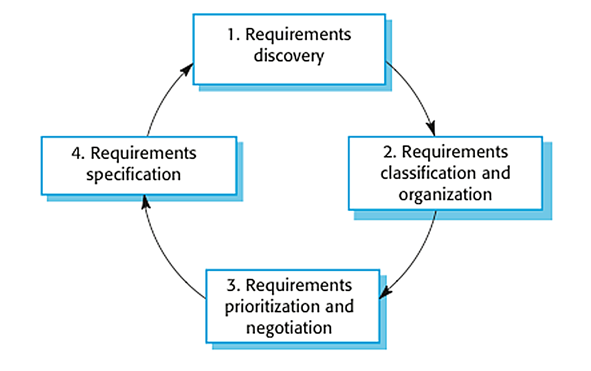
Alexandra P. Veluz

BSIT-4H

**Describe what is requirements elicitation? (Attached an image)**

Requirements elicitation is the method of studying and discovering a system's requirements from users, customers, and other stakeholders in requirements engineering. The term "requirement gathering" is also used to describe the process. We typically start by gathering the requirements, this could be done through a general discussion or interviews with your stakeholders, and also it may involve some graphical notation. Then you organize the related requirements into sub-components and prioritize them, and finally, you refine them by removing any ambiguous requirements that may arise from some conflicts.

1. Requirement Discovery- It's the process of interacting with stakeholders and acquiring requirements for the required system and existing system (if exist). Interviews, scenarios, prototypes, and other tactics can be used to help stockholders understand what the system will look like.

2. Requirements Classification and Organization- It’s very important to organize the overall structure of the system. Putting related requirements together, and decomposing the system into sub components of related requirements. Then, we define the relationship between these components. What we do here will help us in the decision of identifying the most suitable architectural design patterns.

3. Requirements Prioritization & Negotiation- This activity is about prioritizing requirements and identifying and resolving requirements conflicts through conversations until you reach a point where some stakeholders can compromise. We shouldn't get to the point when a stakeholder is dissatisfied because his or her needs aren't being met. Prioritizing your requirements will let you to focus subsequently on the system's basics and key features, allowing you to meet user expectations. It can be accomplished by assigning a priority level to each function. As a result, higher-priority functions require more attention and focus.

4. Requirements Specification- It's the process of compiling user and system requirements into a single document. The requirements should be straightforward, simple to comprehend, comprehensive, and consistent.

**What is a systems requirement in business?**

System requirements, which are generated from business and user requirements, are clearly stated descriptions of what a system must perform in order to satisfy stakeholder demands and requirements. It's also the preparation if a system requires for a hardware or software program to perform smoothly and efficiently. Failure to satisfy these specifications can lead to installation or performance issues. It can prevent a device or application from being installed, but the latter can cause a product to malfunction, perform poorly, hang, or crash. The functional and non-functional types of system requirements should be clearly stated. The system's required behavior and functions are described in functional requirements. Non-functional requirements are particular criteria that can be used to evaluate a system's functioning, such as performance, security, and availability.

**Describe what are the characteristics or features that must be included to satisfy business requirements?**

A Business Requirement needs to meet several criteria to be considered a what is called “good requirement” because better requirements don’t just benefit developers, they help your entire organization’s business value. These are the following characteristics or features that must be included in business requirement:

* **Complete**- Each requirement should include a detailed description of the functionality to be delivered. It must provide all of the information required for the developer to design and implement that feature. There's no rule that says you have to finish all of the prerequisites before starting construction,  but using requirement such as Iterative or incremental development life cycle projects, it should contain a comprehensive set of requirements for each iteration before you proceed to another phases.
* **Correct**- Each need must precisely define the functionality that will be implemented. The source of the demand, such as an actual user or a high-level system requirement, serves as the reference for correctness. It is incorrect to have a software requirement that clashes with its parent system requirement.
* **Feasible**- Each demand must be able to be implemented within the systems and its operating environment's recognized capabilities and constraints. During the elicitation process, have a developer collaborate with marketing or the BA to prevent expressing impossible objectives. To avoid disappointing the owner of the requirement, it is best to identify these requirements as soon as possible.
* **Necessary**- Requirements should document a capability or behavior that is truly required, or specify that the system or product must adhere to certain limits, such as standards.
* **Prioritized**- Each functional need, feature, use case, or user narrative should be assigned an implementation priority to reflect how important it is for a specific product release.
* **Unambiguous**- A requirement should only be able to be interpreted in one way. Ambiguous requirements can cause a project to be delayed, go over budget, or have the wrong functionality or behavior.
* **Verifiable**- A requirement is verifiable if it can be tested to see if it has been met in the implemented system or product. Key to being able to achieve this is knowing which test must be run to verify a particular
* **Atomic**- A single stakeholder demand or a quality trait should be articulated in a requirement. When a requirement has numerous needs, it's impossible to assess each one separately. In the Project Browser, Enterprise Architect can help by allowing modelers to define hierarchies of requirements that can be broken down to an atomic requirement.
* **Cohesive**- The requirements as a whole must be consistent and coherent, and they must express the system's behavior; any gaps must be identified, and overlap between requirements must be resolved.

**References**

<https://medium.com/omarelgabrys-blog/requirements-engineering-elicitation-analysis-part-5-2dd9cffafae8>

<https://www.informit.com/articles/article.aspx?p=1152528&seqNum=4>

<https://www.jamasoftware.com/blog/characteristics-of-excellent-requirements/>