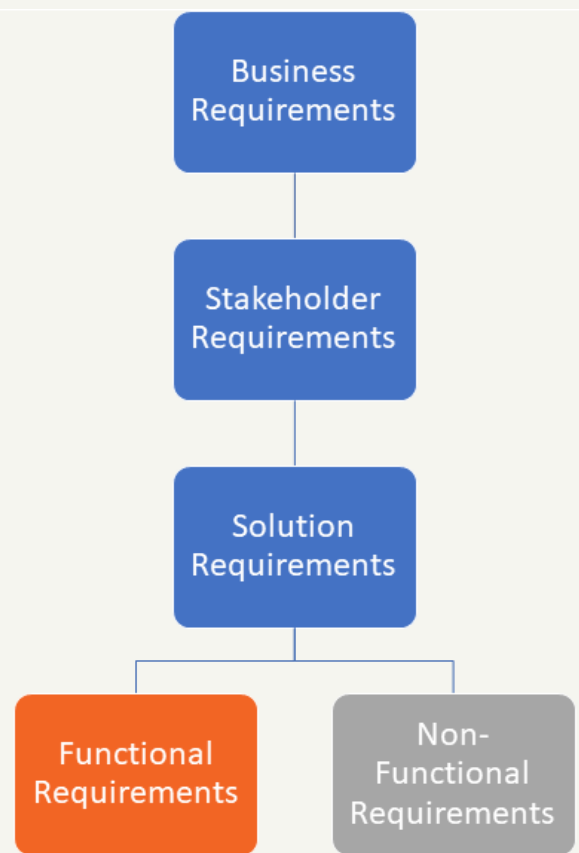


FUNCTIONAL VERSUS NON- FUNCTIONAL REQUIREMENTS

<https://agilecheetah.com>



Define capabilities a product / solution / system must provide its users

Traditional Functional Requirements

Written from the perspective of the product:

The product shall <perform some function>

- Map to higher-level business / mission requirements and stakeholder / customer / end-user requirements
- Business, stakeholder, and solution requirements align but are formulated separately

Agile Functional Requirements

Written from the perspective of stakeholder / user needs. User story example:

As <a user / role / automated client> I want <some capability> so that <desired outcome>

- Stated as what stakeholders / users need to accomplish business outcomes
- Include testable Acceptance Criteria: The results / outcomes functionality must demonstrate when executed

©2021 Agile Cheetah

Define expected product / solution / system-wide behaviors

Describe how the product must behave to satisfy stakeholder / customer / end user needs.

- Behavior attribute examples: Usability, Maintainability, Scalability, Availability, Extensibility, Security, Portability, Quality
- NFRs constrain functional requirements. In other words, capabilities developed to address functional requirements must also support or contribute to meeting NFRs at the system level.
- NFR examples: Specifications, Standards, Policies, Design and Environmental constraints (e.g., authorized technologies, containerized deployments)
- NFRs are realized through implementation of functional requirements and design approaches



AGILE CHEETAH

