

Google Cloud Platform Setup - Complete Walkthrough

For Genesis Provenance AI Authentication

Overview

This comprehensive guide walks you through creating a Google Cloud Platform (GCP) account and configuring it for Genesis Provenance AI authentication. By the end, you'll have:

- ✓ GCP account with \$300 free credits
- ✓ Project configured with Vision API enabled
- ✓ Service account with secure credentials
- ✓ Budget alerts to control costs
- ✓ gcloud CLI installed and configured

Estimated Time: 30-45 minutes

Cost: \$0 (using free tier credits)

Part 1: Create Your Google Cloud Account

Prerequisites

- Google account (Gmail or Google Workspace)
- Credit card for verification (won't be charged)
- Phone number for SMS verification

Step-by-Step Process

1. Visit Google Cloud:

```
https://console.cloud.google.com/
```

Or start the free trial:

```
https://cloud.google.com/free
```

2. Click "Get Started for Free" or "Try for free"

3. Sign in with your Google account

4. Fill in Business Information:

Organization name: Genesis Provenance
 Business type: Small business / Startup
 Country: United States (or your location)
 Business email: [your email]

5. Add Payment Information:

- Add credit card (verification only)
- You get **\$300 free credits** for 90 days
- **Won't be charged** unless you manually upgrade
- All Vision API testing is covered by free credits

6. Complete Verification:

- Verify phone number via SMS
- Confirm email address
- Accept Terms of Service

What You Get

- ✓ **\$300 in free credits** (valid for 90 days)
- ✓ **Access to all GCP services** including Vision AI
- ✓ **No automatic charges** - requires manual upgrade
- ✓ **Perfect for testing** Genesis Provenance AI features

Part 2: Install gcloud CLI

For Linux/Ubuntu (DeepAgent Environment)

```
# 1. Add Cloud SDK distribution URI
echo "deb [signed-by=/usr/share/keyrings/cloud.google.gpg] https://packages.cloud.google.com/apt cloud-sdk main" | sudo tee -a /etc/apt/sources.list.d/google-cloud-sdk.list

# 2. Import Google Cloud public key
curl https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo apt-key --keyring /usr/share/keyrings/cloud.google.gpg add -

# 3. Update and install
sudo apt-get update && sudo apt-get install google-cloud-cli

# 4. Verify installation
gcloud --version
```

Expected Output:

```
Google Cloud SDK 458.0.0
bq 2.0.101
core 2024.01.12
gcloud-crc32c 1.0.0
gsutil 5.27
```

For macOS

```
# Using Homebrew (recommended)
brew install --cask google-cloud-sdk

# Verify
gcloud --version
```

For Windows

1. Download installer: <https://cloud.google.com/sdk/docs/install#windows>
2. Run `GoogleCloudSDKInstaller.exe`
3. Follow installation wizard
4. Verify in PowerShell: `gcloud --version`

Part 3: Initialize gcloud CLI

Login to Google Cloud

```
# Initialize gcloud (opens browser for authentication)
gcloud init
```

Interactive Prompts:

1. **Pick configuration:** Choose `[1] Create a new configuration`
2. **Configuration name:** Enter `genesis-provenance`
3. **Choose account:** Select your Google account or login
4. **Browser opens:** Click “Allow” to grant gcloud access
5. **Return to terminal:** Configuration complete

Verify Login

```
# Check current account
gcloud auth list
```

Expected Output:

```
Credentialed Accounts
ACTIVE ACCOUNT
*       your-email@gmail.com
```

Part 4: Create GCP Project

Method 1: Using gcloud CLI (Recommended)

```
# Set project ID (must be globally unique)
PROJECT_ID="genesis-provenance-ai"

# Create the project
gcloud projects create $PROJECT_ID \
  --name="Genesis Provenance AI" \
  --set-as-default
```

Expected Output:

```
Create in progress for [https://cloudresourcemanager.googleapis.com/v1/projects/genesis-provenance-ai].
Waiting for [operations/cp.1234567890] to finish...done.
```

If Project ID already exists:

```
# Use a unique suffix
PROJECT_ID="genesis-provenance-ai-prod"
# or
PROJECT_ID="genesis-provenance-ai-$(date +%s)"
```

Verify Creation:

```
gcloud projects list
```

Expected Output:

PROJECT_ID	NAME	PROJECT_NUMBER
genesis-provenance-ai	Genesis Provenance AI	123456789012

Method 2: Using Google Cloud Console

1. Visit: <https://console.cloud.google.com/projectcreate>
 2. Fill in:
 - **Project name:** Genesis Provenance AI
 - **Project ID:** genesis-provenance-ai
 - **Organization:** No organization (or your org)
 3. Click **“Create”**
 4. Wait 10-30 seconds for creation
 5. Verify in CLI: `gcloud projects list`
-

Part 5: Set Default Project

```
# Set as default project
gcloud config set project genesis-provenance-ai

# Verify
gcloud config get-value project
```

Expected Output:

```
genesis-provenance-ai
```

Part 6: Enable Billing

Using Console (Easier)

1. Go to: <https://console.cloud.google.com/billing/linkedaccount>
2. Select project: **genesis-provenance-ai**
3. Click **“Link a billing account”**
4. Select your billing account (created during signup)
5. Click **“Set account”**

Using gcloud CLI

```
# List billing accounts
gcloud billing accounts list
```

Expected Output:

ACCOUNT_ID	NAME	OPEN	MASTER_ACCOUNT_ID
01AB23-CD45EF-67890A	My Billing Account	True	

```
# Link billing account to project
BILLING_ACCOUNT_ID="01AB23-CD45EF-67890A" # Replace with your ID

gcloud billing projects link genesis-provenance-ai \
  --billing-account=$BILLING_ACCOUNT_ID
```

Verify Billing:

```
gcloud billing projects describe genesis-provenance-ai
```

Look for: `billingEnabled: true`

Part 7: Enable Vision API

```
# Enable Cloud Vision API
gcloud services enable vision.googleapis.com
```

Expected Output:

```
Operation "operations/acat.p2-123456789012-abcdef" finished successfully.
```

Verify Vision API:

```
gcloud services list --enabled | grep vision
```

Expected Output:

```
vision.googleapis.com    Cloud Vision API
```

Enable Additional APIs:

```
# Enable supporting APIs
gcloud services enable \
  cloudresourcemanager.googleapis.com \
  storage.googleapis.com \
  iam.googleapis.com
```

Part 8: Create Service Account

Step 8.1: Create Service Account

```
# Set variables
SA_NAME="genesis-vision-ai"
SA_DISPLAY_NAME="Genesis Provenance Vision AI"
PROJECT_ID=$(gcloud config get-value project)

# Create service account
gcloud iam service-accounts create $SA_NAME \
  --display-name="$SA_DISPLAY_NAME" \
  --description="Service account for luxury asset authentication"
```

Expected Output:

```
Created service account [genesis-vision-ai].
```

Get Service Account Email:

```
SA_EMAIL="$SA_NAME@$PROJECT_ID.iam.gserviceaccount.com"
echo $SA_EMAIL
```

Verify:

```
gcloud iam service-accounts list
```

Expected Output:

DISPLAY NAME	EMAIL
Genesis Provenance Vision AI	genesis-vision-ai@genesis-provenance-ai.iam...

Step 8.2: Grant Permissions

```
# Grant Cloud Vision permissions
gcloud projects add-iam-policy-binding $PROJECT_ID \
  --member="serviceAccount:$SA_EMAIL" \
  --role="roles/cloudvision.admin"
```

Expected Output:

```
Updated IAM policy for project [genesis-provenance-ai].
bindings:
- members:
  - serviceAccount:genesis-vision-ai@genesis-provenance-ai.iam...
  role: roles/cloudvision.admin
```

Step 8.3: Create and Download Key

```
# Create config directory
mkdir -p /home/ubuntu/genesis_provenance/nextjs_space/config

# Create and download JSON key
gcloud iam service-accounts keys create \
  /home/ubuntu/genesis_provenance/nextjs_space/config/genesis-vision-key.json \
  --iam-account=$SA_EMAIL
```

Expected Output:

```
created key [abc123def456] of type [json] as [/home/ubuntu/genesis_provenance/nex-
tjs_space/config/genesis-vision-key.json]
for [genesis-vision-ai@genesis-provenance-ai.iam.gserviceaccount.com]
```

Secure the Key:

```
# Set proper permissions
chmod 600 /home/ubuntu/genesis_provenance/nextjs_space/config/genesis-vision-key.json

# Add to .gitignore
echo "config/genesis-vision-key.json" >> /home/ubuntu/genesis_provenance/nex-
tjs_space/.gitignore
echo "config/*.json" >> /home/ubuntu/genesis_provenance/nextjs_space/.gitignore

# Verify
ls -lah /home/ubuntu/genesis_provenance/nextjs_space/config/genesis-vision-key.json
```

Expected Output:

```
-rw----- 1 ubuntu ubuntu 2.3K Dec  1 10:00 genesis-vision-key.json
```

Part 9: Set Budget Alerts

Using Console (Recommended)

1. Go to: <https://console.cloud.google.com/billing/budgets>
2. Click **“Create Budget”**
3. **Scope:**
 - Projects: Select `genesis-provenance-ai`
 - Services: All services
4. **Amount:**
 - Budget type: Specified amount
 - Target amount: **\$50.00**
 - Include credits: ☒ Yes
5. **Threshold Rules:**
 - Add rule: 50% (\$25.00)
 - Add rule: 90% (\$45.00)
 - Add rule: 100% (\$50.00)
6. **Actions:**
 - Email alerts: Your email
7. Click **“Finish”**

Part 10: Configure Environment Variables

```
# Navigate to project
cd /home/ubuntu/genesis_provenance/nextjs_space

# Get your project ID
PROJECT_ID=$(gcloud config get-value project)
echo "Project ID: $PROJECT_ID"
```

Update .env file:


```
# Add Google Cloud Vision configuration
cat >> .env << 'EOF'

# ===== Google Cloud Vision AI Configuration =====
GOOGLE_CLOUD_PROJECT_ID=genesis-provenance-ai
GOOGLE_APPLICATION_CREDENTIALS=/home/ubuntu/genesis_provenance/nextjs_space/config/
genesis-vision-key.json

# AI Provider Configuration
AI_PROVIDER=google-vision
USE_REAL_AI=false

# Cost Controls
AI_DAILY_LIMIT=100
AI_MONTHLY_BUDGET=50
EOF
```

Replace with actual project ID:

```
# If your project ID is different, update it:
sed -i "s/GOOGLE_CLOUD_PROJECT_ID=.*//GOOGLE_CLOUD_PROJECT_ID=$PROJECT_ID/" .env
```

Verify .env:

```
cat .env | grep GOOGLE
```

Expected Output:

```
GOOGLE_CLOUD_PROJECT_ID=genesis-provenance-ai
GOOGLE_APPLICATION_CREDENTIALS=/home/ubuntu/genesis_provenance/nextjs_space/config/
genesis-vision-key.json
```

Part 11: Test Your Setup

Quick Test: Verify Credentials

```
# Set credentials environment variable
export GOOGLE_APPLICATION_CREDENTIALS="/home/ubuntu/genesis_provenance/nextjs_space/
config/genesis-vision-key.json"

# Test authentication
gcloud auth application-default print-access-token
```

Expected: A long access token (should NOT show any errors)

Verify Project Access:

```
gcloud projects describe $(gcloud config get-value project)
```

Expected: Project details in YAML format

Part 12: Summary Checklist

✓ Account Setup

- ☐ Google Cloud account created
- ☐ \$300 free credits activated
- ☐ Billing account verified

✓ Project Setup

- ☐ Project created: `genesis-provenance-ai`
- ☐ Project set as default
- ☐ Billing linked to project

✓ API Configuration

- ☐ Vision API enabled
- ☐ Service account created
- ☐ IAM permissions granted

✓ Credentials

- ☐ Service account key downloaded
- ☐ Key file secured (`chmod 600`)
- ☐ Key path added to `.gitignore`

✓ Environment

- ☐ `.env` file updated
- ☐ `GOOGLE_CLOUD_PROJECT_ID` set
- ☐ `GOOGLE_APPLICATION_CREDENTIALS` configured

✓ Cost Management

- ☐ Budget alerts configured (\$50/month)
 - ☐ Threshold rules set (50%, 90%, 100%)
 - ☐ Email notifications enabled
-

Part 13: Quick Reference Commands

```
# Login to gcloud
gcloud auth login

# Set project
gcloud config set project genesis-provenance-ai

# Get project ID
gcloud config get-value project

# List all projects
gcloud projects list

# Check enabled APIs
gcloud services list --enabled

# View service accounts
gcloud iam service-accounts list

# Check billing status
gcloud billing projects describe $(gcloud config get-value project)

# Test credentials
gcloud auth application-default print-access-token
```

Part 14: Troubleshooting

Issue 1: “Permission Denied” when creating key

Error:

```
ERROR: (gcloud.iam.service-accounts.keys.create) Permission denied
```

Solution:

```
# Grant yourself Owner role
PROJECT_ID=$(gcloud config get-value project)
gcloud projects add-iam-policy-binding $PROJECT_ID \
  --member="user:$(gcloud config get-value account)" \
  --role="roles/owner"
```

Issue 2: “Billing not enabled”

Error:

```
ERROR: User project specified in the request is not found
```

Solution:

1. Go to: <https://console.cloud.google.com/billing/linkedaccount>
2. Link billing account manually

Issue 3: “API not enabled”

Error:

```
ERROR: API [vision.googleapis.com] not enabled
```

Solution:

```
# Enable Vision API
gcloud services enable vision.googleapis.com

# Wait for propagation
sleep 30
```

Issue 4: Key file not found

Error:

```
Error: ENOENT: no such file or directory
```

Solution:

```
# Check if file exists
ls -la /home/ubuntu/genesis_provenance/nextjs_space/config/

# If missing, re-download key
PROJECT_ID=$(gcloud config get-value project)
SA_EMAIL="genesis-vision-ai@$PROJECT_ID.iam.gserviceaccount.com"

gcloud iam service-accounts keys create \
  /home/ubuntu/genesis_provenance/nextjs_space/config/genesis-vision-key.json \
  --iam-account=$SA_EMAIL
```

Part 15: Cost Tracking

Free Tier (First 90 Days)

- **\$300 credit** covers all testing
- All Vision API calls included
- No charges during trial

After Free Tier (Production)

Label Detection: \$1.50 per 1,000 images
 Logo Detection: \$2.00 per 1,000 images
 Text Detection: \$1.50 per 1,000 images
 Web Detection: \$3.50 per 1,000 images

Typical analysis: \$0.009 per item

Monthly Estimates:

- 100 analyses: \$0.90
- 1,000 analyses: \$9.00
- 5,000 analyses: \$45.00

Monitor Costs

```
# View billing
gcloud billing accounts list

# Project billing info
gcloud billing projects describe genesis-provenance-ai

# Or visit console
# https://console.cloud.google.com/billing
```

Part 16: Next Steps

✓ You've Completed:

1. ✓ GCP account created with \$300 credits
2. ✓ Project configured: `genesis-provenance-ai`
3. ✓ Vision API enabled
4. ✓ Service account and credentials ready
5. ✓ Environment variables configured
6. ✓ Budget alerts set

▶▶ Next: Implement AI Utility

Now you're ready to proceed to **Phase 1, Step 3**:

Using DeepAgent, create the AI utility library:

1. Review the DeepAgent prompt in `/PHASE_1_AI_RESEARCH_AND_PLAN.md`
2. Use **"Prompt 1: Create Google Vision AI Utility"**
3. DeepAgent will create `/lib/ai-google-vision.ts`
4. Test with sample luxury asset images
5. Integrate into the API route

After Implementation:

1. Test with real images
2. Validate accuracy and cost
3. Deploy to production with `USE_REAL_AI=true`

4. Monitor performance and optimize

Need Help?

Official Documentation

- **Quick Start:** <https://cloud.google.com/vision/docs/quickstart>
- **Service Accounts:** <https://cloud.google.com/iam/docs/creating-managing-service-accounts>
- **gcloud CLI:** <https://cloud.google.com/sdk/gcloud/reference>

Genesis Provenance Docs

- **Phase 1 Plan:** `/PHASE_1_AI_RESEARCH_AND_PLAN.md`
- **AI Implementation Guide:** `/AI_AUTHENTICATION_IMPLEMENTATION_GUIDE.md`

Support

- **Google Cloud Support:** <https://cloud.google.com/support>
 - **Community:** <https://stackoverflow.com/questions/tagged/google-cloud-vision>
-



Congratulations!

You've successfully set up Google Cloud Platform for Genesis Provenance AI authentication. You now have:

- ✓ **\$300 in free credits** for testing
- ✓ **Vision API enabled** and ready to use
- ✓ **Secure credentials** configured
- ✓ **Budget alerts** to control costs
- ✓ **Complete environment** for AI integration

You're now ready to build the world's best luxury asset authentication system!

Document Version: 1.0

Created: December 2024

For: Genesis Provenance AI Authentication Phase 1

Status: Production Ready