**Design goals:**

1. **FLEXIBILITY**

Flexibility embodies the capacity of the system to be reasonably modified, enabling it to evolve and adapt over time according to changing user needs, market dynamics, and to prolong the project's lifespan. Failing to embrace flexibility within the project runs the risk of stagnation and hindering its ability to effectively respond to change.

1. **RELIABILITY**

Reliability refers to the ability of a system to perform its intended function consistently and predictably, without failures or errors, over a specified period of time. It is a measure of how well a system can maintain its operational state and deliver its services or output, even in the presence of various internal or external factors that may cause disruptions or faults.

1. **MAINTAINABILITY**

Maintainability refers to the ease with which a system can be modified, repaired, and enhanced over time. It encompasses design practices and considerations that make it simpler, more efficient, and less error-prone to perform maintenance tasks such as bug fixes, updates, and system improvements

A maintainable system allows for these modifications to be implemented with minimal effort, reducing downtime and associated costs while ensuring the system remains reliable and functional.

1. **PERFORMANCE**

Performance, in the context of system design, refers to the ability of a system to execute its tasks efficiently and deliver optimal response times, throughput, and resource utilization. It involves designing and optimizing the system to ensure it meets or exceeds the performance requirements and user expectations.

1. **SECURITY**

Security assures protection of a system and its data against unauthorized access, malicious attacks, data breaches, and other potential security threats. It involves implementing measures and controls to safeguard the confidentiality, integrity, and availability of the system and its resources.

1. **COMPATIBILITY**

Compatibility refers to the ability of a system to interact, integrate, and coexist with other systems, platforms, or technologies seamlessly. It involves designing the system in a way that enables it to work effectively with various external components, interfaces, or protocols without conflicts or disruptions.

1. **LOW COST**

Low cost refers to designing and implementing a system that achieves its intended objectives while minimizing expenses, both in terms of upfront costs and ongoing operational costs. The goal is to create a cost-effective solution that provides value without compromising quality or functionality.

Designing for low cost is essential because it helps organizations optimize their resources, maximize return on investment (ROI), and make the system more affordable and accessible to a wider range of users.

1. **RAPID DEVELOPMENT**

Rapid development refers to the ability to create and deliver a system quickly, efficiently, and with accelerated development cycles. The focus is on reducing the time it takes to go from the initial concept to a fully functional and deployable system.

Rapid development is important for several reasons. It enables organizations to respond swiftly to market demands, stay competitive, and seize business opportunities. It also allows for faster feedback loops, early user validation, and the ability to iterate and improve the system based on user feedback.