



INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Build A Serverless Website Using AWS Cloud

¹Nikita Shelar, ²Siddhi Kumbhare, ³Abhishek Gorde

¹Student, ²Student, ³Student

¹School of Computing Science and Engineering,

¹Sandip University, Nashik, Maharashtra, India

Abstract: AWS was invented in 2006, it is a cloud service platform which offers compute power, database storage, content delivery and other functionality to help developers to develop and reinvent. AWS provides different services that allows to build full application or a website stacks without managing any server. Serverless denotes to an application framework for building application or a website without servers. The server is managed by cloud provider and takes care of its allocation. This makes application or a website to run in a stateless compute container.

Index Terms – Serverless Computing, Amazon Web Services

I. INTRODUCTION

In this project we have created a simple serverless website that helps user to request the feedback form for college survey. Serverless is a process which shows services, practices and strategies which is used to build a website so as to innovate and develop for faster changes. Serverless computing contains infrastructure management tasks as capacity provisioning and patching. Developers can easily focus on developing code that serves the customer. serverless services like AWS lambda has a quality of auto scaling and pay as per use billing model. Serverless infrastructure does not require any servers. Developers can focus on core product and a business logic. Serverless applications or website doesn't require you to manage any different servers. We can easily focus only on core product and business logic, instead of focusing on the responsibilities like operating system (OS) access control, OS patching, provisioning, right-sizing, scaling, and availability.

II. Objective

- Serverless computing offers a number of advantages over traditional cloud-based or server based infrastructure.
- Serverless architectures offer high scalability, high flexibility, and quicker time to release, all at a reduced cost. Thus, developers do not need to worry about purchasing, provisioning, and managing backend servers.
- However, serverless computing is not a magic bullet for all application developers.

III. LITERATURE REVIEW

Mubashra Sadaqat, R. Colomo-Palacios, L. Knudsen, Published 2018, Computer Science studied serverless computing and create a literature review. This review focuses on introduction of Serverless Computing mainly. Serverless computing is a cloud computing execution model which enables developers to focus more on business logic rather than on infrastructure or maintenance of servers. This new paradigm has become a source of attraction for developers and organizations alike as it does not only reduce but simply eliminates the overhead of scaling, provisioning and infrastructure altogether. Given the novelty of the phenomenon, this paper is meant to study the phenomenon in a systematic way in order to define the core components of serverless computing, its benefits, challenges and what lies in the foreseen future of the serverless concept. To this end, authors conducted a multivocal literature review in order to better comprehend the state-of-art on serverless computing. The study shows that serverless computing is a solution that allows users to create

functions that intercept and operate on data flows in a scalable manner without the need to manage a server, although presents several challenges.

IV. Feedback Form

Feedback form is to take reviews from the students of an university for the university revaluation. For creating a feedback form you can open a new HTML code editing page visual studio code. I have created a single file having HTML code. This is the format of our website. In this we can submit a review to the particular university.

The image shows a web browser window with a feedback form. The browser's address bar shows the URL `s3.amazonaws.com/www.feedform.com/feedbackform-3.html`. The form has an orange header with the text "Welcome to feedback form". Below the header, it says "Please fill the form for our college survey". The form fields are grouped under "Your details:" and include input boxes for "First Name:", "Middle Name:", "Last Name:", "Year:", "University Roll Number:", "Department:", and "Email:". A "submit review" button is at the bottom of the form. The background of the page is decorated with a collage of educational icons: a hand holding a pen, a lamp, an open book, interlocking gears, and a graduation cap. The IJCRT logo is also visible in the bottom right corner of the background.

V. On AWS Platform

You can build a serverless website by using different AWS services. Each service is fully managed and it is not required to manage servers. You have to configure them and upload the code to AWS Lambda, which is also called as serverless compute service.

1. S3 Storage Service (S3 bucket)

Amazon S3 stands for Simple Storage Service. Amazon S3 stores the data in buckets which have capacity of maximum 100 buckets and is used to store and retrieve any amount of data from anywhere such as web sites and mobile apps, corporate applications and data from IoT sensors and devices.

We create and upload a html file of our feedback form. From that an Object URL is generated from that we can access the website and submit the review.

2. API Gateway

An API gateway is an API management tool to connect a client and a collection of backend services. An API gateway used to access the HTTP and HTTPS to the lambda function.

3. AWS Lambda

In this, API gateway triggers the lambda function and the reviews will store into the table named reviews.

4. AWS DynamoDB

The reviews submitted by the students will be stored in the table which we have created in the AWS DynamoDB named as reviews.

VI. Advantages

Server management is not necessary. There is not necessary to manage any server as we have created a serverless website. Pay as per the use, reduce cost. In AWS it has a quality of pay as per the use that means we just have to pay only for that an AWS service which we use to create a serverless website.

1. Scalable

Scalability is one of the quality of AWS cloud in which we can easily scale up or scale down the space as per our requirements.

2. Quick updates are possible

By using serverless infrastructure or an architecture it is easy to deploy or update the web application. This property is generally used by the app developers.

VII. Disadvantages

Testing and debugging become more challenging. In AWS after using the serverless architecture testers face more difficulties to test and debug their web application on AWS cloud platform.

1. New security concern

For the developers working on a backend face more problems regarding security of a code or any other important things which is important to secure.

2. Not for long-running processes

In AWS cloud platform the serverless architecture is not made for long term process or we can call as long running processes.

VIII. Conclusion

Building serverless applications on AWS shows that the responsibilities that servers introduce. Using AWS Lambda as our serverless logic layer used to build faster and focus our development efforts on what differentiates our website. Lambda, AWS provides additional serverless capabilities so that we can build robust, reliable, secure, and cost-effective website. Understanding the capabilities and recommendations described in this research paper can help to ensure our success when building serverless website of our own.

IX. ACKNOWLEDGEMENT

At the end of my dissertation, it is a pleasant task to express my thanks to all those who contributed in many ways to the success of this study and made it an unforgettable experience for me.

We would like to express my sincere gratitude to our mentor **Prof. Ms. Narendra Joshi** for her excellent guidance and continuous encouragement during the course of our work. I truly appreciate her vast knowledge and delight in supervision and advice.

Our special thanks to **Prof Ms. Apurva Bhavsar** Head of Computer Engineering Department, for her constant inspiration and all the facilities provided to successfully complete this work.

We would also like to thank **Dr. Onkar Kemkar, Dean** of the Institute who has provided us this opportunity to present this research as final year submission. I would also like to thank all the faculty members of the department for their valuable guidance and support during the course of my work.

Also, I would like to thank all my friends who have directly or indirectly helped me in my project work throughout the course.

Finally, I would like to thank my parents, from whom I learnt the value of hard work and its fruitful results.

X. REFERENCES

L. N. Venugopal, Research Scholar, Osmania University, Hyderabad, Telangana, India.

Dr. C. R. K. Reddy Professor & Head of the Department computer science, Mahatma Gandhi Institute of Technology

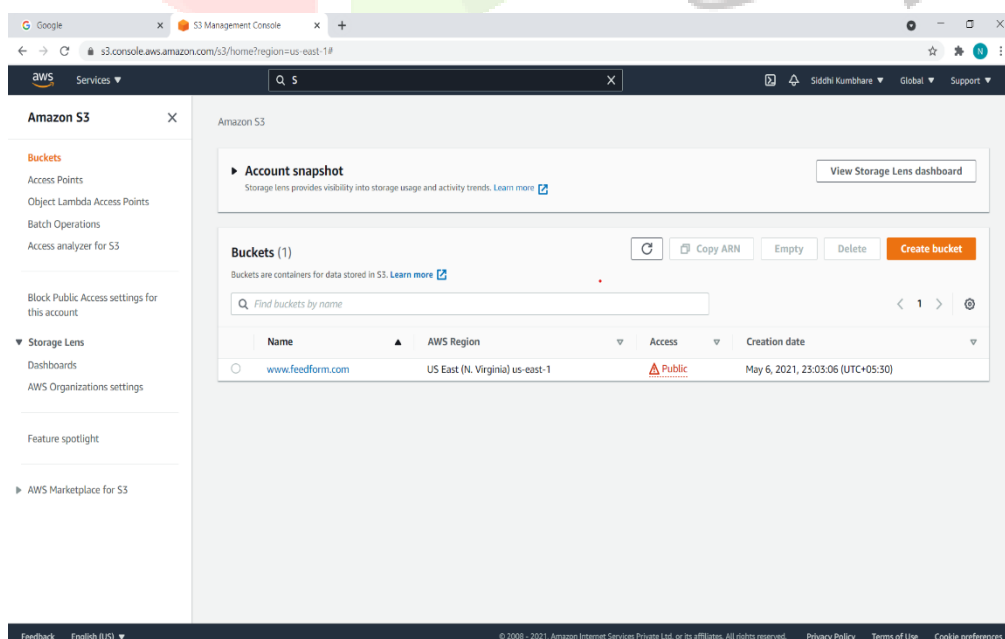
Hyderabad Telangana, India

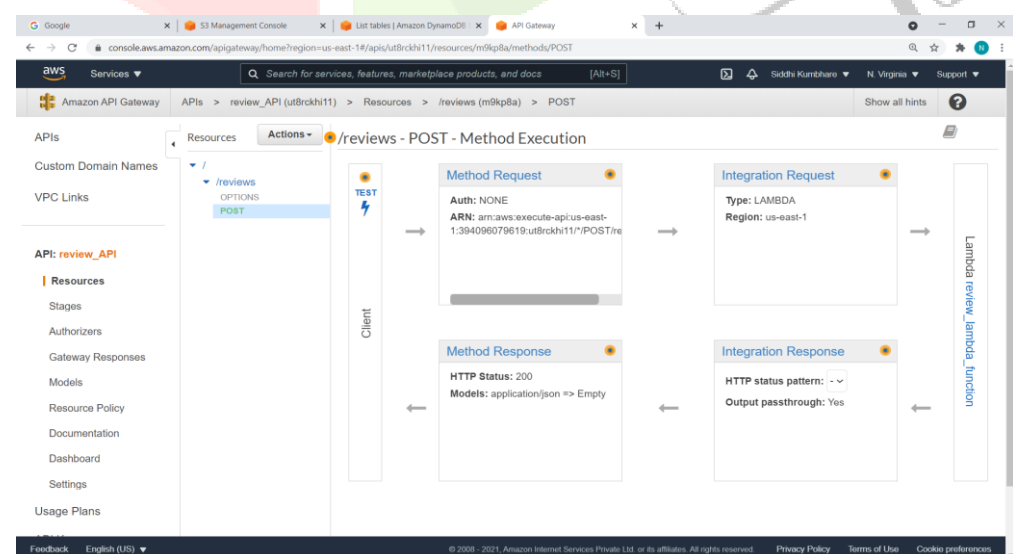
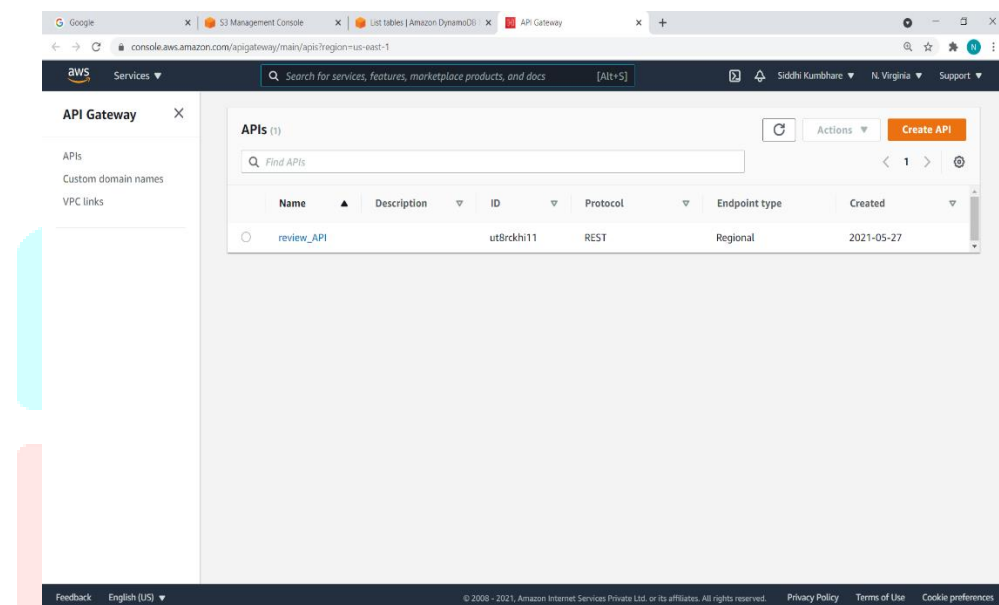
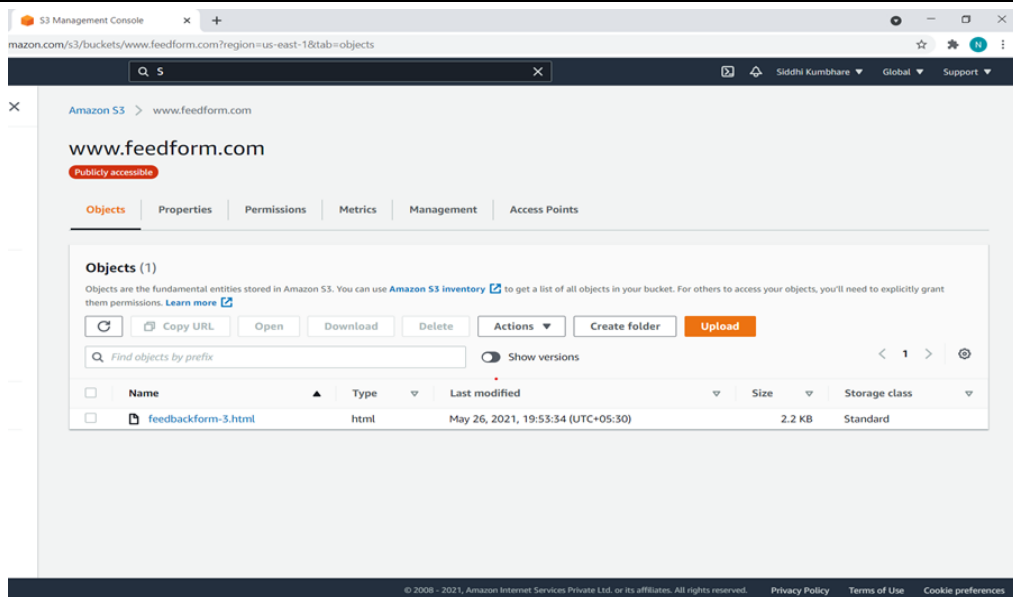
<https://www.ijert.org/serverless-through-cloud-native-architecture>

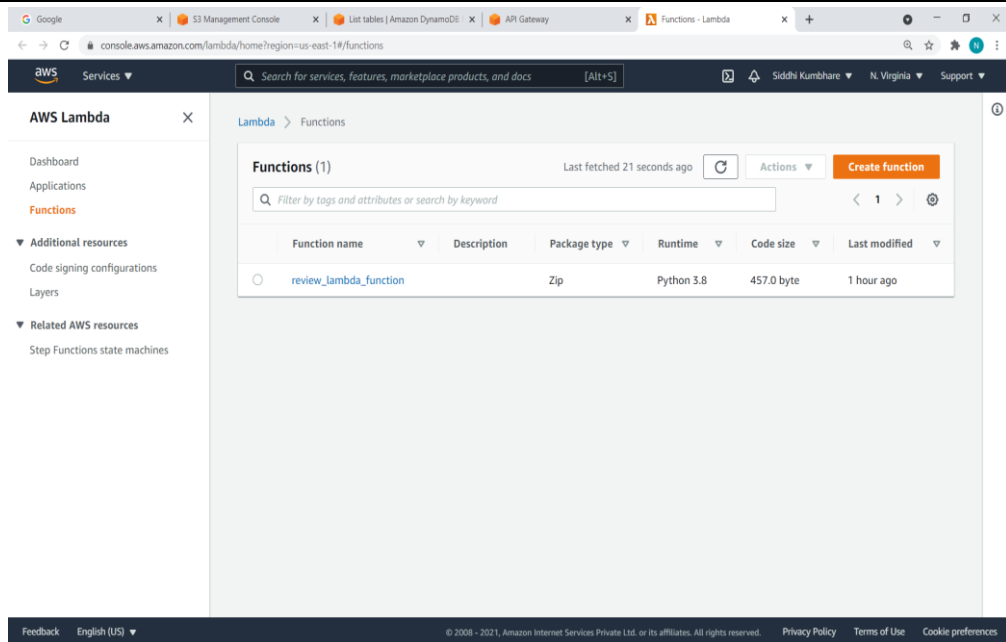
https://www.researchgate.net/publication/340121613_Patterns_for_Serverless_Functions_Function-as-a-Service_A_Multivocal_Literature_Review

<https://aws.amazon.com/console/>

XI. Project Screenshots

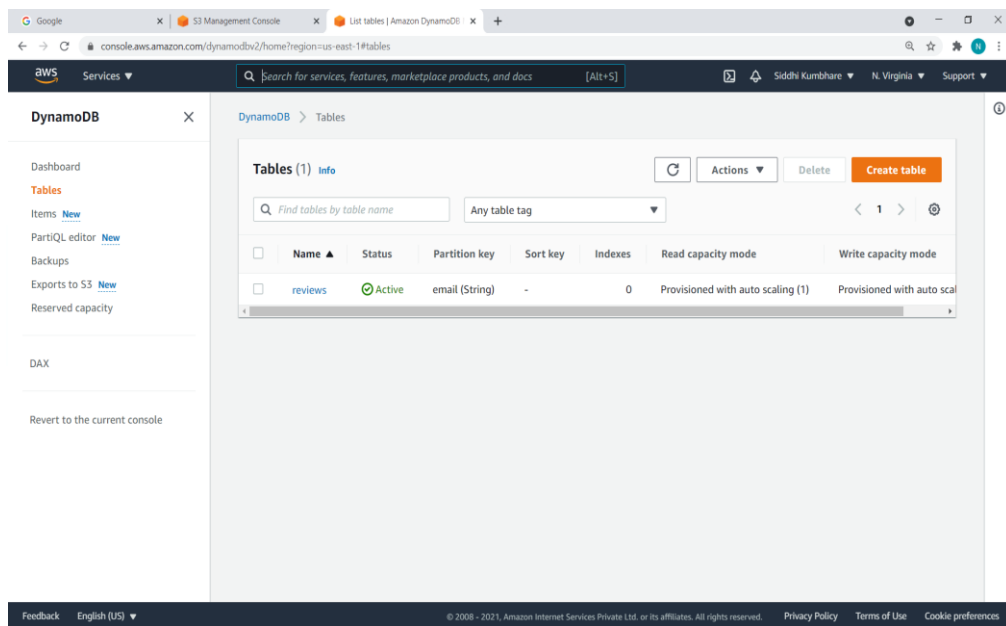






The screenshot shows the AWS Lambda console in the 'Functions' section. The left sidebar contains navigation links: Dashboard, Applications, Functions (selected), Additional resources, Code signing configurations, Layers, and Related AWS resources. The main content area displays a table of functions. A search bar at the top allows filtering by tags and attributes or by keyword. The table lists one function named 'review_lambda_function' with a Zip package type, Python 3.8 runtime, 457.0 byte code size, and was last modified 1 hour ago.

Function name	Description	Package type	Runtime	Code size	Last modified
review_lambda_function		Zip	Python 3.8	457.0 byte	1 hour ago



The screenshot shows the AWS DynamoDB console in the 'Tables' section. The left sidebar contains navigation links: Dashboard, Tables (selected), Items, PartiQL editor, Backups, Exports to S3, Reserved capacity, DAX, and Revert to the current console. The main content area displays a table of tables. A search bar at the top allows finding tables by name. The table lists one table named 'reviews' with an Active status, email (String) partition key, no sort key, 0 indexes, and Provisioned with auto scaling (1) for both read and write capacity modes.

Name	Status	Partition key	Sort key	Indexes	Read capacity mode	Write capacity mode
reviews	Active	email (String)	-	0	Provisioned with auto scaling (1)	Provisioned with auto scaling (1)