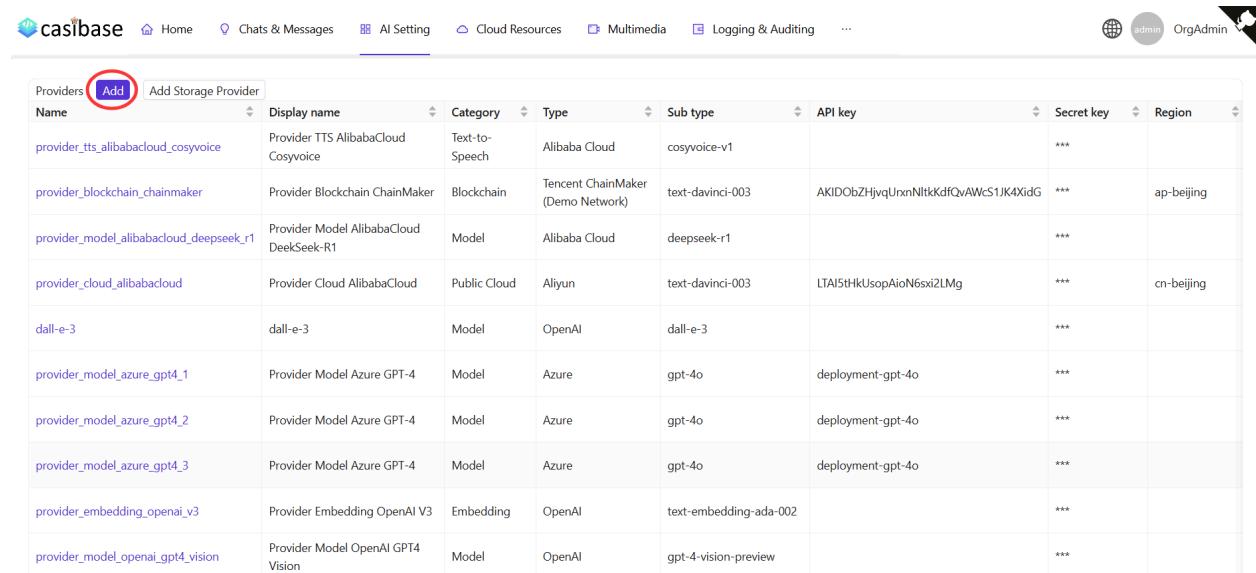
语音识别提供商用将语音识别能力整合到 Casibase 中。您可以按照以下步骤添加它们：

点击主页上的 Providers 按钮。



添加语音识别提供商

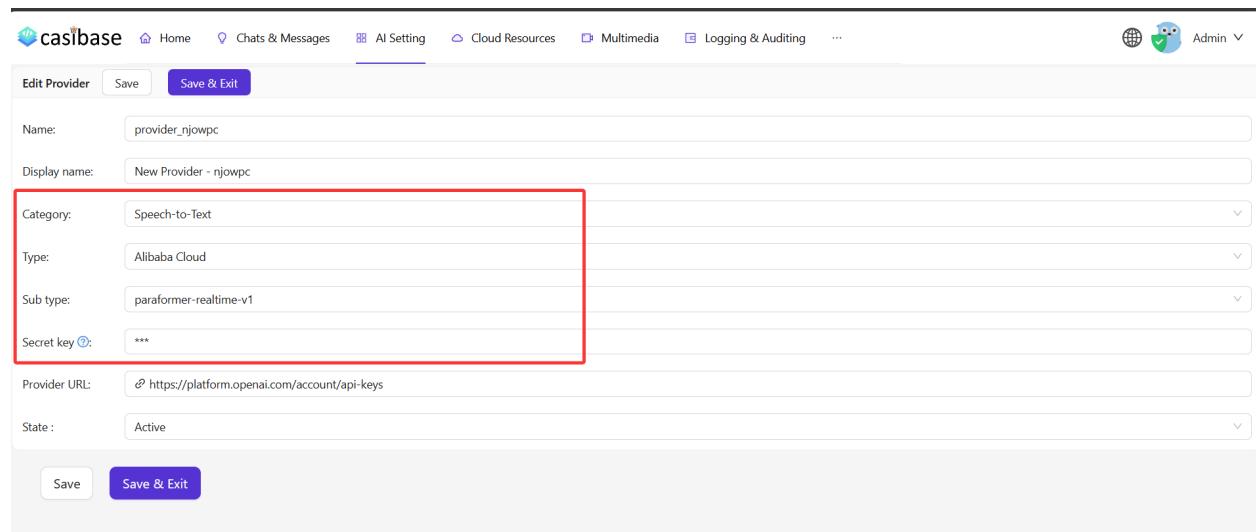
点击 **Add** 按钮来添加存储提供商。



Name	Display name	Category	Type	Sub type	API key	Secret key	Region
provider_tts_alibabacloud_cosyvoice	Provider TTS AlibabaCloud Cosyvoice	Text-to-Speech	Alibaba Cloud	cosyvoice-v1		***	
provider_blockchain_chainmaker	Provider Blockchain ChainMaker	Blockchain	Tencent ChainMaker (Demo Network)	text-davinci-003	AKIDObZHjqUrxnNltkKdfQvAWcS1JK4XidG	***	ap-beijing
provider_model_alibabacloud_deepseek_r1	Provider Model AlibabaCloud DeepSeek-R1	Model	Alibaba Cloud	deepseek-r1		***	
provider_cloud_alibabacloud	Provider Cloud AlibabaCloud	Public Cloud	Aliyun	text-davinci-003	LTAI5tHkUsopAioN6xi2LMg	***	cn-beijing
dall-e-3	dall-e-3	Model	OpenAI	dall-e-3		***	
provider_model_azure_gpt4_1	Provider Model Azure GPT-4	Model	Azure	gpt-4o	deployment-gpt-4o	***	
provider_model_azure_gpt4_2	Provider Model Azure GPT-4	Model	Azure	gpt-4o	deployment-gpt-4o	***	
provider_model_azure_gpt4_3	Provider Model Azure GPT-4	Model	Azure	gpt-4o	deployment-gpt-4o	***	
provider_embedding_openai_v3	Provider Embedding OpenAI V3	Embedding	OpenAI	text-embedding-ada-002		***	
provider_model_openai_gpt4_vision	Provider Model OpenAI GPT4 Vision	Model	OpenAI	gpt-4-vision-preview		***	

填写语音识别提供商详细信息

填写模型提供商详情并点击 **Save & Exit** 按钮。



casibase Home Chats & Messages AI Setting Cloud Resources Multimedia Logging & Auditing ... Admin

Edit Provider Save Save & Exit

Name: provider_njowpc

Display name: New Provider - njowpc

Category: Speech-to-Text

Type: Alibaba Cloud

Sub type: paraformer-realtime-v1

Secret key: ***

Provider URL: https://platform.openai.com/account/api-keys

State: Active

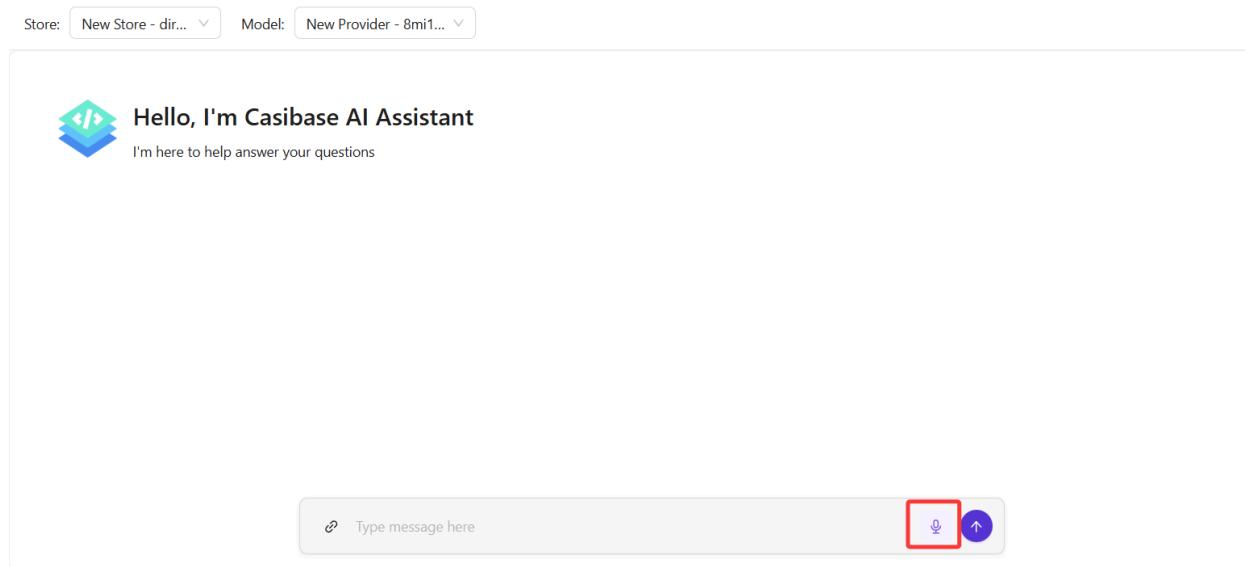
Save Save & Exit

使用语音识别

当您点击 Casibase 应用程序中的语音识别按钮时，将执行以下流程：

1. 浏览器将请求访问您麦克风的权限
2. 一旦授予，系统将开始收听并自动将您的语音转换为文本
3. 在您完成发言后，识别的文本将自动作为消息发送

此功能允许与您的 Casisbase 应用程序进行无操作的互动，使它们更容易访问和使用。



CasiBase 目前支持以下语音识别提供商：

- Alibaba Cloud
 - paraformer-realtime-v1

BlockChain Providers

Introduction

Blockchain technology provides an immutable and transparent ledger for data integrity verification. In Casibase, blockchain providers serve as a crucial security layer by uploading data to blockchain networks, ensuring that your knowledge base data cannot be tampered with or altered maliciously.

By leveraging blockchain's decentralized and cryptographic properties, Casibase can guarantee data authenticity and provide audit trails for all data modifications. This is particularly important for organizations that need to maintain data integrity compliance or require verifiable proof of data authenticity.

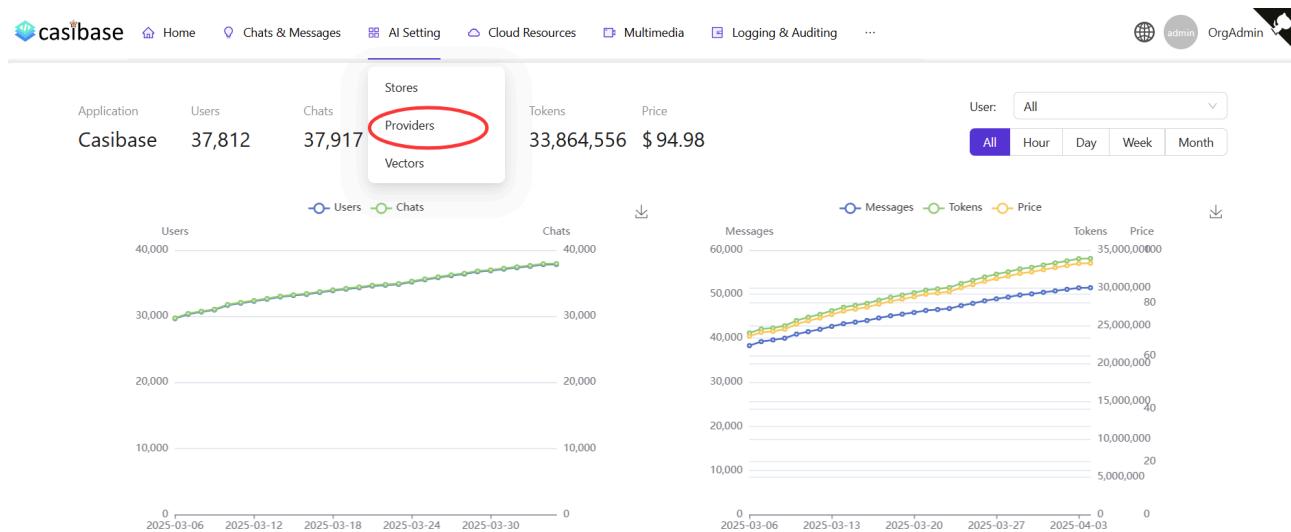
Refer to the [Core Concepts](#) section of our previous documentation for more information about providers.

In Casibase, you can add a blockchain provider by following these steps:

Add a New Blockchain Provider

Blockchain providers are used to integrate blockchain data integrity features into Casibase. You can add them by following these steps:

Click the [Providers](#) button on the page.



Add a Blockchain Provider

Click the [Add](#) button to add a blockchain provider.

Providers	Add	Add Storage Provider						
Name		Display name	Category	Type	Sub type	API key	Secret key	Region
provider_tts_alibabacloud_cosyvoice		Provider TTS AlibabaCloud Cosyvoice	Text-to-Speech	Alibaba Cloud	cosyvoice-v1		***	
provider_blockchain_chainmaker		Provider Blockchain ChainMaker	Blockchain	Tencent ChainMaker (Demo Network)	text-davinci-003	AKIDObZHjvqUrxnNtkKdfQvAWcS1JK4XidG	***	ap-beijing
provider_model_alibabacloud_deepseek_r1		Provider Model AlibabaCloud DeepSeek-R1	Model	Alibaba Cloud	deepseek-r1		***	
provider_cloud_alibabacloud		Provider Cloud AlibabaCloud	Public Cloud	Aliyun	text-davinci-003	LTAI5tHkUsopAioN6sxilMg	***	cn-beijing
dall-e-3		dall-e-3	Model	OpenAI	dall-e-3		***	
provider_model_azure_gpt4_1		Provider Model Azure GPT-4	Model	Azure	gpt-4o	deployment-gpt-4o	***	
provider_model_azure_gpt4_2		Provider Model Azure GPT-4	Model	Azure	gpt-4o	deployment-gpt-4o	***	
provider_model_azure_gpt4_3		Provider Model Azure GPT-4	Model	Azure	gpt-4o	deployment-gpt-4o	***	
provider_embedding_openai_v3		Provider Embedding OpenAI V3	Embedding	OpenAI	text-embedding-ada-002		***	
provider_model_openai_gpt4_vision		Provider Model OpenAI GPT4 Vision	Model	OpenAI	gpt-4-vision-preview		***	

Fill in Blockchain Provider Details

Fill in the required configuration according to the blockchain you use, then click the **Save & Exit** button to save.

More information about the configuration can be found below:

Chainmaker Configuration

ChainMaker is a high-performance, enterprise-grade blockchain platform developed under the leadership of China. It is designed to provide secure, controllable, and sc...

Ethereum Configuration

Ethereum is a decentralized blockchain platform that enables smart contracts and decentralized applications (dApps). It is one of the most popular blockchain platforms...

Edit Provider		Save	Save & Exit
Name	eth_wm_test		
Display name	eth_wm		
Category	Blockchain		
Type	Ethereum		
Private key	***		
Contract Address	D4600b1B04b4FD07194476C35825175B30F0f9Ec		
Invoke method	save		
Browser URL	http://127.0.0.1:5051/txpage?blocknumber={bh}		
Provider URL	http://192.168.31.234:8545		
Is default	<input checked="" type="checkbox"/>		
State	Active		



提示

Casibase supports various blockchain networks for data integrity verification, including:

- [ChainMaker](#)
 - [ChainMaker](#)
 - [Tencent Chainmaker](#)
- [Ethereum](#)
 - Private networks (compatible with Ethereum JSON-RPC)
 - [Geth](#)
 - [Ganache](#)
 - Other Ethereum JSON-RPC compatible chains

Now, you can use the blockchain provider to ensure data integrity and prevent tampering.

After adding a blockchain provider, you can use it in Casibase to create immutable data records. The data records will be committed to the blockchain, providing proof of authenticity and preventing any unauthorized tampering.

Chainmaker Configuration

ChainMaker is a high-performance, enterprise-grade blockchain platform developed under the leadership of China. It is designed to provide secure, controllable, and scalable blockchain infrastructure for industries such as finance, government, and supply chain. ChainMaker supports multiple consensus mechanisms, smart contracts, privacy protection, and other features, meeting the needs of large-scale commercial applications. Its open-source, modular architecture allows developers to flexibly customize and extend functionalities according to actual business scenarios.

In this chapter, you will learn how to configure and use ChainMaker, including setting up storage providers and other operations, to help you quickly get started and apply the features of the ChainMaker platform.

1. Configuration field description

When configuring a ChainMaker provider in Casibase, you need to fill in several key fields. Each field has a specific meaning and is required for the correct integration with the ChainMaker blockchain. The following list explains the purpose of each field:

- `Name`: The unique identifier for this blockchain provider.
- `Display name`: The display name shown in the UI for this provider.
- `Category`: The type of service, here it should be `Blockchain`.
- `Type`: The blockchain type, here it should be `ChainMaker`.
- `orgId`: The organization ID in the ChainMaker network.

- `ChainId`: The chain ID of the ChainMaker blockchain.
- `AuthType`: The account mode. Currently, only `permissionedwithcert` is supported.
- `User key`: The user's private key for authentication in the ChainMaker server.
- `User cert`: The user's certificate for authentication in the ChainMaker server.
- `Sign key`: The user's private key for signing transactions in the ChainMaker server.
- `Sign cert`: The user's certificate for signing transactions in the ChainMaker server.
- `Node address`: The address of the ChainMaker node to connect to.
- `Contract name`: The name of the smart contract to interact with.
- `Invoke method`: The method name to invoke on the contract.
- `Browser URL`: The URL for viewing the blockchain in a browser.
- `Chainmaker endpoint`: The API endpoint for the ChainMaker service. See:
<https://github.com/casibase/chainmaker-server>

Please make sure to fill in each field accurately according to your ChainMaker deployment information. This will ensure that Casibase can successfully connect and interact with your ChainMaker blockchain.

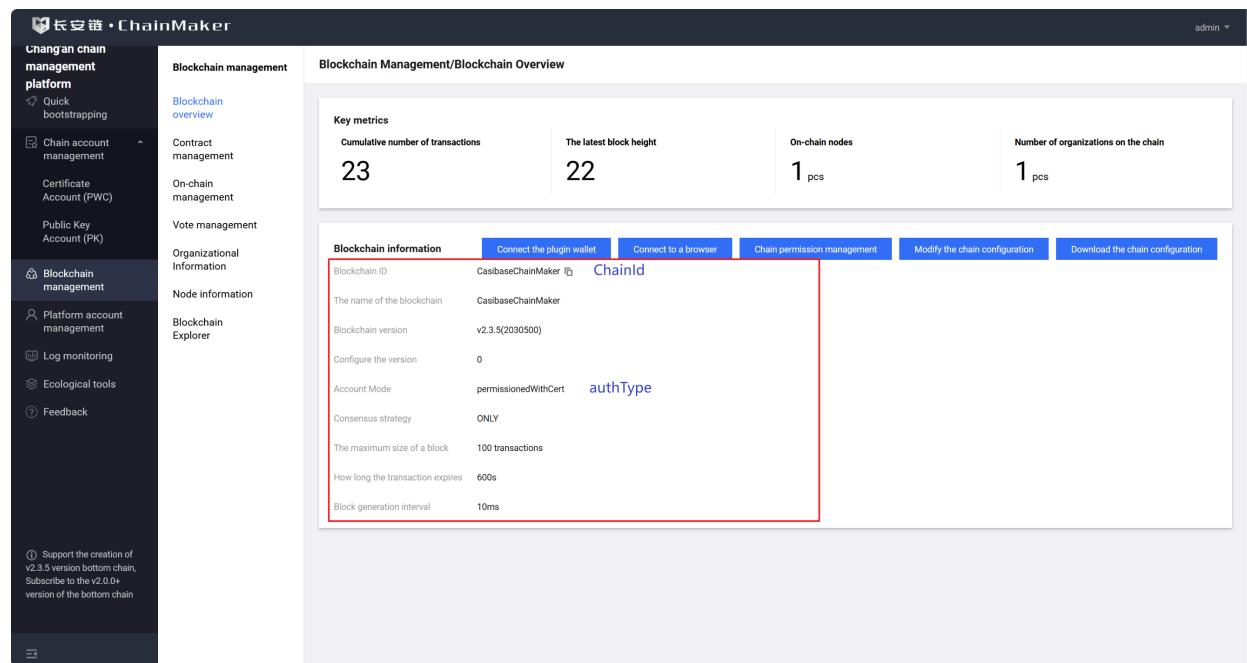
Next, we will use "Deploying ChainMaker via the Management Console" as an example for our introduction. If you have not yet deployed ChainMaker, please follow the [ChainMaker documentation](#) for deployment.

2. Configure ChainMaker

2.1 chainId, orgId and authType configuration

Obtaining Blockchain Information from the Web Panel

To retrieve blockchain information such as `chainId`, `orgId`, and `authType`, log in to the **ChainMaker Management Console** (Web panel). Navigate to the relevant blockchain management section, where these configuration details are displayed. Copy the required values and use them when configuring the ChainMaker provider in Casibase.



The screenshot shows the Chang'an chain management platform's web interface. The left sidebar has sections for 'Chang'an chain management platform' (Quick bootstrapping, Chain account management, Certificate Account (PWC), Public Key Account (PK)), 'Blockchain management' (Blockchain overview, Contract management, On-chain management, Vote management, Organizational Information, Node information, Blockchain Explorer), and 'Platform account management', 'Log monitoring', 'Ecological tools', and 'Feedback'. A note at the bottom left says: 'Support the creation of v2.3.5 version bottom chain, Subscribe to the v2.0.0+ version of the bottom chain'. The main content area is titled 'Blockchain Management/Blockchain Overview' and displays 'Key metrics': Cumulative number of transactions (23), The latest block height (22), On-chain nodes (1 pcs), and Number of organizations on the chain (1 pcs). Below this is a 'Blockchain information' section with fields: Blockchain ID (CasibaseChainMaker), Name of the blockchain (CasibaseChainMaker), Blockchain version (v2.3.5(2030500)), Configure the version (0), Account Mode (permissionedWithCert), Consensus strategy (ONLY), Maximum size of a block (100 transactions), How long the transaction expires (600s), and Block generation interval (10ms). Buttons for 'Connect the plugin wallet', 'Connect to a browser', 'Chain permission management', 'Modify the chain configuration', and 'Download the chain configuration' are also present. The 'ChainId' and 'authType' fields are highlighted with red boxes.

Blockchain information	
Blockchain ID	CasibaseChainMaker
Name of the blockchain	CasibaseChainMaker
Blockchain version	v2.3.5(2030500)
Configure the version	0
Account Mode	permissionedWithCert
Consensus strategy	ONLY
The maximum size of a block	100 transactions
How long the transaction expires	600s
Block generation interval	10ms

Blockchain management

- Blockchain overview
- Contract management
- On-chain management
- Vote management
- Organizational information**
- Node information
- Blockchain Explorer

Blockchain management/organization information

Organization ID	The name of the organization	Creation time	Number of nodes
TestCMorg1 orgId	cmtestorg1	2025-06-10 10:18:33	1

Please enter the organization name to search Q

10 Article/page 1 / 1 page

2.2 Contract name and Invoke method configuration

Blockchain management

- Blockchain overview
- Contract management**
- On-chain management
- Vote management
- Organizational Information
- Node information
- Blockchain Explorer

Blockchain management/contract management

The name of the contract	Current version	Affiliation	Created by	Updated	Voting status	On-chain status	operate
casicbase Contract name	1.0.0	TestCMorg1	cmtestorg1	2025-06-10 13:21:08	normal	normal	freeze logout upgrade edit

Please enter the contract name to search Q

20 Article/page 1 / 1 page

Click the edit button in the "operate" column:

Blockchain management

- Blockchain overview
- Contract management**
- On-chain management
- Vote management
- Organizational Information
- Node information
- Blockchain Explorer

Edit the contract

The name of the contract ① casicbase

state The contract is initialized and deployed

Current version 1.0.0

The type of virtual machine ① WASM

Contract Invocation Method (Optional) ①

save	Invoke method	invoke	file_hash,file_name,time	increase
find_by_file_hash	Inquire	file_hash	Delete	

Are you sure Cancel

2.3 key, certificate, and nodeAddr configuration

The screenshot shows the 'Blockchain Management/Blockchain Overview' page. On the left, there's a sidebar with links: Blockchain management (Blockchain overview, Contract management, On-chain management, Vote management, Organizational Information), Node information, and Blockchain Explorer. The main area has a header 'Blockchain Management/Blockchain Overview'. Below it is a 'Key metrics' section with four cards: Cumulative number of transactions (23), The latest block height (22), On-chain nodes (1 piece), and Number of organizations on the chain (1 piece). Underneath is a 'Blockchain information' section with various parameters like Blockchain ID (CasibaseChainMaker), Version (v2.3.5(2030500)), and Account Mode (permissionedWithCert). A prominent blue button at the bottom right of this section is labeled 'Download the chain configuration', which is also highlighted with a red rectangular border.

Click the "download the chain configuration" button and extract the archive:

2.3.1 node addr

You can find the node_addr configuration in ~\CasibaseChainMaker\sdk_configs\sdk_config.yml.

Alternatively, you can check nodeAddr elsewhere, but note that the displayed port is the p2p port. The one we use should be the rpc port, which is the p2p port + 1000 by default.

Blockchain management

Blockchain overview

Contract management

On-chain management

Vote management

Organizational Information

Node information

Blockchain Explorer

Blockchain management/node information

Node name	Affiliation	Node type	Node ID	Node address	Ledger synchronization type	operate
cmtestnode1	cmtestorg1	Consensus nodes	Qmdcq5NhATkgqEi7q3Tvx...	nodeAddr 39.107.236.48:11301 but port is error	FULL	View
0 1						

Please enter the node name to search

The correct port is the current value plus 1000; i.e.: 12301

10 Article/page | 1 / 1 page |

2.3.2 user certificate and key

You can find the corresponding user certificate and key in ~\CasibaseChainMaker\ sdk_configs\crypto-config\TestCMorg1\user\cmtestuser1. Fill in the provider fields accordingly.

cmtestuser1.sign.crt	2025/6/12 23:49	安全证书	1 K
cmtestuser1.sign.key	2025/6/12 23:49	KEY 文件	1 K
cmtestuser1.tls.crt	2025/6/12 23:49	安全证书	1 K
cmtestuser1.tls.key	2025/6/12 23:49	KEY 文件	1 K

2.4 Browser URL and ChainMaker endpoint

The Browser URL refers to the URL of the ChainMaker management console, which allows for quick access from Casibase in the future:

| <https://manage.casvisor.com/chains/CasibaseChainMaker/nodes?chainMode=permissionedWithCert> **Browser URL**

The screenshot shows the Casibase interface with the 'Logging & Auditing' tab selected. The main area displays a table of records with columns: Organization, ID, Name, Client IP, Created time, Sessions, Action, Block, and Action. The 'Sessions' column shows a list of sessions with their details. A red box highlights the 'Sessions' header, and a red arrow points to the 'Commit' button in the session log. A red callout box contains the text: 'If you have already committed, then the block will appear here, click on it to jump to the admin console'.

Organization	ID	Name	Client IP	Created time	Sessions	Action	Block	Action
casbin	9457	36fd22c8-1771-4083-9bc5-7cc401ed3a40	124.64.124.134	2025-06-13 00:39:	2025-06-13 00:38:57 / provider_blockchain update-provider	signin	Commit	<button>View</button> <button>Delete</button>
casbin	9456	f0bc2228-c10a-420e-90b9-535318658a9a	⋮1		2025-06-13 00:30:03 / provider_blockchain update-provider		Commit	<button>View</button> <button>Delete</button>
casbin	9455	438f014d-b808-40f6-bf52-62289d70d6f5	⋮1		2025-06-13 00:30:03 / provider_blockchain update-provider		Commit	<button>View</button> <button>Delete</button>
casbin	9454	18f40ece-d988-461b-9f4a-5d9cb80ff192	⋮1	2025-06-13 00:27:52	provider_blockchain update-provider		Commit	<button>View</button> <button>Delete</button>

The ChainMaker endpoint refers to the IP and port of the ChainMaker server that Casibase needs to connect to, for example: 127.0.0.1:13900

Ethereum Configuration

Ethereum is a decentralized blockchain platform that enables smart contracts and decentralized applications (dApps). It is one of the most popular blockchain platforms, supporting a wide range of applications from DeFi to NFTs. Ethereum uses a proof-of-stake consensus mechanism and provides a robust ecosystem for developers to build and deploy smart contracts.

① 信息

In this chapter, you will learn how to configure and use Ethereum, including setting up blockchain providers and other operations, to help you quickly get started and apply the features of the Ethereum platform.

1. Configuration field description

When configuring an Ethereum provider in Casibase, you need to fill in several key fields. Each field has a specific meaning and is required for the correct integration with the Ethereum blockchain.

Field Descriptions:

- **Name**: The unique identifier for this blockchain provider.
- **Display name**: The display name shown in the UI for this provider.
- **Category**: The type of service, here it should be **Blockchain**.
- **Type**: The blockchain type, here it should be **Ethereum**.
- **Private key**: The private key of the Ethereum account used for signing

transactions.

- **Contract Address**: The address of the smart contract to interact with on the Ethereum blockchain.
- **Invoke method**: The method name to invoke on the smart contract.
- **Browser URL**: The URL for viewing the blockchain in a browser, with block number template support.
 - Format: `http://127.0.0.1:5051/txpage?blocknumber={bh}` where `{bh}` will be replaced with the actual block number when visiting the block.
- **Provider URL**: The JSON-RPC endpoint URL for connecting to the Ethereum network (e.g., Geth, Ganache, or other node).

Please make sure to fill in each field accurately according to your Ethereum deployment information. This will ensure that Casibase can successfully connect and interact with your Ethereum blockchain.

2. Configure Ethereum

Example

Edit Provider		Save	Save & Exit
Name ⓘ :	eth_win_test		
Display name ⓘ :	eth_win		
Category ⓘ :	Blockchain		
Type ⓘ :	 Ethereum		
Private key ⓘ :	***		
Contract Address ⓘ :	D4600b1B04b4FD07194476C35825175B30F0f9Ec		
Invoke method ⓘ :	save		
Browser URL ⓘ :	http://127.0.0.1:5051/txpage?blocknumber={bh}		
Provider URL ⓘ :	http://192.168.31.234:8545		
Is default ⓘ :	<input checked="" type="checkbox"/>		
State ⓘ :	Active		

2.1 Provider URL Configuration

The Provider URL is the JSON-RPC endpoint that Casibase will use to communicate with the Ethereum network. This is the first and most important configuration as it establishes the connection to your Ethereum network.

Example Provider URL

```
http://127.0.0.1:8545
```

You can use:

- **geth**: A popular Ethereum client that provides a JSON-RPC interface.
- **ganache**: A personal blockchain for Ethereum development that can be used for testing and development purposes.
- **other**: Ethereum JSON-RPC compatible chains. Any other Ethereum-compatible chain that supports the JSON-RPC interface.

Example: Geth Dev Mode

To quickly experiment with Ethereum using Geth, you can start Geth in developer mode. This mode launches a local Ethereum node with instant mining and pre-funded accounts, making it ideal for testing and development.

```
geth --dev --http --http.api eth,web3,net --http.corsdomain  
"https://remix.ethereum.org"
```

This command starts a local Ethereum node with HTTP JSON-RPC enabled and sets the CORS domain to allow cross-origin requests from <https://remix.ethereum.org>. This configuration is suitable for online contract deployment and interaction using Remix web-based tools. You can use the default account (private key can be found in the `geth` console at first launch) and the endpoint (`http://127.0.0.1:8545`) as your Provider URL in Casibase for immediate testing and development.

```

WARN [07-19|02:42:46.376]      stored on a ramdisk, and will be lost if your machine is restarted.
WARN [07-19|02:42:46.376] 4. Mining is enabled by default. However, the client will only seal blocks if trans-
actions
WARN [07-19|02:42:46.376]      are pending in the mempool. The miner's minimum accepted gas price is 1.
WARN [07-19|02:42:46.376] 5. Networking is disabled; there is no listen-address, the maximum number of peers
is set
WARN [07-19|02:42:46.376]      to 0, and discovery is disabled.
WARN [07-19|02:42:46.376]
WARN [07-19|02:42:46.376]
WARN [07-19|02:42:46.376] Running in ephemeral mode. The following account has been prefunded in the genesis
:
WARN [07-19|02:42:46.376]      Account
WARN [07-19|02:42:46.376] -----
WARN [07-19|02:42:46.376]      0x71562b71999873db5b286df957af199ec94617f7 (10^49 ETH)
WARN [07-19|02:42:46.376]      Private Key
WARN [07-19|02:42:46.376] -----
WARN [07-19|02:42:46.376]      0xb71c71a67e1177ad4e901695e1b4b9ee17ae16c6668d313eac2f96
WARN [07-19|02:42:46.376]
INFO [07-19|02:42:46.376] Starting peer-to-peer node           instance=Geth/v1.16.1-stable-12b4131f/linux
-amd64/goi.24.4
WARN [07-19|02:42:46.376] P2P server will be useless, neither dialing nor listening
INFO [07-19|02:42:46.381] IPC endpoint opened           url=/tmp/geth.ipc
INFO [07-19|02:42:46.381] HTTP server started          endpoint=127.0.0.1:8545 auth=false prefix=
cors=https://remix.ethereum.org vhosts=localhost
INFO [07-19|02:42:46.382] New local node record          seq=1,752,864,166,381 id=5a498da1b5df4f0c i
p=127.0.0.1 udp=0 tcp=0
INFO [07-19|02:42:46.382] Started P2P networking        self=enode://2947b9f976fea97f00cf1be7e58b88
995a40f02daacb1eb6052fd298e7acb9e52e7481686d3f6101762a7a48e5b639e1540db8d958baff182b2bfdafb8a79e04@127.0.0.1:0
INFO [07-19|02:42:46.382] Started log indexer

```

Understanding the Console Output:

When you run Geth in dev mode, the console will display important information as shown in the image above:

- Private Key:** The console shows the private key of the pre-funded account that you can use for testing. This key is automatically generated and displayed in the console output.
- HTTP Endpoint:** The console confirms that the HTTP JSON-RPC server is running on `http://127.0.0.1:8545`. This is the endpoint address you should use as your Provider URL in Casibase.
- Account Address:** The corresponding Ethereum address for the generated private key is also displayed.

Important: Copy and save these values immediately as they are essential for configuring your Casibase provider. The private key will be needed for the `Private key` field, and the HTTP endpoint will be your `Provider URL`.

For more details, see the [Geth Dev Mode documentation](#).

2.2 Private key Configuration

The private key is essential for signing transactions on the Ethereum blockchain.



The private key should be provided without the hexadecimal prefix `0x`.

Example Private Key

```
# Example private key in geth dev mode (without 0x prefix)
b71c71a67e1177ad4e901695e1b4b9ee17ae16c6668d313eac2f96dbcda3f291
```



This is just an example private key for demonstration purposes. Never use this key in production or for real funds!

You can obtain your private key from various sources:

- **Initially generated by Geth:** When you start Geth in `dev` mode, it generates a pre-funded account with a private key displayed in the console.
- **Ethereum Clients:** Generate a new account using Ethereum clients like Geth.



Casibase will use `***` to replace the private key on the frontend after the submission.

2.3 Invoke Method and Contract Address Configuration

Invoke Method Configuration

The invoke method is the specific function name in the smart contract that you want to call.

In Casibase, your smart contract should implement specific methods to ensure compatibility:

- `save`: This method is used to store data in the contract. It should accept parameters as a tuple (struct).

```
struct DataItem {
    string key;
    string field;
    string value;
}

// Define event, returns key, field, value in order
event DataSaved(string key, string field, string value);

// Save struct data and emit an event for tracking
function save(DataItem memory _data) public {
    emit DataSaved(_data.key, _data.field, _data.value);
}
```

Method name `save` can be customized, but it should accept a struct as an argument.

You can refer to the [Example](#) to see how to implement the `save` method in your smart contract.

Make sure your contract includes these methods to enable seamless integration with Casibase.

Contract Address Configuration

```
# Example contract address (without 0x prefix)
c36fED2CE2E1Bb14b330465f4498D4892C8ee194
```

The contract address is the deployed smart contract's address on the Ethereum blockchain. You can obtain the contract address after deploying a smart contract.

Example for Contract Deployment Reference

To deploy a smart contract on Ethereum, you can refer to the [Casibase/contract-storage-eth](#). This repository provides sample Solidity contracts and deployment scripts using Go and Remix.

Getting Started with the Repository:

1. **Get Example Code:** Clone or download the repository to access sample Solidity contracts and deployment scripts.
2. **Contract Compilation:** Pre-compiled contract artifacts (ABI and bytecode) are available in the [releases](#) section of the repository.
3. **Setup for Go Script Deployment:** If using the Go deployment script, download the contract artifacts from releases and place them in the `build/` folder within the Go script's working directory.
4. **Deployment Options:** You can deploy the contract using either:
 - **Go Script:** Use the provided Go deployment script in the repository for programmatic deployment (requires contract artifacts in `build/` folder)
 - **Remix IDE:** Deploy contracts online using [Remix](#) with the contract source code

After deployment using either method, you can obtain the contract address from the deployment output.

Use the go script in the reference to deploy the contract

```
$ go run deploy.go
Starting contract deployment...
Connected to Ethereum node: http://192.168.31.234:8545
Deploying from address: 0x71562b71999873DB5b286dF957af199Ec94617F7
Loaded bytecode from: build/SaveContract.bin
Loaded ABI from: build/SaveContract.abi
Gas price: 9 wei
Gas limit: 0
Deploying contract...
Transaction sent: 0xf377a667d3216a1a45b3c3d0944745ea1cbe8ab17745f6e95c44d4f7a5a3fd8f
Contract address: 0xc36fED2CE2E1Bb14b330465f4498D4892C8ee194
Waiting for transaction confirmation...
Contract deployed successfully!
Gas used: 611787
Block number: 198

Running contract test...
Calling save function with: key=test_key_123, field=test_field, value=test_value_456
Save transaction: 0xb060530b6de01bd0537595b86dd0ebcac9007ebab3f24c096c42857ac6fdb3f2
Save function called successfully!
Retrieved data - Key: test_key_123, Field: test_field, Value: test_value_456
Log data - Key: test_key_123, Field: test_field, Value: test_value_456
```

Or you can use the block explorer to find the contract address.

Transaction Dashboard

TOTAL NO. OF TRANSACTIONS 1	BLOCK NUMBER 198	BLOCK HASH 0x58ddd0285c82aca 9676f47e96b2f6f08 7ddc83c4155f5b88 6373a115f466cb5	Transaction Status SUCCESSFUL
--------------------------------	---------------------	---	----------------------------------

Transaction Overview

Transaction Hash	0xf377a667d3216a1a45b3c3d0944745ea1cbe8ab17745f6e95c44d4f7a5a3fd8f
Transaction Gas	617810
Transaction Gas Price	9
Transaction Nonce	197
Transaction To	0xc36fED2CE2E1Bb14b330465f4498D4892C8ee194 [CONTRACT CREATION]
Transaction From	0x71562b71999873DB5b286dF957af199Ec94617F7
Transaction Value [wei]	0
Transaction Status	SUCCESSFUL

2.4 Browser URL Configuration

The Browser URL lets you view specific blockchain blocks and transactions in a web browser. By using a template with the `{bh}` placeholder, Casibase can automatically redirect you to the corresponding block details in your chosen blockchain explorer.

`http://127.0.0.1:5051/txpage?blocknumber={bh}`

Organization	ID	Name	Client IP	Created time	Provider	User	Method	Request URI	Action	Block	Action
built-in	115	0af1c434-a708-4238-a55a-5ae322b2f3f2	127.0.0.1	2025-07-19 03:08:20	eth_win_test	admin	POST	/api/signin?code=f33196946c	signin	200	<button>Query</button> <button>View</button> <button>Delete</button>
built-in	111	aea40549-4bfb-41a4-a9cd-51fb4004fe49	::1	2025-07-14 00:20:36	eth_win_test	admin	POST	/api/update-provider	update-provider		<button>Commit</button> <button>View</button> <button>Delete</button>
built-in	110	88cfef21e-b4e3-4c8b-8b37-8ca5941fed55	::1	2025-07-14 00:19:24	eth_win_test	admin	POST	/api/delete-provider	delete-provider		<button>Commit</button> <button>View</button> <button>Delete</button>
built-in	109	cf93e75c-501d-4aa7-a350-70509a90ba4	::1	2025-07-14 00:19:21	eth_win_test	admin	POST	/api/delete-provider	delete-provider	25	<button>Query</button> <button>View</button> <button>Delete</button>
built-in	108	357977eb-4d44-44d5-8714-13f7aff5a2fe	::1	2025-07-14 00:19:02	eth_win_test	admin	POST	/api/update-provider	update-provider	24	<button>Query</button> <button>View</button> <button>Delete</button>
built-in	107	66b29d6d-2b16-4d6b-9349-6be11748bea7	::1	2025-07-14 00:03:25	eth_win_test	admin	POST	/api/signin?code=77a6534407	signin	23	<button>Query</button> <button>View</button> <button>Delete</button>

💡 TEMPLATE FOR BROWSER URL

When you use the `{bh}` placeholder in the Browser URL template, Casibase will replace it with the actual block number and allow you to jump directly to the relevant block information in your blockchain explorer.

Example: Ganache CLI Block Explorer

To quickly view Ethereum blocks and transactions, you can use the open-source blockchain explorer [casibase/ganache-cli-block-explorer](#). This tool provides a simple web interface for browsing blocks, transactions, and contract events on your local Ethereum node.

The screenshot shows the Ganache Block-Explorer interface. On the left, there's a sidebar with a logo, the title "GANACHE BLOCK-EXPLORER", and two menu items: "Dashboard" and "Menu". The main area is titled "Transaction Dashboard" and contains four cards: "TOTAL NO. OF TRANSACTIONS" (1), "BLOCK NUMBER" (200), "BLOCK HASH" (0x65594c64b90b5805e0c876786b8319f5838179d6d0c314e53e3272b588fc83be), and "Transaction Status" (green). Below this is a section titled "Transaction Overview" with a table:

Transaction Hash	0xd05c3fb8aa26d168ae6d3cdb88948a6487282813afb738f2ec7773d86ae63a6a
Transaction Gas	336909
Transaction Gas Price	9
Transaction Nonce	199
Transaction To	0xD460001B04b4FD07194476C35825175B30F09Ec
Transaction From	0x71562b71999873DB5b286dF957af199Ec94617F7
Transaction Value [wei]	0
Transaction Status	SUCCESSFUL

Note: This explorer is based on [vivekganesan01/ganache-cli-block-explorer](#) and includes additional features contributed by Casibase.

After installation and startup, you can access block details directly from the above address as the Browser URL in Casibase.

Private Cloud Providers

Introduction

In Casibase, Private Cloud Providers act as a bridge, allowing you to connect to and manage various cloud-native resources, such as Docker and Kubernetes (K8s), directly from the Casibase interface. Their core objective is to provide a centralized dashboard for monitoring and operating your containerized services, integrating their management seamlessly into your Casibase workflow.

By configuring a provider, you enable Casibase to communicate with your private cloud or on-premises data center. This provides an ideal solution for organizations that want a unified interface to manage both their knowledge base and the infrastructure it runs on, enhancing operational efficiency and control.

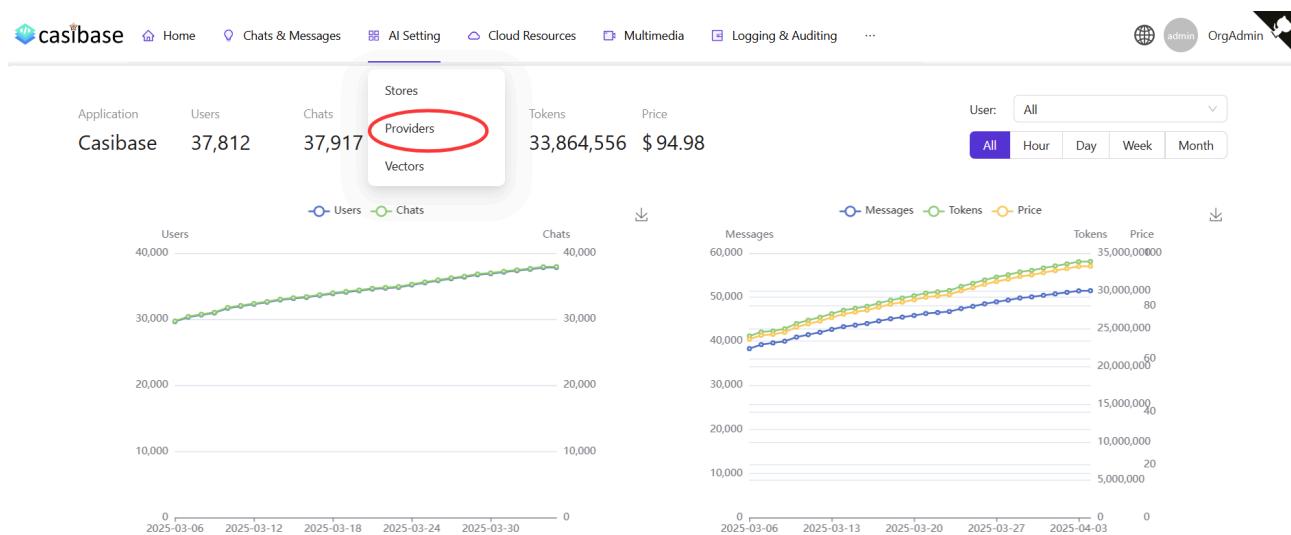
Refer to the [Core Concepts](#) section of our previous documentation for more information about providers.

In Casibase, you can add a private cloud provider by following these steps:

Add a New Private Cloud Provider

Private cloud providers are used to integrate cloud-native management features into Casibase. You can add them by following these steps:

Click the [Providers](#) button on the page.



Add a Private Cloud Provider

Click the [Add](#) button to add a new private cloud provider.

Providers	Add	Add Storage Provider						
Name		Display name	Category	Type	Sub type	API key	Secret key	Region
provider_tts_alibabacloud_cosyvoice		Provider TTS AlibabaCloud Cosyvoice	Text-to-Speech	Alibaba Cloud	cosyvoice-v1		***	
provider_blockchain_chainmaker		Provider Blockchain ChainMaker	Blockchain	Tencent ChainMaker (Demo Network)	text-davinci-003	AKIDObZHjvqUrxnNtkKdfQvAWcS1JK4XidG	***	ap-beijing
provider_model_alibabacloud_deepseek_r1		Provider Model AlibabaCloud DeepSeek-R1	Model	Alibaba Cloud	deepseek-r1		***	
provider_cloud_alibabacloud		Provider Cloud AlibabaCloud	Public Cloud	Aliyun	text-davinci-003	LTAI5tHkUsopAioN6sxilMg	***	cn-beijing
dall-e-3	dall-e-3	Model	OpenAI	dall-e-3			***	
provider_model_azure_gpt4_1		Provider Model Azure GPT-4	Model	Azure	gpt-4o	deployment-gpt-4o	***	
provider_model_azure_gpt4_2		Provider Model Azure GPT-4	Model	Azure	gpt-4o	deployment-gpt-4o	***	
provider_model_azure_gpt4_3		Provider Model Azure GPT-4	Model	Azure	gpt-4o	deployment-gpt-4o	***	
provider_embedding_openai_v3		Provider Embedding OpenAI V3	Embedding	OpenAI	text-embedding-ada-002		***	
provider_model_openai_gpt4_vision		Provider Model OpenAI GPT4 Vision	Model	OpenAI	gpt-4-vision-preview		***	

Fill in Private Cloud Provider Details

Fill in the required configuration according to the cloud-native platform you use, then click the **Save & Exit** button to save.

More information about the configuration can be found below:

Kubernetes Configuration

Kubernetes (K8s) is an open-source container orchestration platform for automating the deployment, scaling, and management of containerized applications. It has bec...

提示

Casibase supports several mainstream cloud-native technologies and platforms, including:

- Docker:** To connect to a Docker host and manage the lifecycle of its containers (e.g., start, stop, view status) directly within Casibase.
- Kubernetes (K8s):** To connect to a Kubernetes cluster and manage its resources, such as Pods and Deployments, providing a high-level orchestration view within Casibase.

Now, you can use this private cloud provider to monitor and manage services in your cloud-native environment.

After adding a private cloud provider, you can use Casibase as a control panel to oversee your containerized applications, simplifying management and providing a unified view of your services alongside your knowledge base.

Kubernetes Configuration

Kubernetes (K8s) is an open-source container orchestration platform for automating the deployment, scaling, and management of containerized applications. It has become the de facto standard for managing applications in modern, cloud-native environments. By providing a robust framework for running distributed systems resiliently, Kubernetes simplifies complex operational tasks.

In this chapter, you will learn how to configure and use a Kubernetes provider in Casibase. This will allow you to connect Casibase to your Kubernetes cluster, enabling you to monitor and manage your cloud resources directly from the Casibase interface.

1. Configuration Field Description

When configuring a Kubernetes provider in Casibase, you need to fill in several key fields. Each field has a specific meaning and is required for the correct integration with your Kubernetes cluster. The following list explains the purpose of each field:

- `Name`: The unique identifier for this private cloud provider.
- `Display name`: The display name shown in the UI for this provider.
- `Category`: The type of service; here it should be `Private Cloud`.
- `Type`: The cloud-native platform type; here it should be `Kubernetes`.
- `Config text`: The raw text content of your `kubeconfig` file, which contains the credentials and endpoint information needed to connect to your Kubernetes cluster.

Please make sure to fill in each field accurately. The `Config text` is crucial for

establishing a successful connection.

2. Configure Kubernetes

The primary method for connecting Casibase to your Kubernetes cluster is by using your `kubeconfig` file.

2.1 Obtain Your Kubeconfig File

Before proceeding, you must ensure that the `kubeconfig` file you intend to use can successfully connect to your Kubernetes cluster. A reliable way to get the raw configuration is to run the following command in your terminal:

```
kubectl config view --raw > kubeconfig.yaml
```

This command will save the complete, flattened configuration into a file named `kubeconfig.yaml` in your current directory. You can then open this file to copy its contents.

You can test your configuration file with a command that checks for pods across all namespaces. This is a more reliable test to confirm connectivity.

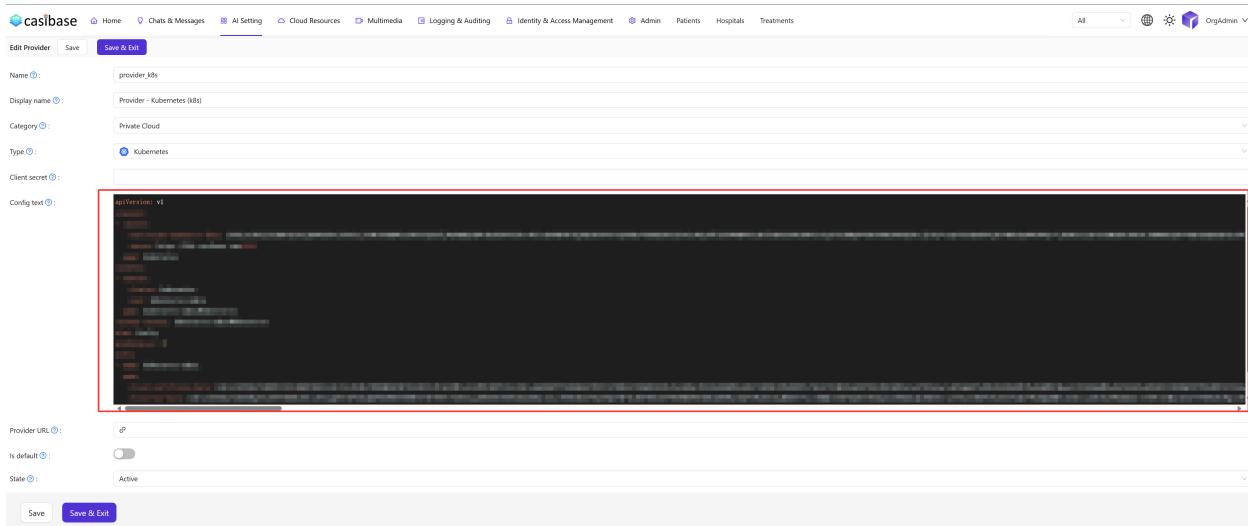
```
kubectl --kubeconfig=./kubeconfig.yaml get pods -A
```

If this command runs successfully (even if it just lists pods from system namespaces), you are ready to proceed.

2.2 Fill in the Provider Details

Copy the entire content of your valid `kubeconfig.yaml` file and paste it into the

`Config text` field in the provider configuration form.



3. Verify the Connection

After you have filled in the details and saved the provider, you can verify if the connection was successful.

Navigate to the Cloud Resources > Applications section within Casibase. Here, you will see a list of your configured providers. Check the status of the Kubernetes provider you just added.

- **Active:** If the status is `Active`, Casibase has successfully connected to your Kubernetes cluster.
- **Inactive:** If the status is `Inactive`, there was an issue with the connection. Please double-check the content of your `Config text` and ensure that there is network connectivity between Casibase and your Kubernetes cluster's API server.

The screenshot shows the Casibase application management interface. At the top, there is a navigation bar with links for Home, Chats & Messages, AI Setting, Cloud Resources, Multimedia, Logging & Auditing, Identity & Access Management, Admin, Patients, Hospitals, and Treatments. On the far right, there are icons for All, a globe, a sun, and OrgAdmin.

The main area displays a table titled "Applications" with the status set to "Active". The table has columns for Name, Display name, Created time, Description, Template, Status, Namespace, and Action. There are three entries:

Name	Display name	Created time	Description	Template	Status	Namespace	Action
application_grekm	New Application - grekm	2025-08-02 21:59:58		template_2	Running	casibase-application-grekm	Edit Undeploy Delete
application_jijkl	New Application - jijkl	2025-08-02 09:43:55		template_2	Running	casibase-application-jijkl	Edit Undeploy Delete
application_x04bi	New Application - x04bi	2025-08-01 23:44:27		template_2	Not Deployed	casibase-application-x04bi	Edit Deploy Delete

At the bottom right of the table, it says "3 in total" and "1 > 10 / page".

Once the connection is active, you can begin to monitor and manage your Kubernetes resources through Casibase.



>

存储

存储

概述

存储概述

存储配置

在添加存储提供商、模型提供商和嵌入提供商后，我们可以配置存储

概述

1. 存储功能概述

在 Casibase 中，存储功能是其核心模块之一，它允许用户集成存储、建模和嵌入服务提供商，用于知识库数据存储、文本向量转换以及与聊天机器人的交互。通过存储功能，用户可以构建一个高效、灵活且强大的 AI 知识管理系统。借助 Stores 功能，用户可以构建一个高效、灵活且功能强大的 AI 知识管理系统。

2. 存储的优势

2.1 多模型集成

Casibase 的存储功能支持多个主流 AI 语言模型，包括 OpenAI（如 GPT-3.5、GPT-4）、Azure OpenAI、HuggingFace、Google Gemini 等。这种多模型支持允许用户根据具体需求选择最合适的 AI 模型，在性能、成本和功能之间找到平衡。这种多模型支持使用户能够根据特定需求选择最合适的 AI 模型，并在性能、成本和功能之间取得平衡。

2.2 多种存储和嵌入选项

用户可以自由选择存储和嵌入服务提供商，以满足不同的数据存储和处理需求。用户可以自由选择存储和嵌入服务提供商，以满足不同的数据存储和处理需求。这种灵活性使用户能够根据其技术栈和业务需求配置最合适的存储和嵌入解决方案。

2.3 多存储模式

Casibase 支持多存储模式，允许用户在不同的存储中使用不同的模型、存储和嵌入服务，为不同场景和用户提供定制化服务。这个功能使用户能够根据不同的业务需求灵活配置和切换存储。此功能使用户能够根据不同的业务需求灵活配置和切换 Stores。

3. 总结

Casibase 的存储功能为用户提供了一个强大的知识管理工具，通过集成多个 AI 模型、存储和嵌入服务，使他们能够灵活地构建和管理知识库。其多存储模式和企业级功能进一步增强了系统的灵活性和安全性，适用于各种应用场景。其多 Stores 模型和企业级功能进一步增强了系统的灵活性和安全性，适用于各种应用场景。

Casibase 是一个开源的 AI 知识库系统，旨在为企业提供高效灵活的知识管理和对话解决方案。其核心功能之一是 Providers，允许用户整合多种 AI 模型和存储服务，以提升系统的功能和性能。Providers 分为三大类：Model Providers、Embedding Providers 和 Storage Providers，分别负责处理 AI 模型和数据存储。

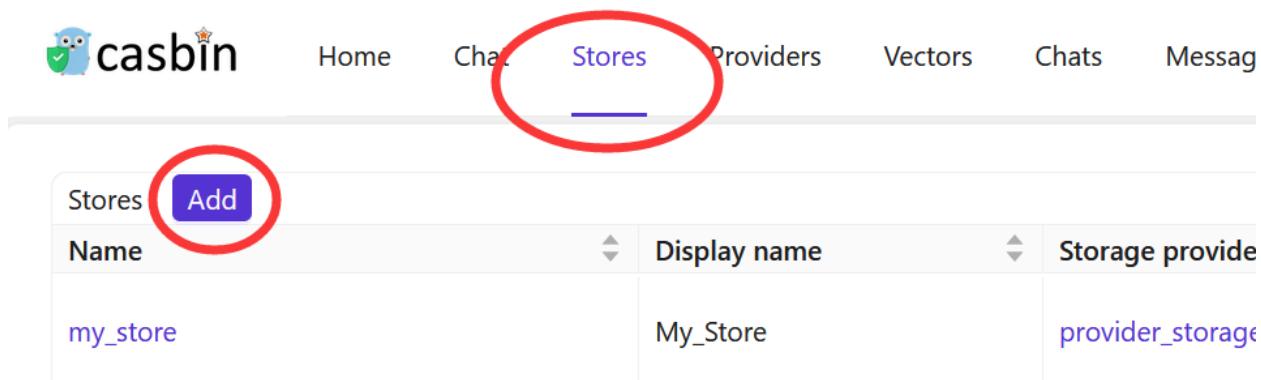
存储配置

在添加存储提供商、模型提供商和嵌入提供商后，我们可以配置存储

1. 添加新存储

存储用于将存储、模型和嵌入提供商集成到 Casibase 中。你可以按照以下步骤添加它们：您可以按照以下步骤添加它们：

点击主页上的 存储 按钮，然后点击 添加 按钮来添加存储。



Name	Display name	Storage provider
my_store	My_Store	provider_storage

填写存储详情

填写存储详情并点击 保存并退出 按钮。

casbin Home Chat Stores Providers Vectors Chats Messages Tasks Resources Permissions Logs Jimmy

Edit Store Save

Name: store_v6c22m

Display name: New Store - v6c22m

Storage provider:

Model provider:

Embedding provider:

File tree:



storage provider is empty

Go to Store

Save

Powered by Casibase

请选择之前添加的存储提供商、模型提供商、嵌入服务提供商、文本转语音服务提供商和语音转文本服务提供商。

Edit Store Save

Name:	<input type="text" value="my_store"/>
Display name:	<input type="text" value="My_Store"/>
Storage provider:	<input type="text" value="Provider_storage_1 (provider_storage_1)"/>
Model provider:	<input type="text" value="Model OpenAI text-davinci-003 (model_openai_text_davinci_003)"/>
Embedding provider:	<input type="text" value="Embedding_OpenAI_Adasimilarity (embedding_openai_adasimilarity)"/>
File tree:	<div style="border: 1px solid #ccc; padding: 5px;"> □ □ □ My_Store <ul style="list-style-type: none"> □ □ □ alibaba_oss <ul style="list-style-type: none"> □ □ □ audio <ul style="list-style-type: none"> □ AC / DC - Highway To Hell.mp3 (8.34 MB) □ □ □ document <ul style="list-style-type: none"> □ casdoor-knowledge.doc (18.0 KB) □ casdoor-knowledge.docx (10.9 KB) □ casdoor-knowledge.html (23.5 KB) □ casdoor-knowledge.md (2.12 KB) □ casdoor-knowledge.pdf (107 KB) □ □ □ image <ul style="list-style-type: none"> □ lena.jpg (105 KB) □ lena.tiff (768 KB) □ □ □ video <ul style="list-style-type: none"> □ my_video.mkv (456 KB) </div>

点击 保存并退出 按钮并返回存储列表页面：

Stores Add

Name	Display name	Storage provider	Model provider	Embedding provider	Action
my_store	My_Store	provider_storage_1	model_openai_text_davinci_003	embedding_openai_adasimilarity	View Refresh Vectors Edit Delete

现在，你可以使用该存储来存储知识库数据、转换文本为向量，并与聊天机器人对话。

在下一节中，我们将学习如何在 Casibase 中与聊天机器人对话。

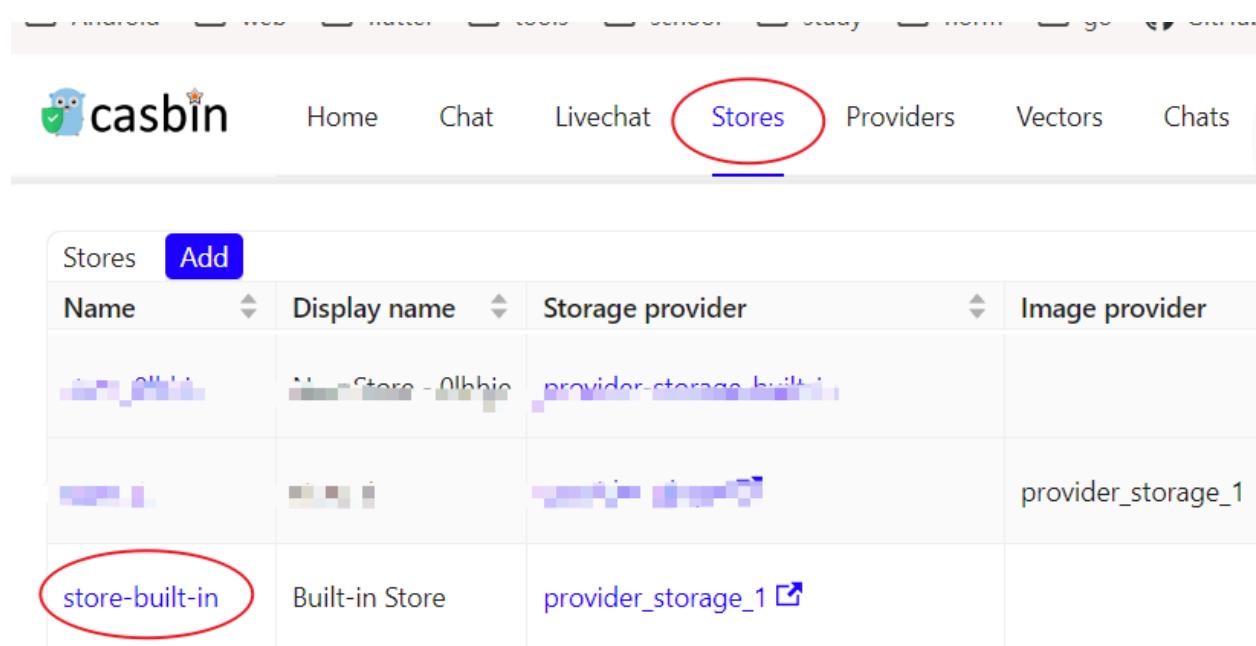
2. 支持多存储

多存储模式为用户在每个不同的存储中提供不同的模型、建议等功能。

启用多存储

首先，你需要在内置存储中启用多存储模式。

点击主页上的 `存储` 按钮，然后点击 `store-built-in` 按钮进入内置存储。



The screenshot shows the Casibase interface with the 'Stores' tab selected. A red circle highlights the 'store-built-in' entry in the table below. The table has columns: Name, Display name, Storage provider, and Image provider. The 'store-built-in' entry has a blue border around its 'Name' field.

Name	Display name	Storage provider	Image provider
store-built-in	Store - Olbbie	provider-storage_builtin	
			provider_storage_1
store-built-in	Built-in Store	provider_storage_1	

向下滚动找到 `可选择存储` 字段，勾选它。

The screenshot shows a software interface with the following elements:

- Suggestion count:** 3
- Theme color:** A blue square.
- Can Select Store:** A checked checkbox, highlighted with a red oval.
- File tree:** A hierarchical list of files and folders under "casibase".
 - Built-in Store
 - casibase
 - user_jaycd9
 - chat_0edmw
 - 20240508_070749-1.png (40.4 KB)
 - 20240508_070749.png (40.4 KB)
 - 20240508_070937-1.txt (0 B)
 - 20240508_070937.txt (40.4 KB)
 - 20240508_071008-1.txt (18 B)
 - 20240508_071008.txt (40.4 KB)
 - 20240508_071037.md (6.21 KB)
 - chat_1gs4l9
 - 20240509_202051.jpg (639 KB)
 - chat_1ikor4

添加可用存储

多店模式仅提供可用的店铺。要使存储可用，您需要配置其存储提供商、模型提供商和嵌入提供商。

选择对话存储

Casibase 提供了一个非常方便的方法来选择存储。



Home Chat Livechat Stores Providers Vectors Chats Messages Usages Frameworks

The screenshot shows the casbin Chat interface. At the top, there's a navigation bar with links: Home, Chat (which is highlighted in blue), Livechat, Stores, Providers, Vectors, Chats, Messages, Usages, and Frameworks. Below the navigation bar, there's a sidebar on the left containing a button labeled "+ New Chat" and a list of storage options: "store_1" and "store-built-in". A red box highlights the "+ New Chat" button. To the right of the sidebar, a main panel titled "New Chat - 7" displays a message from a bot: "You are an expert in your field" followed by a response: "Thank you for recognizing my expertise. Whether it's related to my specific area of knowledge or expertise to provide insightful answers and solutions to any problems you may have." Below the message are five small blue icons representing different actions: a square, a circle, a triangle, a square with a dot, and a circle with a dot.

只需将鼠标悬停在"新建对话"上，然后你就可以从下方出现的列表中选择你想要使用的存储。

如果你点击"新建对话"按钮，系统将为你分配一个默认存储。



>

Forms

Forms



Forms Overview

Introduction to Forms in Casibase



Form Configuration

How to configure and customize forms in Casibase

Forms Overview

Introduction

Forms in Casibase provide a powerful way to customize the display of list pages throughout the application. By configuring forms, you can control which columns are visible, their order, and their width in various list views such as Records, Providers, Stores, and more.

The Forms feature was ported from Casdoor and allows administrators to tailor the user interface to their specific needs, improving usability and focusing on the most relevant information.

Key Features

Customizable Column Visibility

Control which columns appear in list pages by toggling the visibility setting for each form item. This helps reduce clutter and focus on the most important data.

Adjustable Column Width

Set custom widths for each column to optimize the display based on the content type and your screen size preferences.

Column Ordering

Arrange columns in the order that makes the most sense for your workflow by reordering form items.

Multiple Form Categories

Forms support different categories to serve various purposes:

- **Table:** Traditional table-based forms for displaying structured data
- **iFrame:** Forms that can embed external content
- **List Page:** Forms specifically designed to customize list page columns

Form Structure

Each form in Casibase consists of:

- **Organization:** The organization that owns the form
- **Name:** Unique identifier for the form
- **Display Name:** Human-readable name shown in the interface
- **Position:** Placement or order of the form
- **Category:** Type of form (Table, iFrame, or List Page)
- **Type:** Specific form type (e.g., "records" for the Records list page)
- **URL:** Associated URL or endpoint (for Table and iFrame categories)
- **Form Items:** Collection of columns or fields to display

Form Items

Form items define the individual columns in a list page. Each form item includes:

- **Name:** Internal column identifier (e.g., "organization", "name", "createdTime")
- **Label:** Display label shown in the column header
- **Type:** Data type (currently "Text" for list pages)
- **Visible:** Whether the column is displayed
- **Width:** Column width in pixels

Use Cases

Forms are particularly useful for:

1. **Customizing Record Views:** Tailor the Records list page to show only relevant columns for your use case
2. **Simplifying Complex Tables:** Hide technical columns that aren't needed by all users
3. **Optimizing Screen Space:** Adjust column widths to fit more information on screen
4. **Role-Based Views:** Create different forms for different user roles or workflows

Getting Started

To start using Forms in Casibase:

1. Navigate to the Forms section in the Casibase admin interface
2. Create a new form or edit an existing one

3. Configure the form category, type, and items
4. Preview your changes in real-time
5. Save and apply the form to the corresponding list page

For detailed instructions on configuring forms, see the [Form Configuration](#) guide.

Form Configuration

Accessing Forms

Forms can be accessed through the Casibase admin interface:

1. Log in to your Casibase admin dashboard
2. Navigate to the Forms section from the main menu
3. You'll see a list of existing forms organized by category and type

Creating a New Form

Step 1: Basic Information

To create a new form:

1. Click the Add button on the Forms list page
2. Fill in the basic form information:
 - Organization: Select the organization (typically your organization name)
 - Name: Enter a unique identifier for the form
 - Display Name: Provide a human-readable name
 - Position: Set the position or order (optional)

Step 2: Select Form Category

Choose the appropriate category for your form:

- **Table:** For traditional table-based forms
- **iFrame:** For embedding external content
- **List Page:** For customizing list page columns (recommended for most use cases)

Step 3: Configure Form Type

If you selected **List Page** as the category:

1. Choose the **Type** from the dropdown menu
 - Currently supported: **Records**
 - More types will be added in future versions
2. The form will automatically populate with default columns for the selected type

Step 4: Customize Form Items

For each form item (column), you can configure:

Name

The internal identifier for the column. This corresponds to the data field being displayed.

Available columns for Records include:

- `organization`: Organization name
- `id`: Record ID
- `name`: Record name
- `clientIp`: Client IP address

- `createdTime`: Creation timestamp
- `provider`: AI provider name
- `provider2`: Secondary provider
- `user`: Associated user
- `method`: HTTP method
- `requestUri`: Request URI
- `language`: Language
- `query`: Query parameters
- `region`: Geographic region
- `city`: City
- `unit`: Unit information
- `section`: Section
- `response`: Response data
- `object`: Related object
- `errorText`: Error messages
- `isTriggered`: Trigger status
- `action`: Action column
- `block`: Block information
- `block2`: Secondary block information

Visible

Toggle to show or hide the column in the list page. Hidden columns are still available in the form configuration but won't appear in the UI.

Width

Set the column width in pixels. This helps optimize the display based on the content length and your screen size.

提示

- Use narrower widths (90-120px) for short fields like IDs, dates, and status indicators
- Use wider widths (200-300px) for longer text fields like names, descriptions, and URLs
- Adjust widths based on your typical content length to avoid truncation

Step 5: Reorder Columns

Arrange columns in your preferred order:

1. Use the Up arrow button to move a column up
2. Use the Down arrow button to move a column down
3. The leftmost columns will appear first in the list page

Step 6: Add or Remove Columns

- Add Column: Click the Add button to create a new custom column
- Remove Column: Click the Delete button next to a column to remove it
- Reset to Default: Click Reset to Default to restore the original column configuration

Step 7: Preview and Save

1. View the Preview section at the bottom of the form editor
2. The preview shows how your form will appear in the actual list page
3. Click on the preview to open the full list page in a new window
4. Once satisfied, click Save & Exit to apply your changes

Editing Existing Forms

To modify an existing form:

1. Navigate to the Forms list page
2. Click on the form name or the edit button
3. Make your desired changes
4. Click **Save & Exit** to apply the updates

Changes take effect immediately for all users viewing the corresponding list page.

Form Categories in Detail

List Page Forms

List Page forms are the most commonly used type in Casibase. They allow you to:

- Customize which columns appear in list views
- Control column order and width
- Show/hide columns based on user needs
- Create optimized views for different workflows



When configuring List Page forms:

- The **Action** column is always displayed at the end, regardless of form item configuration
- At least one column should be visible for the list page to be functional

- Column names must match the actual data fields available in the backend

Table Forms

Table forms are used for structured data display in table format. Configuration is similar to List Page forms but may have different available fields based on the data source.

iFrame Forms

iFrame forms allow you to embed external content or applications within the Casibase interface:

1. Set the URL field to the external content address
2. Configure display settings as needed
3. The content will be displayed in an embedded frame



>

向量

向量

概述

向量概述

向量生成

向量的生成需要与存储配合使用，这意味着你需要在理解向量之前配置存储。

概述

在 Casibase 中，向量是其核心优势之一。向量技术在知识表示和检索中起着关键作用，通过结合将文本和图像等数据转换为密集向量的 stores 功能，Casibase 实现了高效的相似性搜索和数据分析。

关于向量的定义，请参阅我们之前文档中的 [核心概念](#) 部分。

Casibase 中向量技术的应用

知识嵌入

用户可以上传各种格式的文件（例如 TXT、Markdown、Docx、PDF 等）并选择嵌入方法（例如 Word2Vec、GloVe、BERT 等）生成知识及相应的向量。这些向量存储在向量数据库中，以便快速检索和查询。

相似度搜索

Casibase 将知识转换为向量并存储在向量数据库中。这种向量表示支持强大的相似度搜索功能，使用户能够基于上下文或内容快速找到相关信息。这种向量表示支持强大的相似性搜索功能，使用户能够根据上下文或内容快速找到相关信息。

向量生成

向量的生成需要与存储配合使用，这意味着你需要在理解向量之前配置存储。

向量实际上是嵌入的结果，即将各种数据（例如文本和图像）转换为密集向量表示的过程。此步骤对于促进 Casibase 内高效数据处理和分析至关重要。通过嵌入，聊天中的问题和存储中的知识文件将被转换为用于下一步知识搜索的向量。

1. 刷新向量

在 stores 菜单下的每个存储数据中，都设置了“刷新向量”操作按钮。在 stores 中，由于我们将设置存储提供商，它将为我们提供一个用于存储用户文件的文件树，因此配置完 stores 后，请保存配置并返回主页，您将看到存储提供商的文件树。

通过点击特定存储的刷新向量按钮，它将通过嵌入为该存储的文件树中的所有文件生成相应的向量。下图显示了页面和操作。下图显示了页面和操作。

Name	Display name	Storage provider	Image provider	Model provider	Embedding provider	Memory limit	State	Action
store_08v0xx	New Store - 08v0xx	provider-storage-built-in		dall-e-3	provider_embedding.azure	5	Active	View Refresh Vectors Edit Delete
store_3g89qb	New Store - 3g89qb	provider-storage-built-in		dall-e-3	provider_embedding.azure	5	Active	View Refresh Vectors Edit Delete
store_89ptvi	New Store - 89ptvi	provider-storage-built-in		dall-e-3	provider_embedding.azure	5	Active	View Refresh Vectors Edit Delete
store_kqn8y8	New Store - kqn8y8	provider-storage-built-in		dall-e-3	provider_embedding.azure	5	Active	View Refresh Vectors Edit Delete
store_14hy2t	New Store - 14hy2t	provider-storage-built-in		dall-e-3	provider_embedding.azure	5	Active	View Refresh Vectors Edit Delete
store_coldql	New Store - coldql	provider-storage-built-in		dall-e-3	provider_embedding.azure	5	Active	View Refresh Vectors Edit Delete
store_qc0ptn	New Store - qc0ptn	provider-storage-built-in		dall-e-3	provider_embedding.azure	5	Active	View Refresh Vectors Edit Delete
store_1yvlie	New Store - 1yvlie	provider-storage-built-in		dall-e-3	provider_embedding.azure	5	Active	View Refresh Vectors Edit Delete
store_dpo4j5	New Store - dpo4j5	provider-storage-built-in		dall-e-3	provider_embedding.azure	5	Active	View Refresh Vectors Edit Delete
store_wcjnxk	New Store - wcjnxk	provider-storage-built-in		dall-e-3	provider_embedding.azure	5	Active	View Refresh Vectors Edit Delete

2. 查看向量

之后，我们可以在向量菜单中查看该存储生成的具体向量。

The screenshot shows the 'Vectors' section of the casibase interface. A red box highlights the 'Delete All' button at the top left of the table header. The table lists nine vectors, each with a unique name, store, provider, file, index, text, size, data, dimension, and edit/delete actions. The 'Text' column contains the raw vector data.

Name	Store	Provider	File	Index	Text	Size	Data	Dimension	Action
vector_opfabu	store_08v0xx	provider_embedding_azure	casdoor-website/static/ads.txt	0	google.com, pub-3509678918753247, DIRECT, f08c47fe0c0942fa0	[-0.0111380682,-0.019052684,-0.0176554,-0.009101934,-0.0176554,-0.0021726459,0.013737783,-0.0021617997,-0.025320875,0.009781995,0.015853297,0.011648386,-0.030035077,0.008488]	1536	<button>Edit</button> <button>Delete</button>	
vector_mqhdg8	store_08v0xx	provider_embedding_azure	casdoor-website/docs/webhooks/overview.md	0	... title: Overview description: Adding Webhooks in Casdoor	[-0.0041976073,-0.0032014789,-0.00190431330,0.00293...	1536	<button>Edit</button> <button>Delete</button>	
vector_vagn2q	store_08v0xx	provider_embedding_azure	casdoor-website/docs/user/roles.md	0	... title: User Roles description: Roles assigned to users ...	[-0.005397057,0.0042889416,-0.0072895316,-0.035272...	1536	<button>Edit</button> <button>Delete</button>	
vector_elpwu	store_08v0xx	provider_embedding_azure	casdoor-website/docs/user/permissions.md	0	... title: Permissions description: User Permissions keyword, - 'Baidu' - 'Casdoor' - 'Infoflow' - 'Apple' - 'AzureAD' - ...	[0.0022299571,0.0068149953,0.0028915738,-0.054437...	1536	<button>Edit</button> <button>Delete</button>	
vector_ev350m	store_08v0xx	provider_embedding_azure	casdoor-website/docs/user/overview.md	2	Baidu - Casdoor - Infoflow - Apple - AzureAD - ...	[-0.009141525,0.021197738,-0.19946484,-0.0207195...	1536	<button>Edit</button> <button>Delete</button>	
vector_pj5q7q	store_08v0xx	provider_embedding_azure	casdoor-website/docs/user/overview.md	3	When migrating users from an external database to Casdoor, ...	[-0.004490083,0.010098264,0.0035382845,-0.02122970...	1536	<button>Edit</button> <button>Delete</button>	
vector_7exw5a	store_08v0xx	provider_embedding_azure	casdoor-website/docs/user/overview.md	0	... title: Overview description: Managing Users in Casdoor k...	[0.014196097,0.017991772,0.009771807,-0.03401034...	1536	<button>Edit</button> <button>Delete</button>	
vector_8l70kd	store_08v0xx	provider_embedding_azure	casdoor-website/docs/user/overview.md	1	- 'Gender' - 'Birthday' - 'Education' - 'Score' - 'Karma' - ...	[-0.008841777,0.034623653,0.0250997,-0.0371953...	1536	<button>Edit</button> <button>Delete</button>	
vector_1p401n	store_08v0xx	provider_embedding_azure	casdoor-website/docs/user/multi-factor-authentication.md	12	2. Then enter the code into the "Enter your code" field and ...	[-0.0010608328,-0.0078019355,-0.025427068,-0.0117...	1536	<button>Edit</button> <button>Delete</button>	
vector_vqx3g6	store_08v0xx	provider_embedding_azure	casdoor-website/docs/user/multi-factor-authentication.md	13	3. Above the "Enable" button, copy your recovery codes and S...	[-0.021876315,0.013569352,-0.002898435,-0.0218238...	1536	<button>Edit</button> <button>Delete</button>	

我们可以看到，上一步刷新向量中的存储文件已被转换为向量在此显示。

The screenshot shows the 'Edit Vector' page for the vector named 'vector_opfabu'. The form includes fields for Name, Display name, Store, Provider, File, Text, Size, Dimension, and a large text area for the raw vector data. The 'Text' field is filled with the raw vector data, and the 'Dimension' field is set to 1536.

Form fields:

- Name: vector_opfabu
- Display name: google.com, pub-3509678918753247
- Store: store_08v0xx
- Provider: provider_embedding_azure
- File: casdoor-website/static/ads.txt
- Text: google.com, pub-3509678918753247, DIRECT, f08c47fe0c0942fa0
- Size: 1536
- Dimension: 1536
- Data (Raw Vector):

```
-0.0111380682,-0.019052684,-0.0177048,-0.036407735,-0.03337811,0.0012354781,-0.0094967177,-0.009500225,-0.009101934,-0.0176554,-0.0021726459,0.013737783,-0.0021617997,-0.025320875,0.009781995,0.015853297,0.011648386,-0.030035077,0.008488
174,-0.010557982,-0.02164431,-0.0058601047,-0.0033985341,-0.014873892,-0.012882436,0.019300002,0.0116028,-0.006908675,0.004397527,-0.015695593,-0.00404941,0.0025301287,-0.0298787,-0.023645357,-0.0125233205,-0.0073324763,0.0032565205,-0.008762407,0.009813635,-0.00664466,-0.012955273,-0.0011622269,0.008363244,-0.003951245,-0.02517723,-0.031654558,-0.01241851,0.0072933,-0.009767929,0.00856114,0.045097832,0.01922245,0.01188343,-0.0040351474,0.038340427,-0.01307788,-0.01057516,-0.00465272,0.016610702,0.01219032,0.02223905,-0.01470479,-0.0124651894,0.013236479,0.0028272148,0.0116490491,0.01766846,0.034971274,-0.0008504497,0.00391501,0.00805001,-0.00416247,0.01688569,-0.006092713,-0.006895009,0.0269607,-0.011328447,0.041814048,0.016401745,-0.01463883,-0.029972319,0.0005244117,0.012849799,-0.024615703,-0.01761625,-0.016349528,-0.017133052,0.000318365,0.002393012,-0.030217899,-0.003307123,-0.010629805,0.011119507,-0.010120515,-0.002301601,-0.03721847,0.003625103,0.02382775,-0.027971797,-0.017198347,-0.033267112,0.015461535,-0.024524292,-0.013359797,0.001779948,0.003715616,0.03473622,-0.016976347,-0.019078802,-0.001400207,-0.021298787,-0.0064085713,0.03091894,0.009976167,-0.01062452,-0.001544932,0.0382702,0.01339492,0.034866806,0.013032611,0.008971347,0.014234017,-0.0204107
93,-0.00670567,-0.006842774,0.029920812,0.01254296,-0.002988105,0.0422058,0.010314381,-0.01268655,-0.0046946,13,0.0009172721,0.01988444,-0.01405792,0.004897023,0.028572498,0.000735893,-0.004521948,0.0027737117,-0.019196332,0.004782
759,-0.0037968254,-0.005256138,0.02516756,-0.0209193,0.03207224,0.032516234,-0.006454999,-0.0039372067,-0.022617718,0.006451012,-0.0095478,0.03322140,-0.01024553,-0.01082566,-0.00314876,0.021638313,0.0007088441,-0.012224949,-0.01894
8216,0.02474629,0.012367688,0.007743826,-0.00691997,-0.6815612,-0.02324457,0.01517798,-0.02003209,0.027177143,0.0048036,0.018425867,0.005768935,0.000339922,-0.00120335,-0.00952460,0.00263675,-0.01230782,-0.005868985,0.015004479,0.02747594,0.010113985,0.015148126,0.0099768685,0.00103051,0.0203455,-0.003091654,0.012360087,-0.01796811,-0.001394334,0.031654358,-0.002336677,-0.005931519,0.012902024,-0.019091861,0.010159691,0.035284683,0.005719723,0.03254683,0.027355936,-0.007355893,0.016181943,-0.001465026,0.043407213,-0.0446005,-0.016675996,0.013594137,0.00955899,-0.004733792,0.015082815,0.025595108,-0.009506755,-0.022526307,-0.013711665,0.01474129,-0.00886465,-0.0042147045,0.010342513,0.0040220884,-0.007190667,0.0063857165,-0.0068493034,-0.02357464,0.012307852,-0.0036629734,-0.008279324,-0.01937677,-0.024864028,-0.026072824,0.010257632,-0.014952244,0.026365573,0.02453735,0.012373146,-0.005223497,0.02085479,0.008866876,-0.01486083,0.0156505,-0.023440419,0.018321397,0.001488491,0.007286770,0.026979333,-0.001184801,0.00385235,0.013907546,0.0056201945,0.006666481,0.0024370851,-0.010819157,0.034840688,0.0060037505,-0.015853297,-0.02085479,0.008866876,-0.01486083,0.0156505
```

我的向量编辑页面显示了具体信息，如存储名称、嵌入模型名称、进行嵌入的文件名、文件大小、维度、向量数据等。



>

文本切分器

文本切分器

概述

文本切分器概述

概述

文本切分器 是构建大型语言模型(LLM)应用程序的关键组件。它们的主要作用是将长文本分割成多个较短的段落，从而便于后续任务，例如文本嵌入、增强检索生成(RAG)和问答系统。

在大型语言模型中，文本分割主要基于以下几个原因：

- 提高效率和准确性：通过将大段文本分解为较小的段落，可以优化大型语言模型生成的嵌入向量的相关性和准确性。分块有助于确保嵌入内容在保留语义相关性的同时，噪音最小。例如，在语义搜索中，当为文档库建立索引时，每个文档都包含特定主题的有价值信息。采用有效的分块策略确保搜索结果能够准确捕捉用户查询的本质。
- 限制上下文窗口大小：在使用类似 GPT-4 的模型时，可处理的 token 数量是有限的。例如，GPT-4 的上下文窗口大小限制为 32K 个 token。虽然这一限制通常不会构成问题，但从一开始就考虑分块大小非常重要。如果文本块过大，可能会丢失信息或无法将所有内容嵌入上下文中，从而影响模型的性能和输出。
- 处理长文档：虽然长文档的嵌入向量可以捕捉整体上下文，但它们可能会忽略与特定主题相关的重要细节，导致输出不精确或不完整。分块使得对信息的提取和嵌入有更好的控制，从而降低信息丢失的风险。

Casibase目前提供多种文本切分方法，允许用户针对不同文本场景采用不同的处理策略。

默认文本切分器

默认文本切分器旨在根据标记数量和文本结构高效地分段文本。其分割策略包括：

- 逐行读取和段落识别：文本按行读取，通过连续空行准确确定段落分隔。它还通过标记敏感地识别自然断点，确保文本分割逻辑严谨且精确。
- 代码块的特殊处理：被```符号包围的代码块将被单独处理。代码块内的行数决定了它是否可以独立作为一个段落。这一机制既保持了代码块的完整性，又有效防止单个文本段落超过标记限制。
- 维护句子完整性：在整个分割过程中，严格保持句子完整，确保句子不会被分割。这一特性保证每个文本段落都包含一个完整的信息单元。不論文本多么复杂，分割均精确地在句子边界进行，有效避免因句子被拆分而引起的歧义和信息丢失。

问答切分器

问答切分器专注于对问答格式文本进行精确分割，并提供以下核心优势：

- 问答单元的准确切分：它采用逐行扫描机制，智能识别问答文本的结构。通过确定每行是否以“Q:”或“A:”开头，它能精确定位问题与答案之间的边界，确保每个问答对都被完整分割。这确保了每个问答单元的独立性和完整性，为后续的问答处理和分析提供了清晰的数据。
- 清晰且逻辑性强的实现：代码简洁直观，易于理解和维护。通过管理当前问答对的状态以及指示是否正在收集答案的标志，文本分割过程得到了清晰控制，确保每个问答单元正确配对。



聊天

聊天

概述

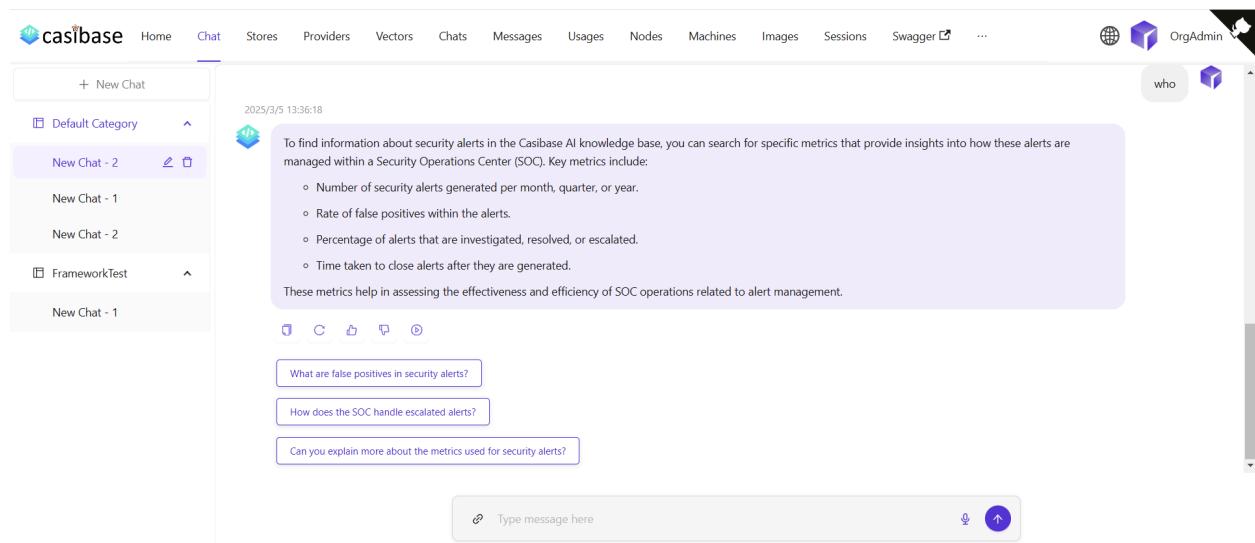
聊天概述

概述

在本节中，我们介绍Casibase最核心的部分：聊天及其管理。

1. 聊天

一旦我们配置好存储，我们就可以与AI进行对话。如下图所示：



2. 聊天管理

我们可以从聊天菜单管理我们的聊天会话。

The screenshot shows the casibase Chats page. At the top, there are navigation links: Home, Chat, Stores, Providers, Vectors, Chats (highlighted with a red box), Messages, Usages, Nodes, Machines, Images, Sessions, Swagger, and more. On the right, there are icons for OrgAdmin and a user profile.

Below the header, there is a search bar with dropdowns for Chats, Users, Chats, Messages, Tokens, Price, and Action. The main table lists three chats:

Name	Updated time	User	Client IP	Count	Token count	Price	Messages	Action
chat_j916c0	2025-03-05 13:37:02	u-0b9800aa	119.164.218.30 中国 山东 济南 Edge 133.0.0 Windows 10	1	1006	\$0.002705		<button>Edit</button> <button>Delete</button>
chat_v67r4z	2025-03-05 13:36:48	u-649ef853	101.129.8.189 中国 台湾 N/A Chrome 133.0.0 Mac OS X 10.15.7	1	1004	\$0.00269		<button>Edit</button> <button>Delete</button>
chat_252ftr	2025-03-05 13:36:18	admin	119.164.218.30 中国 山东 济南 Edge 133.0.0 Windows 10	12	7107	\$0.0203981	<div style="border: 1px solid #ccc; padding: 5px;"> <p>2025/</p> <p>2025/3/4 00:55:33</p> <p>Hello! How can I assist you today?</p> <p>Edit Delete</p> </div>	

A red box highlights the last row of the table, which corresponds to the detailed view shown on the right. This view includes a message history from the bot and a user interaction with a "Delete" button.

该页面允许用户查看已创建的聊天信息，用户也可以点击编辑来查看或编辑它们。它们显示以下信息：它们显示以下信息：

- 名称：创建的聊天名称。
- 更新时间：聊天更新的时间。
- 用户：聊天所属的用户。
- 客户端IP：聊天的客户端IP。
- 计数：此聊天的输入和输出数量。
- 令牌计数：此聊天使用的总令牌数。
- 价格：此聊天花费的总价格。
- 消息：显示聊天的消息内容。
- 存储：显示聊天所属的存储。
- 类别：显示聊天所属的类别。



>

消息

消息



概述

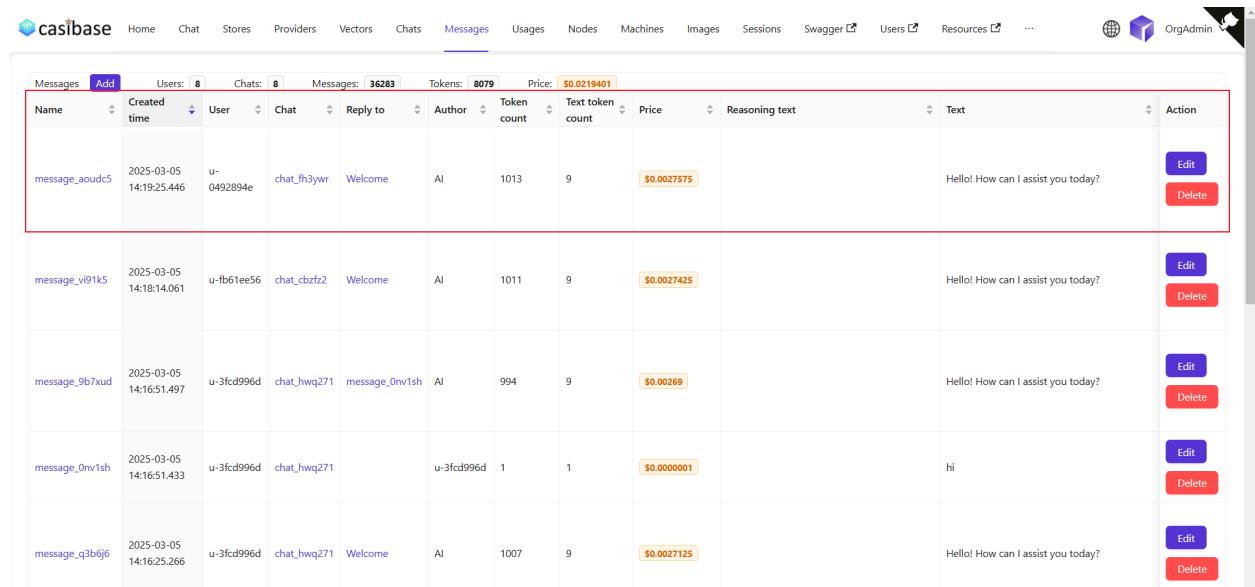
消息概述

概述

在本节中，我们介绍Casibase中消息功能。

消息

消息模块管理我们会话中的所有消息，它显示每条消息的创建时间、所属的聊天、父消息、令牌数量、价格、回复的文本消息、向量、建议等信息。



Name	Created time	User	Chat	Reply to	Author	Token count	Text token count	Price	Reasoning text	Text	Action
message_aoudc5	2025-03-05 14:19:25.446	u-0492894e	chat_fh3yw	Welcome	AI	1013	9	\$0.0027575		Hello! How can I assist you today?	<button>Edit</button> <button>Delete</button>
message_v91k5	2025-03-05 14:18:14.061	u-fb61ee56	chat_cbfzf2	Welcome	AI	1011	9	\$0.0027425		Hello! How can I assist you today?	<button>Edit</button> <button>Delete</button>
message_9b7xud	2025-03-05 14:16:51.497	u-3fcdb996d	chat_hwq271	message_0nv1sh	AI	994	9	\$0.00269		Hello! How can I assist you today?	<button>Edit</button> <button>Delete</button>
message_0nv1sh	2025-03-05 14:16:51.433	u-3fcdb996d	chat_hwq271		u-3fcdb996d	1	1	\$0.0000001		hi	<button>Edit</button> <button>Delete</button>
message_q3b6j6	2025-03-05 14:16:25.266	u-3fcdb996d	chat_hwq271	Welcome	AI	1007	9	\$0.0027125		Hello! How can I assist you today?	<button>Edit</button> <button>Delete</button>



> Container Cloud

Container Cloud

Overview

Container Cloud Overview

Template

In Casibase, a Template is a reusable base configuration for an application. It contains the core Kubernetes manifest files, typically structured for use with Kustomize. Y...

Applications

1 个项目

Overview

Once you have successfully connected Casibase to your private cloud providers (like Kubernetes), this section will guide you on how to manage cloud-native resources directly through the Casibase interface.

Casibase provides a powerful system based on Docker and Kubernetes, designed for individuals and organizations to build their own dedicated container cloud environment. Built on the Casbin permission management engine, it implements fine-grained access control policies for secure and controllable private cloud operations.

Core Concepts

Casibase utilizes a streamlined two-part system for managing container deployments:

1. Templates: Reusable Application Blueprints

Templates are pre-configured Kubernetes manifests that serve as blueprints for your applications. Each template contains:

- **Base Configuration:** Complete Kubernetes resources (Deployments, Services, ConfigMaps, etc.) required to run an application
- **Customizable Parameters:** Configurable fields that can be modified during deployment
- **Version Management:** Template versioning for consistent deployments across environments

2. Applications: Live Application Instances

Applications are running instances created from templates. They represent actual workloads deployed to your Kubernetes cluster:

- **Parameter Customization:** Override template defaults with specific configurations (replicas, image versions, resource limits)
- **Namespace Isolation:** Each application runs in its own dedicated namespace for security and organization
- **Lifecycle Management:** Complete application lifecycle control from deployment to termination

Key Features

Declarative Application Orchestration: Transform from resource-level management to application-level management, simplifying complex multi-resource deployments into simple "select template → configure → deploy" workflows.

Service Governance Integration: Built-in support for service mesh and gateway templates (Istio, Linkerd, Nginx Ingress) enabling one-click deployment of microservice governance capabilities including service discovery, circuit breaking, and rate limiting.

Enhanced Platform Visualization:

- **Application Dashboard:** Monitor and manage all your deployed applications with real-time status updates
- **Resource Insights:** Deep visibility into underlying Kubernetes resources, logs, and events for each application

Kustomize-Powered Flexibility: Leverages Kubernetes-native Kustomize for configuration management, ensuring consistency across development, testing, and production environments while maintaining the ability to customize deployments per environment.

This approach helps you standardize your infrastructure, ensure deployment consistency, and streamline the process of launching and managing containerized services. It eliminates the complexity of manual Kubernetes resource orchestration while maintaining full control over your applications.

Please proceed to the following sections to learn more about managing templates and applications:

- [Kubernetes Templates](#)
- [Kubernetes Applications](#)

Template

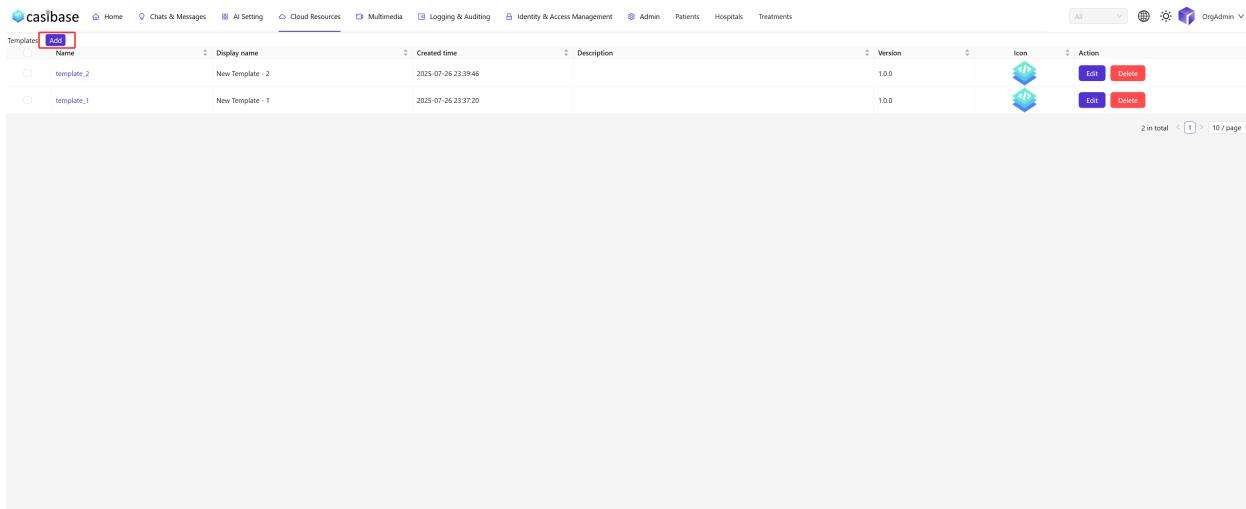
In Casibase, a **Template** is a reusable base configuration for an application. It contains the core Kubernetes manifest files, typically structured for use with Kustomize. You define a template once, and it can then be used as a blueprint to create multiple, customized application instances.

This model allows you to standardize your deployment patterns, ensuring consistency and simplifying the process of launching new services.

This chapter will guide you through creating and managing templates in Casibase.

Create a New Template

First, navigate to the **Cloud Resources > Templates** section and click the  button to open the creation page.



Name	Display name	Created time	Description	Version	Icon	Action
template_2	New Template - 2	2025-07-26 23:39:46		1.0.0		Edit Delete
template_1	New Template - 1	2025-07-26 23:37:20		1.0.0		Edit Delete

You will need to fill in the following fields, which correspond to the template's

properties:

- **Name**: A unique identifier for the template (e.g., `my-app-template`). This is a required field.
- **Display name**: A user-friendly name that will be shown in the UI (e.g., `My App Template`).
- **Description**: A brief description of what this template is for.
- **Version**: The version of the template (e.g., `1.0.0`).
- **Icon**: A URL to an icon image that represents the template in the UI.
- **Manifest**: The raw YAML text of your Kubernetes manifests. This content serves as the base for Kustomize deployments.

The screenshot shows the Casibase interface for editing a template. At the top, there's a navigation bar with links like Home, Chats & Messages, AI Setting, Cloud Resources, Multimedia, Logging & Auditing, Identity & Access Management, Admin, Patients, Hospitals, and Treatments. Below the navigation is a toolbar with 'Edit Template', 'Save', and 'Save & Exit' buttons. The main area has fields for Name (template_2), Display name (New Template - 2), Description, Version (1.0.0), and Icon (a URL to an icon image). Below these is a large text area for the Manifest, which contains the following YAML code:

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
spec:
  replicas: 1
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx
          ports:
            - containerPort: 80
```

After saving, your template will be available in the selection list when you create a new application.

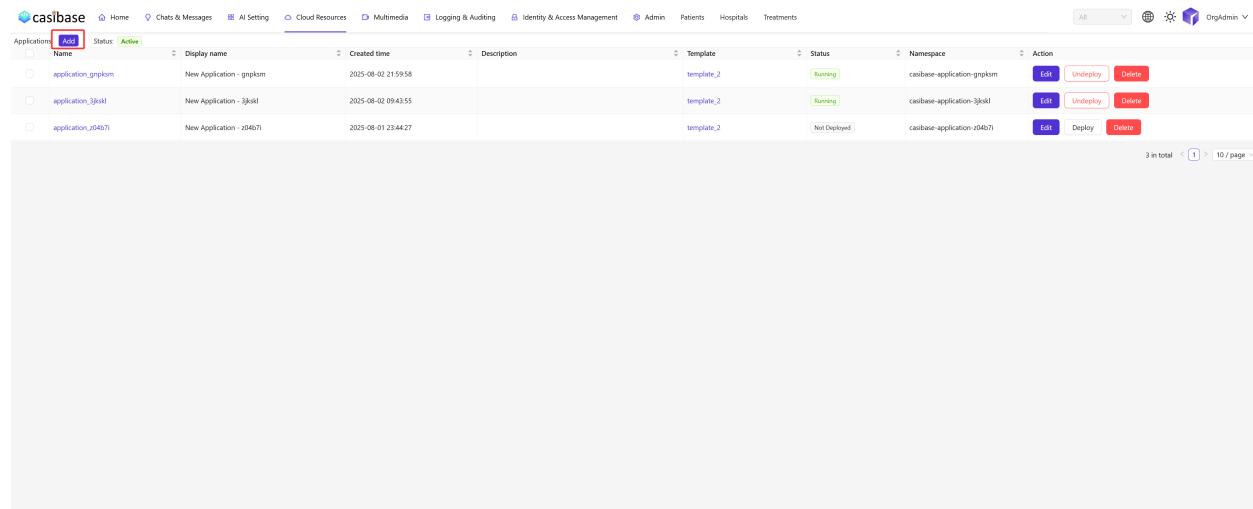
Application

An Application in Casibase is a specific, deployable instance created from a Template. Before you can create an application, you must first have at least one template defined. When you create an application, you select a base template and can then apply specific customizations before deploying it to your Kubernetes cluster.

This chapter will guide you through creating, deploying, and managing applications in Casibase.

Create a New Application

Navigate to the Cloud Resources > Applications section and click the [Add](#) button to open the creation page.



The screenshot shows the Casibase application management interface. At the top, there's a navigation bar with links like Home, Chats & Messages, AI Setting, Cloud Resources, Multimedia, Logging & Auditing, Identity & Access Management, Admin, Patients, Hospitals, and Treatments. Below the navigation is a search bar and a filter dropdown set to 'All'. The main area is titled 'Applications' and features a large 'Add' button with a plus sign. A table lists three existing applications:

Name	Display name	Created time	Description	Template	Status	Namespace	Action
application_grpkxm	New Application - grpkxm	2025-08-02 21:59:58		template_2	Running	casibase-application-grpkxm	Edit Undeploy Delete
application_3k4sl	New Application - 3k4sl	2025-08-02 09:43:55		template_2	Running	casibase-application-3k4sl	Edit Undeploy Delete
application_x04b7l	New Application - x04b7l	2025-08-01 23:44:27		template_2	Not Deployed	casibase-application-x04b7l	Edit Deploy Delete

At the bottom right of the table, it says '3 in total < 1 > 10 / page'.

The key fields for an application are:

- **Name**: A unique name for your application instance (e.g., `my-app-prod`). This is a required field.
- **Display name**: A user-friendly name that will be shown in the UI (e.g., `My App (Production)`).
- **Description**: A brief description of this specific application instance.
- **Template**: Select a pre-existing template from the dropdown list. This will be the base for your application.
- **Parameters**: This field is used for customization. Here you can provide specific Kustomize patches or other variable substitutions in YAML format to override or extend the base **Manifest** from the selected template.

Note: Fields like **Status** and **Namespace** are managed by the system. The **Namespace** is automatically generated based on the application name upon creation and cannot be modified by the user. The **Status** is updated based on its deployment state (e.g., `Not Deployed`, `Running`, `Pending`).

The screenshot shows the 'Edit Application' interface in the casibase platform. The 'Parameters' section is highlighted with a red box, containing the following YAML code:

```

apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
spec:
  replicas: 3

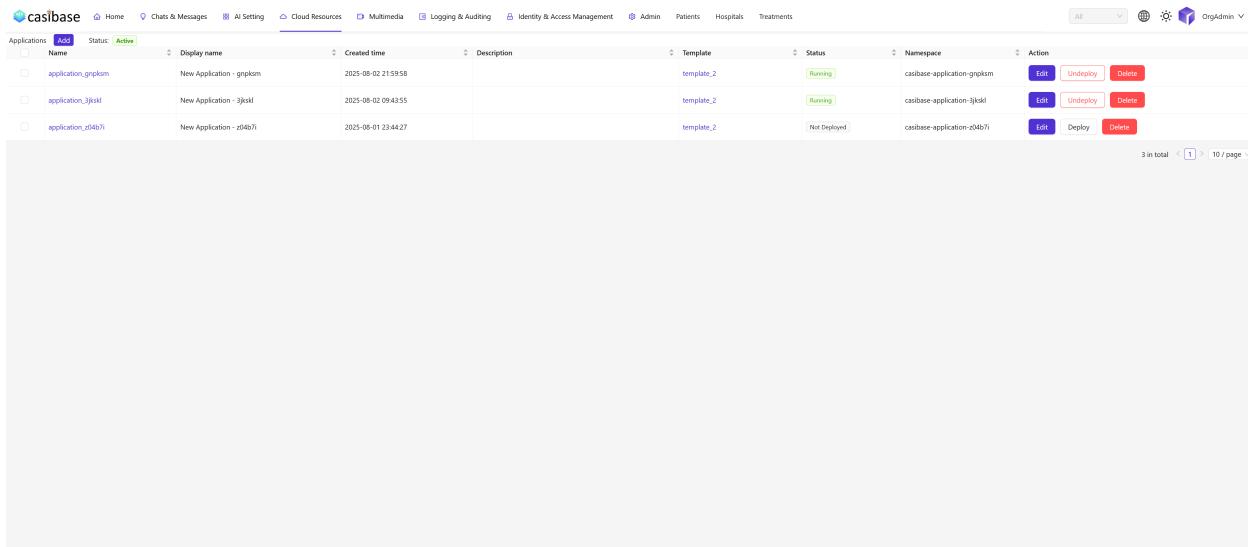
```

Deploying and Monitoring an Application

After creating an application, it will appear in the applications list. From here, you

can manage its lifecycle.

- **Deploy:** Click the `Deploy` button to apply the application's configuration to your Kubernetes cluster. Casibase will use Kustomize to merge the base template's `Manifest` with your application's `Parameters` and run `kubectl apply`.
- **Undeploy:** The `Undeploy` button will remove the application's resources from your Kubernetes cluster.



The screenshot shows the Casibase application management interface. At the top, there is a navigation bar with links like Home, Chats & Messages, AI Setting, Cloud Resources, Multimedia, Logging & Auditing, Identity & Access Management, Admin, Patients, Hospitals, and Treatments. On the far right, there are user profile icons and a dropdown for OrgAdmin.

The main area is a table titled "Applications". The table has columns: Name, Display name, Created time, Description, Template, Status, Namespace, and Action. There are three rows of data:

Name	Display name	Created time	Description	Template	Status	Namespace	Action
application_grpkm	New Application - grpkm	2025-08-02 21:59:58		template_2	Running	casibase-application-grpkm	<button>Edit</button> <button>Undeploy</button> <button>Delete</button>
application_3jekl	New Application - 3jekl	2025-08-02 09:43:55		template_2	Running	casibase-application-3jekl	<button>Edit</button> <button>Undeploy</button> <button>Delete</button>
application_z04b7i	New Application - z04b7i	2025-08-01 23:44:27		template_2	Not Deployed	casibase-application-z04b7i	<button>Edit</button> <button>Deploy</button> <button>Delete</button>

At the bottom right of the table, it says "3 in total" and has a page number "1" with a "10 / page" dropdown.

By using this template-and-application model, you can effectively standardize and scale your Kubernetes deployments through the Casibase interface.



Creating Databases with KubeBlocks

Overview

KubeBlocks is an open-source Kubernetes operator and toolset designed to simplify the complexity of running and managing data infrastructure, such as databases, message queues, and streaming systems, on Kubernetes. It provides a declarative approach to deploying and managing stateful applications, allowing you to manage them as easily as stateless ones.

This guide will walk you through the process of creating a database cluster using KubeBlocks.

Why KubeBlocks?

- **Simplified Management:** Automates the database lifecycle, including deployment, upgrades, scaling, and monitoring.
- **Production-Ready:** Supports high availability, backup and restore, and robust monitoring.
- **Versatile:** Supports a wide range of databases, including MySQL, PostgreSQL, MongoDB, Redis, and more.

Installing KubeBlocks

You can install KubeBlocks using [Helm](#). For more installation options, refer to the [KubeBlocks Installation Guide](#).

This guide will use Helm for the installation.

Step 1: Deploy Snapshot Controller

KubeBlocks requires the Snapshot Controller to manage volume snapshots. First, check if it is already installed in your cluster.

```
kubectl get crd volumesnapshotclasses.snapshot.storage.k8s.io  
kubectl get crd volumesnapshots.snapshot.storage.k8s.io  
kubectl get crd volumesnapshotcontents.snapshot.storage.k8s.io
```

If it is not installed, you can deploy it using the following commands:

```
helm repo add piraeus-charts https://piraeus.io/helm-charts/  
helm repo update  
# Update the namespace to an appropriate value for your  
environment (e.g. kb-system)  
helm install snapshot-controller piraeus-charts/snapshot-  
controller -n kb-system --create-namespace
```

Then, verify the installation:

```
kubectl get pods -n kb-system | grep snapshot-controller
```

The Snapshot Controller should be in the [Running](#) state.

Step 2: Get the Latest KubeBlocks Version

Get the latest stable version tag (e.g. v1.0.1):

```
curl -s "https://api.github.com/repos/apecloud/kubeblocks/releases?per_page=100&page=1" | jq -r '.[] | select(.prerelease == false) | .tag_name' | sort -V -r | head -n 1

# Example output:
# v1.0.1
```

Step 3: Create KubeBlocks CRDs

Create the Custom Resource Definitions (CRDs) required by KubeBlocks.

```
# Replace <VERSION> with the version you selected
kubectl create -f https://github.com/apecloud/kubeblocks/releases/download/<VERSION>/kubeblocks_crds.yaml

# Example: If the version is v1.0.1
kubectl create -f https://github.com/apecloud/kubeblocks/releases/download/v1.0.1/kubeblocks_crds.yaml
```

Step 4: Install KubeBlocks with Helm

1. Add the KubeBlocks Helm repository:

```
helm repo add kubeblocks https://apecloud.github.io/helm-charts
helm repo update
```

2. Install KubeBlocks: This command installs the KubeBlocks chart into the kb-

system namespace.

```
helm install kubeblocks kubeblocks/kubeblocks --namespace kb-system --create-namespace
```

Creating a Database

Once KubeBlocks is installed, you can create a database cluster using `kubectl`.

Currently supported databases include:

- MySQL
- PostgreSQL
- MongoDB
- Redis
- Kafka
- Milvus
- Qdrant
- RabbitMQ
- Elasticsearch

Example: Creating a Demo MySQL Cluster

1. Create a file named `my-mysql-cluster.yaml`:

```
apiVersion: apps.kubeblocks.io/v1
kind: Cluster
metadata:
```

2. Apply the manifest to create the cluster:

```
kubectl apply -f my-mysql-cluster.yaml
```

More Information

For more detailed information, advanced configurations, and troubleshooting with KubeBlocks, please refer to the [official KubeBlocks documentation](#).



>

节点

节点

概述

Casibase节点概述

远程桌面协议

Casibase节点RDP

VNC

Casibase节点VNC

概述

Casibase帮助您管理节点，并通过RDP、VNC、SSH和Telnet远程连接到您的节点。

协议：

- 安全外壳协议
- 远程桌面协议
- 虚拟网络计算
- 远程终端协议

每个节点都具有以下基本属性：

- 组织：节点所属的组织。
- 名称：唯一的节点名称。
- 描述：节点的描述。
- IP：域名或IP地址。
- 协议：协议的端口号。
- 端口：节点的端口号。
- 用户名：连接到节点的用户名，如root、administrator、sa等。
- 密码：连接到节点的密码。
- 操作系统：节点的操作系统，包括Windows和Linux，用于分类节点。
- 标签：节点的标签，用于分类节点。

在本章中，您将学习如何开始连接到您的节点。

让我们一起探索！

远程桌面协议

Casibase支持通过RDP协议连接到您的节点:

RDP连接

1. 启动Guacamole服务器

```
docker run --name guacd -d -p 4822:4822 guacamole/guacd
```

2. 添加一个新节点, 将协议设置为 rdp

Organization	Name	Created time	Description	Protocol	IP	Port	Username	Language	Auto query	Is perm	Action		
casbin	node_eqiwer	2025-03-09 23:37:34		VNC	127.0.0.1	5900	Administrator	en	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<button>Connect</button>	<button>Edit</button>	<button>Delete</button>
casbin	node_apacdj	2025-03-09 23:32:12		VNC	127.0.0.1	5900	Administrator	en	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<button>Connect</button>	<button>Edit</button>	<button>Delete</button>
casbin	node_qf773r	2025-02-25 11:12:14+03:30		RDP	127.0.0.1	3389	Administrator	en	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<button>Connect</button>	<button>Edit</button>	<button>Delete</button>
casbin	node_zbj7av	2025-02-21 17:18:08		RDP	127.0.0.1	3389	Administrator	en	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<button>Connect</button>	<button>Edit</button>	<button>Delete</button>
casbin	node_cy3c9s	2025-02-14 11:59:43		RDP	127.0.0.1	3389	Administrator	en	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<button>Connect</button>	<button>Edit</button>	<button>Delete</button>

The screenshot shows the 'Edit Node' form for a 'host-base' node. The fields include:

- Organization: casbin
- Name: host-base
- Description: 21212
- Protocol: RDP
- IP: 47.93.49.234
- Port: 3389
- Username: administrator
- Password: ***
- OS: Windows
- Tag: (empty)
- Language: en
- Auto query: On
- Is permanent: On

Services section:

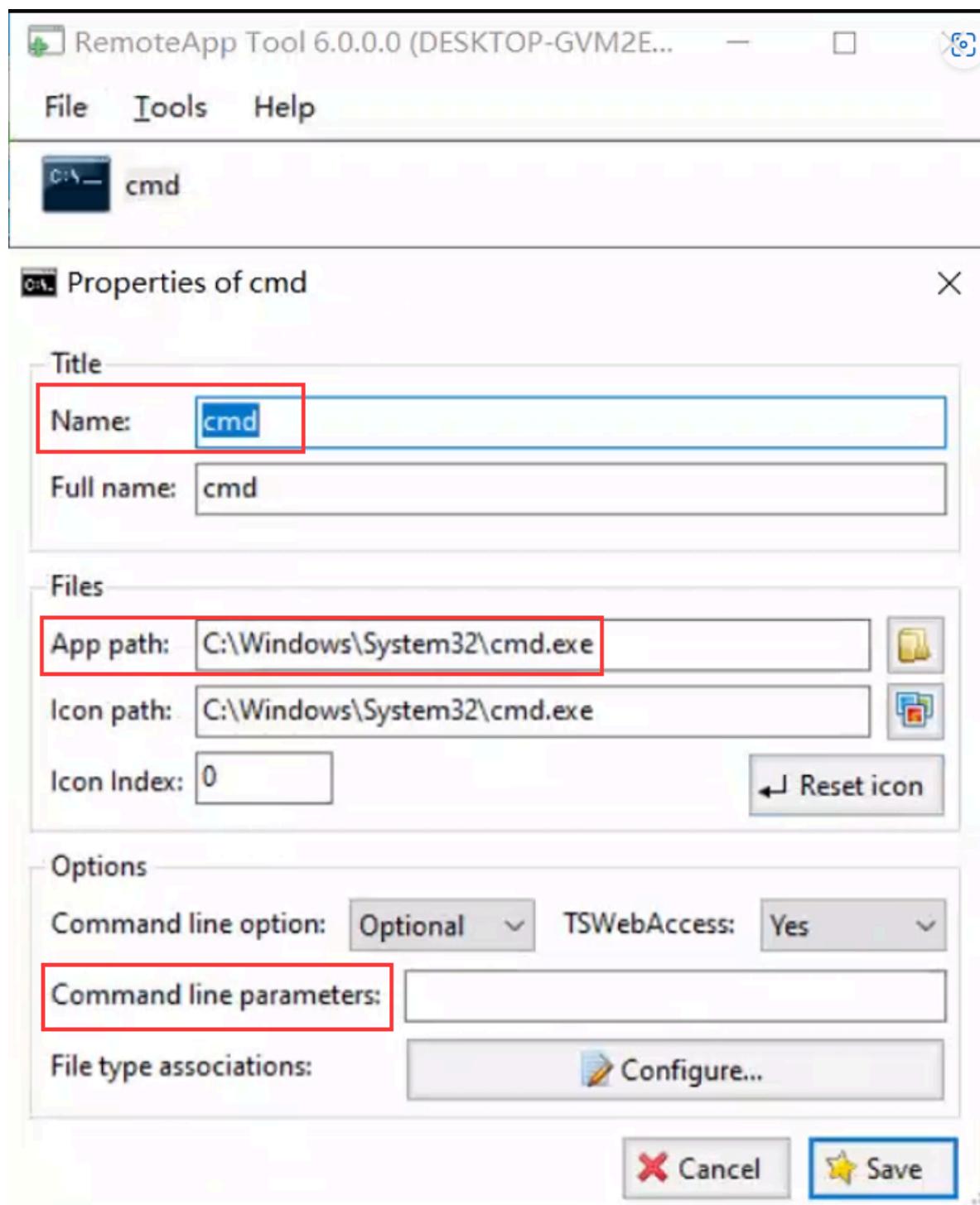
No.	Name	Path	Port	Process ID	Expected status	Status	Message	Action

3. 点击 **连接** 按钮连接到您的节点

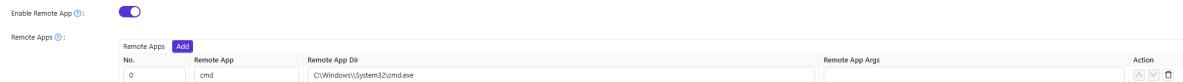
远程应用程序

我们支持在Windows节点上使用远程应用程序，您可以在**节点编辑**页面添加远程应用程序，然后通过点击**连接**按钮连接到您的远程应用程序。

1. 在服务器端配置您的远程应用程序。\\您可以使用 [RemoteApp工具](#) 注册应用。



2. 根据服务器端配置，在节点编辑页面中配置远程应用信息。“remoteAppName”、“remoteAppDir”和“remoteAppArgs”是必需的。



refer to [Configuring Guacamole — Apache Guacamole Manual v1.5.3](#)

3. 连接到您的远程应用程序。

VNC

VNC连接

VNC连接与RDP连接类似。

1. 启动Guacamole服务器

```
docker run --name guacd -d -p 4822:4822 guacamole/guacd
```

2. 添加一个新节点，将协议设置为vnc

Organization	Name	Created time	Description	Protocol	IP	Port	Username	Language	Auto query	Is perm	Action
casbin	node_eqiwer	2025-03-09 23:37:34		VNC	127.0.0.1	5900	Administrator	en	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<button>Connect</button>
casbin	node_apacdj	2025-03-09 23:32:12		VNC	127.0.0.1	5900	Administrator	en	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<button>Connect</button>
casbin	node_qf773r	2025-02-25 11:12:14+03:30		RDP	127.0.0.1	3389	Administrator	en	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<button>Connect</button>
casbin	node_zbj7av	2025-02-21 17:18:08		RDP	127.0.0.1	3389	Administrator	en	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<button>Connect</button>
casbin	node_cy3c9s	2025-02-14 11:59:43		RDP	127.0.0.1	3389	Administrator	en	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<button>Connect</button>

3. 点击connect按钮连接到您的节点。