



Phase-1 Submission Template

Student Name: Aravindhan M

Register Number: 410723243005

Institution: Dhanalakshmi College of Engineering

Department: Artificial Intelligence and Data Science

Date of Submission: 20-04-2025

1.Problem Statement

In monolithic application architectures, scalability, fault isolation, and team autonomy are challenging. Deploying a microservices-based system with robust API management and service-to-service communication can address these issues. This project aims to solve the complexity of managing distributed services, ensuring secure and efficient inter-service communication while simplifying service discovery, observability, and resilience

2.Objectives of the Project

- Design and deploy a microservices architecture on Kubernetes.
- Implement API Gateway for centralized entry-point management.
- Integrate Istio service mesh to manage traffic, security, and observability between microservices.
- Ensure security, scalability, and fault tolerance of services.
- Demonstrate monitoring, load balancing, and traffic routing capabilities.

3.Scope of the Project

Features:

- - Microservices deployment on Kubernetes.
- API Gateway setup (e.g., using Kong, NGINX, or Ambassador).
- Istio integration for service mesh features.



- Observability via Prometheus, Grafana, or Kiali.
- Basic authentication and traffic management.

Constraints:

- - Limited to open-source tools.
- Deployment on local minikube or cloud (time/resources permitting).
- No external paid services.

4.Data Sources

Not applicable (infrastructure-focused project). Synthetic or sample requests will simulate service communication and API interactions for demonstration.

5.High-Level Methodology

- Service Design – Break a sample application into independently deployable microservices.
- Kubernetes Deployment – Containerize and deploy services using YAML files.
- API Gateway Setup – Configure routes, rate limiting, and authentication.
- Istio Integration – Enable mutual TLS, telemetry, tracing, and intelligent routing.
- Monitoring&Logging – Set up dashboards with Grafana, logs with Fluentd/Elasticsearch.
- Testing – Validate inter-service communication and gateway behavior.
- Documentation&Presentation – Summarize the architecture and its benefits.

6.Tools and Technologies

- Programming Language: Go / Python / JavaScript (depending on service)
- IDE: VS Code
- Libraries/Frameworks: Docker, Kubernetes, Istio, Helm, Prometheus, Grafana
- Deployment Tools: Minikube, kubectl, Istioctl, Kong, Kiali

7.Team Members and Roles

- Aravindhan M – DevOps Lead: Responsible for Kubernetes setup, service deployment, and CI/CD integration.
- Jaikerthi R – Service Mesh Architect: Handles Istio installation, configuration, and service communication policies.
- Akash – API Gateway Specialist: Sets up API Gateway routes, authentication, and rate-limiting strategies.
- K Adithya – Monitoring&Documentation: Implements observability tools (Grafana, Kiali) and prepares final documentation and dashboard presentation.

ORACLE



AdroIT Technologies®
Innovative Solutions Pvt LTD