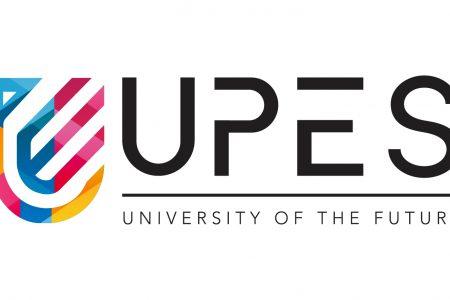
****

**UNIVERSITY OF PETROLEUM AND ENERGY STUDIES, DEHRADUN**

**BACHELOR OF TECHNOLOGY**

in

**COMPUTER SCIENCE**

Specialization in

**CLOUD COMPUTING & VIRTUALIZATION TECHNOLOGY**

**WEEK 2 OF**

**CLOUD APPLICATION DEVELOPMENT.**

*Submitted to*

**PROF. SAURABH SHANU SIR**

*By:*

Anant Garg

R2142201709

**Making Chat Applications using AWS**

**Architectural Style of my Project:**

The architectural style of a real-time chat application is typically a SIMD (Single Instruction Multiple Data) and client-server architecture.

A SIMD architecture, on the other hand, processes one instruction on multiple data items at the same time, making it well-suited for real-time chat applications, which need to handle many chats at the same time. With a SIMD architecture, the system can handle multiple chats in parallel, allowing for fast and efficient processing of messages.

In a client-server architecture, the client application communicates with a central server, which handles all the incoming and outgoing messages. This architecture is also well-suited for real-time chat applications as it provides a central point of control, making it easier to manage and maintain the application.

**Services and Technologies used here:**

1. Front end – React.js and Node.js
2. Lambda Function: Lambda functions provide a serverless computing model, allowing you to run code without having to manage servers.
3. Amazon Simple Storage Service (S3): S3 can be used to store static assets for the front end, such as images and JavaScript files.
4. Amazon DynamoDB: DynamoDB can be used as a scalable, highly-available NoSQL database to store chat messages and user profiles.
5. Amazon API Gateway: API Gateway can create and manage the chat application's APIs, enabling communication between the front and backend services.
6. Amazon Web Sockets: Amazon Web Sockets can be used to handle real-time communication between the frontend and backend services, allowing messages to be sent and received in real time.

**Why did I choose AWS?**

I choose Aws for this project, AWS is currently the market leader in cloud computing and has a larger market share than Azure. This means that AWS has more customers, more resources, and a wider range of services. AWS can be more cost-effective and AWS has a more extensive global infrastructure, with more availability zones and regions, making it easier for organizations with a global presence to use AWS. AWS provides better support for open-source technologies, including Linux and other open-source databases.