

# PSScript Manager

---

A PowerShell Script Management Application with AI Analysis capabilities. This application helps manage, analyze, and document PowerShell scripts using modern web technologies and AI-powered insights.

## Features

---

- PowerShell script management and organization
- AI-powered script analysis and security scanning
- Vector search for finding similar scripts
- Documentation generation
- Multi-agent architecture for advanced analysis
- User authentication and role-based access control

## Tech Stack

---

- **Frontend:** React, TypeScript, Vite
- **Backend:** Node.js, Express
- **AI Service:** Python, FastAPI
- **Database:** PostgreSQL with pgvector extension
- **Caching:** Redis
- **Containerization:** Docker

## Quick Start with Docker

---

The easiest way to get started is using Docker:

```
# Clone the repository
git clone https://github.com/YOUR_USERNAME/psscript.git
cd psscript

# Create and configure environment variables
cp .env.example .env
# Edit .env file to set your configuration

# Start the application in production mode
./docker-deploy.sh

# Access the application
# Frontend: http://localhost:3002
# Backend API: http://localhost:4000
```

For detailed Docker setup instructions, see [DOCKER-SETUP.md](#).

## Development Setup

---

To set up the development environment:

```
# Install dependencies
npm run install:all

# Start all services in development mode
npm run dev
```

# Environment Variables

---

Key environment variables include:

- `OPENAI_API_KEY` : Your OpenAI API key
- `JWT_SECRET` : Secret for JWT authentication
- `DB_PASSWORD` : Database password
- `MOCK_MODE` : Enable mock mode for development without API keys

See `.env.example` for a complete list of configuration options.

# Architecture

---

The application consists of three main services:

1. **Frontend Service:** React application for user interface
2. **Backend API Service:** Node.js/Express API for business logic
3. **AI Service:** Python/FastAPI service for AI analysis

These services are containerized using Docker and can be deployed together using Docker Compose.

# Contributing

---

Contributions are welcome! Please see [CONTRIBUTING.md](#) for guidelines.

# License

---

This project is licensed under the MIT License - see the [LICENSE](#) file for details.

Generated 2026-01-16 21:23 UTC