Document

23. Production Deployment Strategies

Goal: Understand ADK deployment options and implement production-grade agents with custom authentication, monitoring, and reliability patterns.

Prerequisites:

- Tutorial 01 (Hello World Agent)
- Google Cloud Platform account
- Basic Docker knowledge (helpful)
- Understanding of FastAPI (helpful)

What You'll Learn:

- ✅ Deploy agents using ADK's built-in server (5 minutes)
- 🏗️ Build production FastAPI servers with custom patterns (when needed)
- 📊 Implement custom monitoring and observability
- 🔐 Add authentication and security patterns
- 📈 Auto-scale across platforms
- 🛡️ Understand when to use ADK vs custom server

Quick Decision Framework:

- **5 minutes to production?** → Cloud Run ✅
- **Need FedRAMP compliance?** → Agent Engine ✅✅
- **Have Kubernetes?** → GKE ✅
- **Need custom auth?** → Tutorial 23 + Cloud Run ⚙️

- **Just testing locally?** → Local Dev ⚡

Time to Complete: 5 minutes (Cloud Run) to 2+ hours (custom patterns)

🎯 DECISION FRAMEWORK: Choose Your Platform

What's Your Situation?

```
┌──────────────────────────────────────────────────────────────────┐
│ 1. QUICK MVP / MOVING FAST?                                      │
├──────────────────────────────────────────────────────────────────┤
│ Setup: 5 minutes | Cost: ~$40/mo | Security: Auto ✔️
│ → Use: CLOUD RUN ✔️
│ Best for: Startups, MVPs, most production apps                   │
│ Deploy: adk deploy cloud_run --project ID --region us-central1   │
└──────────────────────────────────────────────────────────────────┘

┌──────────────────────────────────────────────────────────────────┐
│ 2. NEED COMPLIANCE (FedRAMP, HIPAA, PCI-DSS)?                    │
├──────────────────────────────────────────────────────────────────┤
│ Setup: 10 minutes | Cost: ~$50/mo | Security: Auto ✔️✔️
│ → Use: AGENT ENGINE ✔️✔️
│ Best for: Enterprise, government, compliance-heavy               │
│ Why: Only platform with FedRAMP compliance                       │
│ Deploy: adk deploy agent_engine --project ID --region us-center  │
└──────────────────────────────────────────────────────────────────┘

┌──────────────────────────────────────────────────────────────────┐
│ 3. HAVE KUBERNETES / NEED FULL CONTROL?                          │
├──────────────────────────────────────────────────────────────────┤
│ Setup: 20 minutes | Cost: $200-500/mo | Security: Configure ⚙️
│ → Use: GKE ✔️
│ Best for: Complex deployments, existing Kubernetes shops         │
│ Deploy: kubectl apply -f deployment.yaml                         │
└──────────────────────────────────────────────────────────────────┘

┌──────────────────────────────────────────────────────────────────┐
│ 4. NEED CUSTOM AUTH (LDAP, KERBEROS)?                            │
├──────────────────────────────────────────────────────────────────┤
│ Setup: 2 hours | Cost: ~$60/mo | Security: Custom + Platform ⚙️
│ → Use: TUTORIAL 23 + CLOUD RUN ⚙️
│ Best for: Custom authentication requirements                     │
│ Why: Platform doesn't support these auth methods natively        │
│ Note: Most users don't need this - use Cloud Run IAM instead     │
└──────────────────────────────────────────────────────────────────┘

┌──────────────────────────────────────────────────────────────────┐
│ 5. JUST DEVELOPING LOCALLY?                                      │
├──────────────────────────────────────────────────────────────────┤
│ Setup: < 1 min | Cost: Free | Security: Add before deploy ⚡
│ → Use: LOCAL DEV ⚡                                                │
│ Best for: Development, prototyping, testing                      │
│ Deploy: adk api_server                                           │
└──────────────────────────────────────────────────────────────────┘
```

→ Pick the box that matches your situation. That's your platform.

⚠️ Important: Understanding ADK's Deployment Model

Key Insight: Security is Platform-First

ADK's built-in server is **intentionally minimal by design**. Here's why:

- ✔️ **ADK provides**: Input validation, session management, error handling
- ✔️ **Platform provides**: TLS/HTTPS, DDoS protection, authentication, compliance
- ✔️ **Result**: Secure production deployment with zero custom security code

See: Security Research Summary (https://github.com/raphaelmansuy/adk_training/blob/main/tutorial_implementation/tutorial23/SECURITY_RESEARCH_SUMMARY.md) for complete analysis of what each platform secures automatically.

Custom Server (Tutorial 23) is ADVANCED & OPTIONAL

You only need the custom FastAPI server if:

- You need custom authentication (LDAP, Kerberos, etc.)
- You need advanced logging beyond platform defaults
- You have specific business logic endpoints
- You're not using Google Cloud infrastructure

Most production deployments use Cloud Run + ADK's built-in. No custom server needed.

Platform Comparison

Platform	Security	Setup	Cost	Best For	Needs Custom Server?
Cloud Run	Auto ✔️	5 min	Pay-per-use	Most apps	❌ No
Agent Engine	Auto ✔️✔️	10 min	Pay-per-use	Enterprise	❌ No
GKE	Configure ⚙️	20 min	Hourly	Complex	❌ No
Custom + Cloud Run	Hybrid ⚙️	2 hrs	Pay-per-use	Special needs	✔️ Yes
Local Dev	Minimal	< 1 min	Free	Development	✔️ Yes (add locally)

See: Complete Security Analysis (https://github.com/raphaelmansuy/adk_training/blob/main/tutorial_implementation/tutorial23/SECURITY_ANALYSIS_ALL_DEPLOYMENT_OPTIONS.md) for detailed security breakdown per platform.

🔐 Security First: What's Automatic vs Manual

Important Discovery: Each platform provides different levels of automatic security.

Security by Platform (Quick Reference)

Security Feature	Cloud Run	Agent Engine	GKE	Local
HTTPS/TLS	✓ Auto	✓ Auto	✓ Manual	❌
DDoS Protection	✓ Auto	✓ Auto	❌	❌
Authentication	✓ Auto (IAM)	✓ Auto (OAuth)	⚙ Manual	❌
Encryption at Rest	✓ Auto	✓ Auto	✓ Manual	❌
Audit Logging	✓ Auto	✓ Auto	✓ Manual	❌
Compliance Ready	✓ HIPAA, PCI	✓✓ **FedRAMP**	✓ All	❌

Key Message: Cloud Run and Agent Engine give you **production-ready security with zero configuration**. All security is automatic.

Read the Full Security Analysis

For comprehensive details on what's secure across all platforms:

- 📄 **SECURITY_RESEARCH_SUMMARY.md** (https://github.com/raphaelmansuy/adk_training/blob/main/SECURITY_RESEARCH_SUMMARY.md) - Executive summary (5 min read)
- What ADK provides vs what platforms provide
- When you actually need custom authentication
- Platform security capabilities comparison
- Real-world use case recommendations
- 📋 **SECURITY_ANALYSIS_ALL_DEPLOYMENT_OPTIONS.md** (https://github.com/raphaelmansuy/adk_training/blob/main/SECURITY_ANALYSIS_ALL_DEPLOYMENT_OPTIONS.md) - Comprehensive (20 min read)
- Detailed security breakdown per platform
- Compliance certifications
- Platform-specific security checklists
- Security verification steps
- When to use custom server

Bottom Line: "ADK's built-in server is secure by design because platform security is the foundation."

Quick Reference: Understanding ADK's Deployment

What Happens When You Run `adk deploy cloud_run`?

```
Your Agent Code
       ↓
[ADK Generates]
├── Dockerfile
├── main.py (using get_fast_api_app() from ADK)
└── requirements.txt
       ↓
[Builds Container]
       ↓
[Deploys to Cloud Run]
       ↓
✔️ Live FastAPI Server
   (with basic endpoints only)
```

What's Inside ADK's Built-In Server?

Provided by `get_fast_api_app()`:

- ✔️ `GET /` - API info
- ✔️ `GET /health` - Health check
- ✔️ `GET /agents` - List agents
- ✔️ `POST /invoke` - Run agent
- ✔️ Session management

NOT Provided:

- ❌ Custom authentication
- ❌ Custom logging

- ❌ Custom metrics
- ❌ Rate limiting
- ❌ Circuit breakers

When You Need a Custom Server

The custom server in this repository (Tutorial 23) adds:

- ✔️ Custom authentication
- ✔️ Structured logging with request tracing
- ✔️ Health checks with real metrics
- ✔️ Request timeouts and circuit breaking
- ✔️ Custom error handling
- ✔️ Full observability

See: `DEPLOYMENT_OPTIONS_EXPLAINED.md` for complete details

Time to Complete: 45 minutes

🌍 Real-World Scenarios: Which Platform for Which Situation?

Scenario 1: Startup Building MVP

Your Situation: Moving fast, limited resources, want to deploy this week.

What You Need:

- Deployment in < 5 minutes
- Automatic security (don't want to manage this)
- Pay only for what you use
- Can iterate quickly

Recommendation: ✔️ **Cloud Run**

Why:

- Fastest time to market (5 minutes!)
- Secure by default (HTTPS, DDoS, IAM)
- Cost-effective (~$40/mo for 1M requests)
- No infrastructure to manage

Deploy:

```
adk deploy cloud_run \
  --project your-project-id \
  --region us-central1
```

Cost: ~$40/month (1M requests) + $0.30/CPU-month

Next Step: As you grow, consider Agent Engine for better compliance.

Scenario 2: Enterprise System (Need Compliance)

Your Situation: Building for enterprise customers, need FedRAMP or HIPAA compliance.

What You Need:

- FedRAMP compliance (government-ready)
- HIPAA/PCI-DSS certifications
- Zero infrastructure management
- Immutable audit logs
- Sandboxed execution

Recommendation: ✔✔ **Agent Engine (ONLY PLATFORM WITH FedRAMP)**

Why:

- Only platform with FedRAMP compliance built-in
- Google manages all security/compliance
- Zero configuration needed
- Best for highly regulated industries

Deploy:

```
adk deploy agent_engine \
  --project your-project-id \
  --region us-central1 \
  --agent-name my-agent
```

Cost: ~$50/month (1M requests) + usage

Benefits:

- FedRAMP compliance
- SOC 2 Type II certified
- Automatic audit logging
- Content safety filters
- No ops burden

Next Step: Already production-ready. Focus on agent safety.

Scenario 3: Kubernetes Shop

Your Situation: Your company runs Kubernetes infrastructure, you want ADK in that environment.

What You Need:

- Deploy in existing Kubernetes cluster
- Full control over configuration
- NetworkPolicy for traffic control
- Workload Identity integration
- Pod resource limits

Recommendation: ✔️ **GKE (or any Kubernetes)**

Why:

- Leverage existing infrastructure
- Full control over security config
- Support for complex networking

- Advanced observability

Deploy:

```
kubectl apply -f deployment.yaml
```

Cost: $200-500+/month (based on cluster size)

Requires:

- Kubernetes expertise
- Manual security configuration
- Pod security setup
- RBAC configuration

Next Step: Use GKE Autopilot to simplify security.

Scenario 4: Custom Authentication Required

Your Situation: You need LDAP, Kerberos, or other custom authentication not available on platforms.

What You Need:

- Custom authentication provider
- Custom API endpoints
- Advanced logging
- Specific business logic

Recommendation: ⚙️ **Tutorial 23 Custom Server + Cloud Run**

Why:

- Cloud Run provides platform security
- Tutorial 23 provides custom authentication
- Combined = secure + custom

Deploy:

```
# 1. Use custom server from Tutorial 23
cd tutorial_implementation/tutorial23

# 2. Deploy to Cloud Run
adk deploy cloud_run \
  --project your-project-id \
  --region us-central1
```

Cost: ~$60/month (on Cloud Run) + custom server complexity

Note: **MOST USERS DON'T NEED THIS**

- Use Cloud Run IAM for standard authentication
- Use Agent Engine OAuth for standards
- Only use this if platforms don't support your auth method

Effort: 2+ hours to implement custom server

Scenario 5: Local Development

Your Situation: Building and testing locally before deploying.

What You Need:

- Fast iteration loop
- Hot reload on code changes
- Easy testing
- No infrastructure needed

Recommendation: ⚡ **Local Dev (add security before deploy)**

Why:

- Zero setup time
- Instant feedback
- Free
- Perfect for development

Run Locally:

```
# Start dev server
adk api_server

# Or use custom server
python -m uvicorn production_agent.server:app --reload
```

Before Production:

- Add authentication layer
- Test with HTTPS (use ngrok)
- Verify security settings
- Move to Cloud Run

Cost: Free (local)

Next Step: Deploy to Cloud Run when ready for production.

Path 1: Simple Deployment (Recommended)

5-Minute Quick Start with ADK's Built-In Server

Want to deploy NOW? Use this command:

```
# Cloud Run
adk deploy cloud_run \
  --project your-project-id \
  --region us-central1 \
  ./your_agent_directory

# GKE
adk deploy gke \
  --project your-project-id \
  --cluster_name my-cluster \
  --region us-central1 \
  ./your_agent_directory

# Agent Engine
adk deploy agent_engine \
  --project your-project-id \
  --region us-central1 \
  ./your_agent_directory
```

✔️ **That's it!** Your agent is live in 5 minutes.

What you get:

- Automatic container build
- FastAPI server with basic endpoints
- Auto-scaling
- Public HTTPS URL
- Session management
- `/health` endpoint
- No custom code needed

🏗️ Advanced: When You Need a Custom FastAPI Server

⚠️ Important: Most Users Don't Need This

First Check: Do you actually need a custom server?

- ✔️ **Use Cloud Run + ADK's built-in** if you need standard authentication (IAM, OAuth)
- ✔️ **Use Agent Engine** if you need compliance/security
- ✔️ **Use GKE** if you need Kubernetes control
- ⚙️ **Use Custom Server** ONLY if you have special needs below

When Custom Server is Actually Needed

You need Tutorial 23's custom server IF:

1. **Custom authentication** (LDAP, Kerberos, API keys)
2. Cloud Run IAM doesn't support it
3. Agent Engine OAuth doesn't work for you
4. You have proprietary auth system
5. **Advanced logging/observability** beyond platform defaults
6. Custom request correlation IDs
7. Business event tracking
8. Custom metrics
9. **Additional API endpoints** for business logic
10. Webhooks
11. Custom health checks
12. Integration endpoints
13. **Non-Google infrastructure**
14. Running on AWS, Azure, on-premises
15. Portable solution needed

If none of these apply: Use Cloud Run or Agent Engine. Much simpler.

What Tutorial 23 Provides

This tutorial includes a **complete, production-ready implementation**:

```
tutorial23/
├── production_agent/
│   ├── agent.py              # Agent with 3 tools
│   └── server.py             # FastAPI server (488 lines)
├── tests/                    # 40 comprehensive tests
├── Makefile                  # Commands: setup, dev, test, demo
├── FASTAPI_BEST_PRACTICES.md # 7 core patterns guide
└── README.md                 # Complete documentation
```

Key Features (If You Need Custom Server):

- ✔️ Custom authentication with API keys
- ✔️ Structured logging with request tracing
- ✔️ Health checks with real metrics
- ✔️ Error handling and validation
- ✔️ Request timeouts and circuit breaking
- ✔️ 40 passing tests (93% coverage)
- ✔️ Production-ready patterns

📖 **Full Implementation**: View on GitHub → (https://github.com/raphaelmansuy/adk_training/tree/main/tutorial_implementation/tutorial23)

Security Note: Tutorial 23 is ADVANCED pattern. It adds application-layer features but depends on platform-layer security from Cloud Run or your infrastructure.

Quick Start (5 minutes)

```
cd tutorial_implementation/tutorial23

# Setup
make setup

# Run development server

make dev

# Run tests
make test

# See demos
make demo-info
```

Open `http://localhost:8000` and select `production_deployment_agent` from dropdown.

Deployment Strategies

ADK supports multiple deployment paths. Choose based on your needs:

Comparison Matrix

Strategy	Setup Time	Scaling	Cost	Best For
Local	< 1 min	Manual	Free	Development
Cloud Run	5 mins	Auto	Pay-per-use	Most apps
Agent Engine	10 mins	Auto	Pay-per-use	Enterprise
GKE	20 mins	Manual	Hourly	Complex

1. Local Development

Perfect for: Quick testing and iteration

```
# Start FastAPI server
adk api_server

# Custom port
adk api_server --port 8090
```

Test it:

```
curl http://localhost:8080/health
curl -X POST http://localhost:8080/invoke \
  -H "Content-Type: application/json" \
  -d '{"query": "Hello!"}'
```

Features:

- 🔄 Hot reload during development
- 📖 Auto-generated API docs at `/docs`
- ⚡ Instant feedback loop

See tutorial implementation (https://github.com/raphaelmansuy/adk_training/tree/main/tutorial_implementation/tutorial23) for custom server code.

---

2. Cloud Run (Recommended for Most Apps)

Perfect for: Serverless auto-scaling with minimal ops

```
# Deploy in one command
adk deploy cloud_run \
  --project your-project-id \
  --region us-central1 \
  --service-name my-agent
```

That's it! ADK handles:

- ✔️ Building container image
- ✔️ Pushing to Container Registry
- ✔️ Deploying to Cloud Run
- ✔️ Setting up auto-scaling

Manual Alternative:

```
# 1. Build
gcloud builds submit --tag gcr.io/YOUR_PROJECT/agent

# 2. Deploy
gcloud run deploy agent \
  --image gcr.io/YOUR_PROJECT/agent \
  --platform managed \
  --region us-central1 \
  --memory 2Gi \
  --max-instances 100
```

Cost: ~$0.40 per million requests + compute

3. Vertex AI Agent Engine

Perfect for: Managed agent infrastructure with built-in versioning

```
# Deploy to managed service
adk deploy agent_engine \
  --project your-project-id \
  --region us-central1 \
  --agent-name my-agent
```

Benefits:

- 📦 Managed infrastructure
- 🎯 Version control
- 🔄 A/B testing
- 📊 Built-in monitoring

- 🔐 Enterprise security

4. Google Kubernetes Engine (GKE)

Perfect for: Complex deployments needing full control

```
# Create cluster
gcloud container clusters create agent-cluster \
  --region us-central1 \
  --machine-type n1-standard-2 \
  --num-nodes 3

# Get credentials
gcloud container clusters get-credentials agent-cluster \
  --region us-central1

# Deploy
kubectl apply -f deployment.yaml
```

When to use GKE:

- Need advanced networking
- Running multiple services
- Existing Kubernetes expertise
- Custom orchestration requirements

See tutorial implementation for full Kubernetes manifests.

Deployment Flow Diagram

```
YOUR AGENT CODE
       |
       v
+--------------------+
| adk deploy XXXX    |
+--------------------+
       |
       +-------+-------+-------+-------+
       |       |       |       |       |
       v       v       v       v       v
     LOCAL  CLOUD-RUN  AGENT-ENG  GKE  CUSTOM
       |       |         |        |      |
       v       v         v        v      v
  localhost  serverless  managed  k8s  your-infra
```

Production Setup

Environment Configuration

Create `.env` file (never commit!):

```
# Google Cloud
GOOGLE_CLOUD_PROJECT=your-project-id
GOOGLE_CLOUD_LOCATION=us-central1
GOOGLE_GENAI_USE_VERTEXAI=1

# Application
MODEL=gemini-2.0-flash
TEMPERATURE=0.5
MAX_TOKENS=2048

# Security
API_KEY=your-secret-key
ALLOWED_ORIGINS=https://yourdomain.com

# Monitoring
LOG_LEVEL=INFO
ENABLE_TRACING=true
```

Health Checks

All deployments should expose `/health` endpoint:

```
GET /health

{
  "status": "healthy",
  "uptime_seconds": 3600,
  "request_count": 1250,
  "error_count": 3,
  "error_rate": 0.0024,
  "metrics": {
    "successful_requests": 1247,
    "timeout_count": 0
  }
}
```

Configure in orchestrator:

- **Cloud Run**: Automatically detected
- **GKE**: Set as liveness probe
- **Agent Engine**: Built-in

Secrets Management

Never commit API keys to code. Use Google Secret Manager:

```python
from google.cloud import secretmanager

def get_secret(secret_id: str) -> str:
    client = secretmanager.SecretManagerServiceClient()
    project = os.environ['GOOGLE_CLOUD_PROJECT']
    name = f"projects/{project}/secrets/{secret_id}/versions/latest"
    response = client.access_secret_version(request={"name": name})
    return response.payload.data.decode('UTF-8')

# Usage
api_key = get_secret('api-key')
```

Monitoring & Observability

Key Metrics to Track

Metric	Target	Alert Threshold
Error Rate	< 0.5%	> 5%
P99 Latency	< 2 sec	> 5 sec
Availability	> 99.9%	< 99%
Request Count	Track	N/A

Structured Logging

All production servers should log JSON to stdout:

```json
{
  "timestamp": "2025-01-17T10:30:45Z",
  "severity": "INFO",
  "message": "invoke_agent.success",
  "request_id": "550e8400-e29b",
  "tokens": 245,
  "latency_ms": 1230
}
```

Cloud Logging automatically parses and indexes these fields.

💰 Cost Breakdown: Choose Based on Budget

Monthly Cost Estimates (at 1M requests/month)

Platform	Base	Per-Request	Setup	Monthly Total	Best For
Cloud Run	$0	~$0.40	5 min	~$40	Most apps
Agent Engine	$0	~$0.50	10 min	~$50	Enterprise
GKE	$50+	Varies	20 min	$200-500+	Complex
Custom + Cloud Run	$0	~$0.40	2 hrs	~$60	Special needs
Local Dev	$0	$0	< 1 min	$0	Development

Notes:

- Costs based on US pricing (may vary by region)
- Includes compute + storage estimates

- Actual costs depend on model, memory, CPU usage
- Agent Engine includes managed infrastructure overhead
- GKE includes cluster base cost + node costs

ROI Analysis:

- **Startup**: Start with Cloud Run ($40/mo), move to Agent Engine ($50/mo) if compliance needed
- **Enterprise**: Start with Agent Engine ($50/mo), includes compliance
- **Existing K8s**: Use GKE ($200+/mo), leverages existing infrastructure

✔️ Deployment Verification: How to Verify It Works

After Deploying to Cloud Run

```
# 1. Get your service URL
SERVICE_URL=$(gcloud run services describe my-agent \
  --region us-central1 \
  --format 'value(status.url)')

# 2. Test health endpoint
curl $SERVICE_URL/health

# 3. Test agent invocation
curl -X POST $SERVICE_URL/invoke \
  -H "Content-Type: application/json" \
  -d '{"query": "Hello agent!", "temperature": 0.5}'

# 4. Check metrics
curl $SERVICE_URL/health | jq '.metrics'
```

After Deploying to Agent Engine

```
# Agent Engine dashboard: https://console.cloud.google.com/vertex-ai/
# Check:
# - ✔️ Agent deployed
# - ✔️ Endpoints responding
# - ✔️ Invocation successful
# - ✔️ Audit logs appearing
```

Security Verification Checklist

- [ ] HTTPS/TLS working (curl shows https://)
- [ ] Authentication enabled (get 401 on unauthenticated call)
- [ ] CORS configured (check headers)
- [ ] Health check responding (GET /health)
- [ ] Logging to Cloud Logging (check console)
- [ ] No API keys in logs (verify secrets not exposed)
- [ ] Request timeouts working (test long-running query)
- [ ] Error handling working (test invalid input)

See: DEPLOYMENT_CHECKLIST.md (https://github.com/raphaelmansuy/adk_training/blob/main/tutorial_implementation/tutorial23/DEPLOYMENT_CHECKLIST.md) for complete verification steps.

✨ Best Practices for Production Deployment

🔐 Security (Platform Provides Most of This Automatically)

What Cloud Run/Agent Engine Provides Automatically:

- ✔️ HTTPS/TLS encryption (handled by platform)
- ✔️ DDoS protection (included)
- ✔️ Encryption at rest (Google-managed)

- ✔️ Non-root container execution (enforced)
- ✔️ Binary vulnerability scanning (included)

What You Must Configure:

- [ ] Use Secret Manager for API keys (never hardcode)
- [ ] Enable authentication in Cloud Run console
- [ ] Configure CORS with specific origins (never use wildcard `*`)
- [ ] Set resource limits (memory, CPU)
- [ ] Store secrets in Secret Manager (not .env)
- [ ] Monitor error rates and latency

For Custom Server:

- [ ] Implement request authentication (see Tutorial 23 examples)
- [ ] Use Bearer token validation
- [ ] Implement timeout protection
- [ ] Validate input sizes
- [ ] Handle errors securely (don't expose internals)

📊 Observability

- [ ] Export logs to Cloud Logging
- [ ] Set up error tracking with Error Reporting
- [ ] Monitor metrics with Cloud Monitoring
- [ ] Use request IDs for tracing
- [ ] Log important business events

⚡ Reliability

- [ ] Set request timeouts (30s recommended)
- [ ] Implement health checks
- [ ] Configure auto-scaling appropriately
- [ ] Use load balancing
- [ ] Plan for disaster recovery

📈 Performance

- [ ] Use connection pooling
- [ ] Stream responses when possible
- [ ] Cache agent configuration
- [ ] Monitor memory usage
- [ ] Use multiple workers

FastAPI Best Practices

This implementation demonstrates **7 core production patterns**:

1. **Configuration Management** - Environment-based settings
2. **Authentication & Security** - Bearer token validation
3. **Health Checks** - Real metrics-based status
4. **Request Lifecycle** - Timeout protection
5. **Error Handling** - Typed exceptions
6. **Logging & Observability** - Request tracing
7. **Metrics & Monitoring** - Observable systems

📖 **Full Guide**: FastAPI Best Practices for ADK Agents → (https://github.com/raphaelmansuy/adk_training/blob/main/tutorial_implementation/tutorial23/FASTAPI_BEST_PRACTICES.md)

This guide includes:

- ✓ Code examples for each pattern
- ✓ ASCII diagrams showing flows
- ✓ Production checklist
- ✓ Common pitfalls (❌ Don't / ✓ Do)
- ✓ Deployment examples

Common Patterns

Pattern: Gradual Rollout

```
Deploy to Cloud Run
       |
       v
Traffic: 5% (canary)
       |
       v
Monitor for 1 hour
       |
       +------ Error Rate High? -----> ROLLBACK
       |
       +------ Healthy? --------> 25% traffic
                                  |
                                  v
                               Monitor
                                  |
                                  +---> 100% traffic
```

Pattern: Zero-Downtime Deployment

Blue-Green Deployment:

```
CURRENT (Blue)          NEW (Green)
   |                        |
   +----> BOTH ACTIVE <-----+
   |           |            |
   +--- LB routes traffic ---+
   |                        |
   +-- Health checks OK? ---|
           |                |
         YES                NO
           |                |
           v                v
        Blue OFF       Rollback (Blue ON)
        Green ON           Green OFF
```

Troubleshooting

Agent Not Found in Dropdown

Problem: `adk web agent_name` fails

Solution: Install as package first

```
pip install -e .
adk web  # Then select from dropdown
```

`GOOGLE_API_KEY Not Set`

```
# Or in Cloud Run: Set env var in Cloud Console
```

High Latency

Check:

1. Request timeout setting
2. Agent complexity (use streaming)
3. Resource limits (increase CPU)
4. Model selection (try `gemini-2.0-flash`)

Memory Issues

- Reduce max_tokens
- Enable request streaming
- Use connection pooling
- Monitor with Cloud Profiler

Quick Reference

CLI Commands

```
# Local
adk api_server --port 8080

# Deploy
adk deploy cloud_run --project PROJECT --region REGION
adk deploy agent_engine --project PROJECT --region REGION
adk deploy gke

# List deployments
adk list deployments
```

Environment Variables

```
GOOGLE_CLOUD_PROJECT       # GCP project ID
GOOGLE_CLOUD_LOCATION      # Region (us-central1)
GOOGLE_GENAI_USE_VERTEXAI  # Use Vertex AI (1 or 0)
MODEL                      # Model name
API_KEY                    # Secret key for auth
REQUEST_TIMEOUT            # Timeout in seconds
```

Endpoints

```
GET  /                  # API info
GET  /health            # Health check + metrics
POST /invoke            # Agent invocation
GET  /docs              # OpenAPI docs
```

Summary

You now know:

- ✔️ Deploy locally for development
- ✔️ Deploy to Cloud Run for most production apps
- ✔️ Use Agent Engine for managed infrastructure
- ✔️ Use GKE for complex deployments
- ✔️ Configure and secure production systems
- ✔️ Monitor and observe agent systems
- ✔️ Implement reliability patterns

Deployment Checklist:

- [ ] Environment variables configured
- [ ] Secrets in Secret Manager
- [ ] Health checks working
- [ ] Monitoring/logging setup
- [ ] Auto-scaling configured
- [ ] CORS properly configured
- [ ] Rate limiting enabled
- [ ] Error handling tested
- [ ] Disaster recovery planned

Next Steps:

- **Tutorial 24**: Advanced Observability (./24_advanced_observability.md) - Deep observability patterns
- **Tutorial 25**: Best Practices & Patterns (./25_best_practices.md) - Production patterns
- 🚀 Deploy your own agent to production!

Supporting Resources

Comprehensive Guides

- 🔐 Security Verification Guide → (https://github.com/raphaelmansuy/adk_training/blob/main/tutorial_implementation/tutorial23/SECURITY_VERIFICATION.md) - Step-by-step verification for each platform
- 🚀 Migration Guide → (https://github.com/raphaelmansuy/adk_training/blob/main/tutorial_implementation/tutorial23/MIGRATION_GUIDE.md) - Safe migration between all platforms
- 💰 Cost Breakdown Analysis → (https://github.com/raphaelmansuy/adk_training/blob/main/tutorial_implementation/tutorial23/COST_BREAKDOWN.md) - Detailed pricing for budget planning
- ✔️ Deployment Checklist → (https://github.com/raphaelmansuy/adk_training/blob/main/tutorial_implementation/tutorial23/DEPLOYMENT_CHECKLIST.md) - Pre/during/post deployment verification

Security Research

- 📋 Security Research Summary → (https://github.com/raphaelmansuy/adk_training/blob/main/SECURITY_RESEARCH_SUMMARY.md) - Executive summary of platform security
- 🔍 Detailed Security Analysis → (https://github.com/raphaelmansuy/adk_training/blob/main/SECURITY_ANALYSIS_ALL_DEPLOYMENT_OPTIONS.md) - Per-platform security breakdown

Additional Resources

- 📚 Tutorial Implementation → (https://github.com/raphaelmansuy/adk_training/tree/main/tutorial_implementation/tutorial23)
- 📖 FastAPI Best Practices Guide → (https://github.com/raphaelmansuy/adk_training/blob/main/tutorial_implementation/tutorial23/FASTAPI_BEST_PRACTICES.md)
- 🌐 Cloud Run Docs (https://cloud.google.com/run/docs)
- 🤖 Agent Engine Docs (https://cloud.google.com/vertex-ai/docs/agent-engine)
- ⚙️ GKE Docs (https://cloud.google.com/kubernetes-engine/docs)
- 🔐 Secret Manager (https://cloud.google.com/secret-manager/docs)

🎉 **Tutorial 23 Complete!** You're now ready to deploy agents to production. Proceed to Tutorial 24 for advanced observability.

Generated on 2025-10-21 09:02:49 from 23_production_deployment.md

Source: Google ADK Training Hub