

HBAI-Ltd / Toonflow-app Public

<> Code Issues Pull requests 1 Actions Projects Security and quality Insights

master 2 Branches 19 Tags Go to file Go to file <> Code

ACT、流星雨 更新Atlas Cloud描述	055e0c6 · last week
.github/workflows	提升健壮性 last week
data	修正滚动问题 last week
docs	更新Atlas Cloud描述 last week
scripts	更新minimax与火山引擎代码 2 weeks ago
src	修正供应商获取代码 last week
.dockerignore	no message 2 months ago
.gitignore	修正基底打包规则，修正版本更新方式、修... 3 weeks ago
Dockerfile	完善docker last month
LICENSE	完善协议 last month
NOTICES.txt	矫正出图方法 2 weeks ago
README.md	更新Atlas Cloud描述 last week
electron-builder.yml	修正打包脚本兼容arm 2 weeks ago
nodemon.json	新增厂商2.0，染上深度思考模块，修正厂... 2 weeks ago
package.json	发版前准备 last week
tsconfig.json	no message 2 weeks ago
yarn.lock	添加默认添加数据 2 weeks ago

README Apache-2.0 license

GitHub | Gitee | AtomGit

简体中文 | 繁體中文 | English | ไทย | Tiếng Việt | 日本語 | Русский



# Toonflow

AI短剧工厂  
动动手指，小说秒变剧集！  
AI剧本 × AI影像 × 极速生成 🔥

STARS 7.4K LICENSE APACHE 2.0 RELEASE V1.1.6

 一站式短剧工程：从文本到角色，从分镜到视频，0门槛全流程AI化，创作效率提升10倍+！

## 多语言支持

Toonflow 支持以下语言界面：

语言	Language
简体中文	Chinese (Simplified)
繁體中文	Chinese (Traditional)
English	English
ไทย	Thai
Tiếng Việt	Vietnamese
日本語	Japanese
Русский	Russian

 更多语言适配中，欢迎贡献翻译！

## 主要功能

Toonflow 是面向短剧生产的 AI 工作台，围绕“策划 → 编剧 → 分镜 → 出片”构建完整闭环，并支持本地化、可编程、可持续迭代的生产流程。

- ✓ **无限画布生产工作台**  
以类无限画布形式组织剧本、角色、分镜、素材与视频节点，支持自由编排、回溯与并行生产，不受线性步骤限制。
- ✓ **三层 Agent 协作体系**  
决策层、执行层、监督层协同工作，覆盖任务拆解、内容生成、质量审阅与修订反馈，提升稳定性与成片一致性。
- ✓ **持久化 Agent 记忆**  
基于本地 ONNX 向量检索的跨会话记忆系统，支持短期消息、长期摘要和语义召回，确保多轮创作连续性。
- ✓ **可编程供应商系统**  
支持在设置中心直接编写供应商 TypeScript 逻辑并即时生效，无需改源码或重启，便于私有化和多模型接入。
- ✓ **章节事件图谱驱动改编**  
自动提取原著章节事件并结构化存储，剧本改编按事件图谱精准调用上下文，减少长文本信息丢失。
- ✓ **Skill 文件化配置**  
ScriptAgent 与 ProductionAgent 的核心提示词外化为 Markdown Skill 文件，支持在线编辑与快速调优。

## 应用场景

- 短视频内容创作
- 小说影视化实验
- AI 文学改编工具
- 剧本开发与快速原型
- 视频素材生成

## 使用指南

### v1.0.8 快速上手

1. 启动应用并登录（默认账号：`admin / admin123`）。

2. 在设置中心完成模型供应商配置（文本/图像/视频模型）。
3. 新建项目并导入原著，执行章节事件提取。
4. 进入 ScriptAgent 生成故事骨架、改编策略与结构化剧本。
5. 切换到 ProductionAgent，在无限画布中组织分镜、素材与视频节点。
6. 对分镜图进行节点化精调后回流工作台，完成视频拼接与导出。

## 视频教程

<https://www.bilibili.com/video/BV1oXD7BqEqJ>



Toonflow 12 分钟快速上手 AI 视频 [👉 点击观看](#)

 手机微信扫码观看



## 安装

### 前置条件

在安装和使用本软件之前，请准备以下内容：

- 大语言模型 AI 服务接口地址
- Sora 或豆包视频服务接口地址
- Nano Banana Pro 图片生成模型服务接口

## 本机安装

### 1. 下载与安装

操作系统	GitHub	Atomgit	夸克网盘下载	说明
Windows	<a href="#">Release</a>	<a href="#">Release</a>	<a href="#">夸克网盘</a>	官方发布安装包
Linux	<a href="#">Release</a>	<a href="#">Release</a>	<a href="#">夸克网盘</a>	官方发布安装包
macOS	<a href="#">Release</a>	<a href="#">Release</a>	<a href="#">夸克网盘</a>	官方发布安装包

#### ⚠ Caution

MacOS 系统请到 [设置-隐私与安全性](#) 配置安全性否则可能因证书问题无法正常打开

参考知乎文档：<https://www.zhihu.com/question/433389276>

因 Gitee OS 环境限制及 Release 文件上传大小限制，暂不提供 Gitee Release 下载地址。

### 2. 启动服务

安装完成后，启动程序即可开始使用本服务。

#### ⚠ 首次登录

账号：`admin`

密码：`admin123`

## Docker 部署

### 前置条件

- 已安装 [Docker](#) (版本 20.10+)

### 方式一：在线部署

待完善，暂时使用本地构建。

### 方式二：本地构建

使用本地已有的源码直接构建，适合开发者或已克隆仓库的用户，这需要你在本地安装 git：

```
# 先克隆项目 (如已有则跳过)
git clone https://github.com/HBAI-Ltd/Toonflow-app.git
cd Toonflow-app

# 使用 docker-compose 本地构建并启动
yarn docker:local

# 或者手动构建
docker build -t toonflow .
docker run -d -p <本地端口>:10588 -v <本地数据路径>:/app/data toonflow

# 此时在相应端口的 /web/index.html 路径即可访问页面
# 例如 http://localhost:10588/web/index.html
```

### 服务端口说明

端口	用途	部署映射
10588	软件界面	10588:10588

环境变量说明：

变量	说明
NODE_ENV	运行环境，prod 表示生产环境
PORT	服务监听端口（默认 10588）
OSSURL	文件存储访问地址，用于静态资源访问

## 云端部署

### 一、服务器环境要求

- 系统：Ubuntu 20.04+ / CentOS 7+
- Node.js：24.x（推荐，最低 23.11.1+）
- 内存：2GB+

### 二、服务器部署

#### 1. 安装环境

```
# 安装 Node.js
curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.39.7/install.sh | bash
source ~/.bashrc
nvm install 24
# 安装 Yarn 和 PM2
npm install -g yarn pm2
```

#### 2. 部署项目

从 GitHub 克隆：

```
cd /opt
git clone https://github.com/HBAI-Ltd/Toonflow-app.git
cd Toonflow-app
yarn install
yarn build
```

从 Gitee 克隆（国内推荐）：

```
cd /opt
git clone https://gitee.com/HBAI-Ltd/Toonflow-app.git
cd Toonflow-app
yarn install
yarn build
```

#### 3. 配置 PM2

创建 pm2.json 文件：

```
{
  "name": "toonflow-app",
  "script": "data/serve/app.js",
  "instances": "max",
  "exec_mode": "cluster",
  "env": {
    "NODE_ENV": "prod",
    "PORT": 10588,
    "OSSURL": "http://127.0.0.1:10588/"
  }
}
```

**环境变量说明：**

变量	说明
NODE_ENV	运行环境，prod 表示生产环境
PORT	服务监听端口
OSSURL	文件存储访问地址，用于静态资源访问

**4. 启动服务**

```
pm2 start pm2.json
pm2 startup
pm2 save
```

**5. 常用命令**

```
pm2 list          # 查看进程
pm2 logs toonflow-app # 查看日志
pm2 restart all   # 重启服务
pm2 monit         # 监控面板
```

**⚠️ 首次登录**

账号：admin

密码：admin123

**6. 部署前端网站**

如需单独部署或定制前端界面，请参考前端仓库：

- **GitHub** : [Toonflow-web](#)
- **Gitee** : [Toonflow-web](#)

💡 说明：本仓库已内置编译好的前端资源，普通用户无需单独部署前端。前端仓库仅供需要二次开发的开发者使用。

**🔧 开发流程指南****⚠️ Caution****🚧 PR 提交规范 🚧**

🚫 master 分支不接受任何 PR | ✅ 请将 PR 提交到 develop 分支

欢迎开发者们共同参与 Toonflow 的共创。如有兴趣加入，请在交流群内联系主理人 ACT

**🔧 技术栈**

类别	技术
运行时	Node.js 23.11.1+
语言	TypeScript 5.x
后端框架	Express 5
数据库	SQLite (better-sqlite3 / knex)
AI 集成	Vercel AI SDK (OpenAI / Anthropic / Google / DeepSeek / 智谱 / MiniMax / 通义千问 / xAI)
本地推理	@huggingface/transformers (ONNX)

类别	技术
实时通信	Socket.IO
桌面客户端	Electron 40
图像处理	Sharp
容器化	Docker

## 开发环境准备

- **Node.js** : 版本要求 23.11.1 及以上
- **Yarn** : 推荐作为项目包管理器

## 快速启动项目

### 1. 克隆项目

从 **GitHub** 克隆：

```
git clone https://github.com/HBAI-Ltd/Toonflow-app.git
cd Toonflow-app
```



从 **Gitee** 克隆（国内推荐）：

```
git clone https://gitee.com/HBAI-Ltd/Toonflow-app.git
cd Toonflow-app
```



### 2. 安装依赖

请先在项目根目录下执行以下命令以安装依赖项：

```
yarn install
```



### 3. 启动开发环境

本项目包含 **后端 API 服务** 和 **前端页面** 两部分，请根据需要选择启动方式：

- **方式一：仅启动后端服务**

```
yarn dev
```



**⚠** 此命令仅启动后端 API 服务（端口 10588），**不包含前端页面**。直接访问 `http://localhost:10588` 只能调用 API 接口，无法看到完整的网页界面。如需同时使用前端页面，请配合前端项目单独启动，或使用下方的 GUI 模式。

- **方式二：启动 Electron 桌面客户端**

```
yarn dev:gui
```



此命令会同时启动后端服务和 Electron 桌面窗口，自带内置前端页面，开箱即用，无需额外配置。适合想要完整体验所有功能的开发者。

- **方式三：生产模式启动**

```
yarn start
```



以生产模式直接运行编译后的服务（需先执行 `yarn build`）。

### 4. 项目打包

- 编译并生成 TypeScript 文件：

```
yarn build
```

- 打包为 Windows 平台可执行程序：

```
yarn dist:win
```

- 打包为 Mac 平台可执行程序：

```
yarn dist:mac
```

- 打包为 Linux 平台可执行程序：

```
yarn dist:linux
```

## 5. 代码质量检查

- 进行全局语法和规范检查：

```
yarn lint
```

## 6. AI 调试面板 (可选)

启动 AI SDK 的可视化调试工具，方便调试 AI 调用：

```
yarn debug:ai
```

## 前端开发

如需修改前端界面，请前往前端仓库进行开发：

- **GitHub** : [Toonflow-web](#)
- **Gitee** : [Toonflow-web](#)

前端构建后，将 `dist` 目录内容复制到本项目的 `data/web` 目录即可集成。

## 项目结构

```

├── build/           # 编译产物
├── data/           # 运行时数据
│   ├── models/    # 本地推理模型 (ONNX)
│   ├── oss/       # 对象存储 (素材/角色/场景)
│   ├── serve/     # 生产环境入口
│   ├── skills/    # Agent 技能提示词
│   └── web/       # 前端编译产物 (内置)
├── docs/          # 文档资源
├── env/           # 环境配置
├── scripts/       # 构建与辅助脚本
├── src/
│   ├── agents/    # AI Agent 模块
│   │   ├── productionAgent/ # 生产 Agent
│   │   └── scriptAgent/    # 剧本 Agent
│   ├── lib/       # 公共库 (数据库初始化、响应格式)
│   ├── middleware/ # 中间件
│   ├── routes/   # 路由模块
│   │   ├── agents/ # Agent 记忆管理
│   │   ├── artStyle/ # 画风管理
│   │   ├── assets/  # 素材管理
│   │   └── assetsGenerate/ # 素材生成

```

├─	└─	cornerScape/	# 分镜管理
├─	└─	general/	# 通用接口
├─	└─	login/	# 登录认证
├─	└─	migrate/	# 数据迁移
├─	└─	modelSelect/	# 模型选择
├─	└─	novel/	# 小说管理
├─	└─	other/	# 其他功能
├─	└─	production/	# 制作管理
├─	└─	project/	# 项目管理
├─	└─	script/	# 剧本生成
├─	└─	scriptAgent/	# 剧本 Agent 接口
├─	└─	setting/	# 系统设置
├─	└─	task/	# 任务管理
├─	└─	test/	# 测试接口
├─	└─	socket/	# WebSocket 实时通信
├─	└─	types/	# TypeScript 类型声明
├─	└─	utils/	# 工具函数
├─	└─	app.ts	# 应用入口
├─	└─	core.ts	# 核心初始化
├─	└─	env.ts	# 环境变量处理
├─	└─	err.ts	# 错误处理
├─	└─	logger.ts	# 日志模块
├─	└─	router.ts	# 路由注册
├─	└─	utils.ts	# 通用工具
├─	└─	Dockerfile	# Docker 构建文件
├─	└─	electron-builder.yml	# Electron 打包配置
├─	└─	skillList.json	# 技能清单
├─	└─	LICENSE	# 许可证 (Apache-2.0)
├─	└─	NOTICES.txt	# 第三方依赖声明
├─	└─	package.json	# 项目配置
├─	└─	tsconfig.json	# TypeScript 配置

## 相关仓库

仓库	说明	GitHub	Gitee
<b>Toonflow-app</b>	完整客户端 (本仓库, 推荐普通用户)	<a href="#">GitHub</a>	<a href="#">Gitee</a>
<b>Toonflow-web</b>	前端源代码 (适合前端开发者)	<a href="#">GitHub</a>	<a href="#">Gitee</a>

 **提示**：如果您只是想使用 Toonflow，直接下载本仓库的客户端即可。前端仓库仅供需要二次开发或定制前端界面的开发者使用。

## 微信交流群

拉群小助手:



也可以点击图标加入 Discord :



或点击邀请连接 : <https://discord.gg/HEjKmpNpAZ>

## 联系我们

 邮箱 : [tlctools@outlook.com](mailto:tlctools@outlook.com)

## 许可证

Toonflow 基于 Apache-2.0 协议开源发布，并附有补充商业协议。

许可证详情 : <https://www.apache.org/licenses/LICENSE-2.0>



## 补充协议

- 若将本软件以产品形式分发给 **2 个及以上独立第三方** 使用，须取得 HBAI-Ltd **书面商业授权**。
- **≤ 5 个法人** 联合运营内部使用，不对外提供服务的，视为内部使用，**无需授权**。
- 不得删除或修改 Toonflow 中的标识或版权信息。

## 永久免费场景

- 用 Toonflow 制作内容并获得平台分账
- 二次开发供自己团队内部使用
- ≤ 5 个法人联合运营内部使用
- 个人学习、研究、非商业用途

## 商业授权定价

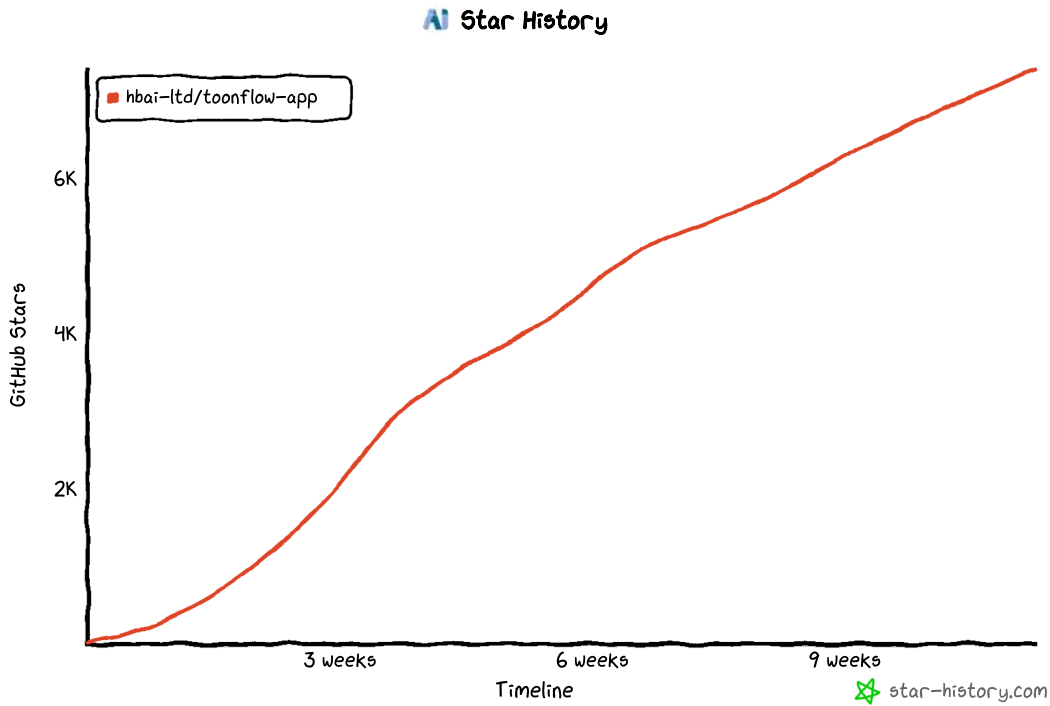
阶段	年销售额	年费
 扶持期	< ¥10 万	申请即可免费授权
 初创期	¥10-50 万	¥5,000/年

阶段	年销售额	年费
成长期	¥50-150 万	¥20,000/年
规模期	¥150-500 万	¥80,000/年
企业级	> ¥500 万	面议

不追溯条款：v1.0.8 发布前基于 AGPL-3.0 使用的用户，继续按 AGPL-3.0 执行，不受本协议变更约束。

完整协议详见 [LICENSE](#) 文件。

## ★ 星标历史



## 🙏 致谢

感谢以下开源项目为 Toonflow 提供强大支持：

### Releases 18

ToonFlow v1.1.6 (Latest)  
last week

+ 17 releases

### Contributors 7



### Languages

