

# Functional requirement

---

In software engineering and systems engineering, a **functional requirement** defines a function of a system or its component, where a function is described as a specification of behavior between outputs and inputs.<sup>[1]</sup>

Functional requirements may involve calculations, technical details, data manipulation and processing, and other specific functionality that define what a system is supposed to accomplish.<sup>[2]</sup> Behavioral requirements describe all the cases where the system uses the functional requirements, these are captured in use cases. Functional requirements are supported by non-functional requirements (also known as "quality requirements"), which impose constraints on the design or implementation (such as performance requirements, security, or reliability). Generally, functional requirements are expressed in the form "system must do <requirement>," while non-functional requirements take the form "system shall be <requirement>."<sup>[3]</sup> The plan for implementing functional requirements is detailed in the system design, whereas *non-functional* requirements are detailed in the system architecture.<sup>[4][5]</sup>

As defined in requirements engineering, functional requirements specify particular results of a system. This should be contrasted with non-functional requirements, which specify overall characteristics such as cost and reliability. Functional requirements drive the application architecture of a system, while non-functional requirements drive the technical architecture of a system.<sup>[4]</sup>

In some cases a requirements analyst generates use cases after gathering and validating a set of functional requirements. The hierarchy of functional requirements collection and change, broadly speaking, is: user/stakeholder request → analyze → use case → incorporate. Stakeholders make a request; systems engineers attempt to discuss, observe, and understand the aspects of the requirement; use cases, entity relationship diagrams, and other models are built to validate the requirement; and, if documented and approved, the requirement is implemented/incorporated.<sup>[6]</sup> Each use case illustrates behavioral scenarios through one or more functional requirements. Often, though, an analyst will begin by eliciting a set of use cases, from which the analyst can derive the functional requirements that must be implemented to allow a user to perform each use case.

## Process

---

A typical functional requirement will contain a unique name and number, a brief summary, and a rationale. This information is used to help the reader understand why the requirement is needed, and to track the requirement through the development of the system.<sup>[7]</sup> The crux of the requirement is the description of the required behavior, which must be clear and readable. The described behavior may come from organizational or business rules, or it may be discovered through elicitation sessions with users, stakeholders, and other experts within the organization.<sup>[7]</sup> Many requirements may be uncovered during the use case development. When this happens, the requirements analyst may create a placeholder requirement with a name and summary, and research the details later, to be filled in when they are better known.