**Functional Requirement Specifications**

Functional User Requirements: high-level statements about system behavior

Functional System Requirements: descriptions of system behavior *in detail*

Functional Requirements Describe:

* *What* the system should do
* *How* the system should react
* *How* the system should behave

Typical Form: “The system shall do…”

**Examples**

*“The system shall allow the user to search the appointments lists for all clinics.”*

*“The system shall generate each day, for each clinic, a list of patients who are expected to attend appointments that day.”*

*“The system shall allow each staff member to uniquely identify him/herself by his or her 8- digit employee number.”*

**Functional Requirements Pitfalls**

**Precision**

* Precision is required to take *functional user requirements* to the more detailed, specific domain of a *functional system requirement*.
* Imprecision or ambiguity can lead to complications and implementing the requirement inadequately

**Completeness**

* Functional requirements should describe *all of the system features* which are required

**Consistency**

* Functional requirements should be harmonious amongst themselves; conflicts or contradictions among requirements should be avoided

**Non-Functional Requirement Specifications**

* The primary concern or purpose of a Non-Functional Requirement, is Quantifiability
* *Validation* of Non-Functional Requirements *is the goal*.
* Constant ask yourself, “how can we validate that this requirement has been satisfied”

Non-Functional Requirements Describe:

* Properties of the system *as a whole*, rather than an individual feature
* Constrains on the services or functionality offered (timing constraints, development process constraints, standards requirements, etc.)

Non-Functional requirements:

* Are often linked with functional requirements
* Act as a guiding principle for overall project development, and may directly contribute to the success/failure of a project

**Examples**

Example of System Workload Capabilities

* *“The system shall be capable of handling a typical workload of 10,000 inquires at the same time, 95% of the time”*

Example of System Response Time Capabilities

* *“The system’s response time must not exceed three seconds under typical workload, 99% of the time”*

**Anatomy of a Good Requirement**

Identifies:

* the system under discussion
* the desired end result
* a specified time period for the result to be measured under

The challenge in creating a quality requirement, is to quantify:

* The specific system under discussion
* The desired end result
* Appropriate success metrics for each individual requirement