# **Google Cloud Functions**

Group 5

# Introduction

"Google Cloud Functions is a lightweight compute solution for developers to create single-purpose, stand-alone functions that respond to Cloud events without the need to manage a server or runtime environment."

Google Cloud Functions is a Function as a Service (FaaS) computing platform similar to AWS Lambda.



# History: Who

Google is one of the largest internet-based companies with many innovations under its belt, including:

- 1. PageRank, algorithm for meaningful web search
- 2. Translate, help people communicate
- 3. Maps, help people find their way
- 4. AutoComplete web search, only need to remember about half of what you're looking for

# History: Why

Competition: Amazon has been the leader in Enterprise cloud solutions and Google is attempting to gain market share.

Internet of Things: provide simple event-based computations for connected devices with minimal power output.

Mobile Apps: provides scalable backend-support with optimization for Android applications using Google Firebase.

# History: How

- Beta released March 2017 to compete against Amazon Lambda.
- General release was in July 2018.

The service evolved to support:

- Google Cloud storage Triggers
- Python (Originally only Node.js)
- Cloud SQL

# Feature Summary

Google Cloud Functions is a FaaS platform.

FaaS provides a number of benefits over more traditional laaS platforms:

- Allows you to easily deploy your code as single functions.
- Infrastructure scaling is done automatically depending on demand.
- No need to manage server software or virtual machines.
- You are only charged when your functions are being used.

#### **Other Features**

- Run functions in response to events
  - Cloud Store: object creation, deletion, archiving and metadata updates.
  - Firebase: DB, Storage, Analytics, Auth.
  - Cloud Pub/Sub: Real time analytics and event processing.
  - StackDriver Logging: Changes to logs can trigger functions.
- Deploy function from Docker container with arbitrary runtime environment, dependencies and Linux distribution.
- Metrics for your functions can be viewed:
  - In the GPC (General Compute Cluster) Console.
  - Using StackDriver Monitoring.
  - API Overview Page (basic call metrics).
- StackDriver Debugger allows to inspect the state of a function.

#### **Example Use Cases**

Some use cases for Google Cloud Functions are:

- Transform and aggregate data in response to storage events
- Processing stream data with automatically scaling infrastructure
- Batch jobs benefit from concurrent execution.
- Use for backend services for mobile applications
- Monitor Database changes to ensure quality standards
- Tradition REST APIs



# **Other Example Use Cases**

Some more use cases for Google Cloud Functions include:

- Sending notifications to users in your app without a server.
- You can use to Google Cloud Functions in response to any Firebase events which include: Databases, Messaging, and Crash Reporting.
- IOT sensors capable of sending events to a Google Cloud Function that sends a text message to a homeowner if:
  - The basement floods.
  - A window is broken or a door is forced open.
  - The smoke alarm goes off.

# **Technology Advantages**

- Google Cloud Platform has a very well designed portal, this makes creating and deploying functions easy.
- Integration with Firebase which is a popular web app and mobile deployment platform.
- The Node.js Emulator allows developers to run, debug and deploy their functions locally.

# Technology Disadvantages

- Late to market in March, 2017
  - $\circ$   $\,$  Compare this with November 2014 for Lambda and March 2016 for Azure Functions.
- Only supporting NodeJS and Python.
- Competitors offer support for many other languages.

# Usability

- We all had very good first impressions.
  - Creating and deploying a function was very easy.
    - Creating "Hello World" takes only a few clicks.
  - Google's Cloud Console has an intuitive layout that looks good.
  - No need to change the default permissions or create an API Gateway to use REST.
- Compared to using Java with AWS Lambda. Node.js is significantly simpler and has fewer steps to get a function deployed and running.

# **Pricing Policy**

#### Invocation

2 million free invocations. \$0.40 per million afterwards.

Memory Provision 400,000 GB-seconds free. \$0.0000025 per GB-s afterwards.

#### **CPU Provision**

200,000 GHz-seconds free. \$0.0000100 per GHz-s afterwards.

Provisioning Sc	hemes
Memory	CPU
128MB	200 MHz
256MB	400 MHz
512MB	800 MHz
1024MB	1.4 GHz
2048MB	2.4 GHz

#### **Cost Example**

Two instances running continuously for 30 days with a function execution time of 100ms.

Calls: 51.84 million Wall clock: 5,184,000 s

CPU: 0.8 GHz Memory: 0.512 GB

CPU Consumption:	4,147,200 GHz-s
CPU Cost:	\$ 39.47
Memory Use:	2,654,208 GB-s
Memory Cost:	\$ 5.64
Invocations:	51,840,000 calls
Invocations Cost:	\$ 19.94
Total Cost:	\$ 65.04



# **Cost Comparison**

Google Cloud Functions: 2 continuous threads for 30 days = \$65.04

Google VM: 2-core VM continuous for 30 days = \$47.88

- If your demand is sequential or continuous then the VM is cheaper.
- Cloud functions offer the benefit of horizontal scaling and no server maintenance or downtime.

# Conclusions

Google offers a very similar FaaS platform to AWS Lambda.

We found it to be easier to use compared to Lambda.

They give out \$300 worth of credits to anyone who signs up so give it a try!







# **Creating a Google Cloud Function**

				O CLOS PARA		e cros Platom					~
Google Cloud Platform			n 🏞/	PI Project 👻	٩	ii 0.	ø	0	• •	3	
(···) Cloud Functions		Ove	rview	CREATE FUNCTI	1		SI	SHOW INFO PANE			
Ŧ	Filter functions								0	Columns	·
	Name ~	Region	Tripger	Runtime	Memory allocated	Executed function	Last de	ployed			
	🥝 caesarCipher	us-central1	HTTP	Node.js 6	256 MB	caesarCipher	11/28/	18, 11:24 AM	. :		
	function-1	us-central1	HTTP	Node js 8 (Beta)	256 MB	helioWorld	11/26/	18, 10:16 AM			
	function/2	us-central1	HTTP	Node js 8 (Beta)	256 MB	helioWorld	11/26/	18, 11:59 AM			
	MelloWorld	us-central1	HTTP	Node js 6	256 MB	helioWorld	11/26/	18, 11:22 AM			

									-
Google Cloud Platforn	n 🔹 API Proje	ect 👻	۹	 ۶.	•	0	٠	1	
) Cloud Functions	← Cre	eate functio	on						
Name 💿									
exampleFunction									
Memory allocated									
256 MB			-						
Trigger									
Trigger									
Trigger HTTP			•						
Trigger HTTP JRL https://us-central1-api-project- 592688302945.cloudfunctions.net/examp	pleFunction		Y						
Trigger HTTP URL https://us-central1-api-project- 952688302945.cloudfunctions.net/examp Source code	oleFunction		×						
Trigger HTTP URL https://us.central1-api-project- 52688302945.cloudfunctions.net/examp Source code Inline editor	oleFunction		•						
Trigger HTTP URL HTTP SvDrcentral1-api-project- to52688302945.cloudfunctions.net/examp Source code infere editor ZP (projektad) ZP (projektad) ZP (projektad)	oleFunction		×						
Trigger HTTP Http://uscentral1-api-project- M92688302945.cloudfunctions.net/examy Source code en Inline editor 2/P from Gloud Storage Cloud Storage	bleFunction		×						
Trigger HTTP URI URI URI Source code Infine editor I proposition Drapidod Storage Cloud Storage Clou	bleFunction		·						
Trigger HTTP URL URL Source code Finder duffer Dir global Science Herbann Source code Finder editor Dir global Sciange Coud Source repository Runtime Node ja 6	bleFunction		•						
Trigger Trigger KTTP URL URL SOurce code Infine store expository ZPI rom Gloud Storage Cloud Storage Cloud Storage Node js 6	pleFunction		•						



# **Calling your Function**

Google Cloud Platform	🐉 API P					8 Ø	0		1
(···) Cloud Functions	←	Functior	n details	🖌 EDIT	T DELETE	е сору		/ LOGS	
Triggering event									
<pre> "message": "Hello World!",     "offmet": 5 }</pre>									
3									
Test the function									
Output Hjqqt Btwqi!			Ð						
								_	



#### **Google Cloud Shell** ••• <> ۍ (۹ 0 6 0 9 0 A I 👰 DASHBOARD ACTIVITY CUSTOMIZE The Google Cloud Shell Project info # RPI APIs Google Cloud Platform status ÷ : can be used to work (api-project-692688302945) % 1x EXDME-cloudshell.txt rhub\_RCordingly\_GoogleFunctionsTest gly8cloudshell:~ (api-project-692688302945)\$ cd functions/ gly8cloudshell:~/functions (api-project-692688302945)\$ lz with your functions.

# Any Questions?