Software Requirements Specification (SRS): Helm Deployment Automation A

1. Introduction

Purpose:

The purpose of this document is to define the functional, non-functional, and technical requirements for the Helm Deployment Automation API.

This API will enable developers to automate Helm chart generation, CI/CD pipeline setup, and application deployment on AWS EKS.

Scope:

- Automate Helm Chart Generation
- Automate GitHub Actions / GitLab CI/CD pipeline creation
- Deploy applications automatically to AWS EKS
- Provide APIs for status monitoring

Assumptions and Dependencies:

- AWS EKS is the target cluster.
- GitHub Actions and GitLab CI/CD will be supported.
- Helm version 3.x will be used.

2. Functional Requirements

API Endpoints:

- /generate-helm (POST): Generates Helm chart from app specs.
- /generate-ci-cd (POST): Creates GitHub Actions/GitLab CI/CD YAML.
- /deploy (POST): Deploys app to AWS EKS using Helm.
- /status (GET): Retrieves application deployment status.
- /webhook (POST): Handles GitHub/GitLab webhooks for auto-deployment.

Use Case Scenarios:

- Developer pushes code with Helm Chart specs.
- Automatic deployment on Git Push triggers the CI/CD pipeline.

3. Non-Functional Requirements

- Scalability: API should handle 1000+ requests per minute.
- Security: Role-Based Access Control (RBAC) for API authentication.
- Availability: 99.9% uptime with AWS auto-scaling.
- Logging & Monitoring: Integrated with AWS CloudWatch.
- Versioning: API versions must be maintained for backward compatibility.

4. System Architecture

High-Level Architecture:

- Developer -> Push Code -> GitHub/GitLab -> Webhook -> API -> Helm Chart -> CI/CD -> AWS EKS.

Tech Stack:

- Backend Framework: NestJS / Express.js (Node.js)
- Kubernetes SDK: @kubernetes/client-node
- Helm Integration: child_process.exec('helm install')
- AWS Deployment: AWS EKS
- CI/CD Integration: GitHub Actions, GitLab CI/CD
- Database (for logs): AWS DynamoDB / PostgreSQL

5. Cost Management

Estimated Cost per Month:

- AWS EKS Cluster: $100
- AWS RDS (PostgreSQL): $50
- AWS Lambda (Webhooks): $10
- AWS S3 (Helm storage): $5
- GitHub Actions (CI/CD): $20
- CloudWatch Logging: $15
- API Hosting & Gateway: $10
- Total Estimated Cost: $210/month

One-Time Development Costs:

- Development (4-6 months): $30,000 - $50,000
- Testing & Security Audits: $5,000 - $10,000
- Infrastructure Setup: $2,000 - $5,000

6. Risks & Mitigation

- API scaling issues: Medium risk, high impact. Mitigation: Use AWS auto-scaling.
- Security vulnerabilities: High risk, high impact. Mitigation: Implement API authentication & encryption.
- Helm misconfigurations: Medium risk, high impact. Mitigation: Validate charts before applying.
- AWS authentication failures: Low risk, high impact. Mitigation: Use IAM roles & permissions.

7. Project Timeline

- Phase 1: API Framework (2 weeks)
- Phase 2: Helm Chart Generation (3 weeks)
- Phase 3: CI/CD Pipeline Integration (3 weeks)
- Phase 4: AWS EKS Deployment (3 weeks)
- Phase 5: Webhook Integration (2 weeks)

- Phase 6: Testing & Optimization (2 weeks)
- Total Duration: ~15 Weeks (~4 months)

8. Deployment Plan

Stages of Deployment:

- Development: Local Kubernetes cluster (Minikube).
- Staging: AWS EKS test cluster for integration testing.
- Production: AWS EKS prod cluster for live deployment.

Deployment Checklist:

- Validate Helm chart and YAML configurations.
- Run CI/CD pipeline and confirm successful deployment.
- Ensure API authentication and security best practices.

9. Success Criteria

- API generates Helm charts successfully and commits them to Git.
- CI/CD pipelines automatically deploy applications.
- The system scales to 100+ concurrent deployments.
- The API achieves 99.9% uptime in production.