

# AgentPM – AI-Powered Product Management Web App

AgentPM is a full-stack web application that simulates the work of a Product Manager using modular AI agents. It automates hiring, roadmap planning, sprint tracking, product launch, and feedback analysis.

---

## Tech Stack

- **Frontend:** React (Vite) + TailwindCSS
- **Backend:** FastAPI (Python 3.10+)
- **LLM & Embeddings:** OpenAI GPT-4o + FAISS
- **Database:** PostgreSQL, mock data files
- **Orchestration (Planned):** CrewAI agent coordination
- **Deployment:** Docker + Docker Compose

---

## Project Structure

```
project-root/
├── backend/app/
│   ├── agents/           # AI modules
│   ├── orchestration/    # CrewAI script (planned)
│   ├── mock_data/        # Sample input files
│   └── tests/             # Pytest agent tests
├── frontend/src/         # React components & pages
├── docker-compose.yml
└── README.md
```

---

## Getting Started

### Prerequisites

- Python 3.10+
- Node.js (v18+)
- Docker & Docker Compose
- OpenAI API key

### 1. Clone the Repo

```
git clone https://github.com/your-username/agentpm.git
cd agentpm
```

## 2. Set Environment Variables

Create a `.env` file in `backend/`:

```
OPENAI_API_KEY=your_openai_key
GOOGLE_API_KEY=your_google_api_key
STRIPE_API_KEY=your_stripe_key
```

## 3. Run with Docker Compose

```
docker-compose up --build
```

- Frontend: <http://localhost:5173>
- Backend API: <http://localhost:8000>

---

## Run Locally Without Docker

### Backend






```
cd backend
python -m venv venv && source venv/bin/activate
pip install -r requirements.txt
uvicorn app.main:app --reload
```

### Frontend

```
cd frontend
npm install
npm run dev
```

---

## AI Agents

Agent	Description
 Talent Agent	Parses resumes, ranks by JD match
 Roadmap Agent	Clusters feedback, builds roadmap
 Progress Agent	Tracks sprint progress
 GTM Agent	Generates launch campaign copy
 Feedback Agent	Analyzes reviews and sales

## Orchestration (Planned)

Located in `backend/app/orchestration/orchestrator.py`, this script (using CrewAI) will:

- Route tasks across agents
- Simulate a full product lifecycle

Run test flow:

```
python backend/app/orchestration/orchestrator.py
```

---

## Testing

```
cd backend
pytest app/tests/
```

Includes unit tests for each agent using mock data files.

---

## Mock Files (For Demo)

- `jd_sample.txt` : Sample job description
- `resumes.json` : Sample candidate resumes
- `feedback.csv` : Feedback data
- `features.json` : Launch content input
- `reviews.json` : Feedback from users

---

## Contact

Made by Sriprakash\ Email: [sriprakash.s.ihub@snsgroups.com](mailto:sriprakash.s.ihub@snsgroups.com)\ GitHub: [github.com/Sriprakash-sns](https://github.com/Sriprakash-sns)

---

## License

MIT © 2025