

# **SOCIALIZE THE VOCABULARY ABOUT THE READING "WHAT ARE THE MOST INNOVATIVE CLOUD ARCHITECTURE PATTERNS** FOR CLOUD COMPUTING?"













 $\cap$ 

### **SERVERLESS ARCHITECTURE:**

**A CLOUD COMPUTING PATTERN ALLOWING CODE EXECUTION WITHOUT** MANAGING SERVERS, WHERE RESOURCES ARE PAID BASED ON **CONSUMPTION. IDEAL FOR EVENT-DRIVEN AND SHORT-LIVED APPLICATIONS, REDUCING OPERATIONAL COMPLEXITY AND COSTS.** 















### **MICROSERVICES ARCHITECTURE:**

A CLOUD COMPUTING PATTERN DIVIDING APPLICATIONS INTO INDEPENDENT SERVICES WITH SEPARATE FUNCTIONALITY, DATA, AND COMMUNICATION. SUITABLE FOR SCALABLE APPLICATIONS LIKE E-COMMERCE, ENHANCING MODULARITY, RELIABILITY, AND ENABLING FASTER FEATURE DELIVERY.

















 $\bigcap$ 

### **EVENT-DRIVEN ARCHITECTURE:**

A CLOUD COMPUTING PATTERN UTILIZING EVENTS TO TRIGGER AND COORDINATE SYSTEM COMPONENT ACTIONS. BASED ON DECOUPLING, ASYNCHRONOUS COMMUNICATION, AND REACTIVE PROGRAMMING, IT IS BENEFICIAL FOR REAL-TIME APPLICATIONS LIKE IOT, IMPROVING SCALABILITY, EFFICIENCY, AND USER EXPERIENCE.







# TIC









### LAMBDA ARCHITECTURE:

A CLOUD COMPUTING PATTERN COMBINING BATCH AND STREAM PROCESSING FOR HANDLING LARGE DATA VOLUMES WITH LOW LATENCY. SUITABLE FOR BIG DATA APPLICATIONS, IT BALANCES LATENCY, THROUGHPUT, AND FAULT TOLERANCE, OFFERING A UNIFIED VIEW OF HISTORICAL AND REAL-TIME DATA.





## TIC







 $\bigcap$ 

### **HEXAGONAL ARCHITECTURE:**

**A CLOUD COMPUTING PATTERN SEPARATING APPLICATION LOGIC FROM** EXTERNAL DEPENDENCIES, ENCAPSULATING LOGIC IN A CENTRAL LAYER, AND ABSTRACTING DEPENDENCIES INTO ADAPTERS. ALSO KNOWN AS PORTS AND ADAPTERS ARCHITECTURE, IT ENHANCES ADAPTABILITY, TESTABILITY, AND MAINTAINABILITY BY ACHIEVING LOOSE COUPLING AND **CLEAR BOUNDARIES.** 









