# 🧠 Product Requirements Document (PRD)

🎯 Product: DevOpsGPT – An AI-Powered SRE/DevOps Agent

Owner: Ganesh  
Version: 1.0  
Date: 2025-06-01

## 1. Executive Summary

DevOpsGPT is an intelligent, command-aware AI agent that helps SREs and DevOps engineers automate repetitive tasks, resolve incidents, provision infrastructure, monitor systems, and maintain compliance. Inspired by the Azure SRE Agent by Microsoft, this agent should integrate with common DevOps tools and cloud providers (AWS, Azure, GCP) and provide a natural language interface to execute operational tasks reliably and securely.

## 2. Goals & Objectives

Goals and their descriptions:

| Goal | Description |  
|------|-------------|  
| Automate Ops | Reduce MTTR by automating common tasks such as log inspection, service restarts, deployment rollbacks, cost analysis, etc. |  
| Natural Language Interface | Accepts natural language commands and converts them into secure executable actions. |  
| Plugin-Based | Modular architecture with pluggable command handlers for extensibility. |  
| Multi-Cloud & On-Prem Support | Support for AWS, Azure, GCP, Kubernetes, and Linux VM environments. |  
| Auditable & Secure | All actions should be logged and optionally require user confirmation before execution. |

## 3. Core Features

### ✅ 3.1 Natural Language Command Parsing  
- Accept user input via chat-like interface or CLI.  
- Understand intent and extract parameters using LLM.  
- Translate intent to executable action (e.g., Bash, Terraform, kubectl, CLI).  
  
### ✅ 3.2 Core Task Categories  
| Category | Sample Tasks |  
|---------|---------------|  
| 🔍 \*\*Troubleshooting\*\* | Fetch pod logs, analyze error patterns, run health checks. |  
| 🚀 \*\*CI/CD Operations\*\* | Trigger pipeline, rollback deployment, list failed jobs. |  
| ☁️ \*\*Cloud Provisioning\*\* | Spin up EC2/GCE/VM, create Kubernetes cluster, attach disk. |  
| 💸 \*\*Cost & Usage\*\* | Analyze team/service cost, list unused resources. |  
| 🔒 \*\*Security & Compliance\*\* | List open ports, check CVEs, run CIS benchmarks. |  
| 📊 \*\*Monitoring & Alerts\*\* | Query Prometheus/Grafana, summarize alerts, restart crashed services. |  
  
### ✅ 3.3 Agent Interface  
- \*\*Input\*\*: Natural language or CLI command ("show logs from payment pod in prod").  
- \*\*Output\*\*: Structured, human-readable result + execution status.  
- Optional dry-run or confirmation prompt before execution.  
  
### ✅ 3.4 Plugin Framework  
- Each category will be implemented as a \*\*plugin module\*\* with:  
 - Command patterns  
 - Required permissions  
 - Execution logic (via shell, API, SDK, or Terraform)  
- Easy to add custom plugins (e.g., internal tools like “AutoHive” or “LicenseLens”).  
  
### ✅ 3.5 Context Awareness (v1.1+)  
- Retain short session memory (e.g., current namespace, service context).  
- Example:  
 - User: "Show logs from frontend pod"  
 - User: "Restart it"  
 - Agent understands context from earlier conversation.