

**University Of Petroleum and Energy Studies, Dehradun**

Diagram

Description automatically generated

**Cloud Application Deployment.**

**Submitted by : Shivam Kumar**

**Batch- B1 (CCVT H)**

**SAP ID- 500083476**

**Enrollment Number- R2142201085**

**Submitted to: Dr. Harvinder Singh**

**A serverless taxi booking web application.**

**Concept Note**

In this project, we will develop a serverless web app taxi booking system that enables customers to reserve rides from a preferred location. ArcGIS.com will be used to display the map for the pickup location that the user can select from the map.

We will use AWS services like the Lambda function, DynamoDB, API Gateway, and AWS Amplify to deploy our project on AWS.

We will upload our website's HTML, CSS, and JS files to AWS Amplify, which will construct a backend server for us.

The option to authenticate will now be available to the user whenever they interact with our website. Therefore, he can do so either by signing in using Google or Gmail or by using his actual existing email address. This ensures that our website and the reservation of the ride can only be accessed by trusted and genuine users.

After the user is authenticated, he or she will be able to select a pickup location. After that, APIs will be called through the API Gateway, which will set off the lambda function to book a ride and store the user's data in the DynamoDB.

The fact that the data is being stored to DynamoDB using the triggering lambda function here gives it a little bit of a dynamic feel.

A display with the booking confirmation will appear once the user has successfully booked the ride.

**Why we are deploying our web-app on cloud.**

1. AWS Amplify: it is a service offered by AWS which helps the users to build and deploy a full-Stack application with highly scalability and security.

So, if we were not using the cloud based solution, then we need to set up our own infrastructure, and need to manage the servers and the backend application.

Also, amplify provides us a public link through which the customer can access our website, so we do not need to purchase a public DNS for our website in the initial phases and hence is cost effective. Also we have used our on premise server, then we need to be worried about setting up the proper environment to run our code and manage the physical servers.

1. AWS Cognito: It is a service offered by AWS which helps us for Authentication and authorisation of the users accessing our website or application.

Cognito maintains a user pool and identity pool. Under user pool, it maintains tha data of the user and is responsible for authentication whereas identity pool will provide the permissions (authorization) to the users after the successful authentication is done.

Whereas, if we have used on premise strategy, then the authorisation and authentication processes are really complex and challenging to perform (or merely impossible to be done for small Startups).

Hence, it increases the security of our website and easily filters the unknown (unauthorized users).

1. DynamoDB: It is the fully managed NoSQL database provided by the AWS to store the data in the form of rows and columns with high scalability. It is highly scalable so we only need to pay for the amount of storage we will use as per our traffic.

So, in our previous days, with low Expenditure cost, we did not need to think for the overhead of storage capacity.

1. Lambda function: it helps us to run our code on high availability compute resources. It is event triggered and has the capacity to automatically scale.

With the help of the lambda function on cloud, we are making our application serverless. The lambda function also provides us the 1 millions of instructions free of cost and even responsible for scaling of resources in case of sudden increase in high request. Whereas, it was very difficult to process, manage and respond to a million of instructions in case of on premise setup and would have required high capex and opex with highly skilled IT professionals.

**Flowchart:**





**Literature Review:**

[1] G. McGrath and P. R. Brenner, "Serverless Computing: Design, Implementation, and Performance," 2017 IEEE 37th International Conference on Distributed Computing Systems Workshops (ICDCSW), Atlanta, GA, USA, 2017, pp. 405-410, doi: 10.1109/ICDCSW.2017.36.

[2] R. A. P. Rajan, "Serverless Architecture - A Revolution in Cloud Computing," 2018 Tenth International Conference on Advanced Computing (ICoAC), Chennai, India, 2018, pp. 88-93, doi: 10.1109/ICoAC44903.2018.8939081.

[3] D. Kelly, F. Glavin and E. Barrett, "Serverless Computing: Behind the Scenes of Major Platforms," 2020 IEEE 13th International Conference on Cloud Computing (CLOUD), Beijing, China, 2020, pp. 304-312, doi: 10.1109/CLOUD49709.2020.00050.

[4] S. Gandhi, A. Gore, S. Nimbarte and J. Abraham, "Implementation and Analysis of a Serverless Shared Drive with AWS Lambda," 2018 4th International Conference for Convergence in Technology (I2CT), Mangalore, India, 2018, pp. 1-6, doi: 10.1109/I2CT42659.2018.9058237.

[5] S. Neela, Y. Neyyala, V. Pendem, K. Peryala and V. V. Kumar, "Cloud Computing Based Learning Web Application Through Amazon Web Services," 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 2021, pp. 472-475, doi: 10.1109/ICACCS51430.2021.9441974.

[6] R. A. P. Rajan, "Serverless Architecture - A Revolution in Cloud Computing," 2018 Tenth International Conference on Advanced Computing (ICoAC), Chennai, India, 2018, pp. 88-93, doi: 10.1109/ICoAC44903.2018.8939081.

[7] J. Surbiryala and C. Rong, "Cloud Computing: History and Overview," 2019 IEEE Cloud Summit, Washington, DC, USA, 2019, pp. 1-7, doi: 10.1109/CloudSummit47114.2019.00007.

[8] J. Surbiryala and C. Rong, "Cloud Computing: History and Overview," 2019 IEEE Cloud Summit, Washington, DC, USA, 2019, pp. 1-7, doi: 10.1109/CloudSummit47114.2019.00007.

[9] S. Namasudra, P. Roy and B. Balusamy, "Cloud Computing: Fundamentals and Research Issues," 2017 Second International Conference on Recent Trends and Challenges in Computational Models (ICRTCCM), Tindivanam, India, 2017, pp. 7-12, doi: 10.1109/ICRTCCM.2017.49.

[10] S. Patidar, D. Rane and P. Jain, "A Survey Paper on Cloud Computing," 2012 Second International Conference on Advanced Computing & Communication Technologies, Rohtak, India, 2012, pp. 394-398, doi: 10.1109/ACCT.2012.15.