

freetodo

找不到“.github/assets/free_todo_banner.png”。

 Stars 1.9k  Forks 116 issues 6 open license FreeU Community python 3.12 FastAPI 0.100+

语言: [English](#) | [中文](#)

 文档 ·  快速开始 ·  功能特性 ·  开发指南 ·  贡献指南

FreeTodo - 放手去做

项目概述

FreeTodo 是一款 AI 驱动的智能待办管理应用，帮助您高效管理任务、提升生产力、达成目标。通过对话式 AI 交互和智能任务拆分，FreeTodo 将复杂项目转化为可执行的行动步骤。

核心功能

AI 智能助手

- **智能任务拆分**: AI 自动将复杂任务分解为可管理的子任务，通过引导式问卷流程完成
- **智能任务提取**: 从 AI 对话响应中提取可执行的待办事项
- **上下文感知建议**: AI 根据当前待办上下文提供任务建议

全面的任务管理

- **层级任务结构**: 支持父子任务关系，无限层级嵌套
- **优先级与状态**: 四级优先级（紧急/高/中/低）和多种状态
- **标签与分类**: 使用自定义标签组织待办，便于筛选
- **截止日期管理**: 设置截止日期，可视化提醒
- **丰富备注**: 为每个待办添加详细备注和描述

多视图日历

- **日/周/月视图**: 灵活的日历视图，可视化您的日程安排
- **拖拽排期**: 轻松拖拽待办到日历时间槽进行排期
- **快速创建待办**: 直接从日历时间槽创建待办

现代化用户界面

- **多面板布局**: 可自定义的面板排列（待办 + 聊天 + 详情）

- **深色/浅色主题**: 精美主题，多种配色方案
- **国际化支持**: 完整支持中英文
- **响应式设计**: 适配各种屏幕尺寸

桌面应用

- **Electron 应用**: Windows 和 macOS 原生桌面体验
- **系统集成**: 原生通知和系统托盘支持

系统架构

FreeTodo 采用前后端分离架构：

- **后端**: FastAPI (Python) - 提供 RESTful API (位于 `lifetrace/` 目录)
- **前端**: Next.js (React + TypeScript) - 现代化 Web 界面 (位于 `free-todo-frontend/` 目录)
- **数据层**: SQLite + ChromaDB (用于 AI 功能)

快速开始

环境要求

后端:

- Python 3.12
- 支持的操作系统: Windows、macOS
- 可选: CUDA 支持 (用于 GPU 加速)

前端:

- Node.js 20+
- pnpm 包管理器

安装依赖

本项目使用 [uv](#) 进行快速可靠的依赖管理。

安装 uv:

```
# macOS/Linux
curl -LsF https://astral.sh/uv/install.sh | sh

# Windows
powershell -c "irm https://astral.sh/uv/install.ps1 | iex"
```

注意：安装完成后，`uv` 可能无法在当前终端中立即使用。要在当前会话中激活它：

- **Windows (PowerShell)**: 运行 `$env:Path = "$env:USERPROFILE\.local\bin;$env:Path"` 来刷新 PATH
- **macOS/Linux**: 运行 `exec $SHELL` 来重新初始化 shell 会话，或重新打开终端

或者，您也可以直接打开一个新的终端窗口，`uv` 将自动可用。

安装依赖并同步环境:

```
# 从 pyproject.toml 和 uv.lock 同步依赖
uv sync

# 激活虚拟环境
# macOS/Linux
source .venv/bin/activate

# Windows
.venv\Scripts\activate
```

启动后端服务

注意：首次运行时，如果 `config.yaml` 不存在，系统会自动从 `default_config.yaml` 创建。您可以通过编辑 `lifetrace/config/config.yaml` 来自定义设置。

启动服务器：

```
python -m lifetrace.server
```

自定义提示词：如果您想修改不同功能的 AI 提示词，可以编辑 `lifetrace/config/prompt.yaml` 文件。

后端服务会自动从 `8001` 端口（构建版为 `8100`）开始查找可用端口。如果默认端口被占用，会自动使用下一个可用端口，并在控制台显示实际使用的端口。

- **默认后端端口**: `http://localhost:8001`
- **API 文档**: 实际 API 文档地址会在控制台显示（通常为 `http://localhost:8001/docs`）

启动前端服务

前端是使用 FreeTodo 的必需组件。启动前端开发服务器：

```
cd free-todo-frontend
```

```
pnpm install  
pnpm dev
```

前端开发服务器会：

- 自动从 3001 端口（开发版默认端口）开始查找可用端口
- 通过检查 /health 端点自动检测运行中的 FreeTodo 后端端口
- 自动设置 API 代理指向检测到的后端端口

实际的前端地址和后端连接状态会在控制台显示。服务启动后，在浏览器中访问控制台显示的前端地址（通常为 `http://localhost:3001`）开始使用 FreeTodo! 🎉

注意：如果端口被占用，前端和后端都会自动查找下一个可用端口。控制台会显示实际使用的端口。

📋 待办事项与路线图

📘 完整路线图：查看详细的 [项目路线图](#) 了解 FreeU 项目的完整愿景和发展规划。

🎯 FreeU 整体项目路线图

1. LifeTrace (v0.2 已完成)

- ✓ 电脑活动流构建：通过截图生成个人活动流
- ⚡ 未来规划：音频获取、视频环境、智能设备集成、本地大模型优化

2. Free Todo (v0.1 当前进行中)

- 📈 当前聚焦：打造极致的 To-Do List
- 🎯 核心使命：固定用户意图、形成个人上下文整理，为主动服务打下基础

3. 主动服务阶段（未来规划）

- 基于 LifeTrace 数据和 Free Todo 意图提供主动服务

🚧 Free Todo 近期计划（专注输入层）

目标：尽可能从用户生活中获取各种各样的信息并收集为 Todo

- 🎨 UI 灵动岛
 - □ 控制语音输入和截图定时任务开关
 - □ 提供便捷窗口访问 Todo 列表和对话界面

-  **Agent 开发**
 -  开发 AI 工具调度能力
 - 从基础对话升级为支持多工具调用的智能 Agent
-

Free Todo 三层次路线图

输入层：减轻输入负担，意念流般的捕获

- 语音输入（灵动岛、快捷键呼出）
- 多模态输入（文字、截图、语音）
- 社交软件集成（微信、飞书等 todo 捕获）
- 智能消息 todo 提取

中间处理层：从“混沌”到“秩序”

- AI 任务拆分（“大石头”变“小石子”）
- AI 意图补全 / 任务详情补全
- 自动分类与组织
- 任务优先级智能规划
- Todo 上下文构建

输出层：心理安全感 + 温暖可靠的秘书伙伴

- AI 秘书人格化
 - 日程提醒（目前正在做）
 - 任务专注模式（只显示部分任务）
 - 已完成任务强化（功劳簿化）
 - 逾期任务重新规划
-

开发中功能

Free Todo 的面板开关栏里有一些正在开发中的面板，这些面板展示了我们未来功能的方向，供社区参考和了解。

社区参与：我们非常欢迎社区成员参与进来！

-  **面板贡献：** 贡献自己的面板设计或提出改进建议
-  **Agent 算法贡献：** 贡献新的 Agent 算法，我们积极合入！

✓ 最近完成

- **AI 任务拆分** - 通过问卷流程实现智能任务分解
- **多面板界面** - 可自定义面板的灵活布局
- **日历集成** - 支持拖拽的日/周/月视图



想要贡献？查看我们的[贡献指南](#)并选择任何你感兴趣的待办事项！

开发指南

项目结构

```
└── .github/          # GitHub 仓库资源
    ├── assets/        # 静态资源 (README 图片)
    ├── BACKEND_GUIDELINES.md # 后端开发规范
    ├── FRONTEND_GUIDELINES.md # 前端开发规范
    ├── CONTRIBUTING.md    # 贡献指南
    └── ...
└── lifetrace/        # 后端模块 (FastAPI)
    ├── server.py      # Web API 服务入口
    ├── config/         # 配置文件
        ├── config.yaml # 主配置文件 (自动生成)
        ├── default_config.yaml # 默认配置模板
        ├── prompt.yaml # AI 提示词模板
        └── rapidocr_config.yaml # OCR 配置
    ├── routers/        # API 路由处理器
        ├── chat.py       # 聊天接口端点
        ├── todo.py       # 待办事项端点
        ├── task.py       # 任务管理端点
        └── ...
    ├── schemas/        # Pydantic 数据模型
    ├── services/        # 业务逻辑服务层
    ├── repositories/   # 数据访问层
    ├── storage/         # 数据存储层
    ├── llm/             # LLM 和 AI 服务
    ├── jobs/            # 后台任务
    ├── util/            # 工具函数
    └── data/            # 运行时数据 (自动生成)
        ├── lifetrace.db # SQLite 数据库
        ├── vector_db/   # 向量数据库存储
        └── logs/          # 应用日志
```

```
free-todo-frontend/          # 前端应用 (Next.js) ★
  app/                      # Next.js 应用目录
  apps/
    todo-list/               # 待办列表模块
    todo-detail/             # 待办详情模块
    chat/                    # AI 聊天模块
    calendar/                # 日历模块
    settings/                # 设置模块
    ...
  components/               # React 组件
  lib/                      # 工具和服务
  electron/                 # Electron 桌面应用
  package.json              # 前端依赖
  README.md                 # 前端文档
  pyproject.toml            # Python 项目配置
  uv.lock                   # uv 锁定文件
  LICENSE                   # FreeU Community License 许可证
  README.md                 # 英文 README
  README_CN.md              # 中文 README (本文件)
```

贡献

FreeTodo 社区的存在离不开像您这样的众多友善志愿者。我们欢迎所有对社区的贡献，并很高兴欢迎您的加入。

最近的贡献：

contributors 11 commit activity 119/month last commit january

贡献指南

我们提供了完整的贡献指南帮助您开始：

- [贡献指南](#) - 完整的贡献流程和规范
- [后端开发规范](#) - Python/FastAPI 编码规范
- [前端开发规范](#) - TypeScript/React 编码规范

快速开始贡献

1.  **Fork 项目** - 创建您自己的仓库副本
2.  **创建功能分支** - git checkout -b feature/amazing-feature
3.  **提交您的更改** - git commit -m 'feat: 添加某个很棒的功能'
4.  **推送到分支** - git push origin feature/amazing-feature
5.  **创建 Pull Request** - 提交您的更改以供审核

您可以贡献的领域

-  **错误报告** - 帮助我们识别和修复问题
 -  **功能请求** - 建议新功能
 -  **文档** - 改进指南和教程
 -  **测试** - 编写测试并提高覆盖率
 -  **UI/UX** - 增强用户界面
 -  **代码** - 实现新功能和改进

开始贡献

- 查看我们的 [贡献指南](#) 了解详细说明
 - 寻找标记为 `good first issue` 或 `help wanted` 的问题
 - 后端开发请遵循 [后端开发规范](#)
 - 前端开发请遵循 [前端开发规范](#)
 - 在 Issues 和 Pull Requests 中加入我们的社区讨论

我们感谢所有贡献，无论大小！ 

加入我们的社区

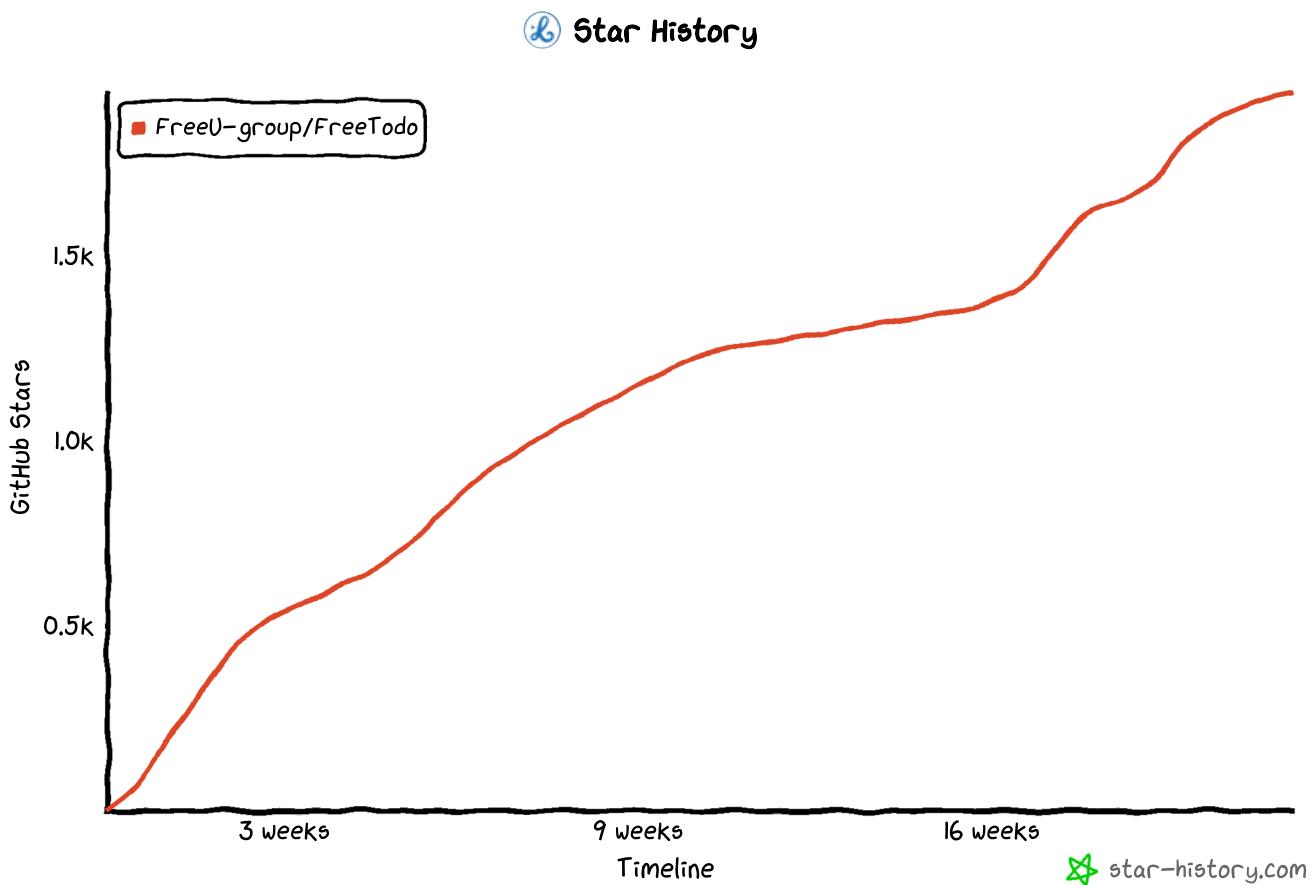
与我们和其他 FreeTodo 用户联系！扫描下方二维码加入我们的社区群组：

微信群	飞书群	小红书
 微信二维码	 飞书二维码	 小红书二维码

文档

我们使用 deepwiki 管理文档, 请参考此[网站](#)。

Star 历史



许可证

版权所有 © 2026 FreeU.org

FreeTodo 采用 **FreeU Community License** 许可证, 该许可证基于 Apache License 2.0, 并附加了关于商业使用的条件。

有关详细的许可证条款, 请参阅 [LICENSE](#) 文件。