# ✅ GitHub Actions 自动构建配置完成

## 🎉 已完成的配置

我已经为您的宝宝成长记录应用配置好了 GitHub Actions 自动构建系统，现在您可以通过云端自动构建 **Windows、macOS 和 Linux** 三个平台的安装包！

## 📁 新增的文件

### 1. .github/workflows/build.yml

**GitHub Actions 工作流配置文件**

* 自动在三个平台上并行构建应用
* 生成安装包并自动发布到 Releases
* 包含详细的版本说明和下载指引

### 2. GITHUB\_ACTIONS\_GUIDE.md

**完整的 GitHub Actions 使用教程**

包含： - 什么是 GitHub Actions - 详细的上传和构建步骤 - 常见问题解答 - 进阶配置说明

### 3. QUICK\_RELEASE.md

**5 分钟快速发布指南**

简化版教程，适合想快速上手的用户： - 创建 GitHub 仓库 - 上传代码 - 触发构建 - 下载安装包

### 4. upload-to-github.sh (macOS/Linux)

**一键上传脚本**

在终端运行，自动完成： - Git 仓库初始化 - 添加和提交文件 - 关联远程仓库 - 推送代码

### 5. upload-to-github.bat (Windows)

**一键上传脚本（Windows 版）**

双击运行，自动完成所有上传步骤

### 6. .gitignore

**Git 忽略文件配置**

确保不会将以下内容提交到 GitHub： - node\_modules - 构建产物 - 缓存文件 - 本地数据库

### 7. 更新的 README.md

添加了 GitHub Actions 构建方式的说明

## 🚀 如何使用

### 最简单的方式（推荐新手）

1. **阅读快速指南**

* 查看 QUICK\_RELEASE.md

1. **创建 GitHub 仓库**
   * 访问 https://github.com/new
   * 创建一个新仓库（public 或 private 都可以）
   * 复制仓库地址
2. **运行上传脚本**

* **Windows**：
  + 双击 upload-to-github.bat
  + 粘贴仓库地址并回车
* **macOS/Linux**：
* chmod +x upload-to-github.sh  
  ./upload-to-github.sh
  + 粘贴仓库地址并回车

1. **触发构建**

* git tag v1.0.0  
  git push origin v1.0.0

1. **等待并下载**
   * 10-20 分钟后，访问 GitHub 仓库的 Releases 页面
   * 下载对应平台的安装包

### 详细教程

如需更详细的说明和问题排查，请查看： - 📖 [GitHub Actions 自动构建指南](GITHUB_ACTIONS_GUIDE.md)

## 🎯 构建产物

构建完成后，您将获得：

| 平台 | 文件格式 | 文件名示例 |
| --- | --- | --- |
| **Windows** | .msi | baby-growth-tracker\_1.0.0\_x64\_en-US.msi |
| **macOS** | .dmg | baby-growth-tracker\_1.0.0\_x64.dmg |
| **Linux** | .AppImage | baby-growth-tracker\_1.0.0\_amd64.AppImage |

## ⚙️ 工作流程

创建 GitHub 仓库  
 ↓  
上传代码  
 ↓  
创建版本标签 (v1.0.0)  
 ↓  
GitHub Actions 自动触发  
 ↓  
并行构建 Windows/macOS/Linux  
 ↓  
自动发布到 Releases  
 ↓  
下载安装包

## 💡 优势

✅ **无需本地环境** - 不需要在本地安装 Rust、Node.js 等开发工具  
✅ **跨平台构建** - 一次构建，三个平台同时生成  
✅ **完全免费** - GitHub Actions 对 public 仓库完全免费，private 仓库每月 2000 分钟免费额度  
✅ **自动化发布** - 自动生成版本说明，自动上传到 Releases  
✅ **持续集成** - 每次打标签都会自动构建最新版本

## 🔄 后续更新流程

当您修改代码后，想发布新版本：

# 1. 提交代码  
git add .  
git commit -m "更新说明"  
git push  
  
# 2. 创建新版本标签  
git tag v1.1.0  
git push origin v1.1.0  
  
# 3. 等待自动构建完成  
# 4. 到 Releases 下载新版本

## 📞 需要帮助？

* 📖 查看 [完整教程](GITHUB_ACTIONS_GUIDE.md)
* 🚀 查看 [快速指南](QUICK_RELEASE.md)
* 💻 参考 [Tauri 官方文档](https://tauri.app/zh-cn/)
* 🤖 参考 [GitHub Actions 文档](https://docs.github.com/actions)

**现在开始，只需几个命令，就能获得跨平台的安装包！** 🎉