

AI Agent Platform: Technical Overview

1. Vision

A no-code AI agent builder platform that allows users to create and customize AI agents via a simple UI. No programming required—just drag, drop, and configure.

2. Tech Stack Overview

Frontend & Dashboard

- **Laravel:** Backend framework for routing, auth, and orchestration.
- **Inertia.js:** Enables single-page app behavior without a separate API.
- **Vue.js:** Frontend UI components.
- **Sanctum:** Token-based SPA authentication.

Agent Execution

- **Basic Agents:** Use Laravel Neuron AI package for simple prompt/response agents.
- **Heavy Agents:** Use AgentForge and FastAPI for complex agents requiring advanced AI logic or chaining.

3. UI & Frontend Architecture

Features

- Landing page
- Pricing page
- User dashboard
- Agent builder UI (drag-and-drop or form-based)

Tooling

- Vite.js for asset bundling
- Vue + TypeScript for safer prop definitions
- Automated tests with PHPUnit, Vitest

4. Laravel-Powered Basic Agents

- Powered by Neuron AI package
- Inline for fast responses, queued for longer processing
- Agents defined via templates and variables stored in DB

5. Python Microservices for Heavy Agents

Why Microservices?

- Scalability and performance (async, multiprocessing)
- Keeps Python dependencies out of Laravel stack

Architecture

- **FastAPI** handles API for AgentForge
- Expose endpoints for executing and monitoring agents
- Use Redis/RabbitMQ for background task queueing

6. Integration Between Laravel & FastAPI

Synchronous Communication

- Use Laravel HTTP client (Guzzle) to call FastAPI endpoints
- Authenticate with JWTs or API keys

Asynchronous Orchestration

- Laravel dispatches heavy agent task to a queue
- Python service consumes, processes, returns result
- Laravel listens for result and updates UI via WebSockets or polling

7. DevOps and Infra

Containerization

- Docker for local development
- Docker Compose or Kubernetes for orchestration

CI/CD

- GitHub Actions / GitLab CI for building and deploying services

Monitoring

- Centralized logs (ELK, Loki)
- Metrics via Prometheus + Grafana

Security

- Rate limit heavy endpoints

- Per-user quotas
- Audit logs of agent runs

8. Suggested Roadmap

1. Proof-of-Concept

- Laravel agent + FastAPI agent
- End-to-end UX test

2. Scalability Tests

- Load test concurrent agent usage

3. Drag-and-Drop Agent Builder

- Modular block-style UI

4. Plugin Architecture

- Let users or devs add plugins/extensions

Summary

Use Laravel for the UI and orchestration layer, while offloading complex AI processing to scalable Python services. This hybrid model ensures user-friendliness, modularity, and performance.