

# Data Engineering System Design Core Concepts

Feel free to follow me  
for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com



# WHAT IS SYSTEM DESIGN?

System design is the process of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements. It involves both high-level architecture and detailed design.

Designing a news feed system for a social media application.

Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A SCALABLE SYSTEM?

A scalable system is one that can handle increased load without compromising performance by adding resources.

Horizontal scaling by adding more servers to handle increased traffic.

Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# HOW DO YOU APPROACH A SYSTEM DESIGN INTERVIEW?

Understand the requirements, define the scope, outline a high-level architecture, dive into detailed design for key components, and address potential bottlenecks and trade-offs.

Designing a URL shortener: Start with the core functionality, then address storage, scalability, and fault tolerance.

Feel free to follow me for more content

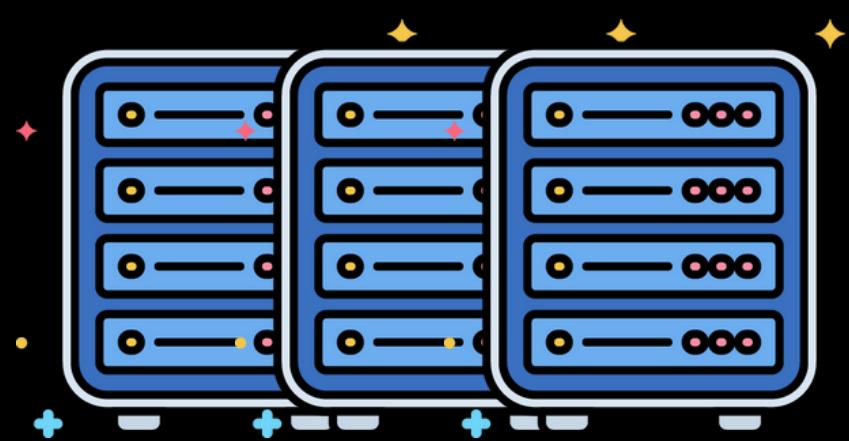
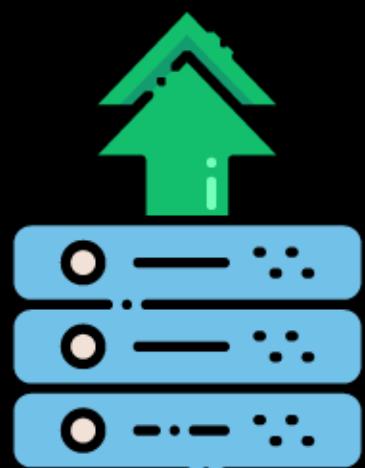


Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS THE DIFFERENCE BETWEEN HORIZONTAL AND VERTICAL SCALING?

Horizontal scaling (scaling out) involves adding more machines to a system, while vertical scaling (scaling up) involves adding more power (CPU, RAM) to an existing machine.

Adding more servers to a web application (horizontal) vs. upgrading the server's hardware (vertical).



Feel free to follow me for more content

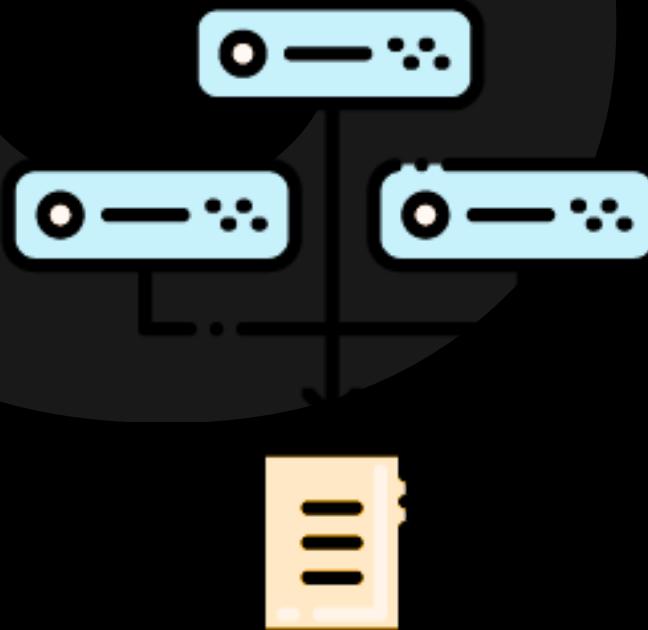


Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT ARE THE ADVANTAGES OF HORIZONTAL SCALING?

It improves fault tolerance, allows for easier load distribution, and avoids the limitations of a single machine.

Distributing web traffic across multiple servers using a load balancer.



Feel free to follow me  
for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A LOAD BALANCER?

A load balancer distributes incoming network traffic across multiple servers to ensure no single server becomes overwhelmed.

Using an AWS Elastic Load Balancer to manage web traffic.



Feel free to follow me  
for more content

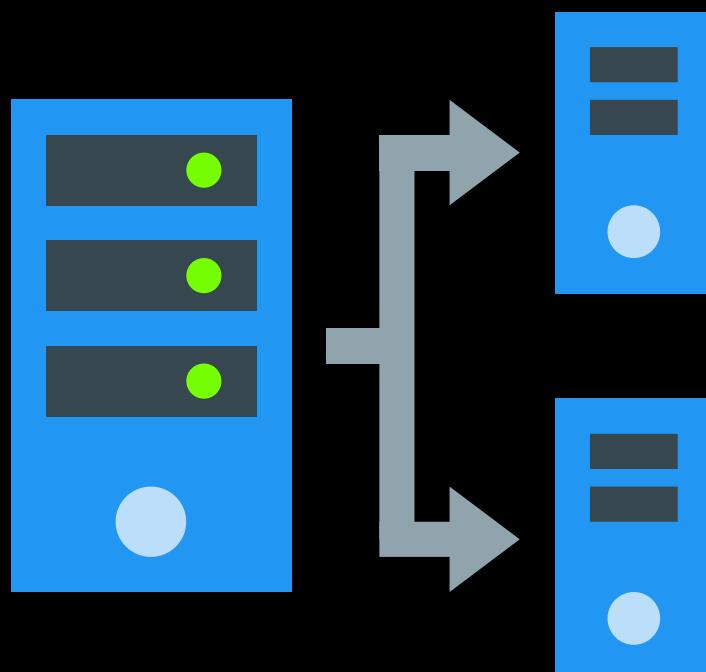


Shwetank Singh  
GritSetGrow - GSGLearn.com

# HOW DOES A LOAD BALANCER IMPROVE SYSTEM RELIABILITY?

By distributing traffic, it ensures that if one server fails, the load balancer can redirect traffic to other available servers, thus maintaining service availability.

In a web application with three servers, if one fails, the load balancer redirects traffic to the remaining two.



Feel free to follow me for more content

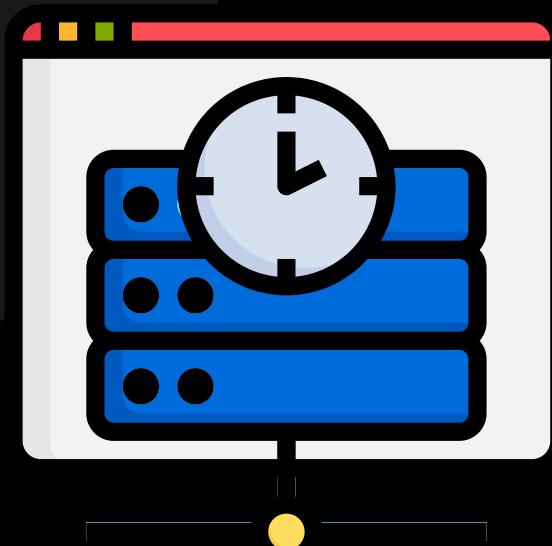


Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS CACHING?

Caching is a technique to store frequently accessed data in a temporary storage location to speed up subsequent data retrievals.

Using Redis to cache database query results.



Feel free to follow me for more content

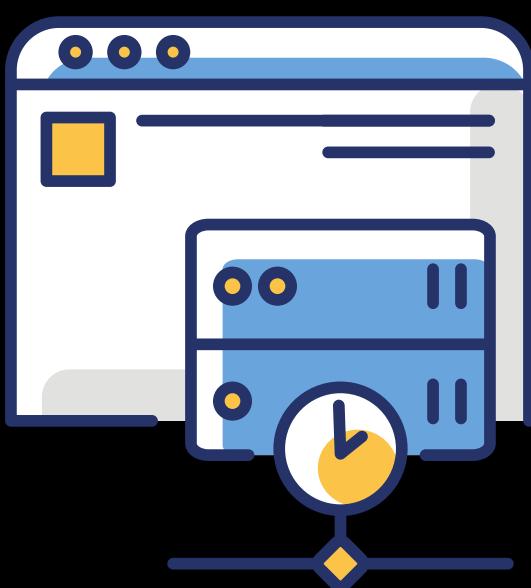
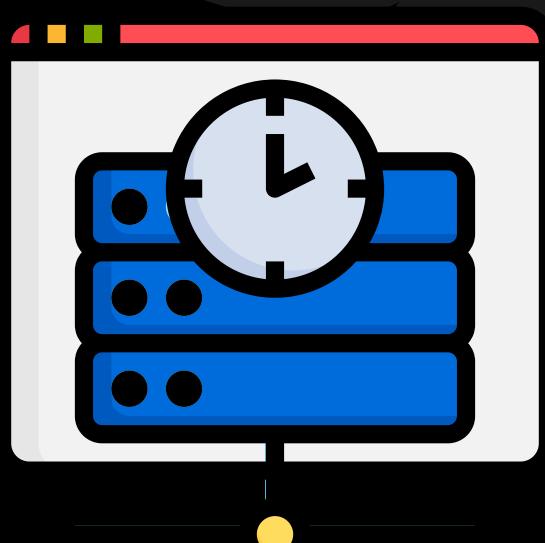


Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT ARE THE TYPES OF CACHES?

There are client-side caches, server-side caches, and distributed caches.

Browser cache (client-side), Memcached (server-side), and Redis (distributed).



Feel free to follow me  
for more content

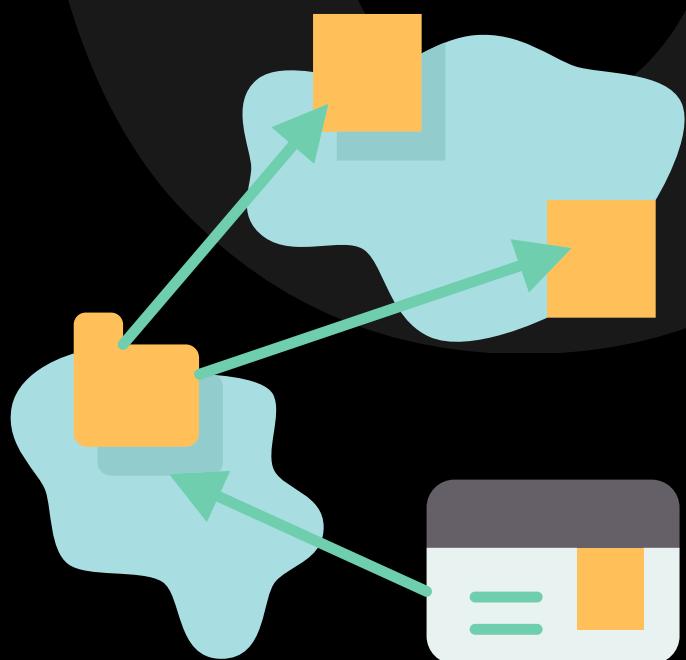


Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A CDN (CONTENT DELIVERY NETWORK)?

A CDN is a network of servers distributed geographically to deliver static content to users from the nearest server location, reducing latency.

Using Cloudflare CDN to serve static assets like images and CSS files.



Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# HOW DOES A CDN IMPROVE PERFORMANCE?

By reducing the distance between users and the content, decreasing load times and reducing bandwidth usage.

Serving a video from a server located closer to the user, resulting in faster loading times.



Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS DATABASE REPLICATION?

Database replication is the process of copying data from one database server (master) to another (slave) to ensure high availability and fault tolerance.

Using MySQL replication to maintain a backup database server.



Feel free to follow me  
for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT ARE THE BENEFITS OF DATABASE REPLICATION?

It improves read performance, ensures high availability, and provides data redundancy.

A master-slave setup where the master handles writes, and slaves handle reads.



Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS SHARDING?

Sharding is a database partitioning technique that divides a large database into smaller, more manageable pieces, called shards, which can be distributed across multiple servers.

Splitting user data across different databases based on user ID ranges.

Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT ARE THE CHALLENGES OF SHARDING?

Challenges include data consistency, complex queries across shards, and re-sharding data when a shard grows too large.

Implementing a consistent hashing algorithm to evenly distribute data across shards.

Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A MESSAGE QUEUE?

A message queue is a component used for communication between processes, allowing them to send and receive messages asynchronously.

Using RabbitMQ to manage background tasks in a web application.

Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT ARE THE BENEFITS OF USING A MESSAGE QUEUE?

It enables asynchronous processing, improves system resilience, and decouples system components.

Processing user signup emails in the background while the main application handles user requests.

Feel free to follow me for more content

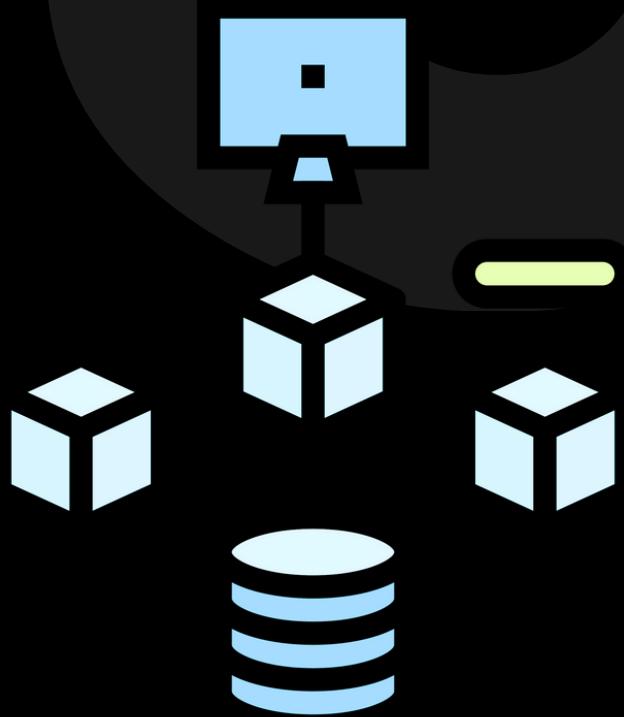


Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A MICROSERVICES ARCHITECTURE?

Microservices architecture is an architectural style that structures an application as a collection of loosely coupled services, each with its own functionality and data storage.

Breaking down a monolithic e-commerce application into individual services for inventory, payment, and user management.



**Feel free to follow me  
for more content**



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT ARE THE BENEFITS OF MICROSERVICES?

Benefits include improved modularity, easier scaling, and better fault isolation.

Like scaling the payment service independently of the inventory service in an e-commerce application.



Feel free to follow me for more content

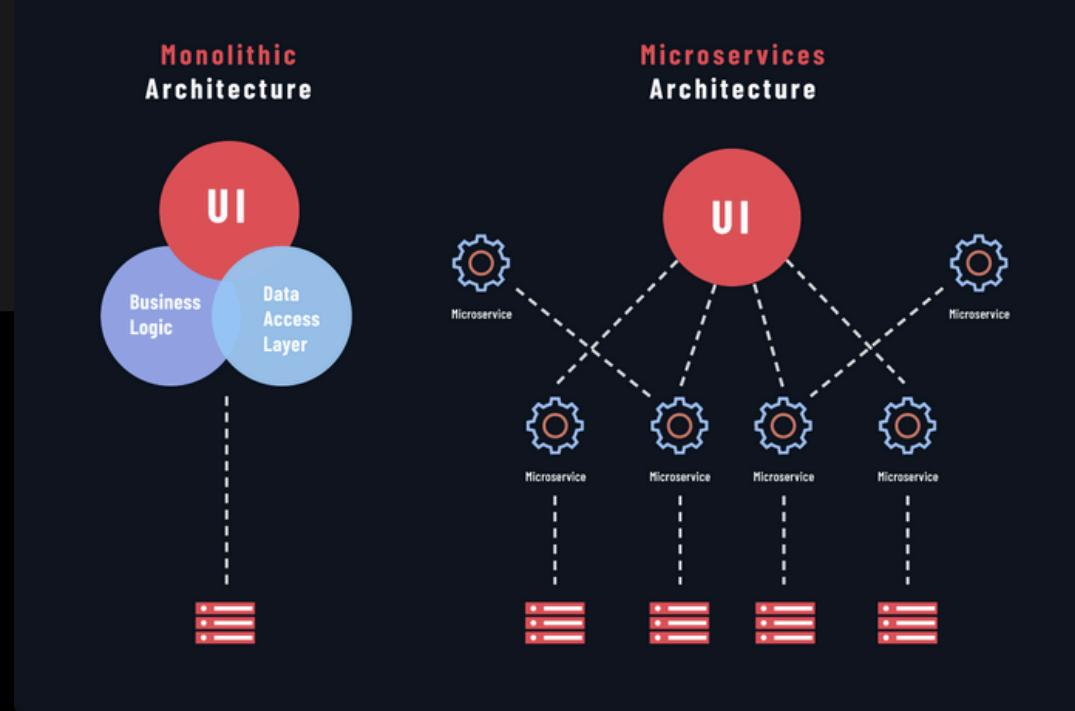


Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A MONOLITHIC ARCHITECTURE?

A monolithic architecture is a single-tier software application where all components are interconnected and interdependent.

A traditional web application where the frontend, backend, and database are tightly integrated.



Feel free to follow me  
for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT ARE THE DRAWBACKS OF A MONOLITHIC ARCHITECTURE?

Drawbacks include difficulty in scaling individual components, tight coupling, and challenges in maintaining and deploying the application.

Updating one part of the application requires redeploying the entire system.

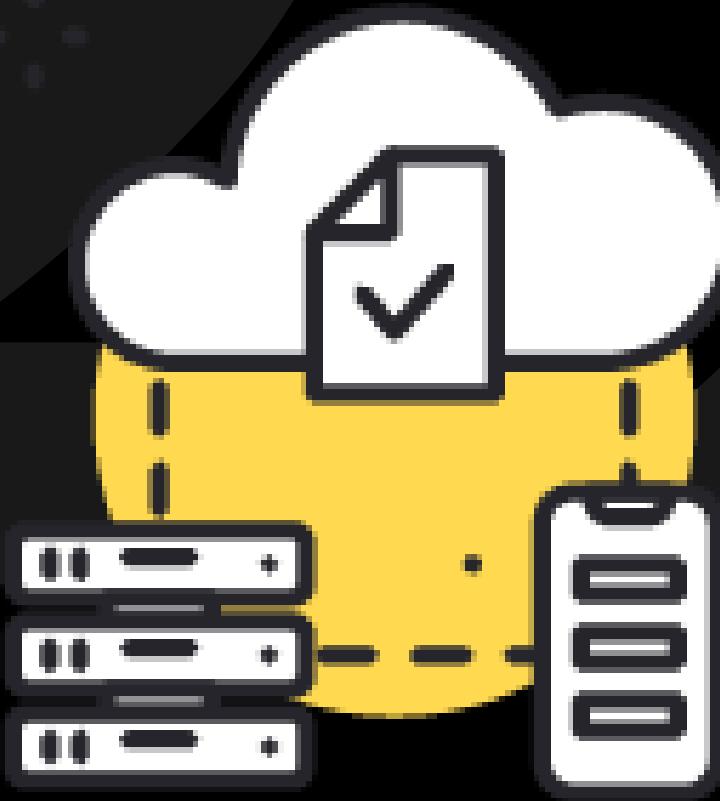
Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS EVENTUAL CONSISTENCY?

Eventual consistency is a consistency model used in distributed systems where updates to a database will propagate to all nodes, but not immediately.



Feel free to follow me  
for more content

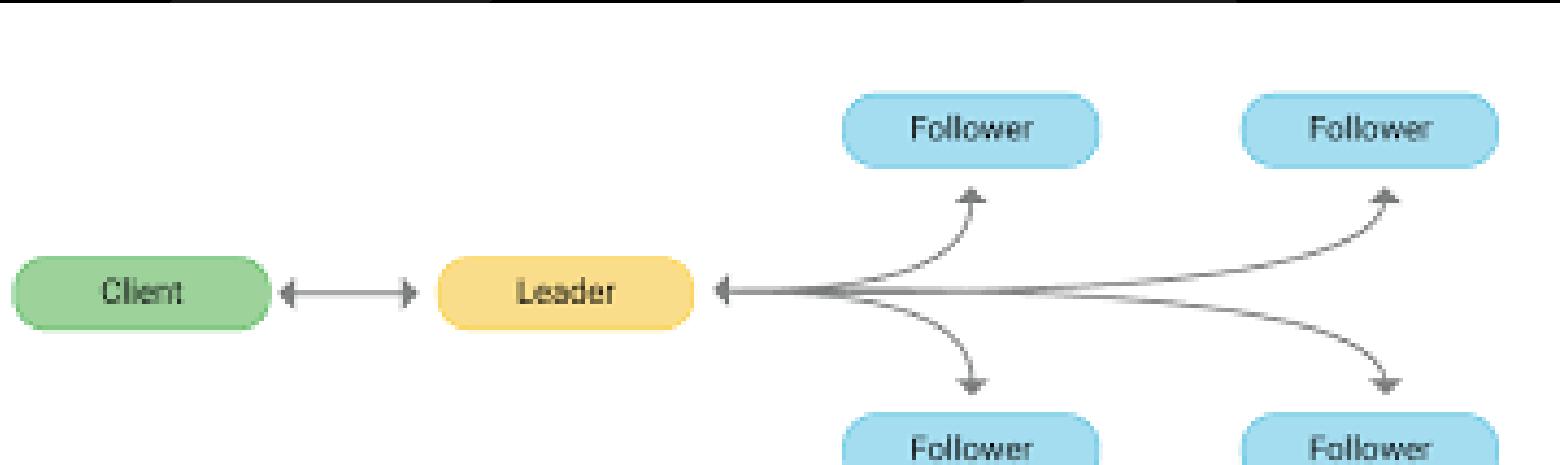


Shwetank Singh  
GritSetGrow - GSGLearn.com

# HOW DO YOU ENSURE DATA CONSISTENCY IN A DISTRIBUTED SYSTEM?

Techniques include using consensus algorithms (e.g., Paxos, Raft), distributed transactions, and ensuring idempotent operations.

The implementation of the Raft algorithm is used to manage leader election and data replication.



Feel free to follow me  
for more content

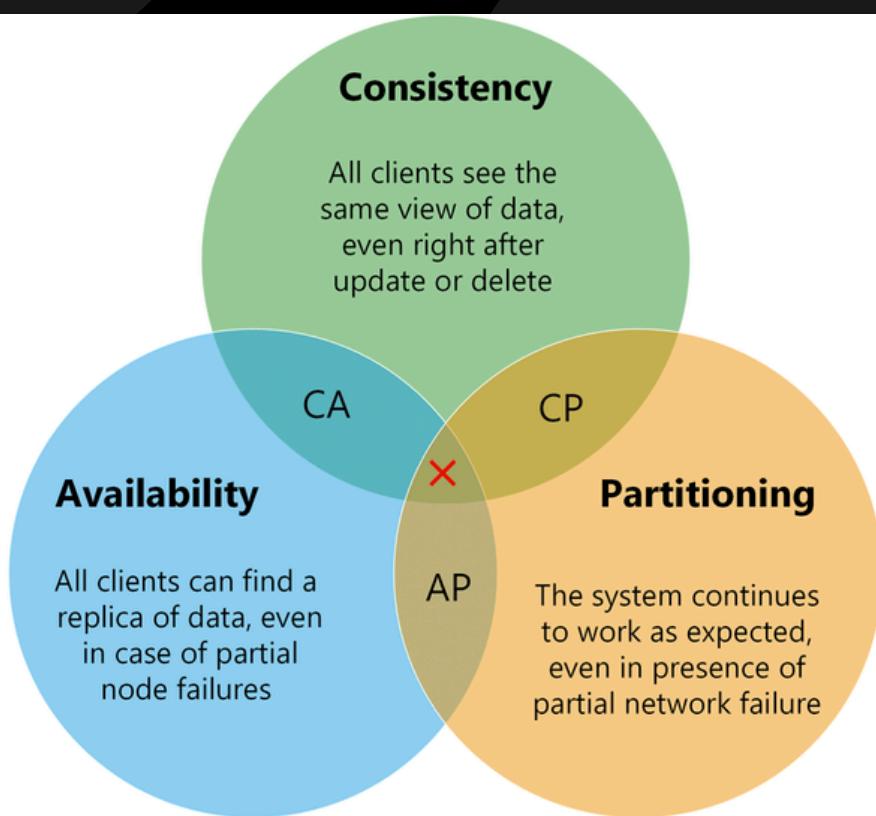


Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS CAP THEOREM?

CAP theorem states that in a distributed data store, you can only achieve two out of the following three guarantees: Consistency, Availability, and Partition tolerance.

Choosing between strong consistency and availability in a distributed database during network partitions.



Feel free to follow me  
for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A KEY-VALUE STORE?

A key-value store is a type of NoSQL database that stores data as a collection of key-value pairs.

Using Redis or Amazon DynamoDB to store session data.



Feel free to follow me  
for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT ARE THE ADVANTAGES OF KEY-VALUE STORES?

They offer fast read and write operations, are easy to scale, and provide flexible data models.

For example storing user preferences in Redis helps in quick retrieval.

Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A DOCUMENT STORE?

A document store is a type of NoSQL database that stores data as documents, typically in JSON or BSON format.

Using MongoDB to store user profiles with varying attributes.

Feel free to follow me for more content

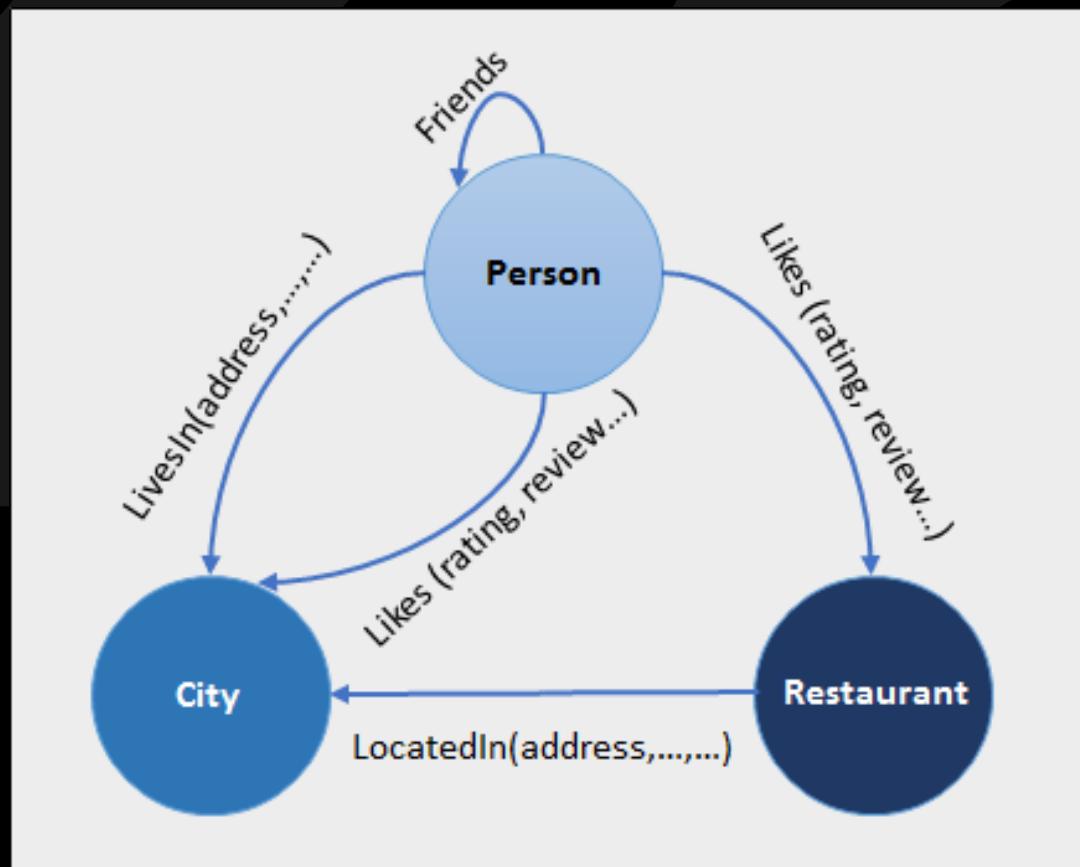


Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A GRAPH DATABASE?

A graph database is designed to store and query data modeled as graphs, with nodes, edges, and properties.

Using Neo4j to manage social network data with complex relationships.



Feel free to follow me  
for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT ARE THE BENEFITS OF GRAPH DATABASES?

They excel at handling complex relationships, enable efficient traversal queries, and provide a flexible schema.

Finding shortest paths and recommendations in a social network using Neo4j.

Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A COLUMN-FAMILY STORE?

A column-family store is a type of NoSQL database that stores data in columns rather than rows, optimized for read and write operations.

Using Apache Cassandra for time-series data storage.

Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT ARE THE BENEFITS OF COLUMN-FAMILY STORES?

They offer high write throughput, horizontal scalability, and are suitable for time-series data and real-time analytics.

Storing and retrieving logs in Apache Cassandra.

Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A DISTRIBUTED FILE SYSTEM?

A distributed file system (DFS) is a file system that allows access to files from multiple hosts, providing redundancy and fault tolerance.

Using Hadoop Distributed File System (HDFS) for big data storage.

Feel free to follow me for more content

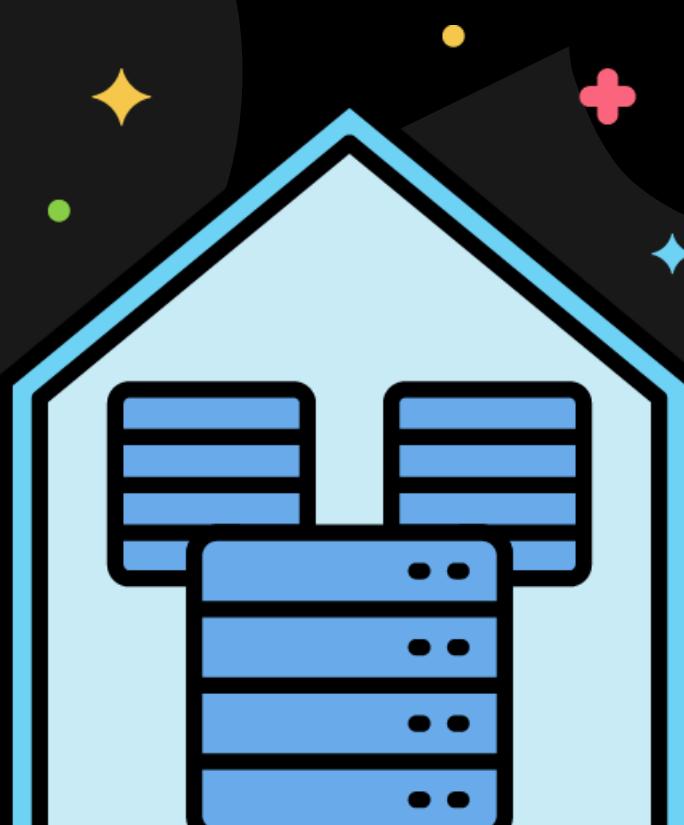


Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A DATA WAREHOUSE?

A data warehouse is a centralized repository for storing large volumes of structured and semi-structured data, optimized for query and analysis.

Example using Amazon Redshift or Google BigQuery for business analytics.



Feel free to follow me  
for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS ETL (EXTRACT, TRANSFORM, LOAD)?

ETL is a process in data warehousing that involves extracting data from various sources, transforming it to fit operational needs, and loading it into a target data store.

Using Apache NiFi to extract data from databases, transform it, and load it into HDFS.



Feel free to follow me for more content

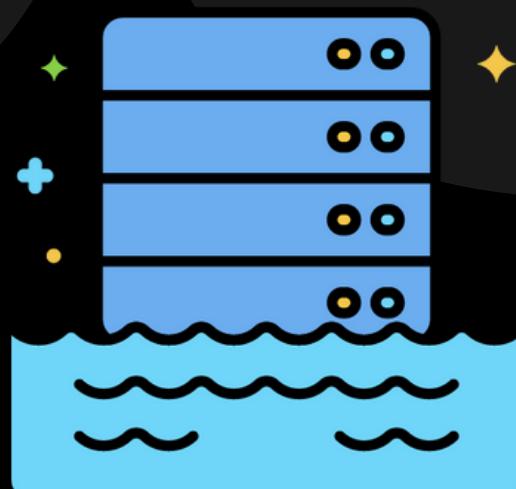


Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A DATA LAKE?

A data lake is a storage repository that holds vast amounts of raw data in its native format until it is needed.

Using Amazon S3 as a data lake for storing structured and unstructured data.



Feel free to follow me  
for more content

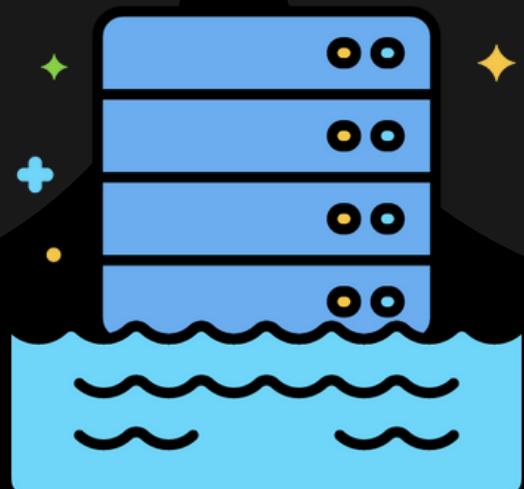


Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT ARE THE BENEFITS OF A DATA LAKE?

Benefits include scalability, flexibility, and the ability to store diverse data types.

Storing log files, images, and structured data together in Amazon S3.



Feel free to follow me  
for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS STREAM PROCESSING?

Stream processing involves processing data in real-time as it flows from one source to another, allowing immediate insights and actions.

Using Apache Kafka and Apache Flink to process real-time clickstream data.



Feel free to follow me  
for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT ARE THE ADVANTAGES OF STREAM PROCESSING?

It enables real-time analytics, reduces data latency, and supports event-driven architectures.

Real-time fraud detection in financial transactions using Apache Kafka and Apache Storm.



Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS BATCH PROCESSING?

Batch processing involves processing data in large blocks or batches at scheduled intervals.

Using Apache Hadoop to run nightly ETL jobs.



Feel free to follow me  
for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS THE DIFFERENCE BETWEEN BATCH PROCESSING AND STREAM PROCESSING?

Batch processing deals with large volumes of data processed at intervals, while stream processing deals with continuous data processing in real-time.

Using Apache Hadoop for batch processing and Apache Kafka for stream processing.

Feel free to follow me for more content

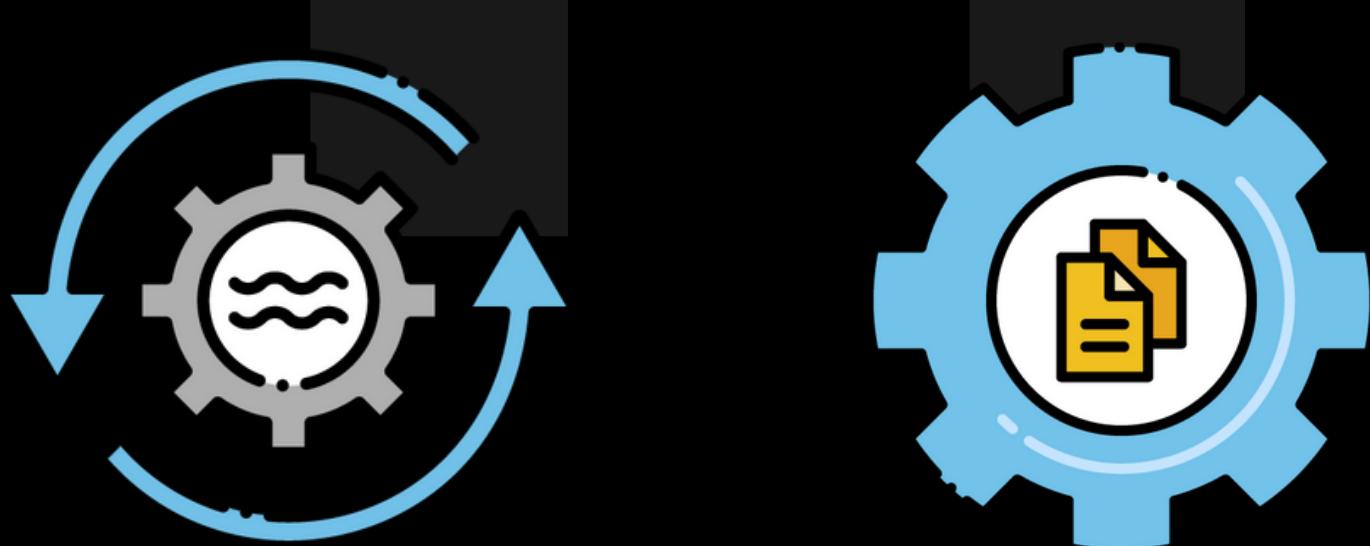


Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A LAMBDA ARCHITECTURE?

A lambda architecture is a data-processing architecture designed to handle massive quantities of data by taking advantage of both batch and stream-processing methods.

Combining Apache Hadoop for batch processing and Apache Storm for real-time processing in a lambda architecture.



Feel free to follow me  
for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT ARE THE COMPONENTS OF A LAMBDA ARCHITECTURE?

The components include a batch layer for processing large datasets, a speed layer for real-time processing, and a serving layer to merge the results.

Using Hadoop for batch processing, Kafka for real-time data, and HBase as the serving layer.



Feel free to follow me for more content

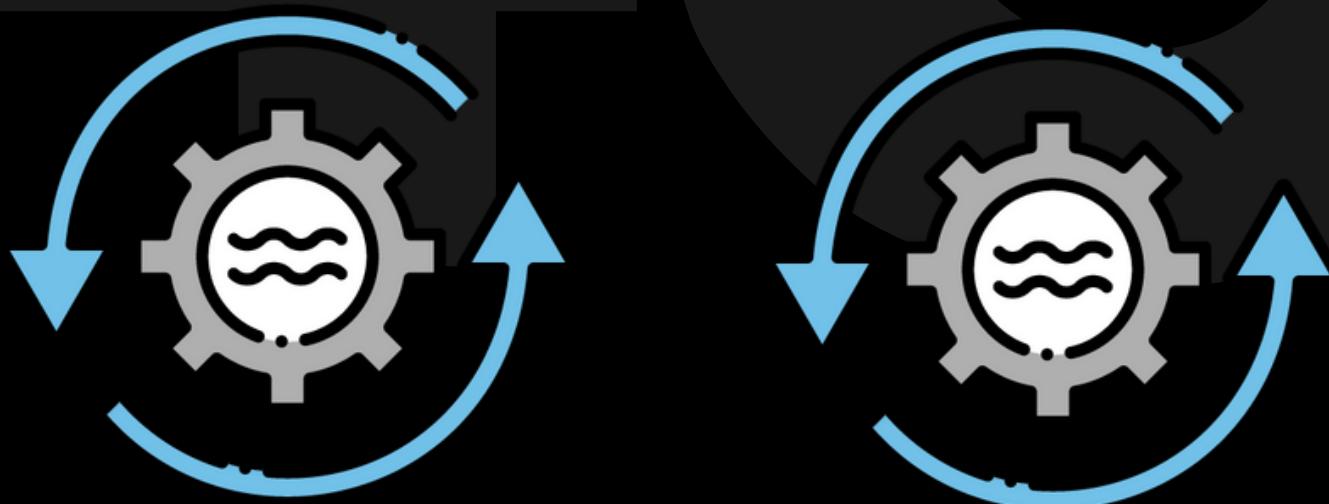


Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS THE KAPPA ARCHITECTURE?

The kappa architecture simplifies the lambda architecture by using only stream processing to handle both real-time and historical data.

Using Apache Kafka and Apache Flink to process all data as streams.



Feel free to follow me  
for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A NOSQL DATABASE?

A NoSQL database is a non-relational database designed to handle large volumes of unstructured or semi-structured data with flexible schemas.

Using MongoDB or Cassandra for storing large-scale data with varying structures.



Feel free to follow me  
for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT ARE THE TYPES OF NOSQL DATABASES?

Types include key-value stores, document stores, column-family stores, and graph databases.

Redis (key-value), MongoDB (document), Cassandra (column-family), Neo4j (graph).

Feel free to follow me for more content

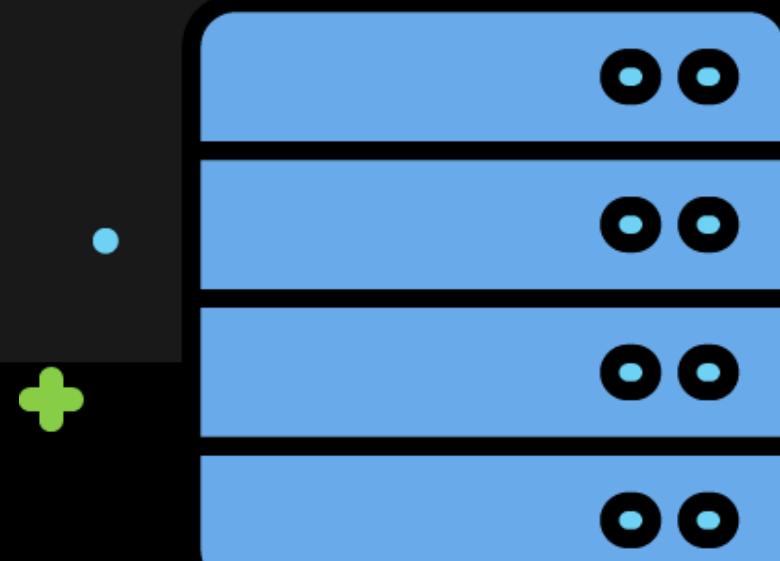


Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A RELATIONAL DATABASE?

A relational database is a type of database that organizes data into tables with rows and columns, using SQL for data management.

Using MySQL or PostgreSQL for structured data storage.



Feel free to follow me  
for more content

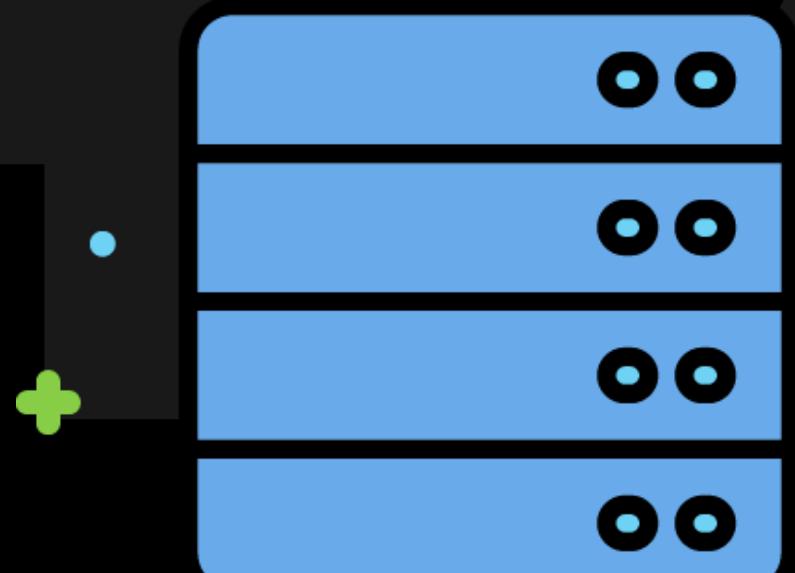


Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT ARE THE BENEFITS OF RELATIONAL DATABASES?

They provide strong consistency, ACID transactions, and support complex queries and relationships.

Using SQL joins to query related data across multiple tables in PostgreSQL.



Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS ACID COMPLIANCE?

ACID stands for Atomicity, Consistency, Isolation, and Durability, which are properties that ensure reliable database transactions.

A banking system using ACID transactions to ensure accurate fund transfers.



Feel free to follow me  
for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS BASE CONSISTENCY?

BASE stands for Basically Available, Soft state, and Eventual consistency, which are properties of some NoSQL databases that prioritize availability over strict consistency.

Using Cassandra for high availability and eventual consistency in a distributed database.

Feel free to follow me for more content

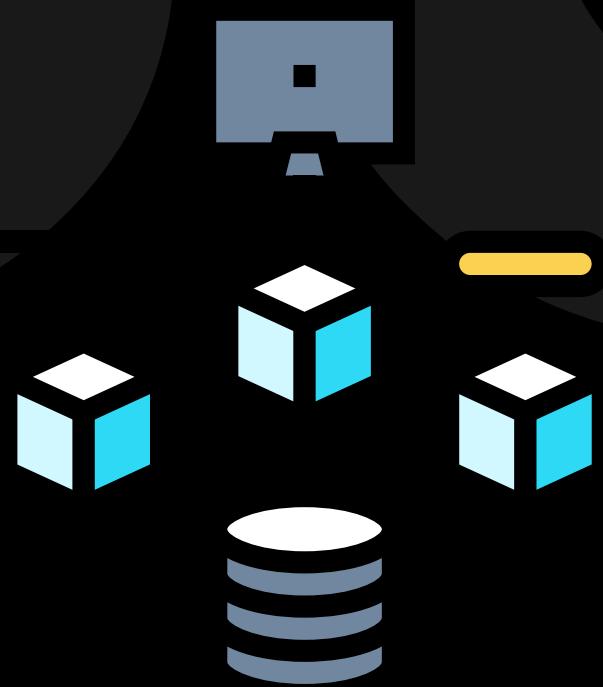


Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A DISTRIBUTED SYSTEM?

A distributed system is a system whose components are located on different networked computers, which communicate and coordinate to achieve a common goal.

Using a microservices architecture with services running on multiple servers.



Feel free to follow me  
for more content

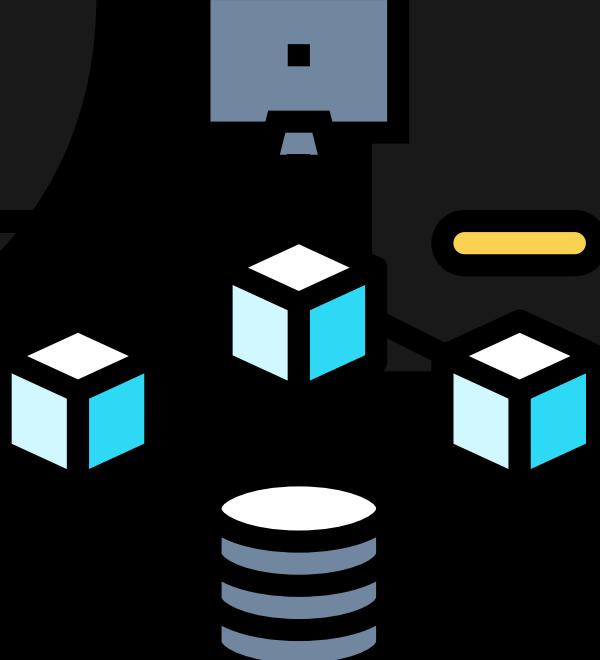


Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT ARE THE CHALLENGES OF DISTRIBUTED SYSTEMS?

Challenges include network latency, fault tolerance, data consistency, and synchronization.

Implementing consensus algorithms like Raft to manage distributed state.



Feel free to follow me  
for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS FAULT TOLERANCE?

Fault tolerance is the ability of a system to continue operating properly in the event of the failure of some of its components.

Using redundant servers and data replication to ensure high availability.

Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# HOW DO YOU ACHIEVE HIGH AVAILABILITY?

Techniques include using load balancers, redundant systems, failover mechanisms, and data replication.

Implementing database replication and load balancing for a web application.



Feel free to follow me  
for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A CONSENSUS ALGORITHM?

A consensus algorithm is a process in computer science used to achieve agreement on a single data value among distributed processes or systems.

Using the Raft algorithm to elect a leader in a distributed system.

Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS THE RAFT ALGORITHM?

Raft is a consensus algorithm designed to be understandable and implementable, used to manage a replicated log in distributed systems.

Implementing Raft for leader election and log replication in a distributed database.

Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS THE PAXOS ALGORITHM?

Paxos is a family of protocols for solving consensus in a network of unreliable or faulty processors.

Using Paxos for achieving consensus in a distributed system with unreliable nodes.

Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A LEADER ELECTION?

Leader election is the process of designating a single node as the organizer of some task distributed among several nodes.

Using ZooKeeper for leader election in a distributed system.

Feel free to follow me  
for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A QUORUM?

A quorum is the minimum number of votes needed for a distributed transaction to be committed.

Using a majority quorum ( $n/2 + 1$ ) in a distributed database to ensure data consistency.



Feel free to follow me  
for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A WRITE-AHEAD LOG (WAL)?

A write-ahead log is a technique used in databases to ensure data integrity by logging changes before applying them to the database.

Using WAL in PostgreSQL to ensure data is not lost during a crash.

Feel free to follow me for more content

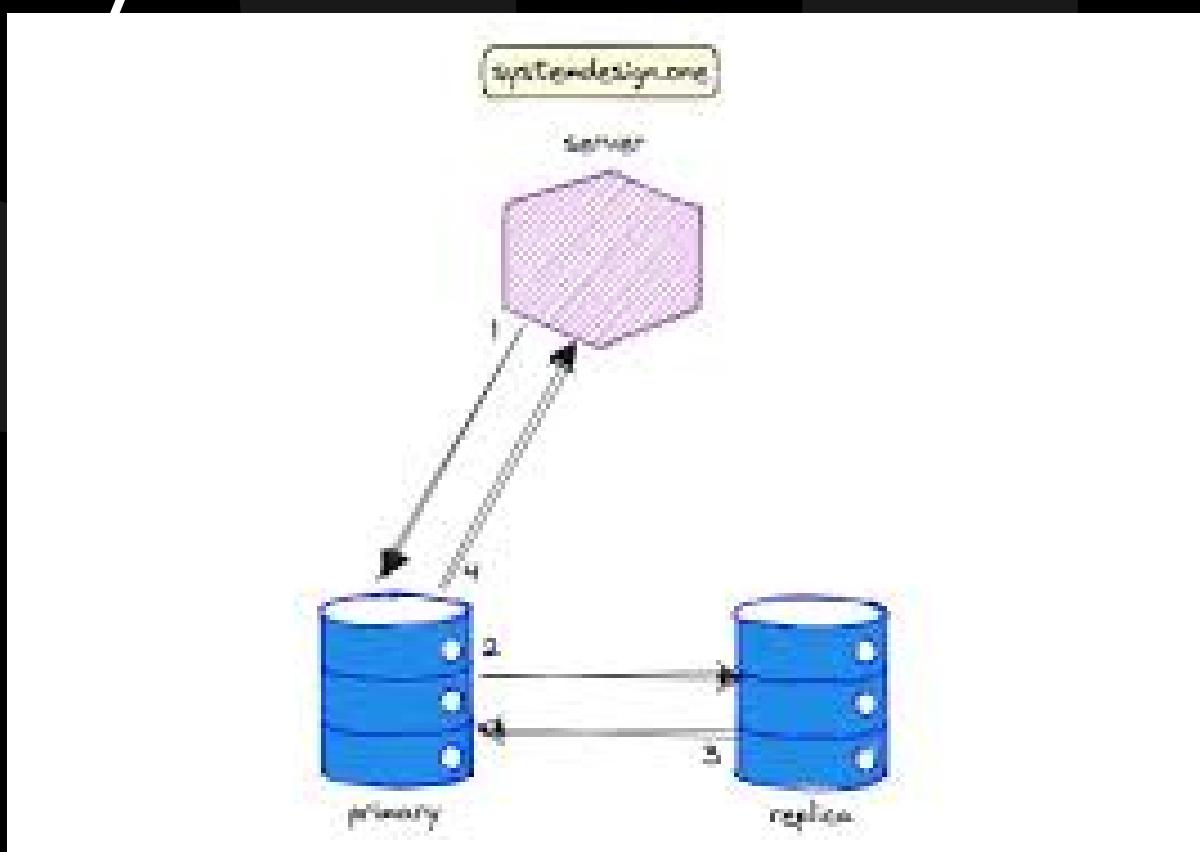


Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS STRONG CONSISTENCY?

Strong consistency ensures that all nodes in a distributed system see the same data at the same time after a write operation.

Using a relational database like PostgreSQL to ensure strong consistency.



Feel free to follow me  
for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A DATA PARTITION?

Data partitioning involves dividing a database into smaller, more manageable pieces, which can be stored across multiple servers.

Sharding a user database by user ID to distribute data across different servers.



Feel free to follow me  
for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A DISTRIBUTED HASH TABLE (DHT)?

A DHT is a decentralized distributed system that provides a lookup service similar to a hash table, where data is distributed across multiple nodes.

Using a DHT for peer-to-peer file sharing in BitTorrent.

Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A GOSSIP PROTOCOL?

A gossip protocol is a method of peer-to-peer communication where nodes periodically exchange state information to achieve eventual consistency.

Using a gossip protocol in Cassandra for node state information exchange.

Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A LEADER-FOLLOWER PATTERN?

The leader-follower pattern is a replication strategy where one node (leader) handles all writes, and other nodes (followers) replicate the leader's data.

Using leader-follower replication in Kafka for high availability.

Feel free to follow me  
for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A DISTRIBUTED LOCK?

A distributed lock is a mechanism to synchronize access to a shared resource across multiple nodes in a distributed system.

Using ZooKeeper to implement distributed locks for coordination.

Feel free to follow me for more content

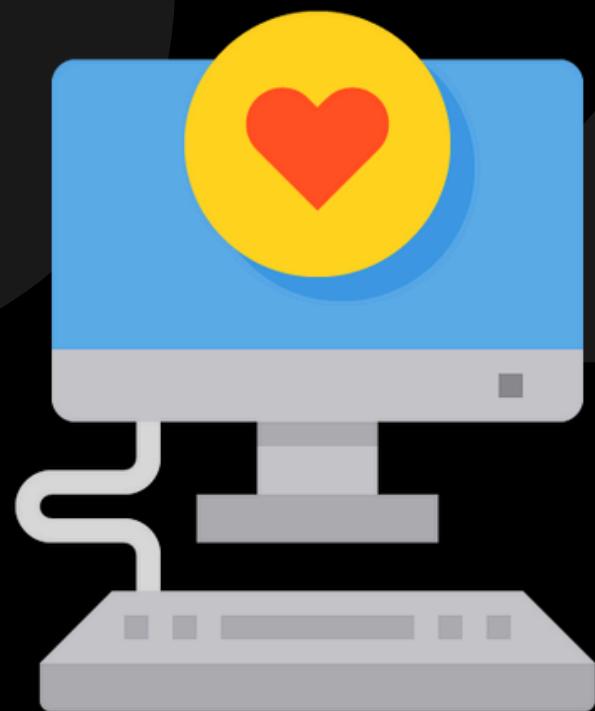


Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A HEARTBEAT IN DISTRIBUTED SYSTEMS?

A heartbeat is a periodic signal sent by a node to indicate its presence and operational status to other nodes in a distributed system.

Using heartbeats in Kubernetes to monitor node health.



Feel free to follow me  
for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A PARTITION TOLERANCE?

Partition tolerance is the ability of a distributed system to continue functioning even when network partitions occur, isolating parts of the system.

Using eventual consistency to handle network partitions in a distributed database.

Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS THE DIFFERENCE BETWEEN SYNCHRONOUS AND ASYNCHRONOUS REPLICATION?

Synchronous replication requires that data be written to all replicas before acknowledging the write, while asynchronous replication allows for acknowledgement before all replicas are updated.

Synchronous replication in a financial transaction system for data integrity vs. asynchronous replication in a content delivery network for performance.

Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A READ REPLICA?

A read replica is a copy of a database that is used to offload read traffic from the primary database, improving performance and availability.

Using read replicas in Amazon RDS to handle increased read traffic.



**Feel free to follow me  
for more content**



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS DATA CONSISTENCY?

Data consistency ensures that data is the same across all nodes in a distributed system after a write operation.

Using strong consistency in a distributed database to ensure accurate data retrieval.

Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A ROLLBACK?

A rollback is the process of undoing changes made to a database during a transaction if an error occurs, ensuring data integrity.

Using rollback in SQL transactions to revert changes if an error occurs during a multi-step process.

Feel free to follow me  
for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A DISTRIBUTED CACHE?

A distributed cache is a caching mechanism where data is stored across multiple nodes to improve scalability and performance.

Using Redis Cluster to distribute cache data across multiple nodes.

Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A CIRCUIT BREAKER PATTERN?

The circuit breaker pattern is a design pattern used to detect failures and prevent cascading failures in distributed systems by stopping the flow of requests to a failing service.

Implementing a circuit breaker in a microservices architecture to handle service failures.

Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A BULKHEAD PATTERN?

The bulkhead pattern isolates components of a system into separate pools to prevent a failure in one component from affecting others.

Using separate thread pools for different services in a microservices architecture to isolate failures.

Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A FALBACK MECHANISM?

A fallback mechanism provides an alternative action or data source when a service call fails, ensuring system resilience.

Returning cached data or a default response when a primary service call fails.

Feel free to follow me  
for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS IDEMPOTENCY?

Idempotency ensures that performing an operation multiple times has the same effect as performing it once, which is crucial for reliable systems.

Using unique transaction IDs to ensure that duplicate payment requests do not result in multiple charges.

Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A SERVICE DISCOVERY?

Service discovery is the process of automatically detecting services and their endpoints in a distributed system, facilitating communication between components.

Using Consul or Eureka for service discovery in a microservices architecture.

Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A PROXY SERVER?

A proxy server acts as an intermediary for requests from clients seeking resources from other servers, providing benefits like load balancing, caching, and security.

Using Nginx as a reverse proxy to distribute traffic to backend servers.

Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A REVERSE PROXY?

A reverse proxy is a type of proxy server that retrieves resources on behalf of a client from one or more servers, often used for load balancing and security.

Using HAProxy to distribute incoming HTTP requests to multiple web servers.

Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A CDN EDGE SERVER?

A CDN edge server is a server located at the edge of a network that stores cached content, delivering it to users from a location closer to them.

Using Cloudflare edge servers to deliver static content to users with low latency.

Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

# WHAT IS A MICROSERVICES GATEWAY?

A microservices gateway is an API gateway that manages and routes requests to the appropriate microservice, often handling cross-cutting concerns like authentication and rate limiting.

Using Kong or Apigee as a microservices gateway to manage API traffic.

Feel free to follow me for more content



Shwetank Singh  
GritSetGrow - GSGLearn.com

Thank  
you!