IREB Certified Professional for Requirements Engineering

- Elicitation and Consolidation, Advanced Level -

Syllabus

Version 1.0
based on German version from December 20 2012

Acknowledgements

This syllabus was created by Oliver Creighton, Dominik Häusser, Kim Lauenroth, Henriette Katharina Lingg, Thomas Mödl, Michael Richter, Chris Rupp, Dirk Schüpferling, Patrick Steiger and Malik Tayeh.

All are thanked for their voluntary commitment.

Foreword

Purpose of the document

This syllabus defines the Advanced Level of the module *Elicitation and Consolidation* of the *Certified Professional for Requirements Engineering* certificate of International Requirements Engineering Board (IREB). The syllabus serves as a basis for training providers to prepare their course materials. Students can use the syllabus to prepare themselves for the examination.

Contents of the syllabus

The module *Elicitation and Consolidation* of the Advanced Level addresses professionals with career profiles like *requirements engineering*, *business analysis*, *business engineering*, *process engineering and organizational design*, who wish to extend their knowledge and skills in the areas of elicitation and consolidation of requirements.

Content scope

In the Advanced Level – as in the Foundation Level– requirements engineering principles are provided that are equally valid for any system – such as embedded systems, safety-critical systems, traditional information systems. This does not mean that the suitability of approaches for the individual areas, accounting for their particularities, cannot be dealt with in a training course. However, it is not the goal to present specific requirements engineering methods of a particular domain.

This syllabus is not based on any specific software development approach and associated process model that makes a statement about the planning, control and sequence of application of the addressed requirements engineering concepts and techniques in practice. It is not intended to particularly emphasize a specific approach, neither for requirements engineering nor for software engineering overall.

It defines what constitutes the knowledge of requirements engineers, but not the exact interfaces with other disciplines and processes of software engineering.

Level of Detail

The level of detail of this syllabus allows internationally consistent teaching and examination. To reach this goal, the syllabus contains the following

- General educational objectives
- Contents with a description of the educational objectives and
- References to further literature (where necessary)

Educational Objectives / Cognitive Knowledge Levels

Each module of the syllabus is assigned a cognitive level. A higher level includes the lower levels. The formulations of the educational objectives (EO) are phrased using the verbs *knowing* for level L1 and *mastering and using* for level L2. These two verbs synonymous with the following verbs:

- **L1 (knowing):** characterize, enumerate, name, recite, recognize
- **L2 (mastering and using):** analyze, apply, assemble, assign, compare, complete, conclude from, describe, design, develop, differentiate, display, elicit, execute, explain, exemplify, formulate, identify, interpret, judge, justify, reflect, suggest, summarize, understand

All terms defined in the glossary have to be known (L1), even if they are not explicitly mentioned in the educational objectives.

In this syllabus the abbreviation "RE" is used for Requirements Engineering**.**

Structure of the Syllabus

The syllabus consists of four main chapters. Each chapter covers one educational unit (EU). Main chapter titles contain the cognitive level of their chapters, which is the highest level of their sub-chapters. Furthermore, the teaching time is suggested that is the minimum a course should invest for that chapter. Important terms within the chapter, which are defined in the glossary (refer to website), are listed at the start of the chapter.

Example: EU 2 Requirements sources (L2)

Duration: 2 hours
Terms: Stakeholder

Educational objectives:
EO 2.1.1 Knowing the classification schema for stakeholders (L1)
EO 2.1.2: Mastering and using tailoring of the stakeholder table for your own project (L2)

EU 2.1 Identify, classify and manage stakeholders

This example shows that in chapter 2 the educational unit *Requirements sources* will be treated in a period of 2 hours. The term *Stakeholder* (i.e. its definition in the glossary) must be known.

The first sub-chapter (EU 2.1) of that educational unit (EU 2) is about identifying stakeholders etc. It contains the educational objective EO 2.1.1 at level L1, which means *knowing* the classification schema for stakeholders. The other educational objective EO 2.1.2 is at level L2, which is about not only knowing the content of how to tailor a stakeholder table but also to be able to master and use that for one's own project.

The Examination

This syllabus is the basis for the examination for the certificate Elicitation and Consolidation, Advanced Level.

A question in the examination can cover material from several chapters of the syllabus. All chapters of the syllabus can be examined.

The format of the examination is multiple-choice as well as an assessed written assignment; the details are set out in the examination regulations.

Examinations can be held immediately after a training course, but also independently from courses (e.g. in an examination center). A list of IREB licensed certification bodies can be found on the website: http://www.ireb.org

Version History

Version	Date	Comment
1.0	December 20 2012	Initial version based on German version 1.0-2 from December 20 2012

Contents

EU 1 Skills of the requirements engineer in the field of elicitation and consolidation (L2)

Duration: 2 hours
Terms: Change Management

Educational objectives:
EO 1.1 Knowing the required skills in the areas of elicitation and consolidation (L1)
EO 1.2 Mastering judging oneself and others on the required skills in the area of elicitation and consolidation (L2)
EO 1.3 Knowing provisions for personal training (L1)
EO 1.4 Mastering and using the communication model of Schulz von Thun (L2)
EO 1.5 Knowing change processes (L1)

EU 1.1 Knowing the required skills in the areas of elicitation and consolidation (L1)

In the Foundation Level, communication skills, analytical thinking, empathy, facilitation skills, confidence and persuasiveness were presented as the required skills or characteristics of a requirements engineer. For the elicitation and consolidation of requirements, the following characteristics are also relevant:

- Motivating nature: beyond the moderation skills called for in the foundation level, the requirements engineer must be able to motivate stakeholders to participate in the elicitation process.
- Leadership: beyond the moderation skills called for in the foundation level, the requirements engineer must be able to guide and lead a group, particularly in conflict situations (for example in the context of a requirements workshop).
- Neutrality: in the elicitation and consolidation process, the requirements engineer must ignore their own wishes and ideas, so that their personal opinion does not become part of the requirements. Furthermore, personal affections and animosities to stakeholders must be avoided.
- Reflective communication: the requirements engineer must be able to reflect continuously on the knowledge acquired from the stakeholders (for example in an interview or in a workshop) in order to identify misconceptions and misunderstandings at an early stage and resolve them.

EU 1.2 Mastering and using judging yourself and others on the required skills in the area of elicitation and consolidation (L2)

The proper assessment of your own abilities and the abilities of others can be made on the one hand by direct behavioral observation and on the other hand by a subsequent analysis. In a direct observation (for example during a workshop) the focus should be placed on one or at most two characteristics, to obtain the most accurate and reliable monitoring result (e.g. observation of one's own reflective communication during an interview).

To assess properly your own skills and the skills of others in a later analysis, the response of other people is an important source (e.g. feedback from customers or colleagues). An assessment sheet on the previously defined capabilities is also a suitable measuring instrument [Smith&Mazin 2004].

EU 1.3 Knowing opportunities for personal training (L1)

The improvement of personal skills for the elicitation and consolidation of requirements is a long-term process, in which personal further development must be regarded as a continuous process. Essential components of this process are:

- Regular measurement of one's own ability profile: an awareness of one's own strengths and weaknesses in terms of the skills profile is the basis for successful further development. Regular analyses (for example through self-assessment questionnaires or conversations with customers or colleagues) of one's own skill profile promote awareness of one's own strengths and weaknesses.
- Training measures: in order to improve your skill profile, further education, training or coaching in relation to one or more of the named characteristics (for example leadership or motivational training) should be carried out.
- Improvement in the everyday work: dedicated training measures are a first step to improving your own skills. Substantial progress, however, is only achieved through application and practice in everyday work. Therefore, the continuous improvement of one's skills must be an integral part in practice. Good progress can be achieved, if over a long period (at least 4 weeks) improvement in a particular skill is focused on (e.g. improving leadership skills in workshops).

EU 1.4 Mastering and using the communication model of Schulz von Thun (L2)

The communication model of Schulz von Thun [Schulz von Thun 2008] describes communication as the *sending* of a *message* from a *sender* to a *receiver* (see Figure 1). In this model, the sent message has four aspects.

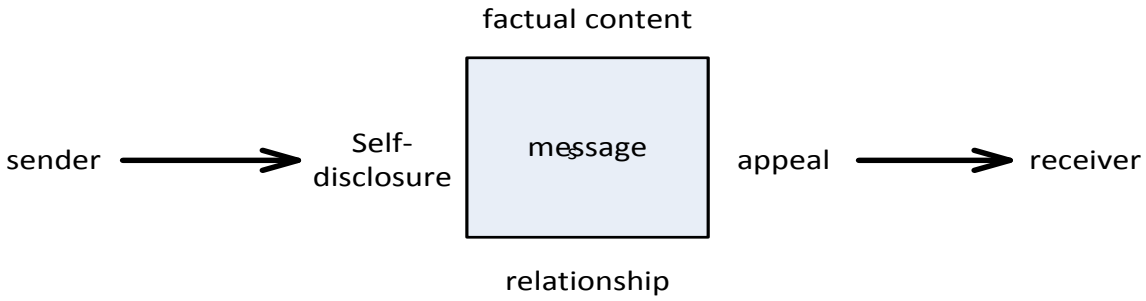

Figure 1 Communication model of Schulz von Thun

The four aspects of a message can be applied as follows to the requirements elicitation:

a) Communication from the stakeholder to the requirements engineer:

- Factual content: the stakeholder expresses information about the requirements for the proposed system.
- Self-disclosure: the self-disclosure expresses the stakeholder's attitude towards a requirement, for instance how important the requirement is for themselves.
- Relationship: the relationship aspect provides information about the stakeholder's attitude to the requirements engineer.
- Appeal: the appeal aspect provides information about what the stakeholder wants to achieve in the current situation, e.g. to deepen the current discussion or terminate it.

b) Communication from the requirements engineer to the stakeholder:

- Factual content: the emphasis of the communication from the requirements engineer to the stakeholder should lie in the factual content, i.e. requirements engineers should communicate their acquired knowledge to the stakeholder (see EU 1.1 Reflective communication).
- Self-disclosure: the communication behavior of the requirements engineer should be as neutral as possible (see EU 1.1 Neutrality).
- Relationship: the requirements engineer should always seek an open and constructive relationship with the stakeholders, to promote an open and content-orientated discussion.
- Appeal: the communication behavior of the requirements engineer should always be geared to motivate stakeholders to provide affirmative or negative feedback about the factual content communicated.

The four aspects of a message can be applied as follows to the consolidation of requirements:

a) Communication from the stakeholder to the requirements engineer:

- Factual content: the stakeholder communicates factual conflict information.
- Self-disclosure: indicates how the stakeholders themselves perceive and evaluate the conflict.
- Relationship: provides information about how the stakeholder perceives the requirements engineer in the conflict situation. This aspect may provide clues that the neutrality of the requirements engineer is not guaranteed.
- Appeal: the appeal aspect provides information on what the stakeholder wants to move the requirements engineer to do in the current conflict situation.

b) Communication from the requirements engineer to the stakeholder:

- Factual content: the communication of the requirements engineer should concentrate as far as possible on the factual portion and represent their actual perception of the conflict.
- Self-disclosure: the communication behavior of the requirements engineer should be as neutral as possible (see EU 1.1 Neutrality).
- Relationship: the requirements engineer should always seek an open and constructive relationship with the stakeholders, to promote an open and content-orientated discussion.
- Appeal: the communication behavior of the requirements engineer should always be geared to motivate stakeholders to provide affirmative or negative feedback about the factual content communicated.

In addition to the previously described aspects of communication between the requirements engineer and stakeholder, the requirements engineer can use the four aspects of a message to analyze the communication between conflicting parties:

- Factual content: the factual content of a message can be examined to uncover factual conflicts.
- Self-disclosure: the self-disclosure aspect can provide information about the extent to which a stakeholder is personally affected by a conflict and thereby give clues to value conflicts.
- Relationship: the relationship aspect of a message can give clues on relationship conflicts and structural conflicts between stakeholders.
- Appeal: the appeal aspect can give clues in a conflict on what the stakeholder wants to achieve and thereby indicate a conflict of interest.

EU 1.5 Knowing change processes (L1)

On the introduction of new systems or processes, friction can be prevented by adequate change management. Already during the elicitation of requirements, the participants are in a process of change. Recognizing this and being able to analyze the behavior of the participants resulting from this, makes it possible to adapt one's own approach and to consider, beside the factual level, the relationship level as well. Thus the elicitation of requirements can happen with less friction and save resources [Cameron&Green 2012], [Reifer 2012].

EU 2 Requirements sources (L2)

Duration: 3.75 hours
Terms: Stakeholder, Usability [1]

Educational objectives:
EO 2.1.1.1 Knowing the classification schema for stakeholders (L1)
EO 2.1.2.1 Mastering and using tailoring of the stakeholder table for your project (L2)
EO 2.1.2.2 Mastering and using assessment of the influence of the project type on the stakeholder relationship management (L2)
EO 2.1.2.3 Knowing stakeholder relationship management for resolving problems with stakeholders (L1)
EO 2.1.3.1 Knowing the outstanding significance of the user as a stakeholder (L1)
EO 2.1.3.2 Mastering and using identification and classification of users as affected by interactive systems (L2)
EO 2.1.3.3 Knowing usability engineering / User-Centered Design (UCD) as a method of user-centered RE (L1)
EO 2.1.3.4 Knowing ISO 9241-210 as a user-centered process model (L1)
EO 2.1.3.5 Knowing user participation in group workshops as a method of user-centered RE (L1)
EO 2.1.3.6 Knowing classical participation methods (L1)
EO 2.1.4.1 Mastering and using creation of personas (L2)
EO 2.2.1.1 Knowing typical document candidates (L1)
EO 2.2.2.1 Mastering and using creation of the documentation schema for the documents used (L2)
EO 2.3.1.1 Knowing typical systems in operation (L1)
EO 2.3.2.1 Mastering and using creation of the documentation schema for the systems in operation being used (L2)

[1] Usability is the outstanding characteristic of a system which best serves its user. The term thus refers to the fitness for use, which is the sum of the system's utility and its ease of use. According to ISO 9241-11 the definition is: *The extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use.*

EU 2.1 Identify, classify, manage stakeholders

Duration: 3 hours

EU 2.1.1 Knowing the classification schema for stakeholders (L1)

A stakeholder is defined as a person (or organization) that has influence on the requirements of the proposed system (see Foundation Level). Stakeholders can be classified on how much they are affected by the new or modified system. That approach is called the onion layer model [Alexander 2005]

- Stakeholders of the system: these stakeholders are directly affected by the new or modified system. Typical examples of that class are users, maintenance personnel and system administrators.
- Stakeholders of the surrounding context: these stakeholders are indirectly affected by the new or modified system. Typical examples of that class are managers of users, project owners or sponsors.
- Stakeholders from the wider context: these stakeholders have an indirect relationship to the new or modified system or to the development project. Typical examples of that class are legislators, standard setting bodies, non-governmental organizations (NGO, e.g. unions or environment protection associations), competitors and those people involved in the development of the system.

Stakeholders are also classified according to their influence on the project (high versus low influence) and their motivation in relation to the project (strong versus weak motivation). Stakeholders with large influence can for example harm the project or advance it. Stakeholders with higher motivation are for example valuable to the project, because they themselves have an interest in advancing the project [Rupp 2009].

EU 2.1.2 Stakeholder relationship management (L2)

The stakeholder table documents relevant information about stakeholders and includes at least the following information (see Foundation Level):

- Name
- Function (role)
- Further personal and contact data
- Time and location availability during the project
- Relevance of the stakeholder
- Their field and depth of knowledge
- Goals and interests in relation to the project

Furthermore, the stakeholder table should contain additional information that is relevant to this project, e.g.:

- Decision-making power / influence on the project
- Motivation in relation to the project
- Relations to other stakeholders
- Preferred form of communication (e.g. by e-mail, telephone or personal meeting)

In defining the additional information, the specifics of the current project are to be considered. Possible influencing factors are:

- Public relevance: in a project with higher public relevance, it may be useful to document how much a stakeholder knows or can influence public opinion.
- Time criticality: in a project with a very strict time frame, the availability or response time of a stakeholder can be very important information when critical decisions are to be taken.

During the project, the stakeholder table must be continuously updated and adapted to the project's circumstances.

Problems with stakeholders typically arise if the rights and obligations of a stakeholder, in respect to the proposed system or the current project, are not clear and transparent. Stakeholder relationship management is an effective technique to counter problems with stakeholders.

An active stakeholder relationship management defines explicitly the rights and obligations of a stakeholder in respect to the project or the proposed system. To the rights of a stakeholder belong, among other things, the right to receive adequate information about the status of the project and about important decisions and developments in the project. To the obligations of a stakeholder belong, for example, adequate availability in time and location for work in the project.

Depending on the nature of the project, it may be useful to formulate a more or less formal stakeholder agreement between the project and the involved stakeholders, to explicitly document the rights and obligations of the stakeholders. In case of problems, this agreement can be called on as a basis for discussion [Bourne 2009], [Rupp 2009].

EU 2.1.3 The user as stakeholder – user-centered RE (L2)

The user as stakeholder

The concept of the user can be defined from different perspectives:

- Business analysis or IT perspective: roles versus users (user groups) of a system
- Marketing perspective: customer, market segments, target groups versus users of a product
- Business or professional perspective: users versus specialist departments of a solution

The characteristics, goals and attitudes of the users are critical to the success of a solution. For example, users of enterprise applications are often the company's specialists, who must use the new system to complete their work and can be trained as needed, whereas users of consumer and mass market products are often laymen who use it by choice and often cannot be trained.

User analysis

The requirements engineer should identify and analyze the users of the system to be specified. This user analysis should capture the characteristics, goals and attitudes of an adequate number of actual users (not their superiors!), aggregate them into user groups and document these, e.g. in the form of a persona per user group (see 0).

If the user groups are known, it can be decided deliberately, which will be considered and how strongly they are considered, or which is the *primary user group* that should be supported by the solution. The system, especially its user interface, will be optimized for these users. For the other user groups it must then be determined whether they should be served *secondarily* through the same system or whether a different solution or a separate user interface must be provided. For some (potential) user groups it may be decided that the system should *not* serve them.

User-centered RE

The following methods support user-centered RE:

- Usability Engineering / User-Centered Design (UCD)
- User participation in group workshops

Usability engineering is a discipline that provides structured methods to guarantee, during the system development, the desired usability [Mayhew 1999]. This includes, in addition to the previously mentioned user analysis, an analysis of the application context and the work processes (e.g. using contextual inquiry, see EU 3.3.3), as well as the definition of the ideal functionality and the design of a suitable user interface [Nielsen 1993].

In usability engineering the user always is in the center, therefore this method is also called user-centered system development or just User-Centered Design (UCD). User-centered RE is a continuous feedback cycle, i.e. an iterative process as described in [InnoSupport 2012], consisting of the analysis of user, task, system and context [Shackel 1991], the elicitation of requirements, their implementation in a prototype and use of the prototype for the evaluation of the analysis and the previously identified requirements as well as the elicitation of new or changed requirements (see following Figure 2):

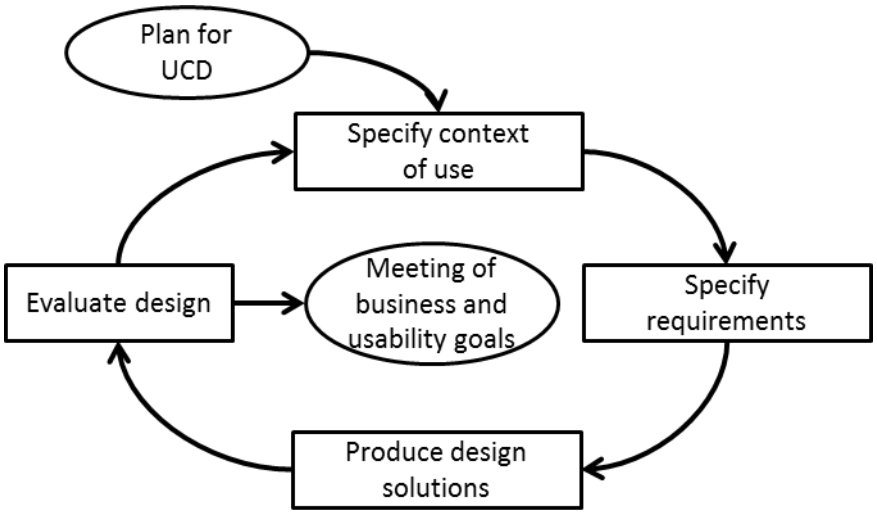

Figure 2 Iterative, user-centered design (UCD) approach according to ISO 9241-210

User participation, i.e. the involvement of users in the requirements elicitation e.g. through group workshops, makes it possible to directly and constructively use their knowledge and commitment. In addition to respectful communication with the participants, methodological knowledge in participatory involvement of groups of various sizes is required, too. This includes on the one hand the design of the context of the group work and on the other hand the nature of the interventions of the facilitator with the group. The aim is that the workshop results are accepted and supported by all participants [Holman,Devane,Cady 2007].

Basic rules of participation:

It is important to prepare the process or intervention correctly, to thoroughly plan the agenda, to conduct it carefully and afterwards to reflect about the process and content. The decision making process and decision rules must be established from the beginning. The course of the process should ensure that everyone can give their input, that it is clear what others have elaborated and that, in long or divided processes, transparency about the state of work should prevail at all times. A basic rule is: Provide room for rational arguments, but also for emotions!

Variants for work in large and small groups:

The decision, which method to apply for a specific participation process depends, among other things, on the number of participants and time available. Also the "participation depth" plays a role, i.e. whether it involves an information, consultation or participation process. The classic methods of participation in organizations are: Open Space, World Cafe, Fish Bowl and dialogue. For further details on these methods refer to [Holman,Devane,Cady 2007].

EU 2.1.4 Persona (L2)

Persona describes a method that provides an analysis and description of the users of the system (see User Analysis in EU 2.1.3). The goal is to concretely communicate to the project team, for whom they are developing the system (clear guidance instead of „elastic" users). The requirements engineer thus knows exactly the primary target group and the secondary target groups and can thus elicit their requirements more effectively and accordingly categories them in a more differentiated way.

In agile software development, personas serve as a tool for creating user stories [Cohn 2004].

Personas are fictitious individuals, representing the typical user groups of the system with similar needs, goals, behaviors or attitudes [Cooper et al. 2007]. The segmentation of the users according to these aspects brings the greatest benefit for the user-centered requirements elicitation.

The data is collected empirically from real users (for instance through interviews, questionnaires or contextual inquiry - see EU 3.2 and 3.3). Through cluster analysis the raw data are aggregated and thereby personas are formed, which are then prioritized (one primary, several secondary personas).

In the communication with stakeholders, also negative personas can be used. These are user groups for which the product is explicitly not being developed.

EU 2.2 Identify, classify, manage documents

Duration: 0.5 hours

EU 2.2.1 Knowing typical candidate documents (L1)

In the requirements elicitation, already existing documents can also be used. With the help of these, various requirements can be extracted. Possible document candidates would include:

- Norms, standards
- Legislation
- Requirements documents
- User manuals
- Strategy papers
- Goal documentation
- Documentation on business processes

EU 2.2.2 Mastering and using creation of the documentation schema for the documents used (L2)

The documents used as requirement sources are documented in a list, similar to the stakeholder list. Here are noted, among other things:

- Document title
- Place where the document is kept
- Version of the document
- Short description (what kind of information the document can provide)

EU 2.3 Identify, classify, manage systems in operation

Duration: 0.25 hours

EU 2.3.1 Knowing typical systems in operation (L1)

Possible sources of requirements are systems in operation. These include among others:

- Competitor systems
- Legacy or predecessor systems
- Systems with similar functions
- Future adjacent systems

EU 2.3.2 Mastering and using creation of the documentation schema for the systems in operation being used (L2)

Those systems used as a requirements source are documented in a list. This list should include, among others, the following points:

- Name of the system
- Type of the system (e.g. competitor system, predecessor system, ...)
- A brief description (what kind of information the system can provide)

EU 3 Elicitation techniques (L2)

Duration: 8.5 hours
Terms: Iterative approach, prototyping, walkthrough, storyboard

Educational objectives:
EO 3.1.1.1 Knowing characteristics of elicitation techniques (L1)
EO 3.1.2.1 Mastering and using classification of elicitation techniques on the basis of characteristics (L2)
EO 3.1.3.1 Mastering and using selection of elicitation techniques on the basis of characteristics (L2)
EO 3.2.1.1 Knowing forms of interview and their differences (L1)
EO 3.2.1.2 Mastering and using formulation of open and closed questions (L2)
EO 3.2.1.3 Mastering and using selected interview questioning techniques (L2)
EO 3.2.1.4 Knowing sources of error and mitigation possibilities (L1)
EO 3.2.2.1 Knowing forms of investigation and their differences (L1)
EO 3.2.2.2 Knowing relevant aspects for creating an inquiry design (L1)
EO 3.2.2.3 Knowing relevant aspects for constructing questionnaires (L1)
EO 3.2.2.4 Knowing the capabilities and limitations of using questionnaires in RE elicitation (L1)
EO 3.3.1.1 Knowing field observation (L1)
EO 3.3.2.1 Knowing apprenticing (L1)
EO 3.3.3.1 Knowing the application, process and goals of the contextual inquiry method (L1)
EO 3.3.3.2 Knowing the aspects observed and collected with contextual inquiry (L1)
EO 3.3.3.3 Knowing the significance of an iterative approach for contextual inquiry (L1)
EO 3.3.3.4 Mastering and using planning of contextual inquiry (L2)
EO 3.4.1.1 Knowing brainstorming and the brainstorming paradox (L1)
EO 3.4.2.1 Mastering and using Method 6-3-5 (L2)
EO 3.4.3.1 Knowing the 6 Thinking Hats Methods (L1)
EO 3.4.4.1 Knowing the analogy technique (L1)
EO 3.4.5.1 Mastering and using Osborn's checklist (L2)
EO 3.5.1.1 Knowing system archaeology (L1)
EO 3.5.2.1 Mastering and using performing perspective-based reading (L2)
EO 3.5.3.1 Knowing re-use of requirements (L1)
EO 3.6.1.1 Knowing the purpose of prototyping for requirements elicitation (L1)
EO 3.6.1.2 Knowing the five most important design aspects for prototypes (L1)
EO 3.6.1.3 Knowing the main eight types of prototypes (L1)
EO 3.6.2.1 Knowing application, process and goals of the method "User Walkthrough" (L1)
EO 3.6.2.2 Knowing the importance of the tasks to be performed in the user walkthrough method on the quality of the results (L1)
EO 3.6.2.3 Mastering and using carrying out a simple user walkthrough (L2)
EO 3.6.3.1 Mastering and using the application, process and goals of scenarios and creation of scenarios (L2)
EO 3.6.3.2 Knowing the characteristics of a good scenario (L1)
EO 3.6.4.1 Knowing application, process and goals of storyboards (L1)
EO 3.6.4.2 Knowing the reasons for visualization with storyboards (L1)
EO 3.6.5.1 Mastering and using application of the elevator pitch (L2)
EO 3.6.6.1 Knowing the importance of goals in the requirements elicitation (L1)
EO 3.6.6.2 Mastering and using goals in requirements elicitation (L2)

EO 3.6.7.1 Knowing the technique *requirements guessing* (L1)
EO 3.6.8.1 Knowing the application, process and goals of the User Stories method (L1)
EO 3.6.8.2 Knowing how acceptance criteria for user stories are formulated and which principle good criteria fulfill (L1)

EU 3.1 Characteristics of elicitation techniques

Duration: 1 hour

EU 3.1.1 Knowing the characteristics of elicitation techniques (L1)

This educational unit is about the characteristics of elicitation techniques. Elicitation techniques are considered as a combination of activities to be performed for the elicitation of requirements in relation to a requirements source.

The activities carried out when applying elicitation techniques can be characterized by the following characteristics: "direction of activity", "activity partner", "nature of the activity" and "frame of reference of the activity".

The characteristic "direction of activity" determines what role is doing the cognitive work of the requirements elicitation. The following types of the characteristic are distinguished:

- Questioning techniques aim to present the stakeholder with questions and tasks that specifically motivate them to formulate requirements. With questioning techniques the cognitive effort lies primarily with the interviewed stakeholders, as they themselves formulate the requirements. Recommended questioning techniques are interviews and questionnaires.
- Demonstrating techniques are aimed at presenting the stakeholder information to encourage them to consent or object, and give them the opportunity to convey their requirements through the demonstration of actions or issues. With demonstrating techniques the cognitive effort lies primarily with the requirements engineers, as they must analyze the explanations and actions of the stakeholders in respect to the requirements. Recommended demonstrating techniques are scenarios, storyboards and prototypes.
- Neutral techniques regarding the direction of activity can be applied as questioning and demonstrating techniques. Examples are workshops and apprenticing.

The characteristic "activity partner" specifies which types of activity partners are addressed by a technique. The following types of the characteristic are distinguished:

- Group-orientated techniques aim at ensuring that requirements are developed within a group in order to take advantage of group-dynamic effects (e.g. mutual inspiration or motivation to participate). Recommended group-orientated techniques are workshops and brainstorming.
- Individual-orientated techniques aim to identify the requirements of individual stakeholders. Recommended individual-orientated techniques are interviews, contextual inquiry and apprenticing.
- Neutral techniques regarding the activity partners can be used both for groups and for individuals. Examples of this are scenarios, field observation and goal elicitation.

The characteristic "nature of the activity" defines in which way the elicitation is carried out. The following types of the characteristic are distinguished:

- Analytical techniques aim to determine the requirements through the analysis of a situation. Recommended analytical techniques are interview, field observation and perspective-based reading
- Involvement techniques aim to identify requirements through the experience of a situation. Recommended involvement techniques are prototyping, apprenticing and storyboard.
- Neutral techniques regarding the nature of the activity can be applied in an analyzing or involving way. Examples are workshops, contextual inquiry and questionnaire.

The characteristic "frame of reference of the activity" defines within which reference system a technique is used. The following types of the characteristic are distinguished:

- Reality-based techniques aim to extract the requirements in the immediate reality of the planned system. Recommended reality-based techniques are contextual inquiry, apprenticing and field observation.
- Projecting techniques aim to elicit requirements out of a fictitious situation. Recommended projecting techniques are scenarios, analogy techniques and storyboard.
- Neutral techniques can be applied in both ways, reality-based as well as projective. Examples are brainstorming and interview.

EU 3.1.2 Mastering and using classification of elicitation techniques on the basis of their characteristics (L2)

The criteria described above can be used for classification of elicitation techniques. In the classification of an elicitation technique it is especially important to observe that an elicitation technique must not necessarily exhibit characteristics in each criterion. Instead, it is also possible that a technique is considered as neutral in respect to one criterion.

For example, the interview is a technique that addresses stakeholders and it is an example of a questioning and individual-orientated technique, because a single stakeholder is questioned in the interview. Furthermore, the interview is an analytical technique, because the factual content of the interview conversation is analyzed. The frame of reference of the interview can be both the reality as well as a fictional situation; hence, regarding this criterion the interview is classified as a neutral technique.

The following table (Figure 3) shows examples of the classification of selected elicitation techniques based on the criteria described above.

	Goal identification	**Apprenticing**	**Brainstorming Paradox**	**Storyboard**
Requirement source	stakeholders	stakeholders, legacy system	stakeholders	stakeholders
Direction of activity	questioning technique	neutral technique	questioning technique	demonstrating technique
Activity Partner	neutral technique	individual-orientated technique	group-orientated technique	neutral technique
Nature of the activity	neutral technique	involvement technique	analytical technique	involvement technique
Frame of reference of the activity	reality-based technique	reality-based technique	neutral technique	projecting technique

Figure 3 Examples for the classification of selected elicitation techniques

EU 3.1.3 Mastering and using selection of elicitation techniques on the basis of their characteristics (L2)

In the selection of appropriate elicitation techniques in a given project situation, the criteria described previously can be used. Depending on the particular project situation, a certain characteristic may be an advantage or a disadvantage, and so elicitation techniques can be particularly recommended or practically unusable in a given project situation. In the following, example project situations are described and their influence on the characteristics of elicitation techniques discussed:

- If stakeholders show little motivation to participate actively in the elicitation process: individual-orientated techniques are likely to help with poorly-motivated stakeholders, because individuals usually are easier to motivate than groups.
- If stakeholders have poor abstraction skills: in this situation, demonstrating techniques are likely to help, since these techniques offer the stakeholder an opportunity to demonstrate their requirements. Furthermore, in this situation involvement techniques can help, because these techniques allow the direct experiencing of requirements.
- High power divide among the stakeholders: in this situation, group-orientated techniques are unsuitable, as these techniques are negatively affected by the imbalance of power among the stakeholders.
- Poor temporal availability of the stakeholders: in this situation, group-orientated techniques are unsuitable because the assembling of the group is difficult due to poor time-availability.
- High complexity of the business matter: in these circumstances demonstrating techniques are less suitable, since the demonstration of complex issues is typically difficult and there is the risk that relevant issues are not recognized.

Based on the assessment of the current project situation, techniques can appear to be particularly suitable or can be explicitly excluded. For example, brainstorming as a group-orientated technique is unsuitable when there is low motivation of the stakeholders.

EU 3.2 Questioning techniques

Duration: 1 hour

EU 3.2.1 Mastering and using qualitative interviews (L2)

Interview in the broadest sense is any form of oral questioning. A distinction is made between the following types of interviews:

- Open (non-standardized): only the topic or rough guidelines are given or unconsolidated collections of questions are used.
- Fully standardized: the order of the questions, their exact wording and their possible answers are given. This interview format is also known as a quantitative interview. It aims to provide comparable results and statistically analyzable quantities.
- Partly standardized: parts of the interviews are given, while others are conducted openly.

In addition to the interview form, two types of potential questions are distinguished:

- With closed-ended questions, the answers are given or defined, either by the question itself or by explicit reference to the available answers.
- Open-ended questions on the other hand allow a free response style, in order to query narrative or argumentative knowledge.

Under the term *qualitative interviews* we group oral interview techniques together that were developed in qualitative social research [Courage&Baxter 2005]. Unlike quantitative interviews, they are not aimed at measuring quantities, but at exploring a specific topic in detail and discovering the background and relationships. In the qualitative interview, the interviewer can deviate from their prepared questions at any time and question answers. Usually specific questioning techniques are used in this context (examples are: dig deeper, filter, summarize, express conclusion). The interviewees can thereby emphasize specific facts and point out important aspects.

The main problems of the qualitative interview technique are:

- Mainly uncontrolled influence of the interviewer
- The risk of interruptions
- Their sometimes narrow interpretational objectivity in the use of content analysis processes.

Frequent sources of errors are:

- Suggestive questions, which imply a specific answer to the interviewee.
- Hypothetical questions, which the interview partner cannot answer from their experience.
- Reply inhibitions and distortions, e.g. due to too personal questions.

These problems can be countered through special training of the interviewer, by using interview guidelines, through organizational measures in the interview preparation as well as through the definition of content analysis evaluation categories prior to the conducting of the interview.

EU 3.2.2 Knowing questionnaires (L1)

Empirical research distinguishes two fundamentally different forms of questioning-based elicitation:

- Quantitative questioning aims to describe numerical occurrences as accurately as possible. For this a representative random sample is questioned and the collected data generalized for the whole population. In general, a hypothesis is set in advance which is to be checked against the results.
- Qualitative questioning aims to determine background, associations and causes. The requirements engineer focusses on the subjective statements of the respondents. In contrast to quantitative methods, the course of action of this method is flexible, open and exploratory.

The conducting of a survey by questionnaire assumes that the enquirer considers exactly, what question is to be answered, how the investigation is to be conducted and how the questionnaire for this must be designed [Kuniavsky 2003], [Robson 2011]. This is also referred to as *research design* and *questionnaire construction.*

Establishing a suitable research design includes a number of key considerations:

- What questions or hypotheses should be answered?
- What is the timetable of the investigation? Is it to be collected only once, will the statements of various groups be compared or will the same users be asked several times at intervals?
- How is the selection of respondents to be made: a random sample or according to certain criteria?
- How many people shall be questioned; should statistically reliable statements be made?
- What are the tools to conduct the investigation? Will an existing questionnaire be used, or will it be elaborated from scratch?
- What conclusions can be drawn from the results as to the requirements for the proposed system?

Construction and use of questionnaires must be methodologically correct to obtain meaningful results. This also applies to simple surveys with a few questions. For the construction of one's own questionnaires for statistically reliable statements, it is worthwhile to involve an expert in questionnaire design.

The following methodological aspects are relevant for the creation of one's own questionnaire, as well as in selecting a standard questionnaire:

- Should open-ended or closed-ended questions be used?
- Are scales (e.g. values of 1-7) to be used? What does the scale mean (e.g. agree - disagree, rates 1-6)?
- How are the instructions given to fill-in the questionnaire?
- Are all the questions understandable for the target group?
- How long does it take to complete the questionnaire? The return rate as well as the quality of response falls with the length of a questionnaire.

Questionnaires can help to clarify the requirements for a new solution. Because requirements analysis is about investigating an area in its entirety, capturing the details of the daily work or exploring causes and connections, this is first and foremost the domain of qualitative methods. However, it may be desirable to clarify certain aspects within a larger set of participants or to target confirmation of questions through quantitative studies. The possibilities of such surveys are however limited. If the questioned people, for example, cannot exactly imagine what the new system is about, the answers are misleading or worthless.

EU 3.3 Observation techniques

Duration: 1.5 hours

EU 3.3.1 Knowing field observation (L1)

In the technique of field observation, the requirements engineer observes the stakeholders at their work activities and so recognizes processes that should be supported by the proposed system. This kind of observation is made preferably in the environment in which the system later will be used. Based on the knowledge gained by the observation, the requirements engineer is enabled to formulate requirements for the proposed system.

EU 3.3.2 Knowing apprenticing (L1)

The technique of apprenticing aims to train the requirements engineer in the work activities and processes that are to be supported or facilitated by the proposed system. Typically, the training is done by an experienced stakeholder and should take place preferably in the environment in which the system will later be used. Based on the knowledge acquired through the training, the requirements engineer is enabled to formulate the requirements for the proposed system.

EU 3.3.3 Mastering and using the use of contextual inquiry (L2)

This technique is a mixture of observation and questioning techniques. A survey in the sense of contextual inquiry is held on-site with the end users and preferably during their work. So the requirements engineer observes the interview partner and makes a point of asking questions about what is being observed. The interview partner should reflect their own course of action and so make explicit the expert knowledge applied. Thereby, the requirements engineer and the interview partner discuss, on the basis of a work task just carried out, problems, factual relationships and opportunities for improvement. The requirements engineer collects any artifacts mentioned in the interview: completed forms, screenshots, sketches of interesting factual relationships, video or audio recordings of conversations and more.

The selection of interview partners needs not be representative in a statistical sense, but should cover a wide spectrum of opinions and needs. It is worth paying attention to a certain distribution with respect to age, gender, position, work location, experience, expertise and cultural background.

Contextual inquiry focuses on the activities of the users and the environment of the application. The following table (Figure 4) shows several aspects that can be captured and documented using this method [Beyer&Holtzblatt 1998]:

Aspect	**Questions posed (example)**
Division of roles and communication	Typical role allocation
	Tasks and responsibilities
	Means of communication
	Communication purpose and content
	Advantages and problems of the division of roles
Strategies and processes	Execution of tasks
	Different approaches
	Strengths and weaknesses
	Incidence, frequency, intensity and duration of execution
	Exceptional circumstances and errors, special cases
Artifacts	Documents, forms, tools, etc. used to complete a task
	Structure and content
	Purpose
	Adaptation to individual needs
	Use for purposes other than intended
	Benefits and problems arising during the work
Cultural and social influences	People with influence
	Effect of social pressure, exercise of power
	Code of conduct
	Goals, values and preferences
	Conflicting influences
	Problems and opportunities at the cultural level
Physical environment	Room layout, work place design
	Available tools
	Routes and distances
	Influence on communication
	Potential for improvement

Figure 4 Various aspects to be elicited in a contextual inquiry

It is important to perform contextual inquiry in an iterative process: the collected findings are analyzed, evaluated and documented in models and RE artifacts (e.g. scenarios, storyboards, etc.). Thus the investigation focus for a next iteration is refined and allows improvement of the already elicited requirements through reflection. Such an iterative approach should already be taken into account during the planning of the method.

EU 3.4 Creativity techniques

Duration: 1.5 hours

EU 3.4.1 Knowing brainstorming and brainstorming paradox (L1)

Brainstorming is a simple method to collect information, but also to create new ideas. Like with most creativity techniques the crucial point of brainstorming is the separation of generating ideas from analyzing the ideas.

Brainstorming is conducted in groups of 5-10 people and lasts about 20 minutes. A moderator ensures the proper conduct of the method.

The process of brainstorming includes the following steps:

- Gathering all ideas (do not judge the ideas in this phase, yet)
- Moderator notes the ideas visibly for all participants
- Carefully analyze/reflect the ideas
- Derive measures

A good overview of brainstorming in RE is provided by Maiden and Gizikis [Maiden&Gizikis 2001].

Brainstorming paradox [Rupp 2009] follows the same procedure as normal brainstorming. The difference is in the brainstorming topic, which is the opposite of the normal brainstorming topic. The aim is to find out, especially, risks and perils regarding a topic.

Brainstorming techniques are particularly well suited to collect many ideas within a short period of time. They are based on the group dynamic effect that the participants inspire each other and therefore come up with new ideas.

EU 3.4.2 Mastering and using the Method 6-3-5 (L2)

The method 6-3-5 [InnoSupport 2012] is a written variant of brainstorming with six people. In this, each of the six participants writes three ideas on a card. After a predetermined time (around 3-5 minutes), each person gives their card to the next person. They let themselves be inspired by the noted ideas and also write three ideas on the topic on the card and pass it on to the next person. This is continued until each person has received each card once. That is, the cards are passed five times. Then the ideas on the cards are evaluated.

This method is recommended especially if it is important to give each participant the same chance to get involved.

EU 3.4.3 Knowing the 6 Thinking Hats Method (L1)

The six thinking hats of Edward de Bono [DeBono 2006] is a multi-perspective creativity technique.

In this technique the six hats represent the different perspectives on a topic:

- Objectivity and neutrality (white): facts and figures
- Personal feelings and subjective opinion (red): feelings, fears, hopes
- Objective, negative arguments (black): doubts, concerns, risks
- Objective, positive arguments (yellow): opportunities, benefits, goals
- New ideas (green): any ideas, similar to brainstorming
- Process control (blue): facilitation and guidance on the idea generation

Each participant receives and wears one of the hats and must argue their point accordingly. These perspectives are taken from each participant in turn. Alternatively, different colored moderation cards can be used.

This technique requires explicitly seeing things from another perspective and therefore helps to overcome rigid ways of thinking.

EU 3.4.4 Knowing the analogy technique (L1)

The analogy technique (e.g. bionics or bisociation [Rupp 2009]) is a creativity technique that helps to come up with ideas for critical and also complex topics. As implied by the name, this technique uses analogies to support thinking and generating ideas.

This technique is applied in the following steps:

- Search for an analogy on the topic
- Present the analogy to the participants
- Introduce the problem for the analogy that is to be solved
- Collect ideas
- Present the original problem
- Apply the ideas to the original problem
- Derive measures

Switching to an analogy removes many inhibitions from the participants, so that even with difficult or sensitive topics good ideas can be developed. However, this technique must be well prepared, as an inappropriate analogy can quickly lead to failure. The inverse transformation of the generated ideas back to the actual topic can also be challenging and can easily lead to wrong conclusions. This technique should not be used by inexperienced requirements engineers.

EU 3.4.5 Mastering and using Osborn's checklist (L2)

Osborn's checklist [OLeary 2008] C. OLeary: Elevator Pitch Essentials. How to create an effective Elevator Pitch. Limb Press LLC, 2008

[Osborn 1979] is a special questionnaire that is intended to inspire the interviewees to come up with new creative ideas. The checklist was originally developed for tangible, material products and is therefore most suitable for such kind of topics.

The checklist includes questions such as:

- Can the product also be used differently?
- Is there something similar to this product, and what would be worthwhile to adopt?
- What can be changed? Is it possible to incorporate other functions?
- How can the product be extended, refined or made more exclusive?
- How can the product be simplified or reduced to basic functions?
- Is it possible to replace the product or parts of it?
- Can you change the product or parts of it, change its sequence or combine it differently?
- Can you also do the opposite with the product?
- Is it possible to combine the product or the idea with something else?
- Can it be used as a building block for something else?
- Can you change its material? Can you compress, liquefy, puncture or otherwise transform it?

These questions lead the participants to take a different perspective on the product.

In the use of the technique not necessarily every question must be answered. It is also recommended that ideas are not necessarily assigned to a specific question.

EU 3.5 Artifact-based techniques

Duration: 1 hour

EU 3.5.1 Knowing system archaeology (L1)

System archaeology is a technique to elicit information regarding a new system from the documentation or the implementation of a legacy or competitor system. The technique is very time-consuming, since existing code or existing documents must be analyzed in order to collect very many detailed requirements. With the help of system archaeology, it can be ensured that no already implemented requirements get lost.

EU 3.5.2 Mastering and using performing perspective-based reading (L2)

Perspective-based reading is a technique for the structured analysis of documents, which can be carried out by both by a requirements engineer or by a stakeholder. In contrast to the usual reading or analyzing of a document, perspectives are defined in advance through which the document should be read. In this way the reader can focus, when studying the document, on the previously defined perspective and identify information in the document in a more targeted way. The perspectives must be defined subject to the document, the proposed system and the goal of the requirements elicitation. Perspectives can, for example, be defined based on requirements types (functional requirements, quality requirements) or based on aspects of the proposed system (e.g. customer database of a web shop, user-friendly functions of a navigation system, etc.) [Pohl 2010].

EU 3.5.3 Knowing re-use of requirements (L1)

Existing, already developed requirements that are, for example, stored in a database or a requirements management tool can be re-used. A particular challenge for re-use is the validation in respect to the validity for the proposed system. By re-use the cost of the requirements elicitation can be significantly reduced. This approach is particularly suitable for non-functional requirements.

EU 3.6 Supporting techniques

Duration: 2.5 hours

EU 3.6.1 Knowing prototyping (L1)

With prototyping, the requirements are visualized (GUI) or physically instantiated (device) and thus made tangible.

The purpose of prototyping in requirements elicitation is the exploration of requirements by encouraging stakeholders to consent or object, or to clarify and amend. Prototyping is therefore a demonstrating elicitation technique (see EU 3.1). In requirements elicitation through prototyping, the following specific aims can be followed:

- Identify new or missing requirements (exploring).
- Search for new possibilities through visualization of innovative, sometimes also crazy or artistic ideas, to provoke reactions and so evolve the initial idea (experimenting).
- Elicit requirements through participation of stakeholders, where they themselves create prototypes (e.g. modeling a device prototype with plasticine or drawing a GUI prototype on a flip chart). They are guided through the creation of the prototype and use it then to derive requirements.
- Elicit contextual requirements or constraints by deploying, e.g., device prototypes in the real context, in order to validate assumptions and to identify non-obvious requirements.

Various methods of prototyping lead to different kinds of prototypes that can be distinguished by the following design aspects:

- Fidelity of the user interface
- Interactional fidelity of the user interface
- Representational fidelity of the data
- Function scope
- Technical maturity of the functionality

This design is carried out in each of the aforementioned aspects as:

- LoFi (Low Fidelity) prototyping:
  Far from the target solution in terms of one of the aspects mentioned above. Without explicit indication, low fidelity prototyping, means in practice, the absence of representational fidelity in the user interface, that is, when a screen mask, for example, is drawn by hand or with wireframes.
- HiFi (High Fidelity) prototyping:
  Close to the target solution in terms of one the aspects mentioned above. Without further specification, high fidelity prototyping means, in practice, a faithfully represented prototype that visually conveys the impression that one has the finished system in front of them.

There are many types of prototypes. The following ones are used frequently in practice:

Physical prototype (device)
If a device is to be developed, already early in the project it can be modeled as a LoFi prototype in regard to visual design, e.g. using plasticine, cardboard, wood or other suitable materials. For a HiFi prototype 3D printers can be used, which automatically create, from shapes designed in the computer, real plastic bodies.

Wizard of Oz / Man behind the curtain
For planned systems with speech-based or tactile user interface, a test user can be led into believe that they are using a realistic system, where actually a person is in an adjacent room (or "behind the curtain") and delivers, on the basis of the user input, sensible answers. So the user tests a system that does not yet exist, without realizing it, and accordingly validates *implemented* requirements under supposedly real conditions and thus discovers still missing requirements [Kelley 1984].

Video-based prototyping
Several observational techniques, especially with the help of the supporting techniques scenario and storyboard, can nowadays be elegantly and clearly combined with videos: users are integrated by performing certain processes in a given situation, which can be filmed and afterwards analyzed in detail. This brings advantages particularly when the processes take place simultaneously, are complex or are located in dangerous or possibly difficult to access environments. In this form of interaction prototyping it is of course possible to use any other type of prototyping as “props”.[Creighton 2006].

Paper and pencil prototyping (hand-drawn sketches)
As soon as the first requirements are on hand, abstract screen designs can be drawn on paper, flip chart or on tablet PCs. The results are hand sketches, i.e. a special kind of wireframe prototype. The advantage of paper and pencil prototyping is the simplicity of the method and that the requirements engineer can focus on the functions, contents and the navigation (rather than the design). So they can, alone or together with the stakeholders in a workshop, quickly create screens that they can discuss with the subject matter experts, or even better with the users, and so validate the requirements and develop the missing requirements.

Clickable mock-up GUI (linked static screens)

With the help of scripting languages (JavaScript, HTML) or with presentation software, static screens are linked and interactivity is simulated through hyperlinks. These mock-ups pursue the same objectives as the paper and pencil prototyping: understanding task flows and application context, validating assumptions and requirements, as well as validating the navigation and the design. With respect to representational fidelity, mock-ups can be implemented either with wireframes (LoFi) or in accordance with the target system (HiFi).

Interactive wireframe GUI prototyping

An interactive wireframe GUI prototype is an abstract prototype of a system, i.e. navigation, headings, lists, functions, etc. are shown in an only rudimentary way (because it is explicitly not about the design), but is made executable in terms of interaction. This type of prototype has the advantage that it is clear to the users that they are looking at a draft and not already a preview of the finished system.

Interactive HiFi GUI prototype

An Interactive HiFi GUI prototype provides, in terms of visual design, a very realistic and in terms of interaction with the user interface, a functional illustration of the future system. This kind of prototype runs the risk that the stakeholders can get the impression that the system is already finished. Moreover, experience shows that a lot of feedback is on details of the visual design, while possibly more fundamental flaws are therefore overlooked.

Programmed, vertical prototyping (technical prototyping)

Only a narrowly limited part of the system functionality is represented, but is implemented throughout all technical layers in order to explore the technical feasibility and thereby reduce risks. Requirements can thus be validated in terms of their practicability and in particular, non-functional requirements are identified and clarified.

EU 3.6.2 Mastering and using application of user walkthrough (L2)

The *user walkthrough* (*usability walkthrough, walkthrough with users, pluralistic walkthrough)* is an open form of evaluation of early prototypes with stakeholders [Sharp et al. 2007]. The main goal of the method is to review and refine requirements using the actual end-user perspective. The method is well suited to study the following aspects of a proposed solution:

- Required information content
- Functionality and processes
- Embedding into the business processes
- Data exchange with other systems and applications
- Representation of tables, graphics, functions, etc.
- Important details of the user interface

The user (also called the test person) in a user walkthrough performs realistic tasks using the system to be tested or a prototype. The moderator (also called test supervisor) has the possibility to intervene directly, ask questions and talk through specific processes with the user.

Other stakeholders (e.g. client, developer) are, depending on the respective agreement, either present as observers or they get actively involved to discuss findings during the walkthrough.

This method is particularly well-suited to evaluate first prototypes early in the process, without already having a running system available. Due to the lack of test supervisor independence, the moderator should know exactly how they can guide the user, without excessively influencing them.

In preparation, the test supervisor and the client develop the tasks (also called test scenarios) that are to be carried out by the users with the prototype. The quality of the results crucially depends on the preparation of these tasks. The preparation of relevant and, from the user's point of view, realistic tasks should therefore be conducted with great care.

The following aspects should be taken into account when developing the tasks:

- The task is a specific scenario and could, from a user's perspective, actually take place in this way.
- A goal is formulated with a view to the application area, not a technical guide for the fulfillment of this goal.
- The tasks should be on a moderate level of difficulty. They should be solvable, but not too trivial.
- Terms and expressions which occur in the application should be avoided

For a user walkthrough, test persons invited should preferably be selected from the application's user group.

EU 3.6.3 Mastering and using scenarios (L2)

Application scenarios, for short scenarios, describe in the form of a realistic example how a user will interact with the proposed system [Rosson&Carroll 2002]. The term scenario should not be confused at this point with the scenario concept in the Use Case methodology (see Foundation Level). Scenarios are created based on the high-level requirements for a new system. They can be developed iteratively or in workshops together with users. A big advantage of scenarios is their ease of understanding. Already at an early stage they can be reviewed, extended or corrected by various stakeholders such as clients, users and subject matter experts. In other words, the requirements engineer models the requirements of a new system using scenarios without anticipating technical details.

The main goals of scenarios are the elicitation and validation of requirements: the reflection on a concrete example allows clients and users to envision requirements in the actual application situation, to review and to supplement them. Scenarios can be considered as the first prototype of a new system.

The following characteristics are typical for a good scenario:

- It is designed for a specific user group, taking into account their characteristics and it meets their needs.
- It presents a concrete use case of the application. Any kind of abstraction should be avoided in order to reveal open issues.
- It shows how the user will use the new system in their real environment.
- It illustrates the relevant aspects for the development of the new solution.
- It is not restricted to the ideal case, but also describes examples of important exceptions and errors to be expected.

EU 3.6.4 Knowing storyboards (L1)

A storyboard shows with the help of the user interface, how a system or product is to be used. It presents visibly important aspects of the application and so serves the communication between all parties involved. Essentially, a storyboard is the visualization of a scenario.

Depending on the elicitation goal, a storyboard can be created in various forms. They range from sketch-like or realistically designed sequences of the user interface (user interface storyboard) to stories in pictures, which also present the application context and persons acting there.

The requirements engineer uses storyboards in situations where text alone is not sufficient. There are two important reasons for such visualization:

- Certain aspects can be conveyed in pictures that cannot, or only with difficulty, be expressed with text, for example, novel concepts for which there are no terms yet.
- Experiences, which are relevant for the application, can be conveyed better into the world of the target audience by using a visual representation.

A storyboard is therefore particularly well-suited to point out the following aspects:

- Dialogue sequences of the user interface
- Difficult to understand concepts and facts
- Important aspects of the application context
- Special or complex environments in which the system will be used

EU 3.6.5 Mastering and using the elevator pitch (L2)

The Elevator Pitch [OLeary 2008], also known as the *Elevator Speech*, is a memorable short presentation of an idea or a request. The goal is to motivate the stakeholder to formulate their requirements concisely and precisely. They are asked to make their presentation in about 30 seconds only verbally and without tools, just like in an elevator. It's not about enumerating the technical details, but about presenting the exceptional aspects of an idea, to arouse emotion and emphasize the benefits.

The given AIDA formula is useful:

- Attention: Attract your target audience
- Interest: Engage their interest
- Desire: Address recipient's need
- Action: Lead recipient towards taking action

EU 3.6.6 Mastering and using goals in requirements elicitation (L2)

Goals are intentional descriptions regarding the proposed system (see Foundation Level). In requirements elicitation a goal describes particularly the additional value which the stakeholder would like to achieve with the proposed system. The determination of goals in requirements elicitation has the advantage that the stakeholders can abstract as much as possible from a proposed solution and concentrate on the added value that should be achieved by the proposed system.

The requirements engineer can support the goal determination by asking "Why?" questions (e.g. "Why is the solution important for you? What should be achieved with it?"). In addition, goals can be categorized into two types:

- Hard goals describe an objectively verifiable additional value.
- Soft goals describe a subjectively verifiable additional value.

In a first step it is often easier to formulate soft goals that can be refined into further soft or hard goals. And/Or trees are suitable for the refinement of goals (see Foundation Level).

For the goal elicitation, the distinction of the AND and OR refinement of goals offers on the one hand the advantage that through the AND refinement absolutely necessary sub goals can be identified. On the other hand the OR refinement offers the possibility of alternative sub-goals and so to identify at an early stage alternative solution paths. At the end of the goal elicitation, whenever practicable, every soft goal should have been refined by hard goals, i.e. in the goal-tree the leaves should only consist of hard goals.

In addition to And/Or refinement there exist numerous other relationships in the goal analysis, for example, goal-supportive relationships or goal-obstructive relationships [Lamsweerde 2009], [Pohl 2010].

An example of a specific goal elicitation technique is the creation of a vision box. Here collectively with the stakeholders, a product packaging is designed and fabricated for the proposed system, even if the target product is not supposed to be sold as shrink-wrapped software in the end. The design is based on common marketing concepts that are used in the packaging of software products. In particular, account is taken of the concise formulation of three or four purchase criteria [Cohn 2004]. Other relevant aspects of the proposed system are described in terms of symbols, pictures, positive user-statements or appropriate success stories and documented on the vision box. In addition to enhanced memorability for the participants, this form of collective, interactive design fulfills many conditions that learning research acknowledges as important for the absorption, evaluation and generation of results [Iles et al. 2002].

EU 3.6.7 Knowing the use of experiential knowledge (L1)

Requirements engineers with sufficient experience in a domain can guess or derive requirements for a proposed system from their accumulated know-how. This technique is applicable, for example, if the proposed system is very similar to already existing systems or if the proposed system is located in a domain that shows strong analogies to the domain of existing systems. Requirements, which are obtained using this technique, must be submitted for a review by stakeholders, in order to detect wrongly defined requirements.

EU 3.6.8 Knowing user stories (L1)

User Stories in agile approaches [Cohn 2004] are widely used both as documentation and elicitation techniques. User stories serve to quickly gain an overview of the planned functionality of the system. User stories themselves are not written specifications, but represent a communication promise. The user story, written on a single card, forms the basis for the communication, which eventually leads to an agreement.

User stories are documented using the following template:

As a <persona/role>

I want <goal>

So that <business value>

Good user stories follow the INVEST principle (Independent, Negotiable, Valuable, Estimable, Small, Testable). Acceptance criteria are defined for every user story. These define under which conditions the user stories are considered to be accepted. Acceptance criteria are documented with the following template:

Given <context>

When <to event occurs>

Then <expected outcome>

Good acceptance criteria fulfill the SMART principle (Specific, Measurable, Achievable, Relevant, Time-bound).

EU 4 Consolidation techniques (L2)

Duration: 3 hours
Terms: none

Educational objectives:
EO 4.1.1 Mastering and using detection of the 5 conflict types on the basis of indicators in the project (L2)
EO 4.1.2 Knowing Glasl's conflict escalation model (L1)
EO 4.1.3 Mastering and using selection of suitable consolidation techniques on the basis of characteristics of the conflict (L2)
EO 4.2.1 Mastering and using use of agreement as a consolidation technique (L2)
EO 4.2.2 Mastering and using use of compromise as a consolidation technique (L2)
EO 4.2.3 Knowing definition of variants (configuration) as a consolidation technique (L1)
EO 4.3.1 Mastering and using voting as a consolidation technique (L2)
EO 4.3.2 Mastering and using overruling as a consolidation technique (L2)
EO 4.4.1 Mastering and using Consider all Facts (CAF) as an analytical support technique (L2)
EO 4.4.2 Mastering and using Plus-Minus-Interesting (PMI) as an analytical support technique (L2)
EO 4.4.3 Mastering and using cost-benefit analysis as an analytical support technique (L2)

EU 4.1 Conflict types

Duration: 1.25 hours

EU 4.1.1 Mastering and using detection of the 5 conflict types on the basis of indications in the project (L2)

Five different types of conflicts are distinguished [Moore 2003]:

- Data conflicts
- Interest conflicts
- Value conflicts
- Structural conflicts
- Relationship conflicts

In the real world, mixed forms of conflict types frequently appear (see Foundation Level).

The first step of consolidation is the detection of conflicts. There are several indicators by which conflicts can be detected. The most commonly encountered indicators are:

- Previously made statements are ignored or changed, as if they had never been made.
- Blind agreement or rejection of statements made by others
- Pedantry
- Statements of others are questioned in minute detail
- Statements are consciously interpreted wrongly
- Information or details are concealed
- The person makes only vague statements, with the demand to others to specify the details.

However, there is not always a conflict present, if one of the indicators occurs.

EU 4.1.2 Knowing Glasl's conflict escalation model (L1)

Friedrich Glasl's model [Glasl 1982] states that a course of conflict can be described by a phase model, which is divided into 9 stages, which in turn are assigned to 3 levels.

Level I: Win – Win

Both parties are still trying to resolve the conflict by mutual consent.

- Stage *hardening*: two opposing viewpoints sometimes harden; both parties assume that the conflict can be resolved through talks.
- Stage *debates and polemics*: black and white thinking sets in, because the positions are perceived as competitive.
- Stage *actions, not words*: the conflicting parties believe that discussions can no longer lead to a consensus. From that moment, the conflict's intensity is ever increasing.

Level 2: Win – Lose

The real conflict has already moved into the background, the assertion of one's own position has become the primary goal.

- Stage *image & coalition*: The parties to the conflict seek supporters and try to tarnish the reputation of the opponent.
- Stage *loss of face:* Compromises become unthinkable for both sides; the opponent's credibility will be undermined by massive damage to their reputation.
- Stage *strategies of threats*: threat and blackmail strategies and the resulting counter-threats lead to escalation of the situation.

Level 3: Lose - Lose

The original conflict theme is already forgotten, the only goal of the dispute left is to harm the opponent.

- Stage *limited destructive blows*: the goal is to inflict more damage on the opponent than they do to oneself.
- Stage *fragmentation of the enemy*: the means of existence of the conflict opponent should be destroyed. Self-preservation is still the ultimate goal.
- Stage *together into the abyss*: fully-fledged war of annihilation, the only goal is to destroy the opponent, no matter how high the costs of this are.

Depending on the degree of hardening of the conflict, different conflict resolution techniques can be used. It is important that all stakeholders involved in the conflict are involved in the solution process in order to prevent a partial resolution of the conflict.

EU 4.1.3 Mastering and using selection of suitable consolidation techniques on the basis of the characteristics of the conflict (L2)

Based on the characteristics of a conflict the right consolidation techniques can be selected. By identifying the circumstances of the real situation, in the majority of cases a sensible technique can be found in the table in
Figure 5, that helps to resolve the conflict quickly and in a way acceptable to all parties.

	Agreement	Compromise	Definition of variants (configurability)	Voting	Overruling
High number of stakeholders	-	-	+	+	+
High criticality of the situation	+	-	-	-	0
Wide distribution of the stakeholders	-	-	+	+	+
High time pressure for conflict resolution	-	-	-	+	+
Clarity of the results is important	+	+	-	+	+
Low social competence of the stakeholders	-	-	+	+	+
Complicated situation	-	+	-	-	-
Long lifetime of the results	+	-	+	-	-
Low motivation of the stakeholders (to take part actively)	-	-	+	+	+
Imbalance of power between the people involved	-	-	0	+	+
Problematic group dynamics	-	-	+	+	+
Many different opinions	-	-	+	+	+
Low communication skills of the stakeholders	-	-	+	+	+
Poor time availability of the stakeholders	-	-	0	+	+
Conflict concerning data	+	+	+	+	-
Conflict concerning relationship	-	-	-	-	-
Conflict concerning value	-	-	+	-	-
Conflict concerning structural	-	0	0	+	+
Conflict concerning interests	-	-	+	0	+

Figure 5 Selection of the suitable consolidation techniques on the basis of the real situation

EU 4.2 Conflict settlement techniques

Duration: 0.5 hours

EU 4.2.1 Mastering and using agreement as a consolidation technique (L2)

Through exchange of opinions, arguments and information, it is possible that all parties to the conflict are convinced by exactly one of the specified solutions. The prerequisite for an agreement is that all people involved disclose their interests and values and are open to the concerns of others. Mutual respect is therefore the indispensable basis of an agreement [Moore 2003]. In RE practice it sometimes helps to establish first agreement about the goals of the proposed system. This makes it then easier to reach agreement on specific requirements [Pohl 2010].

EU 4.2.2 Mastering and using compromise as a consolidation technique (L2)

Through exchange of views and negotiations on solution aspects (a "give and take") a new solution is developed, which takes into account all interests more or less and is therefore acceptable to all sides. The compromise (a "small consent") results – unlike in agreement - in a solution which does not cover all needs to the maximum. It is, however, in some situations the maximum that can be reached, because an agreement is impossible due to legitimate, conflicting requirements.

EU 4.2.3 Knowing definition of variants (configuration) as a consolidation technique (L1)

If neither agreement nor compromise can be reached, it is also possible to implement several solution variants (configurations or parameterizations) in order so to satisfy all stakeholders involved.

However, this results in additional cost for development and additional cost in the subsequent maintenance of each solution variant.

EU 4.3 Voting or instruction methods

Duration: 0.25 hours

EU 4.3.1 Mastering and using voting as a consolidation technique (L2)

The various solution variants are presented to all relevant stakeholders for voting. This should be done secretly. The variant with the highest number of votes is selected as the "winner" and shall be implemented.

Voting is suitable with a large number of participants. Because it is a democratic means of decision making, the result is usually accepted, although with this technique there are always losers.

EU 4.3.2 Mastering and using Overruling as a consolidation technique (L2)

The party with the higher organizational rank prevails. If both parties are on an equal footing, a higher authority must make the decision.

This technique is particularly useful under time pressure or in conflicts at level III (lose - lose) of the Glasl model, if the parties to the conflict can no longer find a constructive solution themselves. This is an authoritarian form of decision making.

EU 4.4 Analytic methods

Duration: 1 hour

EU 4.4.1 Mastering and using Consider All Facts (CAF) as an analytical support technique (L2)

When the support technique *Consider all Facts (CAF)* [DeBono 2006] is applied, all possible influencing factors of a conflict are identified, in order to collect as much information as possible about the conflict and to use it for the requirements consolidation. Then the criteria found are weighted by all of the participants together. Since the analytical techniques primarily suit complex factual conflicts, this support technique is often used for the evaluation of decision criteria. A list of influencing factors / criteria can be created with the help of creativity techniques such as brainstorming.

EU 4.4.2 Mastering and using Plus-Minus-Interesting (PMI) as an analytical support technique (L2)

In *Plus-Minus-Interesting (PMI)* [DeBono 2006] all influencing factors/criteria that were identified for a certain solution with the previous support technique CAF, are examined and their impacts assigned a positive or a negative valuation (a plus or a minus). Impacts, which cannot be assigned to either of the two categories, receive the mark Interesting. Unresolved issues (Interesting) should be discussed in order to determine whether they more concern positive or negative influencing factors.

EU 4.4.3 Mastering and using cost-benefit analysis as an analytical consolidation technique (L2)

In this consolidation technique the best solution variant is determined from the results of the above-described analytical techniques: for each influence factor/criterion a weighting/priority is set and for each solution variant a numeric grading (mark) is given, typically by various parties. The weighting is determined by the conflict parties through conflict settlement processes; the assessment is best carried out by independent experts. Then for each criterion the weight and mark are multiplied and these points are summed-up for each alternative solution. The alternative solution with the most points wins.

	Weight	Solution Variant 1		Solution Variant 2	
		Mark	Points	Mark	Points
Factor / criterion 1	3	1	**3**	6	**18**
Factor / criterion 2	1	-1	**-1**	-4	**-4**
Factor / criterion 3	2	4	**8**	0	**0**
			10		**14**

Figure 1: Example of a cost-benefit analysis

The decision of a cost-benefit analysis can be validated using sensitivity analysis. By means of targeted, slight changes to the marks, it can be identified how stable the original decision is. If the decision would turn out differently through a slight change of a single mark, then the acceptance of such a random decision is at risk.

List of references

[Alexander 2005] I. F. Alexander: A Taxonomy of Stakeholders - Human Roles in System Development. International Journal of Technology and Human Interaction, Vol 1, 1, 2005, pages 23-59.

[Reifer 2012] Donald J. Reifer: Software Change Management: Case Studies And Practical Advice, MICROSOFT PRESS; 1st edition 2012.

[Beyer&Holtzblatt 1998] H. Beyer & K. Holtzblatt: Contextual Design: Defining Customer-Centered Systems, Morgan Kaufmann, San Francisco, 1998.

[Bourne 2009] L. Bourne: Stakeholder Relationship Management: A Maturity Model for Organizational Implementation. Gower Publishing Ltd., 2009.

[Cameron&Green 2012] Esther Cameron and Mike Green: Making Sense of Change Management: A Complete Guide to the Models Tools and Techniques of Organizational Change, Kogan Page; 3rd edition 2012.

[Cohn 2004] M. Cohn: User Stories Applied: For Agile Software Development. Addison-Wesley, 2004.

[Cooper et al. 2007] A. Cooper, R. Reimann, D. Cronin: About Face 3: The Essentials of Interaction Design. Wiley Publishing, 2007.

[Courage&Baxter 2005] C. Courage, K. Baxter: Understanding Your Users: A Practical Guide to User Requirements Methods, Morgan Kaufmann, 2005.

[Creighton 2006] O. Creighton, M. Ott, B. Bruegge. Software Cinema: Video-based Requirements Engineering. 14th IEEE International Requirements Engineering Conference (RE'06), 2006.

[DeBono 2006] E. DeBono: Edward DeBono´s Thinking Course: Powerful Tools to Transform Your Thinking. BBC Active, Harlow, 2006.

[Glasl 1982] F. Glasl: The process of conflict escalation and roles of third parties in G. B. J. Bomers and R. B. Peterson, (eds) Conflict management and industrial relations, (pp. 119-140) The Hague: Kluwer Nijhoff Publishing

[Holman,Devane,Cady 2007] P.Holman, T.Devane, S.Cady: The Change Handbook. The Definitive Resource on Today´s Best methods for Engaging Whole Systems, McGraw-Hill Professional Pub Group West 2007.

[Iles et al. 2002] A. Iles, D. Glaser, M. Kam, J. Canny and E. Do: Learning via Distributed Dialogue: Livenotes and Handheld Wireless Technology, Proc. Conf. Computer Support for Collaborative Learning, January 2002.

[InnoSupport 2012] InnoSupport: Supporting Innovation in SMEs, available for Download at http://www.innovation.lv/ino2/publications/leonardo_manual/en/www.innosupport.net/webhelp/wso/user/firma_3/en04-3.pdf

[ISO 9241-210] International Organization for Standardization (ISO): ISO 9241-210 - Ergonomics of human-system interaction -- Part 210: Human-centred design for interactive systems, International Organization for Standardization, Geneva, 2010 (replaces ISO 13407-210:1999).

[Kelley 1984] J.F. Kelley: An iterative design methodology for userfriendly natural language office information applications. ACM Transactions on Office Information Systems, March 1984, 2:1, S. 26-41.

[Kuniavsky 2003] Kuniavsky, M.: Observing the user Experience: A Practioners's Guide to User Research, Morgan Kaufmann, 2003.

[Lamsweerde 2009] A. van Lamsweerde: Requirements Engineering: From System Goals to UML Models to Software Specifications, John Wiley & Sons, 2009.

[Maiden&Gizikis 2001] N. Maiden, A. Gizikis: Where Do Requirements Come From? IEEE Software 18, 5, 2001, S.10-12.

[Mayhew 1999] D. J. Mayhew: The Usability Engineering Lifecycle, Morgan Kaufmann, San Francisco, 1999.

[Moore 2003] C. Moore: The Mediation Process – Practical Strategies for Resolving Conflicts, 3.Auflage, Jossey-Bass, San Francisco, 2003.

[Nielsen 1993] Nielsen, J.: Usability Engineering. Morgan Kaufmann, 1993.

[OLeary 2008] C. OLeary: Elevator Pitch Essentials. How to create an effective Elevator Pitch. Limb Press LLC, 2008

[Osborn 1979] A.F. Osborn: Applied Imagination, Charles Scribner´s Sons, 1979.

[Pohl 2010] K. Pohl: Requirements Engineering – Foundations, Principles, and Techniques. Springer, 2010

[Robson 2011] Robson, C.: Real World Research. John Wiley & Sons, 2011

[Rohrbach 1969] B. Rohrbach: Kreativ nach Regeln – Methode 635, eine neue Technik zum Lösen von Problemen., Absatzwirtschaft 12, Heft 19, S. 73-75, 1969.

[Rosson&Carroll 2002] M. Rosson & J.M. Carroll: Usability Engineering: Scenario-Based Development of Human Computer Interaction, Morgan Kaufmann, San Francisco, 2002.

[Rupp 2009]: C. Rupp: Requirements-Engineering and –Management –professional, iterative Requirementsanalysis in practice (chapters 4, 5 and 18). The book chapters are available for download at no charge at http://www.sophist.de/en/infopool/downloads/

[Schulz von Thun 2008] S. von Thun: Six Tools for Clear Communication**:** The Hamburg Approach in English Language, 2008.

[Shackel 1991] B. Shackel: Usability – Context, Framework, Definition, Design and Evaluation, in B. Shackel & S. Richardson (Eds.): Human Factors for Informatics Usability (p. 21-37), University Press, Cambridge, UK, 1991.

[Sharp et al. 2007] H. Sharp, Y. Rogers, J. Preece: Interaction Design: Beyond Human-Computer Interaction, Wiley, 2007.

[Smith&Mazin 2004] S. Smith JD, R. Mazin: The HR Answer Book: An Indispensable Guide for Managers and Human Resources Professionals. Amacom Books, 2004.