

CẨM NANG CÀI ĐẶT & SỬ DỤNG PASCAL EDITOR BY PASCALORG

Trình Dựng Kiến Trúc 3D Local-First • Đồ Họa WebGPU & React Three Fiber • Tích Hợp AI Agent Qua Giao Thức MCP

★ GitHub Repository: <https://github.com/pascalorg/editor> (Mã Nguồn Mở MIT)

👑 Tác giả / Tổ chức: pascalorg

Giấy phép: MIT License

Ngôn ngữ & Runtime: TypeScript / Node.js 22.13+ (hoặc pnpm / Bun)

Công nghệ lõi: WebGPU, React Three Fiber, SQLite local database

Giao thức AI Agent: Model Context Protocol (MCP) • skills.sh • Claude Code • Codex

1. PASCAL EDITOR LÀ GÌ & TẠI SAO LẠI ĐẶC BIỆT?

Pascal Editor là trình biên tập kiến trúc không gian ba chiều (3D building editor) mã nguồn mở, hoạt động theo triết lý Local-first (ưu tiên cục bộ) và được xây dựng trên nền tảng React Three Fiber kết hợp WebGPU. Phần mềm cho phép các kiến trúc sư, kỹ sư và lập trình viên thiết kế, bóc tách cấu trúc nhà ở, đồng thời kết nối trực tiếp với các AI Agent thông qua giao thức MCP (Model Context Protocol).

- Đồ họa WebGPU siêu tốc trên trình duyệt:** Tận dụng tối đa phần cứng GPU máy tính để dựng khung cảnh 3D thời gian thực với đổ bóng mượt mà và tốc độ phản hồi 60 FPS.
- Khởi chạy tức thì qua Local CLI:** Chỉ cần lệnh NPX đơn giản mà không cần clone cả kho mã nguồn; tự động thiết lập cổng kết nối nội bộ và cơ sở dữ liệu SQLite tại `~/pascal/data/pascal.db`.
- Cầu nối Model Context Protocol (MCP):** Cung cấp schema công cụ 3D chuẩn hóa cho Claude Code, GPT và các AI Agent tự động đọc hiểu tọa độ, thêm bớt vách ngăn và bố trí đồ đạc theo ngôn ngữ tự nhiên.
- Chế độ bóc tách chuyên sâu (Next Preview):** Tích hợp Exploded view (nhấc mái tách tầng), Cut view (mặt cắt đa góc) và X-ray view (soi rõ hệ thống cơ điện, đường ống nước ngầm).
- Agent Skills chuyên dụng:** Cung cấp sẵn 2 kỹ năng chính thức pascal-3d (thao tác 3D an toàn) và furniture-fit (kiểm tra diện tích và khả năng vừa vặn đồ nội thất thực tế).

2. YÊU CẦU HỆ THỐNG & ĐIỀU KIỆN TIỀN QUYẾT

- Node.js:** Phiên bản Node.js 22.13 trở lên (kiểm tra bằng lệnh `'node -v'`).
- Trình duyệt hỗ trợ WebGPU:** Google Chrome, Microsoft Edge hoặc Brave phiên bản mới nhất đã bật tăng tốc phần cứng WebGPU.
- Trình quản lý gói:** npm, pnpm hoặc bun đi kèm Node.js.

- **AI Agent Client (Tùy chọn):** Claude Code CLI, OpenAI directory hoặc Codex IDE nếu muốn kích hoạt tính năng tự động hóa CAD cùng AI.

3. HƯỚNG DẪN CÀI ĐẶT & KHỞI CHẠY TỪNG BƯỚC

Người dùng không cần clone toàn bộ repository để trải nghiệm. Gói npm `@pascal-app/cli` đóng gói sẵn công cụ dòng lệnh và dịch vụ xác thực MCP nội bộ.

Cách 1: Khởi chạy trực tiếp không cần cài đặt (Khuyến dùng)

```
npx @pascal-app/cli editor
```

Cách 2: Cài đặt CLI toàn cục vào hệ thống

```
npm install --global @pascal-app/cli

# Sau khi cài xong, chỉ cần gõ lệnh:
pascal editor
```

Khởi động dịch vụ kết nối MCP nền

```
# Kết nối dịch vụ MCP cho agent mà không cần mở giao diện web:
pascal mcp connect
```

4. TÍCH HỢP AI AGENT & CÀI ĐẶT AGENT SKILLS

Để các trợ lý AI (như Claude Code hoặc Codex) trực tiếp thao tác không gian 3D, hãy tích hợp bộ Agent Skills chính thức của Pascal từ kho mã nguồn mở.

Cài đặt Agent Skills qua skills.sh

```
npx skills add pascalorg/editor \
  --skill pascal-3d \
  --skill furniture-fit
```

Cài đặt Plugin trên Claude Code

```
# Thêm marketplace và cài đặt plugin pascal-agent-skills:
/plugin marketplace add pascalorg/editor
/plugin install pascal-agent-skills@pascal
```

Lưu ý quan trọng cho Claude Code (Tránh xung đột cổng)

```
# Xóa kết nối thủ công cũ nếu đã tạo trước đó để plugin tự quản lý vòng đời:
claude mcp remove --scope user pascal
```

Cài đặt Plugin trên Codex IDE

```
codex plugin marketplace add pascalorg/editor
codex plugin add pascal-agent-skills@pascal
```

5. SỬ DỤNG GÓI THƯ VIỆN NHÚNG CHO DEVELOPER (SDK)

Nếu bạn là lập trình viên muốn nhúng trình xem 3D Pascal vào ứng dụng web riêng, hãy sử dụng hệ sinh thái package chính thức trên npm:

Cài đặt bộ thư viện Pascal Web Packages

```
npm install @pascal-app/core @pascal-app/viewer @pascal-app/editor  
@pascal-app/nodes
```

Mã nguồn TypeScript nạp plugin và viewer

```
import { loadPlugin } from '@pascal-app/core';  
import { builtinPlugin } from '@pascal-app/nodes';  
  
// Nạp plugin node dựng sẵn trước khi mount thẻ <Viewer>  
await loadPlugin(builtinPlugin);
```

6. CÁC LỖI THƯỜNG GẶP & CÁCH KHẮC PHỤC TRIỆT ĐỂ

- **Lỗi Node.js cũ (< 22.13):** Pascal yêu cầu tính năng runtime mới nhất của Node.js 22. Hãy nâng cấp Node bằng nvm (Node Version Manager) hoặc tải bản mới nhất từ nodejs.org.
- **Lỗi WebGPU không khả dụng trên trình duyệt:** Vào `chrome://flags`, tìm kiếm 'Unsafe WebGPU Support' hoặc 'WebGPU' và chuyển sang Enabled, sau đó khởi động lại trình duyệt.
- **Xung đột nhiều Agent Client cùng lúc:** Pascal quy định nguyên tắc một client hoạt động cho mỗi tiến trình local CLI. Nếu cần chạy nhiều agent độc lập, hãy cấu hình các biến môi trường PASCAL_HOME riêng biệt.
- **Không kết nối được MCP Server:** Kiểm tra xem cổng loopback có bị tường lửa chặn không, hoặc gõ lệnh 'pascal mcp status' để kiểm tra tình trạng dịch vụ nội bộ.

Tài khoản githot mới được tặng 5 xu để dùng thử công cụ AI Repo Explainer hoặc Repo Health Check.

Đăng ký: githot.io.vn/register · Tìm thêm hướng dẫn cài đặt: githot.io.vn

**Cẩm nang được biên soạn tự động và lưu trữ chính thức tại kho tài liệu githot.io.vn •
Hãy theo dõi kênh để cập nhật các công cụ AI mã nguồn mở mới nhất mỗi ngày!**

© Kênh AI Mỗi Ngày | Hướng dẫn AI & Linux Tự Động Hóa

Nội dung kỹ thuật tổng hợp từ tài liệu chính thức của repo <https://github.com/pascalorg/editor>