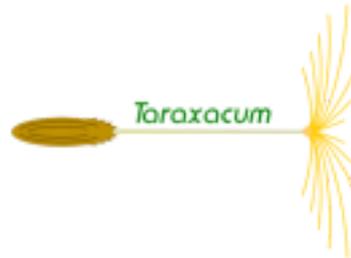




The Nine Principles of Good Requirements Engineering

Q&A from presentation on October 5, 2021 for IIBA Brussels

- **(Q)** What is the best way to document requirements? Office tools or any other?
(from J. P. Malo)
(A) It depends. In early project phases and small projects, office tools like Word and Excel are sufficient. In larger and complex projects, dedicated tool suites are indispensable, preferably tools that are used by analysts, developers and testers alike. In that way, the output of the analyst is directly used as input for the other team members. For instance, in many Agile projects I often see Jira being used as a common tool.
- **(Q)** Regarding NFR - where would you put response speed - in quality or constraints? *(from Elena Nikolskaya)*
(A) Typically, we consider things as response speed to be a quality requirement. IREB refers to the ISO 25010 standard for the categorization of requirements. Constraints refer to requirements that are imposed on the system despite of the needs of the stakeholders. A good example is 'The system must comply with current privacy legislation'.
- **(Q)** How to connect requirements elicitation with the testing process? Use case scenarios or any better way? *(from J. P. Malo)*
(A) The testing process needs clear, detailed requirements. Use case diagrams, use case scenarios and user stories are very suitable to document requirements specifications as an input for the testing process. My personal favorite are acceptance criteria in the BDD-format '**GIVEN** defined preconditions, **WHEN** a specific event occurs, **THEN** an observable behavior is shown'. If an analyst can add such acceptance criteria to the user stories, this is a great help for the tester.
- **(Q)** We mainly focused on principle requirements and good practices, what about bad ones we should avoid? *(from Gianluca Fino)*
(A) In my experience, the worst thing to do is to start from a solution by your client, instead of from a true understanding of the underlying problems and goals. Clients are often in a hurry and think that their proposed solution is best. But this might prevent you from finding better solutions. You should only start thinking about possible solutions after you have completely investigated the whole 'landscape' of interrelated problems and goals.
- **(Q)** Can you suggest some effort estimation methods for requirements elicitation? *(from Aishwarya Rajendran)*
(A) Estimating elicitation is notoriously difficult because you won't have sufficient information at the start. A timeboxed approach where you adjust the estimation after each timebox may be useful. For the first timebox you can refer to experiences from similar projects as an indication.



The IREB Advanced Level Elicitation handbook contains a chapter on estimation, see https://www.ireb.org/content/downloads/13-cpre-advanced-level-elicitation-handbook/advanced_level_elicitation_handbook_en.pdf

- **(Q)** What can you use to better understand the root need of a stakeholder?
Sometimes the "why" isn't answered truly. *(from Bogdan Carp)*
(A) Indeed, this is often the case. Use the '5 Why'- method to dig deeper. Ask questions like "What current problems inhibit you to reach this future goal", or "What benefits do you expect after this current problem is resolved?". See also our article 'The Goal Is to Solve the Problem', <https://www.modernanalyst.com/Resources/Articles/tabid/115/ID/3859/The-Goal-Is-to-Solve-the-Problem.aspx>
- **(Q)** From your presentation, I get the impression that Requirements Engineering is only focussing on the development of software systems. Could it be applied beyond this scope too? *(additional question after the webinar)*
(A) In any situation where one develops a solution based on the needs of stakeholders, good requirements engineering is relevant, be it an insurance system, a funeral or a holiday trip to Ibiza. In the last examples, some techniques may not apply, for example, elaborating a class diagram, but on the other hand, an activity diagram would be suitable. And the nine principles should be adhered to in all cases.
- **(Q)** Your look on the Requirements Engineering process, as a sequence of elicitation, documentation and validation, seems to be quite waterfallish to me. How about Requirements Engineering in Agile? *(additional question after the webinar)*
(A) In principle, Requirements Engineering is process agnostic; it can and should be applied in all types and flavors of development. You always will start with the gathering of information about the needs of the stakeholders: elicitation. As this information has to be shared between clients, users, developers, testers, and other stakeholders, you will always capture it in a structured way: that's documentation. And as you want to be sure that the quality of your documentation is sufficient, a validation step is necessary before you deliver it to the developers. In a waterfall project, one might do this in large trunks of work by specialized analysts with lots of detailed documentation; Agile will do the same in small iterations, based on user stories on a product backlog collected by a Product Owner. The way of working and the techniques may vary, but the principles remain the same.
- **You want more information?**
Download the Handbook for the CPRE Foundation Level for free at https://www.ireb.org/content/downloads/5-cpre-foundation-level-handbook/cpre_foundationlevel_handbook_en_v1.0.pdf

Taraxacum BV, Kockengen | KvK Utrecht 30.142.444 | VAT NL8059.97.209.B.01 | Bank NL18 RABO 0332 8113 87