iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 9 **Issue:** IX **Month of publication:** September 2021

DOI: https://doi.org/10.22214/ijraset.2021.37986

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue IX Sep 2021- Available at www.ijraset.com

Cloud Functions Using Server less Computing

Vijay Kumar[1], Talwinder Kaur[2]
[1]*M.Tech (CSE), Indo Global College, Abhipur Mohali, Punjab, India*
[2]*Department of Computer Science Engineering, Indo Global College, Abhipur Mohali, Punjab, India*

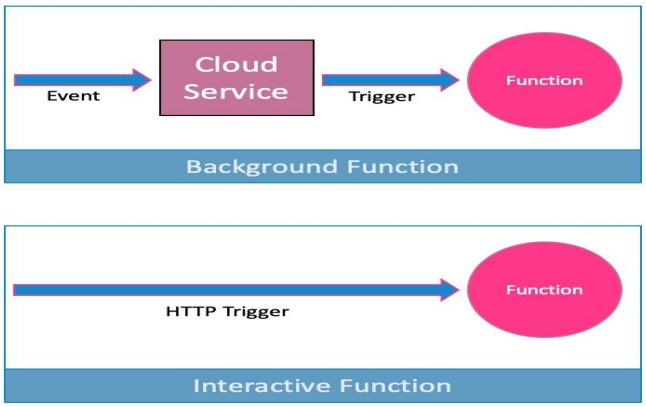
Abstract: Cloud Function in the simplest words is a Function-as-a-Service (FaaS). FaaS is actually a family of the serverless computing category. "Serverless" means that the user can focus on its application logic without dealing with infrastructure at all. Painless development, deployment, and maintenance of a web API is still not a turn-key solution, although modern web application frameworks have improved dramatically in the last few years. Serverless is without any doubt a game-changer. The event-driven approach combined with a scalable and robust cloud ecosystem offered by the main top cloud vendors opens endless opportunities. Cloud Functions is Google Cloud's event-driven serverless compute platform. FaaS is a real NoOps technology, it completely abstracts away servers. Cloud Functions are developed with the sole purpose to build event-driven architectures. These can react to events like file changes in storage, messages in the queue, or any HTTP request.
Index Terms: Cloud Computing, Serverless Technology, Function as a Service, Lambda Functions and Functions

I. INTRODUCTION

Cloud Functions is a serverless execution environment for building and connecting cloud services. With Cloud Functions you write simple, single-purpose functions that are attached to events emitted from your cloud infrastructure and services. Your function is triggered when an event being watched is fired. Your code executes in a fully managed environment. There is no need to provision any infrastructure or worry about managing any servers. Cloud Functions can be written using JavaScript, Python 3, Go, or Java runtimes on Google Cloud Platform. You can take your function and run it in any standard Node.js (Node.js 10 or 12), Python 3 (Python 3.7 or 3.8), Go (Go 1.11 or 1.13) or Java (Java 11) environment, which makes both portability and local testing a breeze.
Cloud Functions provides a connective layer of logic that lets you write code to connect and extend cloud services. Listen and respond to a file upload to Cloud Storage, a log change, or 418an incoming message on a Pub/Sub topic. Cloud Functions augments existing cloud services and allows you to address an increasing number of use cases with arbitrary programming logic. Cloud Functions have access to the Google Service Account credential and are thus seamlessly authenticated with the majority of Google Cloud services, including Cloud Vision, as well as many others.

II. LITERATURE WORK

With Cloud Functions, there are no servers to provision, manage, patch, or update. Features are automatically scaled and highly available and fault tolerant. Cloud capabilities are ideal for building serverless backends, processing real-time data, and building smart applications. Cloud events are *things* that happen in your cloud environment. These might be things like changes to data in a database, files added to a storage system, or a new virtual machine instance being created.
Events occur whether or not you choose to respond to them. You create a response to an event with a *trigger*. A trigger is a declaration that you are interested in a certain event or set of events. Binding a function to a trigger allows you to capture and act on events. For more information on creating triggers and associating them with your functions



©IJRASET: All Rights are Reserved

III. FUNCTION-AS-A-SERVICE

This section provides an overview of cloud functions that was built on Goggle's Cloud Functions. In functions, we can define environment variables, configuration types, trigger types and then execute on serverless platform.

A. Basic Steps

1) Click Create function.
2) Name your function.
3) In the Trigger field, select HTTP.
4) In the Authentication field, select Allow unauthenticated invocations.
5) Click Save to save your changes, and then click Next.
6) In the Source code field, select Inline editor. In this exercise, you will use the default function provided in the editor.
7) Use the Runtime dropdown to select the desired Node.js runtime.
8) At the bottom of the page, click **Deploy**.
9) On the testing page, click **Test the function**.

B. Approach

1) Define environment variables for related information like API details, database connection, date and time check etc.
2) Add appropriate dependency in package.json
3) Write Code for accessing data:
a) Pass the credentials to connect to API
b) Apply filters to fetch the data
c) Format the output accordingly
d) Insert the data into database
e) Release the connections/pool
4) Check the data in database. Find the time in achieving this.

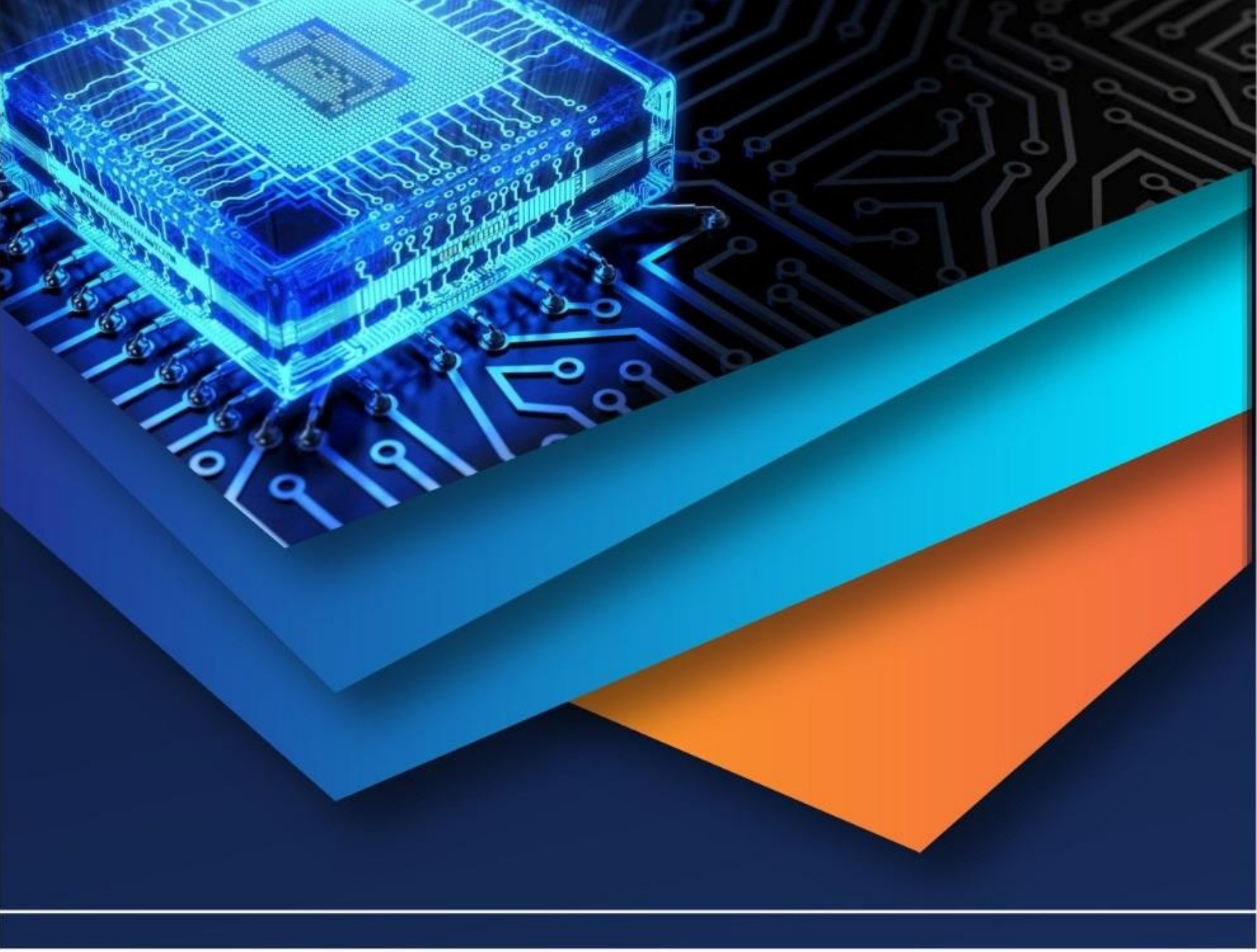
IV. CONCLUSION

A. Scalable pay-as-you-go functions as a service (FaaS) to run your code with zero server management.
B. No servers to provision, manage, or upgrade
C. Automatically scale based on the load
D. Integrated monitoring, logging, and debugging capability
E. Built-in security at role and per function level based on the principle of least privilege
F. Key networking capabilities for hybrid and multi-cloud scenarios

REFERENCES

[1] Baldini, Ioana, et al. "Serverless computing: Current trends and open problems." Research Advances in Cloud Computing. Springer, Singapore, 2017. 1-20.
[2] McGrath, Garrett, and Paul R. Brenner. "Serverless computing: Design, implementation, and performance." IEEE 37th International Conference on Distributed Computing Systems Workshops (ICDCSW), 2017. 405-410
[3] Fox, Geoffrey C., et al. "Status of serverless computing and function-as-a-service (faas) in industry and research." arXiv preprint arXiv:1708.08028 (2017)
[4] Hellerstein, Joseph M., et al. "Serverless computing: One step forward, two steps back." arXiv preprint arXiv:1812.03651 (2018).
[5] Jonas, Eric, et al. "Cloud programming simplified: A berkeley view on serverless computing." arXiv preprint arXiv:1902.03383 (2019).
[6] Adzic, Gojko, and Robert Chatley. "Serverless computing: economic and architectural impact." Proceedings of the 2017 11th joint meeting on foundations of software engineering. 2017. pp. 884-889
[7] Castro, Paul, et al. "The rise of serverless computing." Communications of the ACM 62.12 (2019): 44-54.
[8] Lee, Hyungro, Kumar Satyam, and Geoffrey Fox. "Evaluation of production serverless computing environments." 2018 IEEE 11th International Conference on Cloud Computing (CLOUD), 2018, pp. 442-450.
[9] Lloyd, Wes, et al. "Serverless computing: An investigation of factors influencing microservice performance." IEEE International Conference on Cloud Engineering (IC2E). 2018. 159-169
[10] Akkus, Istemi Ekin, et al. "{SAND}: Towards High-Performance Serverless Computing." 2018 {Usenix} Annual Technical Conference ({USENIX}{ATC} 18). 2018.

©IJRASET: All Rights are Reserved



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)