### What is a requirement?

A requirement in the context of Business Analysis is simply a statement provided by a stakeholder about what they believe they need in order to solve a particular business problem or respond to a specific business need.  Once this requirement has been raised by the stakeholder it is the business analyst’s role to further define, analyze, validate and prioritize the requirement statement as it is now included within the business analysis context of requirements management. In real life, the stakeholder will typically state their business problem or need and then provide a whole range of individual requirements throughout the requirements management process managed by the business analyst.

### Why does the requirement type matter?

There are many different types of these requirements, which are raised by the stakeholders of a particular project. Although most of the requirements are raised typically at the start of the initiative or project, it is also likely that further requirements will be raised during later stages of the project. In the case of an Agile based project, it is likely that requirements will be raised on an ongoing basis although these will also be initiated with most of the significant and larger requirements to be stated early on in the project.

This means that all the different types of requirements that are provided need to be classified within its category in order to be able to manage the requirements process effectively. Once you understand the different types of requirements very clearly, this becomes a natural and easy task to firstly identify when a requirement is in fact a requirement and then to determine what type of requirement it is.

#### There are two sub-types of Solution Requirements:

* **Functional Requirements:** This type of solution requirement describes how the solution must behave. In the example of the house, it describes how the house must look (colors, size of bedroom) and perform (have an air-conditioning unit in each bedroom). In a system related scenario example, the different functions that you want a system to perform is typically described as functional requirements (in an Agile Project context, it is referred to as a ‘user story’). An easy way to remember this type of solution requirement is to think about what do you want to the system to be able to do. Another example in the context of the house would be that a functional requirement exists to have internal doors, which can be opened and closed but not locked. This is something that you want the house to be able to do, a function you want the house to be able to perform.
* **Non-functional Requirements:** The non-functional requirement type of solution requirement describes the characteristics that you want the system to have. In the context of the house example, solution requirement 4 is describing a characteristic that is required of the house walls. It is not a function of the house but rather a characteristic of the walls. In the scenario of a system an example that compares to this house analogy would be a non-functional requirement describing the need to have a backup-system installed to be used in the event of a disaster to prevent unnecessary data loss. The non-functional type of solution requirement therefore describes the attributes a system or process should possess and not a function that the system must perform.

**JAD:-**

JAD stands for Joint Application Development. It's one of those software engineering techniques that some folks with lots of time on their hands sat around and dreamed up. All the design methodologies like this are complicated replacements for a huge dollop of common sense. Sit down with the client and design a paper UI that they can see what the application will look like and behave like. Give the user a chance to work through common scenarios and see if the application will work for them. Keep refining until the user feels the application is doing what they want it to do. As you get functionality implemented, bring the user in and have them work through those scenarios and see if it still works.  
  
On a more serious note, According to Wiki "JAD is a popular Fact-finding technique that brings users into the development process as active participants"

**A Typical JAD session agenda:**

**Project leader:** **1)** Introduce all JAD team members **2)** Discuss ground rules, goals, and objectives for the JAD sessions **3)** Explain methods of documentation and use of CASE tools, if any

**Top management :** Explain the reason for the project and express top management authorization and support.

**Project Leader:** **1)** Provide overview of the current system and proposed project scope and constraints **2)** Present outline of specific topics and issues to be investigated.

**Open discussion session, moderated by project leader:** **1)** Review the main business processes, tasks, user roles, input, and output **2)** Identify specific areas of agreement or disagreement **3)** Break team into smaller groups to study specific issues and assign group leaders.

**JAD team members working in smaller group sessions, supported by IT staff:** **1)** Discuss and document all system requirements **2)** Develop models and prototypes.

**Group leaders:** **1)** Report on results and assigned tasks and topics **2)** Present issues that should be addressed by the overall JAD team

**Open discussion session, moderated project leader:** **1)** Review reports from small group sessions **2)** Reach consensus on main issues **3)** Document all topics

**Project leader:** **1)** Present overall recap of JAD session **2)** Prepare report that will be sent to JAD team members

**Typical JAD's :**

* Business Process Modeling JAD
* Business Rules Definition JAD
* Business Data Modeling JAD
* Requirements Gathering JAD
* Quick Fix Design JAD
* Test Planning JAD

**To sum it up :**

Joint Application Development (JAD) enables a group of people to grasp complex issues quickly and make informed decisions.

The core component of JAD is a structured, facilitated workshop that focuses on creating specified deliverables based on the group's input.

It is an effective tool for planning a project, designing a solution, defining requirements, or any other process that requires consensus-based decision making across functional areas.

**Source / Credits :--Resources on web**