Building A Serverless Web Application Using API Gateway

Abstract:

Serverless computing allows you to build and run applications and services without thinking about servers. With serverless computing, your application still runs on servers, but all the server management is done by AWS. Using AWS and its Serverless Platform, you can build and deploy applications on cost-effective services that provide built-in application availability and flexible scaling capabilities. This lets you focus on your application code instead of worrying about provisioning, configuring, and managing servers.

Existing System:

In existing system, developers manage servers and due to that they had over-burden on performance metrics of server and thus, when AWS launched serverless applications it has created a mega buzz in the developer community.

Proposed System:

Building a serverless application allows developers to focus on your application code instead of managing and operating infrastructure. Developers do not think about provisioning or configuring servers since AWS handles all of that for developers.

This reduces the infrastructure management burden and helps the developers builds applications faster time-to-market.

Here, developers get add-ons like No Server Management, Flexible Scaling, High Availability, and No idle capacity. We are going to create a RESTful API using Amazon API Gateway for the user to perform actions in the web application. These API's will trigger Lambda Events running on AWS to automate the storage, or even push data to Queue's for processing.

Host a web application in the virtual server of amazon and authenticate the users through Amazon Cognito service and host HTML, CSS, JavaScript through AWS Amplify service.

Software Tools:

1. AWS Cognito
2. AWS API Gateway
3. AWS S3
4. AWS DynamoDB
5. AWS SQS

6. AWS Amplify
7. AWS Lambda
8. AWS IAM
9. AWS SAM
10. AWS CLI
11. Python3
12. VS Code

Hardware Tools:

1. Laptop
2. Operating System: Windows 11
3. RAM: 16GB
4. ROM: 4GB
5. Fast Internet Connectivity