# Google Cloud Run Deployment Guide

## Prerequisites

1. **Google Cloud Account** with billing enabled
2. **Google Cloud SDK** installed locally
3. **Docker** installed locally
4. **Project setup** in Google Cloud Console

## Step-by-Step Deployment

### 1. Setup Google Cloud Project

```bash
# Login to Google Cloud
gcloud auth login

# Set your project ID
export PROJECT_ID="your-project-id"
gcloud config set project $PROJECT_ID

# Enable required APIs
gcloud services enable cloudbuild.googleapis.com
gcloud services enable run.googleapis.com
gcloud services enable containerregistry.googleapis.com
```

### 2. Prepare Your Application

Create the following files in your project root:

- `Dockerfile` (provided above)
- `requirements.txt` (provided above)
- `.dockerignore` (provided above)
- `.env.example` (for environment variables)

### 3. Environment Variables Setup

Create a `.env.example` file:

```bash
# API Keys
OPENAI_API_KEY=your_openai_key_here
ANTHROPIC_API_KEY=your_anthropic_key_here

# Database
CHROMA_PERSIST_DIRECTORY=/app/data/chroma

# Workflow Config
MAX_ITERATIONS=3
DEFAULT_URL=https://example.com

# Logging
LOG_LEVEL=INFO
```

## 4. Build and Deploy Options

### Option A: Direct Deployment (Recommended)

```bash
# Deploy directly from source
gcloud run deploy ai-book-publisher \
  --source . \
  --platform managed \
  --region us-central1 \
  --allow-unauthenticated \
  --port 8080 \
  --cpu 2 \
  --memory 4Gi \
  --timeout 3600 \
  --max-instances 10 \
  --set-env-vars="PYTHONUNBUFFERED=1,PORT=8080"
```

### Option B: Build and Push to Container Registry

bash

```
# Build the Docker image
docker build -t gcr.io/$PROJECT_ID/ai-book-publisher .

# Push to Google Container Registry
docker push gcr.io/$PROJECT_ID/ai-book-publisher

# Deploy to Cloud Run
gcloud run deploy ai-book-publisher \
  --image gcr.io/$PROJECT_ID/ai-book-publisher \
  --platform managed \
  --region us-central1 \
  --allow-unauthenticated \
  --port 8080 \
  --cpu 2 \
  --memory 4Gi \
  --timeout 3600 \
  --max-instances 10
```

## 5. Set Environment Variables

bash

```
# Set environment variables after deployment
gcloud run services update ai-book-publisher \
  --region us-central1 \
  --set-env-vars="OPENAI_API_KEY=your_key,ANTHROPIC_API_KEY=your_key"
```

## 6. Configure Service Settings

bash

```
# Update service configuration
gcloud run services update ai-book-publisher \
  --region us-central1 \
  --cpu 2 \
  --memory 4Gi \
  --timeout 3600 \
  --concurrency 80 \
  --max-instances 10 \
  --min-instances 0
```

# Advanced Configuration

## Custom Domain Setup

bash

```
# Map custom domain
gcloud run domain-mappings create \
  --service ai-book-publisher \
  --domain your-domain.com \
  --region us-central1
```

## Service Account Setup

bash

```
# Create service account
gcloud iam service-accounts create ai-book-publisher-sa \
  --display-name "AI Book Publisher Service Account"

# Deploy with service account
gcloud run deploy ai-book-publisher \
  --service-account ai-book-publisher-sa@$PROJECT_ID.iam.gserviceaccount.com \
  --region us-central1
```

## VPC Connector (if needed)

bash

```
# Create VPC connector for private resources
gcloud compute networks vpc-access connectors create ai-book-connector \
  --region us-central1 \
  --subnet default \
  --min-instances 2 \
  --max-instances 10

# Deploy with VPC connector
gcloud run deploy ai-book-publisher \
  --vpc-connector ai-book-connector \
  --region us-central1
```

# Monitoring and Logging

## View Logs

```bash
# View service logs
gcloud run services logs tail ai-book-publisher --region us-central1

# View specific logs
gcloud logs tail "resource.type=cloud_run_revision AND resource.labels.service_name=ai-book-publisher"
```

### Monitoring Setup

```bash
# Create uptime check
gcloud monitoring uptime create ai-book-publisher-check \
  --display-name "AI Book Publisher Uptime" \
  --http-check-path "/_stcore/health" \
  --hostname your-service-url.run.app
```

## Troubleshooting

### Common Issues and Solutions

1. **Memory Issues**

```bash
# Increase memory allocation
gcloud run services update ai-book-publisher \
  --memory 8Gi --region us-central1
```

2. **Timeout Issues**

```bash
# Increase timeout
gcloud run services update ai-book-publisher \
  --timeout 3600 --region us-central1
```

3. **Cold Start Issues**

```bash
# Set minimum instances
gcloud run services update ai-book-publisher \
  --min-instances 1 --region us-central1
```

### Debug Commands

```bash
# Get service details
gcloud run services describe ai-book-publisher --region us-central1

# List all revisions
gcloud run revisions list --service ai-book-publisher --region us-central1

# Check service URL
gcloud run services list --filter="metadata.name=ai-book-publisher"
```

## Security Best Practices

1. **Use Secret Manager for sensitive data**
2. **Enable authentication if needed**
3. **Implement proper IAM roles**
4. **Use VPC connectors for private resources**
5. **Regular security updates**

## Cost Optimization

- Use `--min-instances 0` for cost efficiency
- Set appropriate `--max-instances` based on expected load
- Monitor usage with Google Cloud Billing alerts
- Use `--cpu-throttling` for cost-sensitive workloads

## Deployment Scripts

Create a `deploy.sh` script for easy deployment:

bash

```
#!/bin/bash
export PROJECT_ID="your-project-id"
export SERVICE_NAME="ai-book-publisher"
export REGION="us-central1"

gcloud run deploy $SERVICE_NAME \
  --source . \
  --platform managed \
  --region $REGION \
  --allow-unauthenticated \
  --port 8080 \
  --cpu 2 \
  --memory 4Gi \
  --timeout 3600 \
  --max-instances 10 \
  --set-env-vars="PYTHONUNBUFFERED=1,PORT=8080"
```

Make it executable: `chmod +x deploy.sh`