

## DAILY DSA | DAY-26 | System design principles | -GOPALKRISHNA A

From the past 25 days in our Daily DSA, we have revised the various data structures to efficiently store, access, and retrieval the data and approaches to implement the algorithms and their applications in day-to-day work.

Along with writing better code, it is also equally important to start to nurture the thought process of developing and designing architecture, components, and interfaces to build a solution/system that meets the end-user requirements.

Have you ever wondered thinking about the apps and services that we use daily be it social media like Instagram, Facebook, and search engines are accessed by a billion people worldwide concurrently? This is where system design comes in...

System design is a process of defining the architecture, modules, interfaces, data for a system to satisfy specified requirements. It a multi-disciplinary field that involves trade-off analysis, balancing conflicting requirements, and making decisions about design choices that will impact the overall system.

### **Purpose of system design:**

How do we architect a system that supports the functionality and requirements of a system in the best way possible? The system can be “best” across several different dimensions in system-level design. These dimensions include:

- **Scalability:** A system is scalable if it is designed so that it can handle additional load and will still operate efficiently.
- **Reliability:** A system is reliable if it can perform the function as expected, it can tolerate user mistakes, is good enough for the required use case, and it also prevents unauthorized access or abuse.
- **Availability:** A system is available if it can perform its functionality (uptime/total time). Note reliability and availability are related but not the same. Reliability implies availability but availability does not imply reliability.
- **Efficiency:** A system is efficient if it is able to perform its functionality quickly. Latency, response time and bandwidth are all relevant metrics to measuring system efficiency.
- **Maintainability:** A system is maintainable if it is easy to make operate smoothly, simple for new engineers to understand, and easy to modify for unanticipated use cases.

## Steps for approaching a system design.

1. Understand the requirements: Before starting the design process, it is important to understand the requirements and constraints of the system. This includes gathering the information about the problem space, performance requirements, scalability needs, and security concerns.
2. Identify the major components: Identify the major components of the system and how they interact with each other. This includes determining the relationships between different components and how they contribute to the overall functionality of the system.
3. Choose appropriate technology: Based on the requirements and components, choose the appropriate technology to implement the system. This may involve choosing the hardware and software platforms, databases, programming languages, and tools.
4. Define the interface: Define the interface between different components of the system, including APIs, protocols, and data formats.
5. Design the data model: Design the data model for the system, including the schema for the database, the structure of data files, and the data flow between components.
6. Consider scalability and performance: Consider scalability and performance implications of the design, including factors such as load balancing, caching, and database optimization.
7. Test and validate the design: Validate the design by testing the system with realistic data and use cases, and make changes as needed to address any issues that arise.
8. Deploy and maintain the system: Finally, deploy the system and maintain it over time, including fixing bugs, updating components, and adding new features as needed.

It's important to keep in mind that system design is an iterative process, and the design may change as new information is gathered and requirements evolve. Additionally, it's important to communicate the design effectively to all stakeholders, including developers, users, and stakeholders, to ensure that the system meets their needs and expectations.