# Product Requirements Document (PRD)

## Project Name: AutoStacks

## Overview

AutoStacks is an AI-powered development assistant designed to streamline the workflow of Stacks developers. It integrates AI agents for code analysis, generation, and optimization, offering an interactive chat interface and an IDE powered by Monaco. The platform enables seamless GitHub integration for project creation, analysis, and enhancement.

## Objectives

- Provide an AI-powered development assistant for Stacks blockchain developers.
- Offer a real-time chat interface for AI interactions.
- Enable seamless GitHub integration for repository management.
- Provide an in-browser IDE (using Monaco) for editing and enhancing smart contracts.
- Implement real-time AI-powered code analysis and suggestions.
- Support RAG (Retrieval-Augmented Generation) for website indexing and response generation.
- Utilize FastAPI for backend services and Next.js for frontend development.

## Key Features

### 1. AI Chat Interface

- Real-time interaction with an AI assistant.
- AI-powered code review, suggestions, and debugging.
- Natural language smart contract writing and deployment.

### 2. Integrated IDE (Monaco Editor)

- Code editing with syntax highlighting and auto-completion.
- AI-assisted contract writing and analysis.
- Version control and history tracking.

### 3. GitHub Integration

- OAuth-based GitHub authentication.
- Ability to create new repositories and manage existing ones.
- Direct file editing and commits through the platform.

### 4. Project Workflow

- **New Project Flow:** Users can create a new repository and set up a development environment.
- **Project Analysis Flow:** Users can analyze existing repositories for improvements.

### 5. Real-time Communication

- WebSocket-based real-time collaboration and updates.
- AI can suggest changes and apply them instantly.

### 6. Backend & AI Processing

- **FastAPI backend** handling AI inference, code processing, and GitHub interactions.
- **Clarinet CLI** for smart contract analysis and debugging.
- **GitHub CLI (gh CLI)** for repository management.
- **FAISS-powered RAG** for intelligent responses based on indexed website data.

## Technology Stack

### Frontend:

- Next.js (React-based framework)
- TailwindCSS (Custom theme inspired by the AutoStacks logo)
- Monaco Editor (Browser-based IDE)

### Backend:

- FastAPI (Lightweight, high-performance API framework)
- WebSockets (Real-time interaction)
- Clarinet CLI (Smart contract analysis)

- GitHub CLI (Repository management)
- FAISS (RAG for contextual responses)

### Database & Storage:

- PostgreSQL or MongoDB (User data and project metadata)
- FAISS Index (For RAG-based AI responses)

## User Flow

1. **User logs in via GitHub.**
2. **User selects** either "Create New Project" or "Analyze Existing Project."
3. **In the IDE**, users can edit, analyze, and test smart contracts.
4. **AI Agent provides** real-time feedback, suggestions, and code enhancements.
5. **User can commit** changes directly to GitHub.
6. **Users interact with AI** through a chat interface for guidance and debugging.

## Design Considerations

- **Dark-mode friendly UI** with neon cyberpunk aesthetics.
- **Minimalistic and developer-focused interface.**
- **Optimized AI response times** to ensure a seamless experience.

## Success Metrics

- User adoption rate and engagement.
- Number of AI-assisted code improvements.
- Speed and accuracy of AI-generated suggestions.
- Frequency of GitHub commits made through the platform.

## Timeline & Deliverables (5 Days Sprint)

1. **Day 1:** UI/UX Design + Initial Backend Setup
2. **Day 2:** GitHub Authentication + IDE Integration
3. **Day 3:** AI Chat + WebSocket Integration
4. **Day 4:** Clarinet & FAISS Integration + RAG Implementation
5. **Day 5:** Testing + Final Deployment

## Conclusion

AutoStacks aims to revolutionize smart contract development on Stacks by providing an**AI-powered**, **real-time**, and **intelligent** development environment. By integrating **GitHub, Monaco, FAISS, Clarinet CLI, and WebSockets**, the platform will significantly **boost productivity** and **enhance developer experience**.