E Requirements Elicitation

The process of discovering the requirements for a system by communicating with the persons involved in a system is called Requirements Elicitation. The persons involved with the system are for example the customers, system users and all others who have a stake in the system and its production.

To elicit the requirements properly one needs several different areas of domain knowledge as well as knowledge over the organizational structure.

Actions

E.a1 ***Requirements Reuse*** ***REPM* 5**

One should not forget, when developing a new system, to reuse the requirements from other systems developed in the same application area when possible. The positive aspects of reusing requirements are that you save time and money and that you reduce risk since the requirements have already been implemented successfully before. Other areas like design and testing are also to large extent finished when it comes to the reused requirement.

Relation: M.1, M.3

E.1 Stakeholder Identification

The stakeholder identification process is a part of the elicitation process. It means to explicitly identify all potential stakeholders.
Stakeholders are people or organizations who are affected by the system and have a direct or indirect influence over the system requirements. An example of stakeholders are end-users, customers and system users. It is important to remember that the developers of the system, managers etc. also are considered as stakeholders.

Actions

E.1.a1 ***Ask Executive Stakeholders*** ***REPM* 1**

This action is consists of asking the executive stakeholders (the ones ordering the system) who the stakeholders are.

Relation: none

E.1.a2 ***Research Stakeholders*** ***REPM* 2**

Doing your own investigation of who the different stakeholders are.

Relation: none

E.2 Stakeholder Consulting

Another area of the elicitation process is to consult the stakeholders. This is of course directly connected to the process of identifying the stakeholders. After identifying the stakeholders the consulting process is a rather natural step forward to elicit the requirements.

Actions

E.2.a1 ***Executive Stakeholders*** ***REPM*** **1**

These are the stakeholders that have the authority when it comes to the order/purchase of the system being developed. Typically one or two persons represent a group of executives that have taken the decision to order/purchase the system. The representatives are the ones that normally contact you and with whom you speak during the day-to-day work.

Relation: E1

E.2.a2 ***General Stakeholders*** ***REPM*** **3**

General Stakeholders is a much larger group of people. The group is comprised of all parties having a stake in the system that are associated with the customer or the world surrounding the system. Basically all stakeholders are represented here (including Executive Stakeholders), the only exception being stakeholders represented in the developing organization/organizations. End-users are a large group in the General Stakeholder group.

Relation: E1

E.2.a3 ***In-house Stakeholders*** ***REPM*** **2**

The stakeholders grouped here are the ones involved in the development/management of the system within the developing company/companies, e.g. programmers, designers, managers, legal, marketing and so on. This is a group that has a very high stake in the development and management of the system, but they are often overlooked as being stakeholders due to the fact that they are part of the development team.

Relation: E1

E.3 Domain Knowledge

Domain knowledge is the general knowledge of all the different aspects and viewpoints of the system. This area is divided into several sub-areas depending on viewpoint.

Actions

E.3.a1 ***Human Domain Consideration*** ***REPM*** **4**

When talking about the human domain knowledge area one must consider the influence that comes from for example political and organizational factors. These factors can to a very high degree influence the requirements sources. An example of this can be an stakeholder who is reluctant to give you accurate information due to the fact that she feels pressure from the organization that employs her.

Relation: E.2

E.3.a2 ***System Domain Consideration*** ***REPM*** **3**

To be sensitive to the system domain knowledge leads to regarding the requirements coming from the application domain of the system. The system domain should be studied under the elicitation process. Examples of this can be in a banking system there may be accounting regulations that have to be followed. Basically look for domain constraints.

Relation: E.2

E.3.a3 ***Technical Domain Consideration*** ***REPM*** **1**

Technical domain knowledge involves the knowledge of the system's operating environment. The system's operating environment consists of all hardware and other software systems that will interact with the system being developed. This includes other systems present and third-party products like database systems.

Relation: E.2

E.3.a4 ***Business Domain Consideration*** ***REPM*** **4**

General information about the business concerns - how the system will make a contribution to the organization is called the business domain. The knowledge of this area will help you to drive the elicitation process forward when having higher goals to strive for.

Relation: E.2

E.3.a5 ***Operational Domain Consideration*** ***REPM*** **3**

When developing a computer based system the objective is often to support other business processes. These business processes might be something like systems producing customer reports or technical activities such as navigating an aircraft.

When analyzing, understanding and documenting this area one completes the domain knowledge area and drives forward the elicitation process.

Relation: E.2, E.4

E.4 Scenario Elicitation

Using the interaction sessions, known as scenarios, helps the developers see what kind of information is needed and it's presentation - all from the users perspective. Basically what the customers expect, need and how they want it.

Actions

E.4.a1 ***Scenario Elicitation - Executive Stakeholders*** ***REPM*** **2**

To elicit scenarios from Executive Stakeholders and assume that they know what the end-user scenarios look like.

Relation: E.2.a1

E.4.a2 ***Scenario Elicitation - General Stakeholders*** ***REPM*** **4**

To elicit scenarios from General Stakeholders, e.g. end-users (often the largest group), system managers/administrators - in short all parties affected by the system.

Relation: E.2.a2

E.4.a3 ***In-house Scenario Creation*** ***REPM*** **1**

To make scenarios from the developer(s) experiences and the gathered information. This is without directly eliciting the scenarios from stakeholders. The scenarios here are based on what the developer(s) think.

Relation: E.2

A Requirements Analysis and Negotiation

When the initial set of requirements has been elicited the analyzing part begins. You should analyze the requirements for conflicts, overlaps, omissions and inconsistencies.

After the analysis the information received should be reviewed by the different stakeholders and through negotiation a set of requirements should be decided up on. Conflicts must be resolved and the requirements should be prioritized.

Actions

A.a1 ***Analysis Through Checklists*** ***REPM* 1**

Checklists are useful and fairly simple tools that can help you in many areas, of which the analysis is one. Basically it is a list with steps that are more or less mandatory to go through. This way it is easy and clear what should be done for each and every requirement and you do not miss any steps during the analysis.

Relation: M.1.1

A.a2 ***Requirements Classification*** ***REPM* 2**

It is beneficial to classify requirements during the RE process. The two main reasons for this is traceability and identifying related requirements.
Dividing the requirements into classes or groups makes it easier to see which classes/groups are affected if e.g. a change in one requirement takes place. Furthermore you can use a standard classification policy to avoid missing requirements. Let us say that you normally have at least five requirements under every class in a normal case, if one is empty on a project chances are that you have missed something.

Relation: M.2.a2

A.a3 ***Interaction Matrices*** ***REPM* 3**

Interaction between requirements are often many and not always plain to see. The use of matrices makes it easier to discover and document dependencies, conflicts and take problems and benefits into consideration.
It is also possible to observe independent requirements. Interaction matrices can also be useful during the negotiation process as they can show which requirements can cause problems or be difficult to implement due to the interaction with other requirements.

Relation: M.3

A.a4 ***Ambiguous Requirements refinement*** **Opt** ***REPM* 4**

Requirements that are not clear can be clarified by documenting information about what they do not cover. Important to notice is that this should be done when the developers feel that it is appropriate. An example can be a printing function in the system. If a stakeholder assumes that it can print to both paper and file - and the developer knows this but the only thing being implemented is the paper function one could specify that the "print to file" option will not be included in the requirement. This is a rather over simplified example.

Relation: none

A.1 System Boundaries Definition

When starting the Analysis and Negotiation process one should define the system boundaries. This could be done by assessing the initial set of requirements. You should effectively decide which requirement lie in the scope of the system and which are outside.

Actions

A.1.a1 ***Boundary definition through categorization*** ***REPM*** **3**

To evaluate if a requirement is within the boundary of the system (the computer based system you are developing) you can categorize (in association with the Executive and General Stakeholders) the requirements into different groups, preferably System Requirements (inside the system scope), Requirements for the operational processes associated with the system and Requirements clearly outside the scope.
Operational processes requirements are defined by the fact that they are in need of information that lies outside the scope of the designed database of the system and subsequently in need of human decision making.

Relation: E.2.a1, E.2.a2, A.a2

A.2 Requirements Prioritization

During the Analysis and Negotiation one should divide time to go through the requirements with the intention to prioritize them. The priority should be set to reflect the importance the requirement has to the stakeholders and the overall success of the system.

Actions

A.2.a1 ***Prioritizing Requirements*** ***REPM*** **3**

It is imperative that you have a clear and complete picture of the requirements before this step.
Putting the requirements into a fairly small number of priority classifications is preferable to having many (10 vs. 100), e.g. critical, important, useful and desirable.
It is important to realize that different stakeholders may be partial to certain requirements and setting higher priority to them. Logical and Economical arguments must be presented for each and every requirement.

Relation: A.1.a1, A.3, E.2.a1, E.2.a2, E.2.a3

A.2.1 Requirements Re-prioritization

Requirements priority change over time. Change can stem from any number of reasons, of which the most common are covered by the Actions below. It is important to realize this change and to re-prioritize requirements as needed.

Actions

A.2.1.a1 ***Re-prioritization - New Requirements*** ***REPM*** **3**

When a new requirement is introduced there is a need to give it a priority and to take the requirement's impact on other requirements into consideration. It may be a simple matter of just inserting a new requirement and adding additional resources to the project to compensate. On the other hand there may be a need for re-prioritizing all of the requirements due to resource management and/or the fact that a new requirement may change the importance of other requirements.

Relation: A.2.a1

A.2.1.a2 ***Re-prioritization - New Release*** ***REPM*** **3**

New releases can give rise to changes in requirements' priority. New release is a broad term and means everything from a new release of the system being developed, new releases from third-party products to new releases of rules and regulations governing the domain of the system.

Relation: A.2.a1, E.3.a2, E.3.a3, E.3.a5

A.2.1.a3 ***Re-prioritization with Regularity*** ***REPM*** **5**

During a projects life time requirements' priority often change as a rule rather than as an exception. The change does not have to be linked to a certain clear factor, like new releases or new requirements, but may come as a result of changes over time. These changes to environment and human domain factors are hard to spot and require that requirements be re-prioritized regularly. This Action denotes the fact that re-prioritization is done regularly as a point of practice and is not as such not dependent on a change being noticed.

Relation: A.2.a1, E.3

A.2.1.a4 ***Re-prioritization due to Change*** ***REPM*** **4**

This Action denotes re-prioritization due to all changes (except new requirements and new releases covered in previous Actions). The key point being that a re-prioritization is done when a change is noticed, not on a regular basis as a point of practice (covered in A.2.1.a3).
Changes may stem from human domain factors such as changes in the organization (both in the developing company and customer company).

Relation: A.2.a1, E.3

A.3 Requirements Risks

After eliciting the requirements and starting the analysis a risk analysis should be carried out on individual and sets of requirements. This is an excellent way of evaluating possible problems that may arise in the implementation of the requirements. You should also assess the probabilities of these problems arising and the effects of the problems if the do arise.
Important to notice is that risk assessment of individual, sets and selected requirements are not necessarily mutually exclusive but may rather complement each other, e.g. the risk for a set of requirements may differ from the risk of the individual requirements making up the set in question.

Actions

A.3.a1 ***Risk Assessment - individual*** **OG1.01** ***REPM*** **4**

Each individual requirement is assessed according to certain types of risk.
The different types of risk could be; performance risk, safety and security risk, process risk, implementation technology risk, database risk, schedule risk, external risk and stability risk.

Relation: A.a1

A.3.a2 ***Risk Assessment - sets*** **OG1.02** ***REPM*** **4**

Sets of requirements, e.g. predefined classes of requirements (e.g. done in A.a2 Requirements Classification), can be assessed as a group/set. The assessment should be done according to certain types of risks, an example of these was covered in A.3.a1.

Relation: A.a1, A.a2

A.3.a3 ***Risk Assessment - selected*** **OG1.03** ***REPM*** **3**

Certain selected requirements are assessed due to the fact that they are considered within areas of acknowledged risk, such as requirements subject to frequent change.

Relation: A.a1, A.a3, M.4.a1

M Requirements Management

The management process encapsulates all of the Requirements Engineering areas. Management in RE can be compared to Project Management in projects. Documentation, Traceability policies and Change Policies are all examples of areas covered in Requirements Management.

Actions

M.a1 ***Requirements Origin Specification*** ***REPM 2***

It is important to have a clearly documented specification which links the natural language stakeholder requirement to the more detailed models specifying the system. This makes it easier to crosscheck models and associated requirements and reduces the chance for specification drift (the developer specifying the requirements can lose sight of what the system stakeholders really need/want).

Relation: M.1.1, M.3, M.1.2.a1

M.a2 ***Global System Requirements Identification*** ***REPM 3***

The Global System Requirements are requirements that are true for the system as a whole, not any individual system part or subsystem. The information gathered in the areas of E.2.a2 System Domain Consideration, E.2.a3 Technical Domain Consideration and E.2.a4 Business Domain Consideration comprise the foundation for the data needed for identification of Global System Requirements.

Relation: E.3.a2, E.3.a3, E.3.a4

M.a3 ***Rejected Requirements Documentation*** ***REPM 5***

It is important to document all of the requirements, not only the ones to be implemented. Some requirements are dropped or rejected for a wide variety of reasons, e.g. cost and time considerations. They are to be documented. This documentation offers clarity (you have what shouldn't be implemented on paper) as well as material for future reference.

Relation: M.1.1, M.2.a2, M.3

M.1 Requirements Document

The requirements document is a document that communicates the requirement to the customers, system users, managers and system developers. It might be of interest to show the document to all other stakeholders, but that is something that is voluntary.
The Requirements Document is basically a gathering of all the documentation produced during the RE process.

Actions

M.1.a1 ***Record Requirements Rationale*** ***REPM 3***

The rationale is the basis for the requirement. Information about why the requirement was specified in the first place and what function it has should be specified. This specification should be made at an early stage so that the initial rationale is documented.

Relation: M.1.2.a1, M.3

M.1.1 Standardized Document Structure

All the documents written during the requirements engineering process should be of a set standard structure. The specific structure can be directed by a company standard and should be checked as a part of the document quality assurance process.
The Actions below are parts that should be present in a Requirements document.

Actions

M.1.1.a1 ***Document Summary*** ***REPM* 1**

The Requirements Document should have a summary. The summary is a way to get a quick overview of the entire document without having to read it all. A reader can use the summary to quickly decide what parts he/she needs to read further.

Relation: none

M.1.1.a2 ***Document Usage Description*** ***REPM* 2**

This section is included in the beginning of the Requirements Document and gives an explanation of how the Requirements Document should be used/read. Different user groups are taken into consideration and the section should be divided into different parts addressing each user group. Each group should receive the information of how they should use the document for optimal benefit. A section about what is needed of the reader (expert-knowledge etc.) should also be present.

Relation: E.1

M.1.1.a3 ***Business Case*** ***REPM* 3**

The Business Case is basically an explanation of why the system is needed in the business at hand, what the purpose is, what problems are solved and what opportunities are taken advantage of through the system. This does not necessarily include a thorough description of the world/organization surrounding the system but rather the business case where the system is a part.

Relation: E.3.a4

M.1.1.a4 ***Term Definition*** ***REPM* 1**

This is a very important part of the Requirements Document. There are many types of users of the document and they often have different backgrounds. A Term Definition part (i.e. a data dictionary) will give everybody an equal chance to understand the vocabulary used in the document. Also the Term Definition can eliminate some ambiguity when it comes to the interpretation of the document and certain terms.

Relation: none

M.1.2 Describing Requirements

The natural language used to describe requirements should be concise, understandable and unambiguous. The requirements needs to be written in a manner that will help all the readers to immediately understand the requirements meaning and placement.

Actions

M.1.2.a1 ***Requirements Description Template*** ***REPM*** **2**

The use of a template as a way to organize the description of requirements makes for a standardized specification. If a certain way of describing requirements (and what information should be present) is used at all times the reader will be familiar with the way the information is written and can more efficiently absorb the contents.

Relation: none

M.1.2.a2 ***Quantitative Requirements Description*** ***REPM*** **2**

This mainly applies to non-functional requirements such as e.g. user load, hardware utilization and uptime. It is beneficial to quantify (put a figure on) these requirements at an early stage so that they can be taken into consideration when specifying the other requirements e.g. avoiding conflicts between them.

Relation: E.3.a3

M.1.2.a3 ***Unambiguous Requirement Description*** ***REPM*** **1**

Each Requirement should be described using clear and unambiguous natural language. As far as possible a description should not be open to interpretation but rather be clear to a reader.

Furthermore it is important that every requirement be specified separately, i.e. several requirements should not be described in one text body.

Relation: none

M.1.2.1 Descriptive Complements

To best describe a requirement one might have to use other sorts of notation to make the descriptions. This could include mathematical formulas, specialized notation, decision tables or programming language. These complements should always be present if there is a need for a more precise explanation of the requirement.

Actions

M.1.2.1.a1 ***Descriptive Diagrams usage*** **Opt** ***REPM*** **3**

The use of diagrams is another way of structuring the information in a readable way. Diagrams can be a good complement in the Requirements Document. It can visually describe several things like summation of numeric information or sequences of events and activities, e.g. WBS, Gantt, Interaction diagrams and Architectural diagrams.

Relation: none

M.1.3 System Modeling

Another way of simplifying the presentation and description of requirements is to use system modeling. A system model can also help to put the requirements in their proper environment and define new requirements. There are several different types of models that can be used, e.g. Prototypes, System models, Environmental models and Architectural models

Actions

M.1.3.a1 ***Prototyping*** ***REPM* 2**

A prototype is a demonstration system that may show what facilities the system can provide. This can be successfully used in two ways first as a way of refining poorly understood requirements and at a second stage to verify that the requirements are adequately specified and/or correctly understood.
Important to notice is that all of the stakeholders (i.e. the customer) understands that a prototype is only a cosmetic piece of software and not a product that is complete in any way, this to avoid misunderstandings.

Relation: E.2.a3, E.4, M.2.a2

M.1.3.a2 ***System Models*** ***REPM* 3**

System models are models of system specification information. Several models should be made of different parts of the system to simplify the overview. Examples of system models are data-processing models, composition models, classification models, stimulus-response model and process model.

Relation: E.2.a3, E.4, M.2.a2

M.1.3.a3 ***Environmental Models*** ***REPM* 4**

The Environment model is a model of the systems environment. The model includes information of other automated systems that are interfaced with the system being developed, as well as business processes that may use the system. In short information about what lies outside the system.

Relation: E.3.a2, E.3.a3, E.3.a4, M.2.a2

M.1.3.a4 ***Architectural Models*** ***REPM* 5**

Architectural models are overviews of sorts depicting the entire system, the sub-systems and how they are linked. Communication between subsystems is a crucial part of the architectural description.

Relation: M.2.a2

M.1.4 Requirements Validation

After producing the requirements document the requirements should be formally validated. This process is concerned with checking the requirements for omissions, conflicts and ambiguity. It is also the process fo confirming that the right requirements are followed and that the requirement coincide with what the stakeholders have agreed up on.
A large part of requirements validation is concerned with the task of validating that standards are followed.

Actions

M.1.4.a1 ***Requirements Inspection*** ***REPM*** **4**

An inspection is a technique for the detection of errors, violation of development standards and other problems. It is basically a walkthrough. A Requirements Engineer leads the involved through a segment of documentation which are commented upon and then analyzed by the author. The inspection can be a peer-inspection (conducted by people not previously involved in the RE process) or not.

Relation: E.2.a3, M.1.1, M.1.3

M.1.4.a2 ***Requirements Review*** ***REPM*** **3**

Reviews are always conducted by peers (independent from the developement at hand) and they are often contractually bound, e.g. at certain stages (milestones) you decide to have a review to ensure that standards are followed, errors are not present and so on.

Relation: E.2.a3, M.1.1, M.1.3

M.1.4.a3 ***System Model Paraphrasing*** ***REPM*** **5**

This is a sort of reverse engineering. Use the system models, diagrams and so on and convert the requirements into natural language. This to let the stakeholders (primarily the General Stakeholders) understand and comment on the requirements.

Relation: M.1.2.1, M.1.3.a2

M.1.4.a4 ***User Manual Draft*** ***REPM*** **2**

Using the requirements document as a system specification rather than a blueprint and write an initial user manual draft. This can unravel problems and omissions in the requirements specification.

Relation: none

M.1.4.a5 ***Requirements Test Cases*** ***REPM*** **2**

For each requirement there should be at least one proposed test case. This to check for incompleteness and ambiguity – in short to test if the requirement in question is met and implemented in a correct and adequate way in the system.

Relation: M.1.3.a1

M.2 CARE Tool Utilization

With CARE Tool we mean all Computer Aided Requirements Engineering tools, that can help the analyzing and negotiation. This can include graphic tools, text editor programs or communication programs. When encouraging the use of electronic systems to exchange information about the requirements you can make communication easier and faster. You have to observe that the use of CARE tools does not by itself solve communication problems or errors.

Actions

M.2.a1 ***Information Interchange Through CARE*** ***REPM*** **1**

The use of Computer Aided tools in the RE process can be a powerful tool to improve and facilitate communication between stakeholders. Examples of such tools can be everything from simple e-mail communication to video conferencing systems that allow more frequent meetings between all types of stakeholders with minimizing the costs of such meetings.

Relation: none

M.2.a2 ***Information handling Through CARE*** ***REPM*** **1**

The use of Computer Aided tools in the RE process can be a powerful tool to improve and facilitate information handling, e.g. using databases for storing information and modeling tools.

Relation: none

M.3 Traceability Policies

To define what traceability information and how the information should be represented is something that should be stated in the traceability policies and is a part of the management process. Traceability information is information which allows you to find relationships and dependencies between requirements. The relationships can also be defined between the requirements and the system design, components and documentation. In addition information about author/creator of the requirement, and the one responsible for it should be available.
It is important to note that the area of traceability is not covered in detail in this version of the REPM Model - parts like the need for a traceability manual and so on.

Actions

M.3.a1 ***Requirements Identification*** ***REPM*** **2**

Every requirement should have a unique identifier. All information pertaining to a certain requirement should be tagged with this identifier.

Relation: M.2.a2, M.1.2.a1

M.3.a2 ***Backward-from traceability*** ***REPM*** **2**

Links requirement to their source in other documents or people.

Relation: M.3.a1, E.4, M.a1, M.1.a1, M.1.3, E.a1

M.3.a3 ***Forward-from traceability*** ***REPM*** **3**

Links requirements to the design and indirectly implementation components.

Relation: M.3.a1

M.3.a4 ***Backward-to traceability*** ***REPM*** **2**

Links design and implementation components back to requirements.

Relation: M.3.a1

M.3.a5 ***Forward-to traceability*** ***REPM*** **4**

Links other document (which may have preceded the requirements document, e.g. pre-study documents) to the relevant requirements.

Relation: M.3.a1

M.3.a6 ***Version traceability*** ***REPM*** **5**

Handles the different versions and variations of a specific requirement and links them together.

Relation: M.3.a1

M.4 Requirements Change Policies

A requirement change policy should consist of the information of how to manage a change of a requirement, how the change should be proposed, analyzed and reviewed. All this to simplify and effectively change the requirements in the way the stakeholder or other persons involved wants them changed.

Actions

M.4.a1 ***Volatile Requirements Identification*** ***REPM*** **3**

Volatile Requirements are the requirements that are likely to change during the RE process. By identifying these it is possible to foresee and anticipate change and possible problems beforehand.

Relation: A.a2, A.a3, E.3

M.5 Documentation Deliverables

Before, during and after system developement documentation is produced. This documentation can be everything from system models, diagrams, design documents, management documents to user manuals. These are deliverables just like system modules and should be present in the requirements document specified just like a system requirement.
I.e. if you are to deliver certain documentation the requirements that this documentation should satisfy must be specified along with what documents should be present.

Actions

M.5.a1 ***User documentation*** ***REPM*** **3**

This group is comprised of all documents used by the users of the system in question, e.g. user manuals, user dictionaries and so on.

Relation: M.1.1, M.1.2

M.5.a2 ***System documentation*** ***REPM*** **3**

This group covers all system documentation from prestudy to complete system design with all pertaining documents, e.g. Design documentation, technical specifications, use case diagrams and so on.

Relation: M.1.1, M.1.2

M.5.a3 ***Management documentation*** ***REPM*** **4**

This group contains all management documentation to handle the finished system for all kind of upgrades or administrative actions, e.g. how to maintain the system, run diagnostics and optimize the system.

Relation: M.1.1, M.1.2