

快速开始

服务器端的安装与配置

MySQL

你需要安装 MySQL 数据库，测试使用的数据库是 MySQL 8.0。

你需要新建一个数据库，并给予用户访问这个数据库的权限，包括 `SELECT`、`INSERT`、`UPDATE`、`DELETE` 权限。

非常不建议使用 `root` 用户来访问数据库。

非常建议使用 `utf8mb4` 字符集。

需要创建以下的表

```
CREATE TABLE `files` (  
  `id` int NOT NULL,  
  `size` int NOT NULL,  
  `sha256` varchar(64) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;  
CREATE TABLE `tokens` (  
  `token` varchar(64) NOT NULL,  
  `time_accessed` bigint NOT NULL,  
  `UNIQUE_ID` tinytext NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;  
CREATE TABLE `users` (  
  `username` varchar(32) NOT NULL,  
  `password` varchar(64) NOT NULL,  
  `salt` varchar(32) NOT NULL,  
  `id` int NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;  
CREATE TABLE `user_files` (  
  `UNIQUE_ID` varchar(32) NOT NULL,  
  `fileid` int NOT NULL,  
  `name` text NOT NULL,  
  `seq` int NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;  
CREATE TABLE `user_projects` (  
  `UNIQUE_ID` varchar(32) NOT NULL,  
  `project_id` int NOT NULL,  
  `project_name` text NOT NULL,  
  `paragraphs` mediumtext NOT NULL,  
  `deleted` tinyint(1) NOT NULL DEFAULT '0'  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;  
CREATE TABLE `user_settings` (  
  `UNIQUE_ID` varchar(32) NOT NULL,  
  `setting` varchar(32) NOT NULL,
```

```
`value` text NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
```

需要创建以下索引:

```
ALTER TABLE `files`  
  ADD PRIMARY KEY (`id`);  
ALTER TABLE `tokens`  
  ADD UNIQUE KEY `token` (`token`);  
ALTER TABLE `users`  
  ADD PRIMARY KEY (`id`),  
  ADD UNIQUE KEY `username` (`username`);  
ALTER TABLE `user_files`  
  ADD PRIMARY KEY (`seq`),  
  ADD KEY `UNIQUE_ID` (`UNIQUE_ID`);  
ALTER TABLE `user_projects`  
  ADD PRIMARY KEY (`project_id`);  
ALTER TABLE `user_settings`  
  ADD PRIMARY KEY (`UNIQUE_ID`,`setting`);
```

设置自增属性:

```
ALTER TABLE `files`  
  MODIFY `id` int NOT NULL AUTO_INCREMENT;  
ALTER TABLE `users`  
  MODIFY `id` int NOT NULL AUTO_INCREMENT;  
ALTER TABLE `user_files`  
  MODIFY `seq` int NOT NULL AUTO_INCREMENT;  
ALTER TABLE `user_projects`  
  MODIFY `project_id` int NOT NULL AUTO_INCREMENT;  
COMMIT;
```

Python

你需要安装 Python 3.11+, 并建议使用 `venv` 创建一个虚拟环境。

然后, 你需要确保安装在 `requirements.txt` 中的所有依赖。

```
pip install -r requirements.txt
```

你需要修改几个文件

`backend/dbpassword.py`, 其模板是 `backend/dbpassword.py.default`, 你需要将其复制一份, 并修改其中的数据库连接信息。

`algo/keys.py`, 其模板是 `algo/keys.py.default`, 你需要将其复制一份, 并修改其中的向量化 API 密钥信息。

此外, 如果你不涉及使用 ssl, 需要修改 `backend/server.py` 与 `algo/server.py`, 删除 `ssl_keyfile=` 与 `ssl_certfile=` 两个参数。

注意, 如果不使用 SSL, 你还需要修改 `src/components/endpoints.js` 中的 `https://` 为 `http://`。

如果你需要使用 ssl, 需要在上述两个文件中, 设置 `ssl_keyfile=` 与 `ssl_certfile=` 的值为你自己的证书文件。

启动服务器

对于 Windows, 我们提供了一个 `run.bat` 文件, 你或许需要修改其中的 python 路径。

对于 Linux 和 macOS, 你需要手动运行 `backend/server.py` 和 `algo/server.py`。

前端运行

这非常简单, 你只需要安装 NodeJS 18+, 然后运行以下命令:

```
npm install
```

要运行, 你只需要运行以下命令:

```
npm run serve
```

要构建, 你只需要运行以下命令:

```
npm run build
```