Air Canada Flights Routes API Deployment Report

Table of Contents

1. Deployment Overview

This document summarizes the deployment details for the Air Canada Flight Routes API solution.

The API is deployed to a cloud-hosted environment and exposed via a public endpoint for testing and integration.

2. Deployment Target

- **Platform:** Render.com - Cloud-based PaaS (Platform as a Service)
- **Deployment Mode:** Container-based deployment (Node.js app)
- **Hosting Region:** Render default region (US-based)
- **Operating System:** Linux container
- **Node.js Version:** 16.x or later

3. Deployment Methodology

- The project is hosted on GitHub.
- CI/CD is enabled:
  - Pushes to the main branch trigger auto-deployment to Render via GitHub integration.
- Deployment pipeline:
  - Build step → NPM install + run
  - App runs as Express server with API endpoints.

4. Deployed URL

The API is available at:

Base URL: https://aircanada-partner-integration.onrender.com

5. Exposed API Endpoints

Endpoint	Full URL	Description
`/flightRoutes`	https://aircanada-partner-integration.onrender.com/flightRoutes	Retrieve flight offers
`/flightRoutesWithHotels`	https://aircanada-partner-integration.onrender.com/flightRoutesWithHotels	Retrieve flight offers + hotels
`/health`	https://aircanada-partner-integration.onrender.com/health	Health check endpoint

`/api-docs`	https://aircanada-partner-integration.onrender.com/api-docs	Swagger API documentation

6. Monitoring & Logging

- Render provides live application logs via the Render dashboard.
- Application implements structured logging using Winston.
- Logs include:
  - o Request correlation IDs
 - o Errors and exceptions
  - o Incoming requests

7. Health Checks

- /health endpoint implemented and monitored.
- Verified to return {"status": "ok"} when system is healthy.

8. Security & Rate Limiting

- API enforces rate limiting via express-rate-limit middleware.
- Current configuration:
  - o Window: 60 seconds
  - o Max requests: 100 requests per window per IP
- Request timeout enforced:
  - o Timeout: 10 seconds (REQUEST_TIMEOUT_MS)

9. Deployment Verification

- Deployment successfully verified on Render.
- Manual testing performed for:
  - o /flightRoutes
  - o /flightRoutesWithHotels
  - o /health
  - o Rate limiting behavior
  - o Timeout handling
- Swagger documentation verified and accessible.
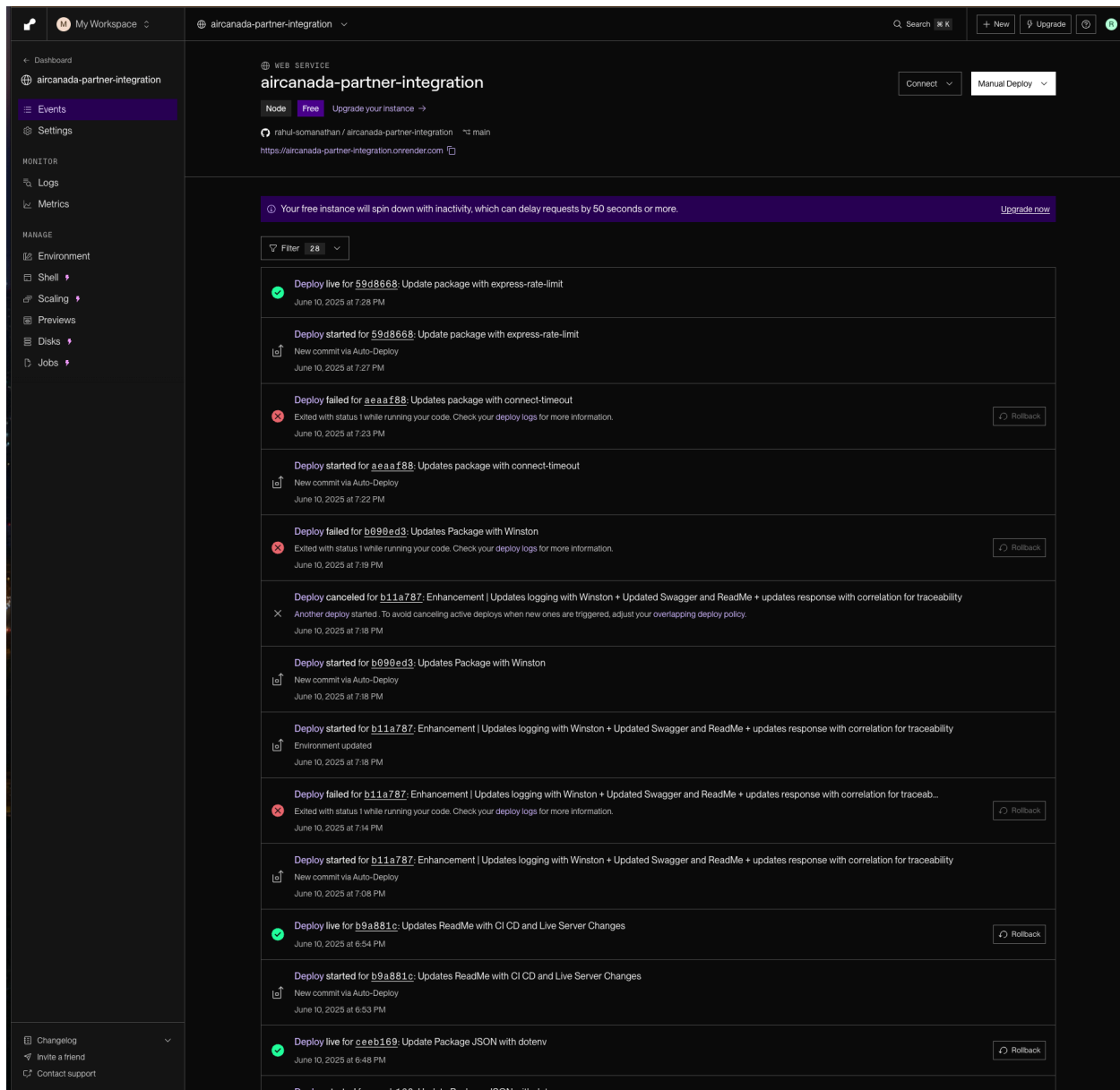
10. GitHub Repository

The full source code and documentation for the project is publicly available at:

https://github.com/rahul-somanathan/aircanada-partner-integration

The repository includes:

- Full source code
- System Architecture Document
- Deployment Report

- README.md with API documentation
- Swagger API documentation exposed at /api-docs

11. Conclusion

The Air Canada Flight Routes API is fully deployed and available at the public endpoint.

It follows production readiness best practices:

- CI/CD enabled
- Rate limiting
- Timeouts
- Health checks
- Structured logging