Technical Report: AI Trip Advisor on Kubernetes

1.0 Project Overview

Project Title: AI Trip Advisor: Multi-Pod Multi-Container Pipeline on Kubernetes
Team: Cloud Innovators
Date: 09/28/25

2.0 Team Composition and Roles

Project Lead & DevOps Engineer: Sultan Fahim- Kubernetes architecture, deployment automation

Backend: James McKim - AI planning service and architecture

AI Specialist: James McKim - API development and Engineering

Frontend & UX Engineer: [Name 3] - User interface, web application

Data & Storage Architect: [Name 4] - Database design, persistence layer

QA & Documentation Specialist: Robert Iacovella - Testing strategy, technical documentation

3.0 Architecture Design

3.1 System Architecture

[Include detailed architecture diagram showing all pods, services, and data flow]

3.2 Component Specifications

Frontend Service: React application served via Nginx

API Gateway: Python/Flask request routing and orchestration

Request Processor: Multi-container pod (validation + metrics sidecar)

AI Planner: Scalable Python service with mock AI intelligence

Storage Service: Multi-container pod (SQLite + backup sidecar)

3.3 Kubernetes Service Design

Service Types: LoadBalancer (frontend), ClusterIP (internal services)

Networking: Internal service discovery via Kubernetes DNS

Persistence: PVCs for database and backup storage

4.0 API Contracts

4.1 Trip Planning Endpoint

yaml
POST /api/trip/plan

Request:

```
{
  "destination": "string",
  "budget": "number",
 "duration_days": "number",
  "travelers": "number",
  "preferences": "string[]",
  "constraints": "string[]"
}
```

Response:

```
{
  "trip_id": "string",
  "itinerary": "object[]",
  "total_cost": "number",
  "status": "success"
}
```

5.0 Persistence Plan

5.1 Storage Requirements

Trip Data: SQLite database with trip itineraries and metadata

Backups: Automated database snapshots with retention policy

PVC Configuration: 1GB primary storage, 2GB backup storage

5.2 Data Model

sql

```
CREATE TABLE trips (
    trip_id TEXT PRIMARY KEY,
    destination TEXT,
    duration_days INTEGER,
travelers INTEGER,
    budget REAL,
    preferences TEXT,
    constraints TEXT,
    itinerary TEXT,
    total_cost REAL,
    created_at TIMESTAMP
);
```

6.0 Risk Assessment and Mitigation

6.1 Technical Risks

AI Service Performance: Implement timeout handling and circuit breaker pattern

Database Scalability: SQLite suitable for MVP; PostgreSQL migration path defined

Resource Contention: Kubernetes resource limits and quality of service classes

6.2 Test Plan

Unit Testing: Individual service functionality

Integration Testing: Inter-service communication

End-to-End Testing: Complete user journey validation

Load Testing: Scalability and performance under load

7.0 Project Timeline

7.1 Milestone Schedule

Milestone 1: Architecture and Design (Week 1-2)

Milestone 2: Core Implementation (Week 3-6)

Milestone 3: Persistence and Demo (Week 7-8)

7.2 Definition of Done

All services containerized and deployed to Kubernetes

Multi-container pods functioning correctly

End-to-end trip planning workflow operational
Persistent storage surviving pod restarts

Comprehensive documentation completed

8.0 Conclusion

This technical report outlines our approach to building a cloud-native AI Trip Advisor platform that demonstrates advanced Kubernetes concepts including multi-pod architecture, multi-container pods, persistent storage, and scalable deployments. Our design prioritizes resilience, scalability, and maintainability while delivering a compelling user experience.