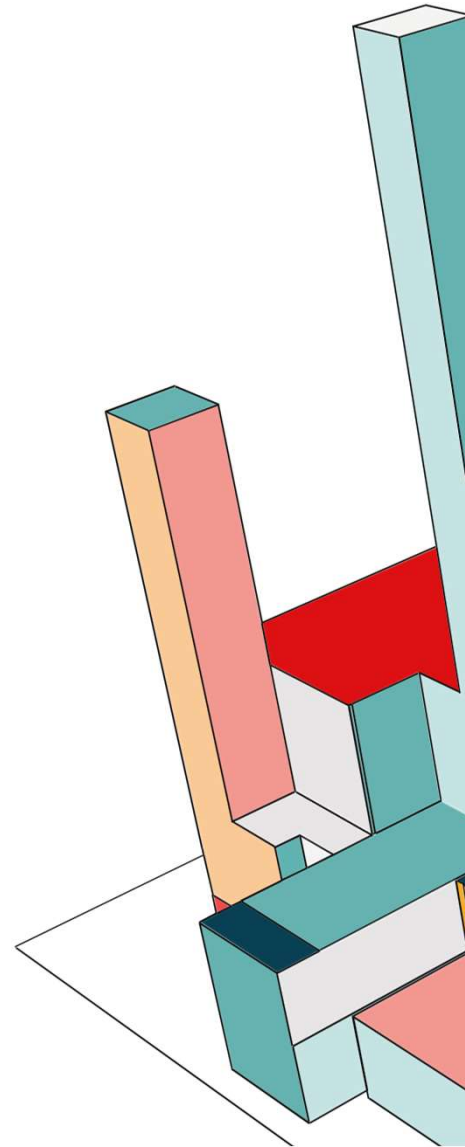


# **SYSTEM REQUIREMENTS**

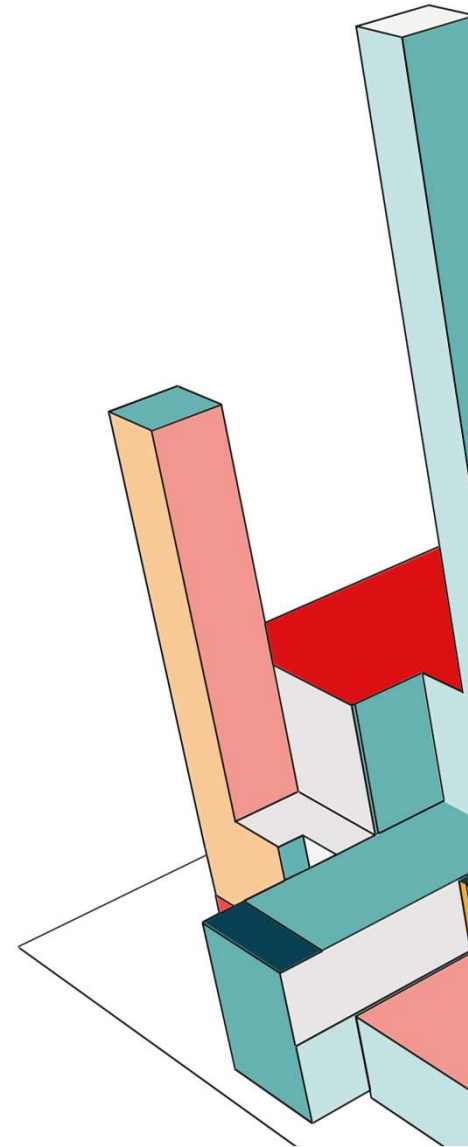
# WHAT IS THE DIFFERENT BETWEEN USER REQUIREMENTS AND SYSTEM REQUIREMENTS?

- User requirements are statements about what the user needs the system to do to solve a specific problem or meet a specific need.
- They are generally written in the language of the user and describe the desired outcomes, features, or functions of the system.
- On the other hand, system requirements are technical descriptions of how the system should be built to meet the user requirements.
- They define the specifications for the system components, interfaces, performance, security, and other technical aspects necessary to build the system.



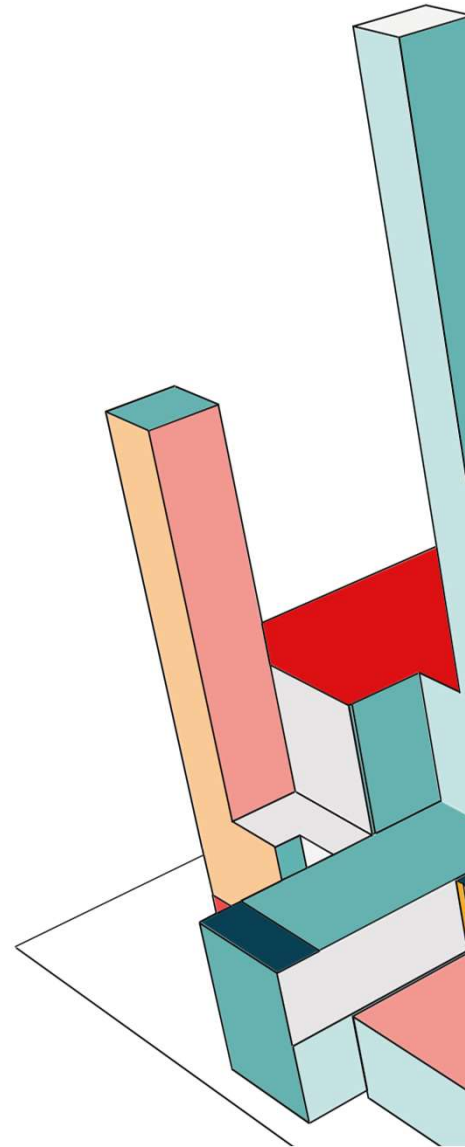
## WHY DISTINGUISH?

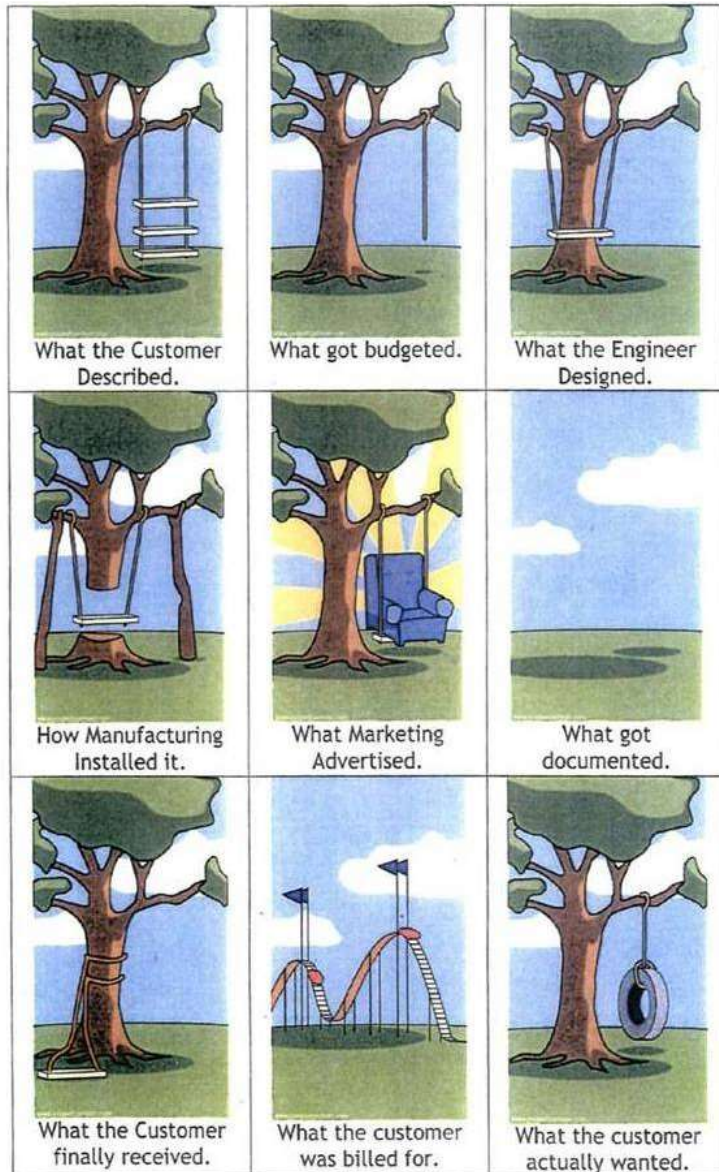
- It is important to distinguish between user requirements and system requirements because they serve different purposes and are addressed to different stakeholders.
- User requirements describe what the user needs and wants from the system, while system requirements describe the technical specifications and constraints that the system must meet to fulfill those user requirements.
- Understanding the difference between the two can help ensure that the system meets the needs of its users while also being technically feasible and efficient.



## KEY REASONS FOR DISTINGUISHING

- Have precision around the perspective, which helps in making more effective trade-offs, as well as ensuring the appropriate lens is used when needed.
- This can lead to better outcomes in software development projects.





## TWO ENDS OF THE SPECTRUM

User Requirements – Wish List,  
what the User Wants, Needs

System Requirements – Builder  
defines exactly what it is that is  
being built.

# DIFFERENTIATE USER AND SYSTEM REQUIREMENTS TO BUILD BETTER SOFTWARE

- Understanding the differences between user requirements and system requirements can be crucial to the success of a software project.
- Failing to accurately capture and distinguish between these requirements can lead to miscommunication, wasted effort, and ultimately project failure.
- By taking the time to properly identify and prioritize both types of requirements, software teams can ensure that they are building the right system for their users and delivering a high-quality product that meets their needs.

