

宝宝成长记录工具

基于 Tauri + React 开发的高性能桌面应用，专为记录宝宝成长瞬间设计。

✨ 核心功能

📁 智能文件管理

- 自动扫描指定文件夹中的所有图片和视频
- 支持多种格式：
 - 图片：JPG、PNG、GIF、BMP、WebP、HEIC
 - 视频：MP4、MOV、AVI、MKV、WMV、FLV、WebM

📅 智能月龄分类

- 根据宝宝出生日期和文件创建时间自动分类
- 直观的月龄展示（如"0岁3月"、"1岁2月"）
- 支持手动调整文件分类

🚀 高性能体验

- Rust后端，文件扫描速度极快
- 智能缩略图生成和缓存
- 懒加载技术，流畅浏览大量文件
- 首次扫描后，后续启动秒开

🎨 现代化界面

- 清爽的网格布局

- 便捷的侧边栏月龄导航
- 图片/视频预览功能
- 内置视频播放器



便利贴功能

- 为每张照片/视频添加文字描述
- 记录特殊时刻和美好回忆
- 数据本地存储，永不丢失



技术栈

- 后端: Rust + Tauri 2.0
- 前端: React 18 + TypeScript + Vite
- 样式: TailwindCSS
- 数据库: SQLite
- 图像处理: image-rs
- 并发处理: Rayon



安装与构建

前置要求

1. **Node.js** (v18+)
 - 下载地址: <https://nodejs.org/>
2. **Rust** (1.70+)
 - ```bash
 - # Windows
 - 下载并安装: <https://www.rust-lang.org/tools/install>

```
# macOS/Linux
curl --proto '=https' --tlsv1.2 -sSf https://sh.rustup.rs | sh
```
```

## 1. 系统依赖

### Windows:

- Microsoft Visual Studio C++ Build Tools
- WebView2 (Windows 10/11 自带)

### macOS:

```
bash xcode-select --install
```

### Linux (Ubuntu/Debian):

```
bash sudo apt update sudo apt install libwebkit2gtk-4.1-dev \ build-essential \ curl \ wget \ file \ libxdo-dev \ libssl-dev \ libayatana-appindicator3-dev \ librsvg2-dev
```

## 开发模式

### 1. 克隆或复制项目

```
bash cd baby-growth-tracker
```

### 2. 安装依赖

```
bash npm install
```

### 3. 运行开发服务器

```
bash npm run tauri:dev
```

首次运行会编译 Rust 依赖，需要几分钟时间。

## 构建生产版本

```
构建可执行文件
npm run tauri:build
```

构建完成后，安装包位于：

- **Windows:** `src-tauri/target/release/bundle/msi/` 或 `nsis/`

- **macOS:** `src-tauri/target/release/bundle/dmg/`
- **Linux:** `src-tauri/target/release/bundle/deb/` 或 `appimage/`

## 使用说明

### 首次使用

#### 1. 启动应用

- 双击运行构建好的可执行文件

#### 2. 配置设置

- 首次启动会自动打开设置对话框
- 设置宝宝的出生日期
- 选择存放照片/视频的文件夹

#### 3. 扫描文件

- 点击"选择文件夹"按钮
- 应用会自动扫描并按月龄分类
- 后台生成缩略图以提升浏览速度

### 日常使用

#### 浏览照片

- 左侧边栏显示所有月龄分组
- 点击月龄查看该阶段的照片/视频
- 点击"全部文件"查看所有内容

#### 查看详情

- 点击任意图片/视频卡片
- 进入全屏预览模式
- 查看详细信息（拍摄时间、月龄等）

## 添加便利贴

- 在预览模式下点击"编辑"按钮
- 输入文字描述
- 点击"保存"

## 修改设置

- 点击右上角"设置"按钮
- 可修改出生日期或更换文件夹
- 保存后会重新扫描文件

## 性能优化

### 缩略图缓存

- 首次扫描时生成缩略图
- 缩存位置：应用数据目录 `/cache/thumbnails/`
- 后续打开无需重新生成，秒开

### 并行处理

- 使用 Rayon 并行处理文件扫描
- 多核 CPU 充分利用，速度提升数倍

### 数据库优化

- SQLite 存储文件信息
- 快速检索和过滤
- 支持数万张照片流畅浏览

## 项目结构

```

baby-growth-tracker/
├── src/ # React 前端
│ ├── components/ # React 组件
│ │ ├── Sidebar.tsx
│ │ ├── FileGrid.tsx
│ │ ├── SettingsModal.tsx
│ │ └── FilePreview.tsx
│ ├── App.tsx # 主应用
│ ├── main.tsx # 入口文件
│ ├── types.ts # TypeScript 类型
│ └── index.css # 样式
├── src-tauri/ # Rust 后端
│ ├── src/
│ │ ├── main.rs # 主入口
│ │ ├── commands.rs # 命令处理
│ │ └── db.rs # 数据库操作
│ ├── Cargo.toml # Rust 依赖
│ └── tauri.conf.json # Tauri 配置
├── package.json # Node.js 依赖
└── README.md # 本文件
```

## 常见问题

### Q: 首次构建很慢?

A: Rust 首次编译需要下载和编译所有依赖，正常需要 5-10 分钟。后续构建会快很多。

### Q: 缩略图在哪里?

A: 缩略图存储在系统应用缓存目录:

-

**Windows:**

C:\Users\<用户名>

```
>\AppData\Local\com.baby.growth.tracker\cache\thumbnails\
- macOS: ~/Library/Caches/com.baby.growth.tracker/thumbnails/
- Linux: ~/.cache/com.baby.growth.tracker/thumbnails/
```

## Q: 可以删除缩略图缓存吗?

A: 可以。删除后，下次打开应用会自动重新生成。

## Q: 支持网络相册吗?

A: 当前版本仅支持本地文件夹。后续版本会考虑添加云同步功能。

## Q: 数据存储在哪里?

A: 所有数据（设置、笔记）存储在本地 SQLite 数据库：

```
- Windows: C:\Users\<用户名>\AppData\Roaming\com.baby.growth.tracker\baby_growth.db
- macOS: ~/Library/Application Support/com.baby.growth.tracker/
baby_growth.db
- Linux: ~/.local/share/com.baby.growth.tracker/baby_growth.db
```

## Q: 如何备份数据?

A: 只需备份上述数据库文件即可。建议定期备份到云盘或移动硬盘。



## 开发说明

### 添加新功能

#### 1. 添加 Rust 命令

- 在 `src-tauri/src/commands.rs` 中添加函数
- 在 `src-tauri/src/main.rs` 的 `invoke_handler` 中注册

#### 2. 调用命令

- 在 React 组件中使用 `invoke('命令名', { 参数 })`

## 调试

```
开启 Rust 调试日志
RUST_LOG=debug npm run tauri:dev

开启 DevTools
在 tauri.conf.json 中设置 "devtools": true
```

## 贡献

欢迎提交 Issue 和 Pull Request!

## 许可证

MIT License

## 作者

MiniMax Agent

---

祝您的宝宝健康成长! 🎉👶✨