Connected Care for Elderly Persons Suffering from Dementia

D2.1./D2.2 Performance Requirements for user-centred ICT solutions:

Task 2.1 and Task 2.2.

Classification: internal

Abstract

The goal of work package 2 is to establish performance requirements for interoperable technology solutions for assistive living provision to the elderly in the EU. In this deliverable we describe the performance and quality requirements on interoperable technology solutions for assistive living provision to the elderly. We describe an update of the CCE use cases, the results of an elicitation and prioritization of quality attributes / nonfunctional requirements that was performed within the consortium and with external experts. In this prioritization, efficiency, usability, reliability, maintainability, portability, and security as the most important quality attributes in our context were analyzed.
We also give system level use cases specifying the technical requirements and describe requirements on home automation solutions from the technical point of view.

History of changes

Date	*Version*	*Editor*	*Change description*
2010-11-16	0.0.0	A. Groß, (IESE)	Draft Version of document structure
2010-11-22	0.1	I.John, (IESE)	Document structure first version, Chapter 3 and 4 partially filled
2010-12-20	0.9	S.Hess, A.Gross, I.John (IESE)	Input from different partners integrated, chapter 5,6,7 integrated
2010-12-28	1.0	Michael Eisenbarth (IESE)	Finalization

Table of contents

List of figures

List of tables

1. Introduction

The goal of work package 2 is to establish performance requirements for interoperable technology solutions for assistive living provision to the elderly in the EU. In this deliverable we describe the performance and quality requirements on interoperable technology solutions for assistive living provision to the elderly. We describe an update of the CCE use cases, the results of an elicitation and prioritization of quality attributes / nonfunctional requirements that was performed within the consortium and with external experts. In this prioritization, efficiency, usability, reliability, maintainability, portability, and security as the most important quality attributes in our context were analyzed.

1.1 Non-functional Requirements

Non-functional or quality requirements of products are essential for a successful product development and a high customer acceptance is a key differentiator between a company and its competitors. Requirements Engineering (RE) aims at defining a product that satisfies the customer needs. But in addition to the definition of the functional properties, the customers also request for quality characteristics of the product. They typically ask for products to be secure, stable, usable, and showing high performance. During the system and software development phase, non-functional characteristics are important for deciding on the architecture and for planning and conducting quality assurance activities. To successfully address non-functional characteristics in these phases, it is essential to elicit and capture the non-functional requirements (NFR) during the requirements engineering phase.

Several approaches for dealing with non-functional requirements exist [1][2], but most approaches do not address an integrated approach of modeling functional and non-functional requirements. In practice, the elicitation of non-functional requirements, functional requirements and architecture is not intertwined, leading to missing or inconsistent requirements. In addition, specific approaches exist that address one specific quality attribute (QA), e.g., for usability or security like [3] or [4]. Fraunhofer IESE has developed a systematic, experience-based method to elicit, document, and analyze NFR in conjunction with functional requirements. The goal of the method is to achieve a minimal and sufficient set of measurable and traceable NFR [5]. In the chapter 3, we will briefly describe the Fraunhofer IESE NFR method and present the major quality reference models of the CCE non-functional requirements.

1.2 Outline

In the remainder of this deliverable, we describe an update of the CCE use cases, the results of an elicitation and prioritization of quality attributes / nonfunctional requirements that was performed within the consortium and with external experts. We also give system level use cases specifying the technical requirements and describe requirements on home automation solutions from the technical point of view.

This Deliverable presents the content of Task 2.1 and Task 2.2. Chapter 2-6 are related to Task 2.1., Chapter 7 describes the contents of Task 2.2.

2. Summary of functional use cases

This chapter gives a summary of functional use cases. The use cases have been described in [Del1.3], but the requirements and use cases have been updated meanwhile, so in this deliverable we give an update. Before we describe the use cases we give a short sketch of the CCE solution layout that will be deepened in [Del3.1], [Del4.1] and [Del5.1] and describe the application on a high level.

2.1 CCE Solution Layout

The system generally consists of the following hardware and software components:

- Phillips netTV that provides a user interface for a digital corkboard
- A digital corkboard application
- A set of sensors that monitor the behavior and the activities of the assisted person
- A medical dispenser
- A dementia diary that documents daily activities for the assisted person
- A middleware platform that integrates all of the data

The high level components of the system are depicted in Figure1 which illustrates the current CCE solution layout. Further information on the architecture and technical features of the CCE solution can be found in [Del3.1], [Del4.1] and [Del5.1].

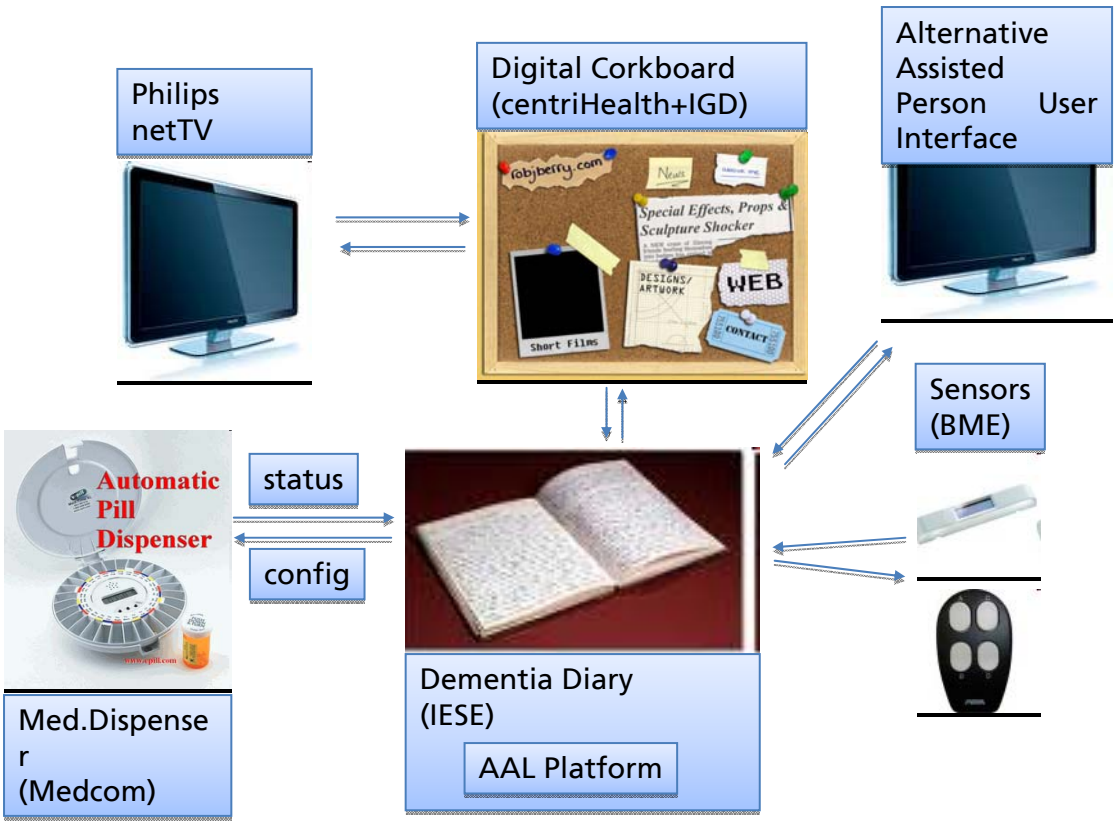

Figure1: CCE Solution Layout

2.2 CCE Stakeholder

Within the scope of the requirements engineering activities, a list of stakeholders (i.e., end users) has been identified and described (see [Del1.3] for more details) that are to be supported by the CCE solutions. These stakeholders comprise patients ("Assisted Persons") and caring persons (such as "Relatives" and "Caregivers"). As we will refer to these stakeholders in the use cases in the next chapters, these stakeholders are shortly introduced by means of a stereotypical persona description.

2.2.1 Patient ("Assisted Person")

Role	senior patient with Dementia
Name	Peter Miller
Age	82
Marital Status	Married
Living Situation	Lives together with his wife Martha
Medical Case	Medium progressed form of dementia combined with visual impairment
Additional context information	Lives in his own apartment but felt strange in his own rooms in the last months, his persecution mania has increased a lot in the last weeks

Table 1: Persona Description "Assisted Person" ([Del1.3])

2.2.2 Caring Person (Relative)

Role	Relative of patient with Dementia / Guardian, caring person
Name	Paula Matthews
Age	33
Marital Status	Married, 2 children
Living Situation	Lives with her family and her mother
Medical Case	-
Additional context information	Paula lives together with her husband, her two children and her mother. Three days per week she works in a clothes shop for about 5 hours. During that time she organized a caregiver from the care centre for Dementia to stay with her mother. In other cases when she has to leave the house, she calls Emma's brother Harry to take care of Emma.

Table 2: Persona Description "Caring Person" (Relative) ([Del1.3])

2.2.3 Caring Person (Caretaker)

	**Name**	Ben Meyers
	Gender	Male
	Birth date	June 26th , 1970
	Marital Status	Married
	Job Title / Role	Out-patient caregiver, Group Leader
	Job Experience	12 years
	Additional Context Information	Ben works at a local care center organization "Senior MobiCare". Every day he visits his patients at their apartments and provides them with a pre-defined set of individual care tasks such as preparing meals, washing activities, provide medication, etc.

Table 3: Persona Description "Caring Person" (Caregiver) (derived from [EMERGE])

2.3 CCE Components

In the following, the CCE high level components and related use cases are described in more detail comprising Dementia Diary (see section 2.3.1), Digital Cork Board (see section 2.3.2), and Medication Dispenser (see section 2.3.3).

2.3.1 Dementia Diary

Target Group: early and advanced stages depending on the function

- The device can register specific daily activities through sensors (e.g. getting up, personal hygiene, preparation of food, leaving the house, etc. This happens via sensors, which are installed in certain areas of the apartment
- On an internet-enabled television set, the caretaker/family members can then see what the patient has done throughout the day.
- The system can also ask the patient about specific activities, e.g. when leaving the house. When the patient returns the system can ask: Where were you? What did you do? The individual can leave a short comment if he/she wishes.
- Caretaker/ family members can remind the patient about different things that should be done via the system, e.g. set up a daily schedule.
- Registered activities and reminders are accessible via an internet connection, e.g. working family members can check from their office what the patient does and send him/her a message.

2.3.2 Digital Corkboard

Target Group: early dementia stage

- The devices functions like a traditional pinboard, only that this one is electronic and the notices are displayed on an internet-enabled television.
- On the electronic pinboard can be shown the following:
  - Time (digital or analog)
  - when, which medications should be taken
  - Reminders of appointments, visitors (calendar function)

- Reminders of where different things can be found (e.g. keys)
- simple and clear demonstrations of how to use simple devices at home (e.g. remote control)

- Pictures, illustrations and videos will be used as often as possible when displaying notices on the pinboard in order to simplify the information as much as possible for the patients

2.3.3 Medication Dispenser

Target Group: Early dementia stage

- The intelligent medication dispenser
  - will be programmed by the doctor depending on the given prescription
  - makes the right dose available to the dementia sufferer depending on the time of day
  - registers the time the medication was taken by the patient
  - prevents the patient from taking more medication than was prescribed for a given time
  - gives an alarm signal when the patient has forgotten to take his/her medication too often
  - Data records of the medication intake are automatically sent to the doctor/clinic; caretaker and family members can access this data with a password.

2.4 Use Cases

As agreed in the technical meeting, we will focus our further activities on the reminder. Reminders can be attached to an activity (e.g. check-mark "send me a reminder"). Reminders can be attached to to-do /task (connected to an activity) and to to-happen (just a reminder of something which will happen). An Activity or To-happen has a reminder.
Caretaker, Family Members, Doctor, and Dementia Patient can enter an activity or a reminder. The digital corkboard should act as an interface for the caregivers and realize the reminder use cases.
Apart from the reminders, we also envision an alarm functionality. How the alarm is treated can be found in the second part of this deliverable that covers the CCE Task 2.2.

For specifying the use case, we used the following use case template (see also [Del1.3]):

Table 4: Use Case Template

Use Case Number	< Use case 1 etc.>
Use Case Name	<Verb+ noun (e.g. order product)>
Actors	<stakeholder name, role name; if applicable: distinguished as primary actor, secondary actor…>
Version Number + Author	<Version Number and name of author of use case description>
Iteration	<Level of description: draft; reviewed; final>
Summary	<brief description>
Trigger/ intent	<system or user action that starts use case/ user's intent

	for use case>
Supported goal(s) from User Needs (if applicable)	<described from user perspective, including use context>
Preconditions	<state of the system that is required for the use case to be applicable, e. g. user has to be logged on to the system>
Flow of events: (Main Flow)	<all steps of the interaction between the actor and the system that are necessary for achieving a goal, incl. exchange of information> 1. 2. 3. …
Alternate flows	<e.g. At step 3. ….OR 3a…>
Exceptional flows	<e.g. if user has forgotten password to log on to system>
Assumptions/ rules	<conditions that are necessary for completing the use case successfully>
Displayed information	<information displayed in the GUI and that is necessary for the actor to execute this use case>
Postconditions	<state of the system after use case is finished successfully; corresponding to user's intent>
NFRs	<NFRs specified for this particular use case>
Services Required	<extracted from flow of events; number corresponding to number in system function list>
Relation to other use cases	<e.g. sub-use case>
Open issues	<e. g. What if patient is unconscious?; N/A for iteration "finished">

In the following, we will list the following use cases:

- Enter planned activity
- enter to happen
- check activity (check if activity has been done)
- reminder goes off
- snooze reminder (remind me later
- mark off reminder (hide or delete) BRE
- revise reminder

The use cases are similar to the use cases in Deliverable D1.3 but they reflect our further system understanding by being more concrete and more aligned to the architecture.

2.4.1 Enter Activity

Table 5: Use Case Description "Enter Activity"

Use Case Number	UC_2.1.1
Use Case Name	**Enter Activity**
Actors	Caregiver, relative, assisted person

Version Number + Author	V1 Özgür Ünalan; V2 Isabel John
Iteration	Pre-final
Summary	This Use-Case describes the creation of a planned activity (an activity that should be done, e.g. cooking in a given time period). For example, the assisted person should brush her teeth regularly but often forgets this.
Trigger/ intent	The caregiver wants to specify activities that should be done by the assisted person
Supported goal(s) from User Needs (if applicable)	• Safety of Peter, • Not feeling guilty, • Reliable Structures, • Time for himself in the future (as he can e.g. leave the house, as Peter will be able to cope on his own with his scheduled activities), • Information (which other activities are scheduled for the day?)´
Preconditions	Caregiver should be authenticated
Flow of events: (Main Flow)	1. The caregiver chooses to create a planned activity in the reminder 2. The system provides a UI for creating a to-be activity 3. The caregiver fills out the UI and adds detailed information about the planned activity (time, activity type, recurrence, alarm trigger, reminder type, e.g. visual, sound) 4. The system saves the planned activity inside the diary and shows a confirmation that the activity has been saved
Alternate flows	4.a If an activity with the same name has already been saved, the system asks for overwrite permission
Exceptional flows	<e.g. if user has forgotten password to log on to system>
Assumptions/ rules	N/A
Displayed information	• Name and description of planned activity • Timeframe (morning, midday, afternoon, etc.) • Frequency of activity (daily, weekly, other)
Postconditions	The planned activity is saved
NFRs	The creation of a to-be activity should take no longer than 1 minute in average.
Services Required	*This will be added later*
Relation to other use cases	N/A
Open issues	Who will use which device? This use case will probably not work on the TV (too much typing), so it should work on a PC/Handheld/Touch device

2.4.2 Enter to happen

Table 6: Use Case Description "Enter to happen"

Use Case Number	UC_2.1.2
Use Case Name	**Enter to happen**
Actors	Caring Person (probably Family Member, maybe Professional Caregiver), Patient (Assisted Person)
Version Number + Author	V1 Lenja Sorokin, UID
Iteration	Pre-final
Summary	This Use-Case describes the creation of an event, which is going to take place (e.g. birthday of a family member, purely informative character, no action of the dementia patient required).
Trigger/ intent	User interaction: The caregiver wants to specify an information that the assisted person should be aware of at a certain time (e.g. an event)
Supported goal(s) from User Needs (if applicable)	• Safety of Peter, • Not feeling guilty, • Reliable Structures, • Time for himself in the future (as he can e.g. leave the house, as Peter will be able to cope on his own because he has all information he needs in a given moment), • Information (which other activities are scheduled for the day?)´
Preconditions	• Caring Person (Family Member/Professional Caregiver) has access to the system (physical access and right to access) • Caring Person should be authenticated
Flow of events: (Main Flow)	1. The actor chooses to create an event (to happen) 2. The system provides a UI to enter the details 3. The actor adds the details, like name, description, priority, reminder with attributes (time, reminder style, location …) etc. 4. The system saves the event (to happen) inside the diary and shows a confirmation that it has been saved
Alternate flows	4.a If an event with the same name has already been saved, the system asks for overwrite permission
Exceptional flows	<e.g. if user has forgotten password to log on to system> Activity already entered (Step 1 changes to: Family Member Open Activity)
Assumptions/ rules	User does not have to confirm reminder
Displayed information	Event (= to happen)
Postconditions	Planned activity is saved, “reminder set” status
NFRs	*This will be added later*
Services Required	*This will be added later*

Relation to other use cases	N/A
Open issues	N/A

2.4.3 Check Activity

Table 7: Use Case Description "Check Activity"

Use Case Number	UC_2.1.3
Use Case Name	**Check Activity**
Actors	Caregiver, Relative, (Assisted Person)
Version Number + Author	V1, Özgür Ünalan
Iteration	Pre-final
Summary	This use case describes how the monitored and confirmed activities can be checked by caregivers, relatives or the assisted person themself
Trigger/ intent	The actor wants to check the status of the monitored and confirmed activities
Supported goal(s) from User Needs (if applicable)	Get an overview about the activities of the assisted person
Preconditions	The user is authenticated to check the activities of the assisted person
Flow of events: (Main Flow)	1. The actor chooses to check the activities of the assisted person 2. The system asks the actor to provide authentication information (e.g. user name and password) 3. The actor provides their security information 4. The system authenticates the actor and offers different views for the monitored and confirmed activities of the assisted person. 5. The actor views the activities by looking at the different views (monitored and confirmed activities, chronological view, as-is and to-be activities comparison, etc.) offered by the system. After a while, the actor chooses to log out. 6. The system logs the actor out.
Alternate flows	N/A
Exceptional flows	2a. The system recognizes an unsuccessful login more than 3 times. 3a. The system sends a message to the person responsible for the administration of the system, that there might be an attempt of breaking the security of the system 4a. The administrator takes countermeasures 5b. If the actor does not log out and there is a time-period of 5 minutes of inactivity, the system logs out the user automatically.
Assumptions/ rules	The assisted person has given permission that the caregiver or relatives can check his/her activities.
Displayed information	• List of activities • Confirmed activities

	• Monitored activities • Chronological view of activities • As-is and to-be activity comparison
Postconditions	The actor has gained an overview about the activities of the assisted person
NFRs	This use case should have a high security level. The actor needs to be authenticated.
Services Required	*This will be added later*
Relation to other use cases	N/A
Open issues	N/A

2.4.4 Reminder goes off

Table 8: Use Case Description "Reminder goes off"

Use Case Number	UC_2.1.4
Use Case Name	**Reminder goes off**
Actors	Assisted Person
Version Number + Author	V1 Patrick Röder, Fh-IGD
Iteration	Pre-final
Summary	The assisted person (dementia patient) is reminded of a planned activity/to-happen, e.g. cooking. The reminder task selects an appropriate means of reminding (message on TV set, audio notification…) and an appropriate "intensity", i.e. at first, the person is "nudged", then, if no reaction is monitored, it get more explicit.
Trigger/ intent	Timeout of the specified reminder due date/time interval
Supported goal(s) from User Needs (if applicable)	• Coping with daily life • Short-term memory • Organization (dated, todo's)
Preconditions	Activity has not yet been recognized or confirmed (still in "planned" stage), i.e. still not "done"
Flow of events: (Main Flow)	1. The system detects timeout of reminder due date 2. The system notifies the actor via appropriate means 3. The system keeps on reminding in given interval until confirmed or maximum intensity reached
Alternate flows	3a. If the activity is monitored, the system detects user reaction within certain time frame as confirmation
Exceptional flows	N/A
Assumptions/ rules	User does not have to confirm reminder
Displayed information	Associated activity
Postconditions	"reminder went off" status
NFRs	N/A
Services Required	*This will be added later*
Relation to other use cases	N/A
Open issues	It needs to be discussed if confirmation is necessary. Also, reminder should work in conjunction with "guiding" the patient to accomplish the task.

2.4.5 Snooze Reminder

Table 9: Use Case Description "Snooze Reminder"

Use Case Number	UC_2.1.5
Use Case Name	**Snooze Reminder**
Actors	Assisted Person
Version Number + Author	V1 Isabel John, IESE
Iteration	Pre-final
Summary	The assisted person (dementia patient) was reminded of a planned activity/to-happen, e.g. cooking. The message has recently been displayed (message on TV set, audio notification…) and the user wants to snooze the reminder for a certain time.
Trigger/ intent	Get reminded later.
Supported goal(s) from User Needs (if applicable)	Coping with daily life, short-term memory, organization (dated, todos)
Preconditions	A reminder is currently open/has been activated
Flow of events: (Main Flow)	1. The actor interacts with the system to put the reminder to sleep for a certain amount of time (the time duration is set in the system setting) 2. The system protocols the snooze activity 3. The system keeps on reminding after the specified interval until confirmed or maximum intensity reached
Alternate flows	N/A
Exceptional flows	N/A
Assumptions/ rules	User is able to appropriately interact with the system
Displayed information	Associated activity + snooze time
Postconditions	"reminder snoozed" status
NFRs	Usability-related requirements should be addressed.
Services Required	*This will be added later*
Relation to other use cases	N/A
Open issues	It needs to be discussed if confirmation is necessary. Also, reminder should work in conjunction with "guiding" the patient to accomplish the task.

2.4.6 Mark off Reminder

Table 10: Use Case Description "Mark off Reminder"

Use Case Number	UC_2.1.6
Use Case Name	**Mark off reminder**
Actors	Assisted Person
Version Number + Author	V1 Ranjit Bassi, BRE
Iteration	Pre-final
Summary	The assisted person (dementia patient) is reminded of a planned activity, e.g. to take their medications. The reminder task selects an appropriate means of reminding (message on TV set, audio notification…) and once the task has been completed the user has to verify that they have completed the task e.g., press button on medication

	dispenser to take medication, which sends a sms text to verify that user has taken medication
Trigger/ intent	Completion of a task at specified time or duration
Supported goal(s) from User Needs (if applicable)	Coping with daily life and undertaking key tasks, such as medications
Preconditions	Define critical tasks that user has to undertake
Flow of events: (Main Flow)	1. The system detects timeout of reminder due date 2. The system notifies the actor via appropriate means 3. The actor verifies that critical tasks have been completed
Alternate flows	If the activity is monitored, system detects user reaction within certain time frame as confirmation
Exceptional flows	N/A
Assumptions/ rules	User has to confirm critical tasks have been undertaken
Displayed information	Associated activity
Postconditions	"reminder went off" status
NFRs	N/A
Services Required	This will be added later
Relation to other use cases	N/A
Open issues	It needs to be discussed if confirmation is necessary. Also, reminder should work in conjunction with "guiding" the patient to accomplish the task.

2.4.7 Revise Reminder

Table 11: Use Case Description "Revise Reminder"

Use Case Number	UC_2.1.7
Use Case Name	**Revise reminder**
Actors	Caregiver, Relative, Assisted Person
Version Number + Author	Isabel John
Iteration	Pre-final
Summary	This Use-Case describes the modification of a reminder, e.g. changing the reminder interval
Trigger/ intent	The actor wants to modify the reminder
Supported goal(s) from User Needs (if applicable)	Coping with daily life and undertaking key tasks, such as medications
Preconditions	Caregiver should be authenticated
Flow of events: (Main Flow)	1. The actor chooses to modify a reminder 2. The system provides a UI displaying the reminder data 3. The actor modifies the UI (e.g. time, activity/reminder type, recurrence, alarm trigger, reminder type, e.g. visual, sound) 4. The system saves the reminder inside the diary and shows a confirmation that the reminder has been saved
Alternate flows	N/A

Exceptional flows	N/A
Assumptions/ rules	N/A
Displayed information	• Name and description of reminder • Timeframe (morning, midday, afternoon, etc.) • Frequency of activity (daily, weekly, other)…
Postconditions	The reminder is saved.
NFRs	N/A
Services Required	*This will be added later*
Relation to other use cases	N/A
Open issues	Who will use which device? This use case will probably not work on the TV (too much typing), so it should work on a PC/Handheld/Touch device

3. Quality Attributes and Nonfunctional Requirements

First, this chapter motivates the importance of nonfunctional requirements within the context of software engineering in general. This motivation is supplemented by a description of a method for eliciting nonfunctional requirements where the prioritization process followed within the CCE project has been derived from. Afterwards, relevant quality attributes are introduced that are considered to be of relevance within the context of CCE, comprising efficiency, reliability maintainability, portability, security, and usability.

3.1 Elicitation of Nonfunctional Requirements

Requirements engineering is the first activity in engineering a software-based product. Making mistakes in such an early phase has a strong impact on all subsequent software development phases. Especially nonfunctional requirements (NFRs) play an important role for the success of a project or product. In today's practice, essential information on a system's NFRs has often not been elicited properly and is thus incomplete. As a result, architectures have to be changed in subsequent development phases leading to increased project or platform development costs and increased time to market. Alternatively, missing NFRs are not incorporated into the product in later phases, leading to low product quality.

For eliciting NFRs we used the elicitation method described in [6]. The basic procedure for the algorithm used in this method is as follows: We have the information about relevant qualities on the one side (QA_InScope) and functional conceptual elements (FR_E) and information on subsystems (SYS) that are characterized by the qualities on the other side. The basic idea is to have a complete pair-wise comparison between the elements in the quality dimension with the elements in the functional and subsystem dimension, asking whether NFRs exist for the quality - function/subsystem pair. The algorithm ends when the pair-wise comparison is complete. One quality attribute does not characterize all types of functional conceptual elements or subsystems; rather, each quality attribute has one specific object (user task, system task, data or system) it characterizes.
Therefore, it is unnecessary to compare quality attributes of a certain type with elements of another type (like asking for Usage Time NFRs for a component). Figure 2 shows the comparison matrix that illustrates this situation. The gray areas are the ones that do not need to be covered by the algorithm as the quality attributes do not relate to the type of functional conceptual element.
The simplified algorithm compares one by one the quality elements of a specific type with the related functional or subsystem elements. The algorithm has to be designed in a way that takes into account the specifics of the refine-relationship between user tasks and system tasks. As user tasks are often refined into system tasks and quality attributes are refined in parallel, the elicitation algorithm needs to address this fact. The assumption is that the probability for a customer to identify and state a corresponding NFR is higher if he or she thinks in refinements rather than being asked for a new NFR.

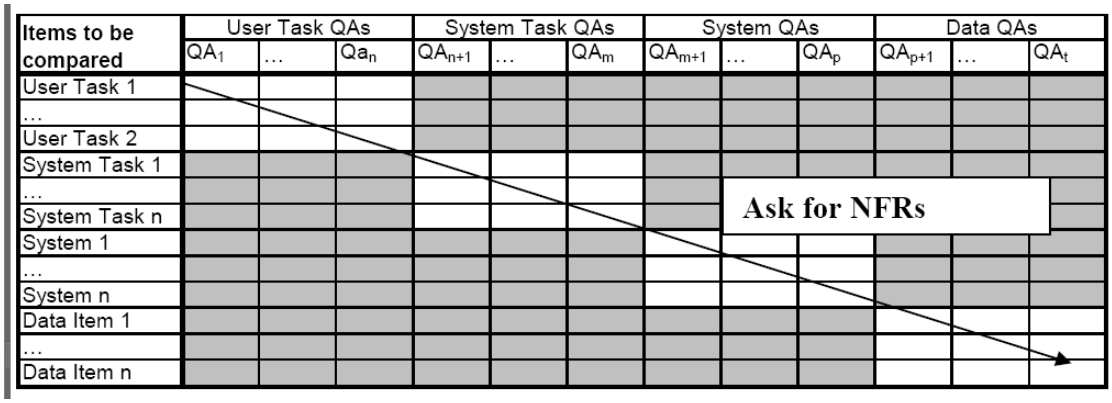

Figure 2: Elicitation Algorithm Comparison Matrix

The three tasks in the algorithm are:

- asking for NFRs: The stakeholders decide and specify whether an NFR exists or not.
- refining an NFR: If a higher level NFR exists, the stakeholders decide and specify whether lower level NFRs exist or not.
- marking pairs as done: Pairs of functional conceptual elements/ subsystems and elementary quality attributes get marked as done.

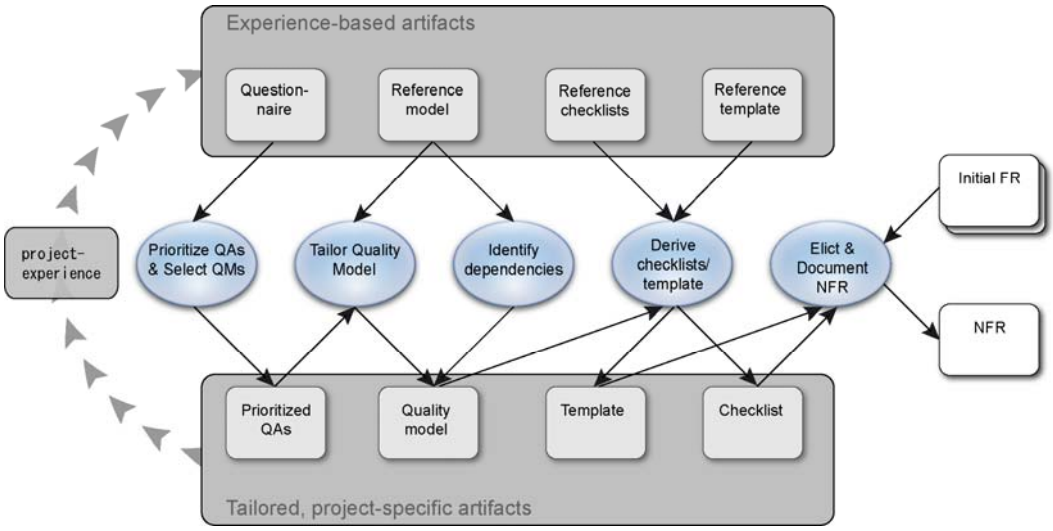

Figure 3: Overview of the NFR methodology

This is done in order to allow an incremental approach: We know exactly which pairs have already been compared, so if the functional range is extended (e.g., due to changing requirements) or new qualities should be checked, the algorithm does not require the requirements analyst to repeat already performed comparisons.

Figure 3 shows an overview of the described methodology. This methodology has been tailored to CCE needs for the purpose of prioritizing relevant quality attributes (the tailored processes both for the prioritization within the consortium as well as with the DRK are described in more detail in section 4.1 and section 4.2, respectively).

3.2 Quality Attributes in CCE

As a pre-analysis step, we identified relevant quality attributes for CCE (this step can basically be referred to “Prioritize QAs and Select QMs” as illustrated in Figure 3). For the analysis, we used material from earlier workshops with different stakeholders in the CCE context (dementia patients, caretakers etc.) and identified relevant quality attributes from the user/stakeholder needs. This analysis resulted in a list of relevant quality attributes comprising efficiency, reliability, maintainability, portability, security, and usability. We will further describe and refine these 6 quality attributes in the remainder of this section. The refinement follows the ISO9126 Norm [7] on qualities.

Efficiency

Efficiency is used to describe properties of a program relating to how much of various types of resources it consumes. Efficiency can be thought of as analogous to engineering productivity for a repeating or continuous process, where the goal is to reduce resource consumption, including time to completion, to some acceptable, optimal level. We mainly address timing issues here, comprising the following relevant sub-attributes:

- Response Time
- Usage Time
- Workload
- Transmission Time

Reliability

Reliability can be defined as the ability of a system or component to perform its required functions under stated conditions for a specified period of time. Reliability can further be refined into:

- Integrity
- Ease of Recovery
- Updates
- Last saved status

Maintainability

Maintainability is the ease with which a product can be maintained in order to: correct defects, meet new requirements, make future maintenance easier, or cope with a changed environment. Maintainability can further be refined into:

- Analyzability
- Changeability
- Stability
- Testability

Portability

Portability can be defined as the ability of a system or component to be transferred from one environment to another. Portability can further be refined into:

- Adaptability
- Installability
- Co-Existence
- Replaceability

Security

Security is the degree of protection against danger, damage, loss, and criminal activity. Security as a form of protection comprises structures and processes that provide or improve security as a condition. In our context, Data Security is the main issue that can further be refined into the following sub-attributes:

- Vulnerability
- Privacy
- Access / Authorization
- Awareness

Usability

Usability is the capability of the software product to be understood, learned, used and attractive to the user, when used under specific conditions. Usability can further be refined into:

- Understandability
  - Simplicity
  - Lucidity
  - Self-descriptiveness
- Productivity
- Learnability
  - Conformity with user expectation
  - Synthesizability
  - Predictability
- Attractiveness
- Error Tolerance
- Operability
  - Accessibility
  - Controllability
- Satisfaction
- Effectiveness
- Usability Compliance

4. CCE Prioritization Processes

This section deals with descriptions of the tailored processes (derived from the methodology introduced in section 3.1) for prioritizing relevant quality attributes (QAs) within the context of the CCE project.

4.1 Prioritization in CCE Consortium

As illustrated in Figure 4, the process underlying the prioritization basically comprised two phases: a *preparation phase* where a prioritization guideline has been developed and a *prioritization phase*, where members of the CCE Consortium prioritized a given set of QAs related to their particular CCE component (see section 2.3 for an overview on the components). Both phases are described in more detail in the following sections.

Figure 4: Prioritization Process

4.1.1 Preparation Phase

Goal of this phase was to prepare a prioritization guideline (in form of an MS Excel Sheet) that could then been used by project members to prioritize quality attributes during the prioritization phase. The preparation phase has been conducted by Fraunhofer IESE and comprised the three activities (1) Identify Relevant QAs, (2) Transfer QAs into Statement, and (3) Provide instructions (see also Figure 6) that are described in more detail in the following.

Activity 1: Identify Relevant QAs

One of the first steps towards the development of a guideline was to decide upon a list of quality attributes that are relevant within the context of the CCE project and that should be prioritized within the CCE consortium. This activity resulted in a list of 6 quality attributes comprising efficiency, reliability, maintainability, portability, security, and usability that have further be refined by following ISO9126 Norm [7] on qualities. The selected quality attributes and refinements have already been introduced and described in section 3.2. Figure 5 illustrates an extract of the guideline where the quality attributes are listed in column A and the respective refinements are listed in column B.

Please note: The refined attributes related to “usability” as introduced in section 3.2 have slightly been modified, further refined and supplemented with additional attributes that we considered to be of relevance but that may not be associated to usability according to [7]. This adaptation based on an elicitation guideline that has been developed to elicit nonfunctional requirements within EMERGE[1] - an ambient

[1] http://www.emerge-project.eu

assisted living project funded by the EU [8]. That is, the final list of prioritized attributes related to usability comprised the following refined attributes:

- Mobility(Operability)
- Avoid information overload (Simplicity)
- Avoid lack of information (Productivity)
- Effectiveness (Effectiveness)
- Localization (Conformity with user expectations)
- Understandability (Understandability)
- Appropriate feedback of system activities (Productivity/Understandability)
- Comprehensible icon language (Self-descriptiveness)
- Traceability (Operability)
- Navigation support (Operability)
- Consistency (Satisfaction/Usability Compliance)
- Appropriate wording for each target group (Conformity with user expectation)
- Undo / abort feasibility (Controllability)
- Accessibility (Accessibility)
- Adaptability (Productivity)
- Experience / Familiarity (Conformity with user expectations)
- Availability of a help system (Understandability)
- Relevant functionalities (Productivity/Effectiveness)
- Costs (Satisfaction)
- Perceived usefulness (Satisfaction)
- Unobtrusiveness of system components (Satisfaction/Usability Compliance)
- Satisfaction (Satisfaction)

All attributes are mapped to the usability quality attributes shown in chapter 3.2 (see attribute in bracket). Because there is not always a one to one mapping possible sometimes two attributes appear in the bracket. We think that the attributes we chose here are more suitable in the CCE-context to perform a prioritization according to usability.

Activity 2: Transfer QAs into Statements

In order to facilitate the prioritization process, each refined QA (i.e., all attributes listed in column B of the guideline partly illustrated in Figure 5) has been translated into a statement that reflects the definition of the respective attribute according to [7] in an understandable manner. These statements have then been prioritized by the members of the CCE consortium that participated in the prioritization phase which is described in more detail in section 4.1.2.

A	B	C
		Statement
Efficiency	Response Time	The system reacts to a given input within a short amount of time.
	Usage Time	For the success of the application, time is critical
	Workload	(Multiuser): The system supports a number of users submitting work through individual terminals.
	Transmission Time	Data is transmitted within a short amount of time.
Reliability	Integrity	If the system crashes, critical information must not be lost.
	Ease of recovery	In case the system fails, it is easy to take the system back to normal operation
		The time for a system to be down must be exactly specified.
	Updates	The system guarantees a proper operation during and after update procedures.
	Last saved status	In case of system abort it is easy to re-establish the last saved and stable status of the system

Figure 5: Quality Attributes (column A), refined qualities (column B) and corresponding statements (column C)

Activity 3: Provide instructions

Finally, the guideline has been supplemented by columns to (1) indicate priorities for each QA statement (2) classify the QAs and (3) give some rationales supporting the prioritization. Corresponding instructions of how to perform the particular activities have been prepared and provided in form of comments in the particular cells of the heading row (see for example comment in Figure 8). The prioritization process itself as well as particular prioritization activities are described in more detail in the next section.

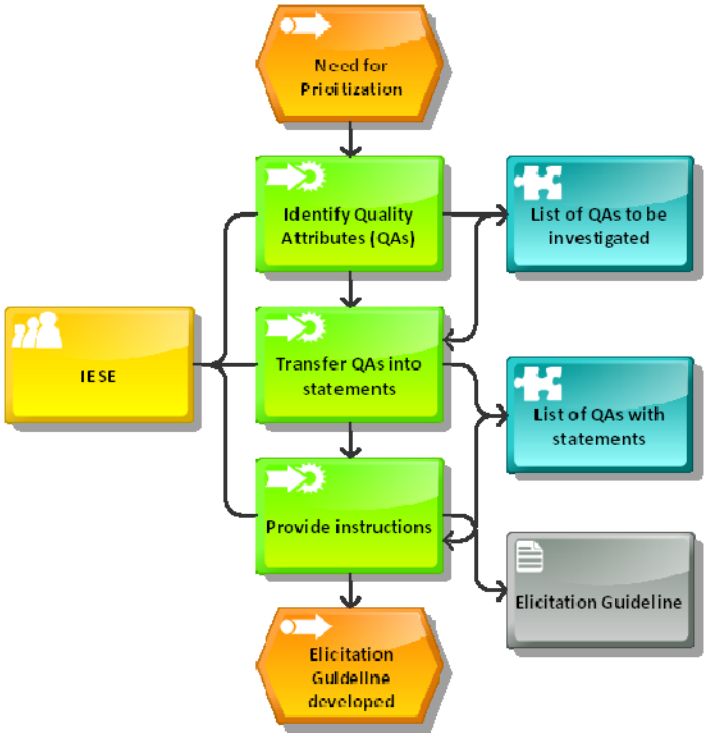

Figure 6: Activities, Outcomes and Responsibilities of the Preparation Phase

4.1.2 Prioritization Phase

Goal of this phase was to prioritize selected quality attributes by using the guideline developed during the preparation phase (see section 4.1.1). As illustrated in Figure 10, the prioritization process itself comprised the three activities (1) Prioritize QA Statements, (2) Classify Relevance of QAs, and (3) Indicate Rationale that are described in more detail in the following.

Activity 1: Prioritize QA Statements

Within the preparation phase, each relevant QA has been transferred into statements reflecting the definition of the respective QA. These statements were then prioritized on a scale from "Priority 1" to "Priority 3" with

- Priority 1 = very important
- Priority 2 = important
- Priority 3 = not of importance

		Statement	Priority		
			1	2	3
Efficiency	Response Time	The system reacts to a given input within a short amount of time.			
	Usage Time	For the success of the application, time is critical			
	Workload	(Multiuser): The system supports a number of users submitting work through individual terminals.			
	Transmission Time	Data is transmitted within a short amount of time.			
	Integrity	If the system crashes, critical information must			

Figure 7: Prioritize Statements (Activity 1)

Activity 2: Classify Relevance of QAs

According to the provided instruction (see Figure 8), the goal of this activity was to indicate whether those QAs that were rated to be "very important" (Priority 1) or "important" (Priority 2), are relevant for *the whole system component* that was subject to the prioritization (i.e., the CCE component where the respective member was responsible for) or *for one or more particular use case(s)* within the prioritized system component. In the latter case, the use cases should also be indicated.

C	D	E	F	G	H
Statement	Priority			Relevant for:	In case that you rated the statement with Priority 1 or 2, please indicate whether this quality attribute is relevant for the **whole system (globally)** or for **particular use case(s)**. In the latter case, please specify the particular use case name.
	1	2	3		
The system reacts to a given input within a short amount of time.					
For the success of the application, time is critical					
(Multiuser): The system supports a number of users submitting work through individual terminals.					
Data is transmitted within a short amount of time.					
If the system crashes, critical information must not be lost.					

Figure 8: Classify Relevance (Activity 2)

Activity 3: Indicate Rationale

In addition to the classification of those QAs with Priority 1 or Priority 2 (see previous step), a rationale had to be given by means of a short statement indicating why the particular QA has such a high importance (see also instruction in Figure 9).

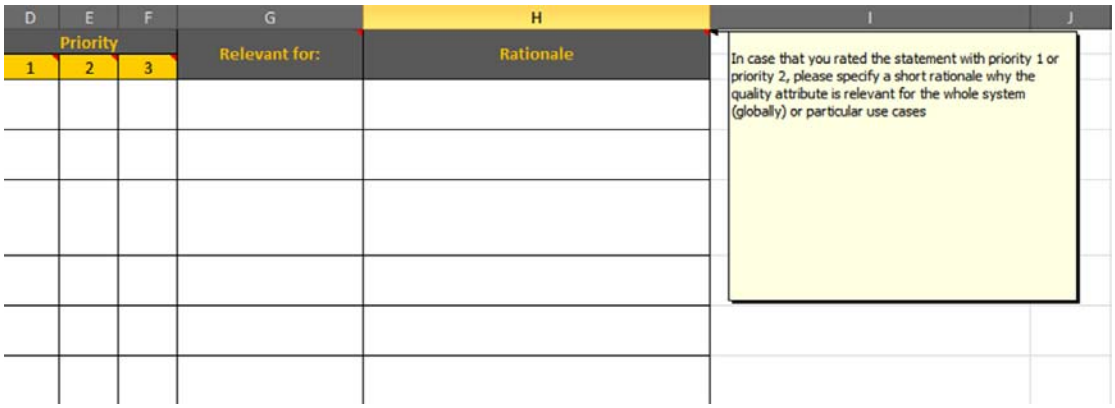

Figure 9: Indicate Rationale (Activity 3)

The following figures summarizes the activities, inputs and responsibilities of the prioritization

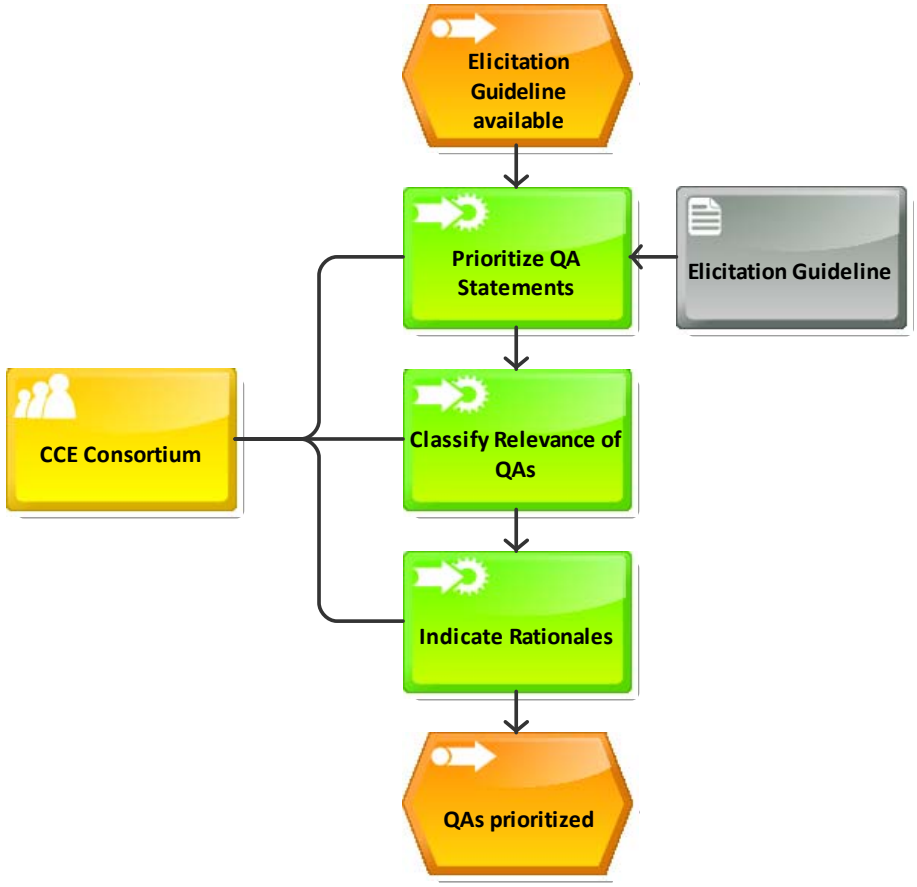

Figure 10: Activities, Outcomes and Responsibilities of the prioritization process within the CCE consortium

4.2 Prioritization with DRK

Within the scope of the prioritization activities in CCE, an interview has been performed at the DRK ("Deutsches Rotes Kreuz") that runs a dementia house located in Kaiserslautern. The interviewee and contact person at the DRK was the medical/psychological manager of the dementia house who has good insight into the goals and needs of caretakers and dementia patients. Goal of this interview was to identify relevant quality attributes related to use cases of the Dementia Diary (as introduced in section 2.3.1) from an end-users perspective, in particular assisted persons and relatives / caregiver (for a description of these stakeholders, please refer to section 2.2).

Similar to the prioritization in the CCE project consortium (as described in section 4.1), there was a *preparation phase* followed by a *prioritization phase*. However, the activities within the particular phases were different. In the following, we briefly describe the phases and related activities. The interview results can be found in section 5.3.

4.2.1 Preparation Phase

Goal of this phase was to derive a guideline that could be followed throughout the interview with the medical/psychological manager of the DRK dementia house. The guideline should enable to prioritize QAs related to the Dementia Diary both for each use case related to the Dementia Diary as well as for the Dementia Diary in general. Investigated use cases comprised:

- Enter Activity
- Confirm Activity
- Check Activity
- Leave Message
- Create Report

Within the preparation phase we basically conducted two activities: (1) Identify Relevant QAs and (2) Map QAs to Use Cases and Transfer QAs to statements.

Activity 1: Identify Relevant QAs
Out of the set of quality attributes (as introduced in section 3.2) we selected the quality attributes efficiency, usability as well as security as we considered these to be the most relevant quality attributes related to the Dementia Diary that can be prioritized and concretized from the end users' perspective by the interviewee.
These attributes have also been refined and supplemented by additional attributes similar to the preparation process for prioritizing quality attributes in the consortium (see section 4.1.1). This activity finally resulted in the following list of (refined) quality attributes that were subject to the prioritization interview with the DRK.

Efficiency
- Usage Time

Usability
- Avoid information overload
- Avoid lack of information

- Understandability
- Appropriate feedback of system activities
- Comprehensible icon language
- Consistency
- Appropriate wording for each target group
- Undo / abort feasibility
- Accessibility
- Adaptability
- Experience / Familiarity
- Availability of a help system
- Relevant functionalities
- Costs
- Perceived usefulness
- Unobtrusiveness of system components
- Satisfaction
- Traceability

Security

- Access / Authorization

Activity 2: Map QAs to Use Cases and Transfer QAs to Statements

In a second step we mapped the QAs identified in the previous activity to the investigated use cases, i.e., we decided which of the QAs might be of relevance for each of the use cases. Afterwards we transferred the mapped QAs into statements that reflect the definition of the respective attribute according to [ISO 01] in an understandable manner and, at the same time, have a direct reference to the investigated use cases. This enabled the interviewed person to directly prioritize the relevance of each QA from the viewpoint of the particular use cases and stakeholders. Figure 11 illustrates the statements related to UC 1 Enter Activity.

	Quality Attribute		Statement
UC 1: Enter Activity (Caregiver, Relative, Assisted Person)	Efficiency	Usage Time	Entering an activity doesn't take long.
	Usability	Understandability	Straightaway I know what kind of information I need to specify in order to enter an activity
		Appropriate Feedback	I receive appropriate system feedback as soon as I successfully entered an activity
		Comprehensible Icon Language	Icons used in the user interface are consistently used and easy to understand.
		Consistency	The specification of the various information required to enter an activity (i.e., what, when, description, reoccurrence etc.) happens similarly (e.g., only via pull down menü and/or text entering)
		Appropriate Wording for each target group	The wording used in the user interface is appropriately chosen and hence easy to understand
		Undo, abort feasability	It is easy to undo incorrect specifications of information related to the definition of an activity.
		Avoid lack of information / avoid information overload	Entering an activity should require specification of information that is really required (and not more and not less).
	Security	Access / Authorization	Only authorized persons are allowed to enter activities.

Figure 11: Transfer QAs to Statements related to Use Cases

Activity 3: Identify "System" QAs and Transfer QAs to Statements

Out of the list of quality attributes collected in activity 1, we identified quality attributes that might be of relevance for the whole Dementia Diary component (so-called "System QAs") rather than on particular use cases. This activity resulted in the following list of System QAs that have also been transferred into suitable statements as it is illustrated in Figure 12.

Usability

- Traceability
- Accessibility
- Adaptability
- Experience / Familiarity
- Availability of a help system
- Relevant functionalities
- Perceived usefulness
- Satisfaction

Tracability	When using the system it is easy to trace back the navigation path.
Accessibility	I can use all system functionalities even if I have disabilities (e.g., low vision, tremor).
Adaptability / Customizability	The user interface can be modified according to my preferences (e.g. colour schemes, font size).
Experience / Familiarity	Thes system is similar in use to systems / devices I am already familiar with.
Availability of a help system	The system offers a help system
	The usage of system is so easy that I do not need any help system.
Relevant functionalities	The system provides only those functionalities that I really need and not more
Perceived usefulness	The system and its functionalities are useful for fulfilling my tasks and goals.
Satisfaction	I feel satisfied when using the system

Figure 12: System QAs and Statements

Activity 4: Extend statements on further questions and/or metrics

Where applicable, all statements have been supplemented by further central questions that – depending on the given priority – have been asked during the interview to elicit further details on the quality attributes, such as concrete non-functional requirements and/or to enable discussions of whether QAs (especially the QAs related to usability) are currently fulfilled or should be improved in the mock-ups.

Quality Attribute	Statement	Priority			NFR / Metric
		1	2	3	
Usage Time	Entering an activity doesn't take long.				How long at most? Please specify the maximal duration in seconds
Understandability	Straightaway I know what kind of information I need to specify in order to enter an activity				Do you think that this requirement is currently (in the mock-ups) fulfilled? What problems do exist (if any)?
Appropriate Feedback	I receive appropriate system feedback as soon as I successfully entered an activity				n/a
Comprehensible Icon Language	Icons used in the user interface are consistently used and easy to understand.				Do you think that this requirement is currently (in the mock-ups) fulfilled? What problems do exist (if any)?
Consistency	The specification of the various information required to enter an activity (i.e., what, when, description, reoccurrence etc.) happens similarly (e.g., only via pull down menü and/or text entering)				Do you think that this requirement is currently (in the mock-ups) fulfilled? What problems do exist (if any)?
Appropriate Wording for each target group	The wording used in the user interface is appropriately chosen and hence easy to understand				Do you think that this requirement is currently (in the mock-ups) fulfilled? What problems do exist (if any)?
Undo, abort feasability	It is easy to undo incorrect specifications of information related to the definition of an activity.				n/a
Avoid lack of information / avoid information overload	Entering an activity should require specification of information that is really required (and not more and not less).				Do you think that this requirement is currently (in the mock-ups) fulfilled? Is there any missing information or information that is not necessary?
Access / Authorization	Only authorized persons are allowed to enter activities.				

Activity 5: Organize interview

Finally, within the preparation phase the interview itself has been organized which included activities like schedule date and time, print material, prepare mock-ups, etc.

4.2.2 Prioritization Phase

As already stated in the introduction of this chapter, the prioritization has been done by means of an interview with the medical/psychological manager of the DRK dementia house. In the interview, three persons were involved:

- 1 interviewed person (interviewee)
- 1 interviewing person that guided through the interview and asked questions by following the interview guideline
- 1 observer taking notes on interesting comments

The interview itself basically followed the following structure:

- Welcome and introduction to purpose of the interview by the interviewing person
- For each use case
    - Explain and demonstrate the functionality of the use case with the help of the clickable mock-ups on a laptop
    - Let the interviewee prioritize statements related to use case
    - Depending on given priority (i.e., in case of priority 1 or 2) ask further questions related to rated quality attributes to elicit concrete non-functional requirements and/or identify usability problems and improvement suggestions in the mock-ups.
- For System QAs
    - Let the interviewee prioritize statements related to System QAs
    - Depending on given priority (i.e., in case of priority 1 or 2) ask further questions related to rated quality attributes to elicit concrete non-functional requirements and/or identify usability problems and improvement suggestions in the mock-ups.

5. Results of the Prioritization

In this chapter we describe the results of the different prioritizations that have been done. The results we got from members of the consortium and the results we got from interviews with an external organization, the "Deutsche Rote Kreuz" in Kaiserslautern are shown. Although the results are not comparable because of the different prioritization approaches we show common aspects and basic differences.

5.1 Prioritization of BME

This section deals with the results of the prioritization by BME. We first provide an overview for the rating of the top level attributes. Then we show into detail, each attributes prioritization and deal with significant sub attributes to show the rational for the rating decision.

Table 12 shows the summary of the prioritization. It is significant, that security is rated far ahead of portability, usability, reliability and maintainability and shown as the most important attribute from BME's point of view. Efficiency is rated with least importance, way behind previous attributes.

Attribute:	**Priority:**
Security	Very important (1,25)
Portability	Important (1,53)
Usability	Important (1,75)
Reliability	Important (1,8)
Maintainability	Important (1,8)
Efficiency	Important (2,25)

Table 12: Prioritization of BME

5.1.1 Efficiency

Table 13 shows the prioritization for efficiency. The only sub attribute that is rated as very important is the response time although it is not exactly defined what short means in case of emergency situations. Workload and transmission time are less important sub attributes.

Table 13: Prioritization BME - Efficiency

		Statement	Priority			Rationale
			1	2	3	
Efficiency	**Response Time**	The system reacts to a given input within a short amount of time.	Y			Emergency situations. What does "short" mean?
	Usage Time	For the success of the application, time is critical		Y		In a general term, it is true.
	Workload	(Multiuser): The system supports a number of users submitting work through individual terminals.			Y	Not in this phase, functionality has not been decided; performance optimization should be done in a later phase.

	Transmission Time	Data is transmitted within a short amount of time.			Y	Not a typical problem in these type of systems

5.1.2 Reliability

Table 14 shows the prioritization for reliability. Last saved status and integrity which is especially of relevance in emergency situations are rated as very important.

Table 14: Prioritization BME - Reliability

		Statement	Priority			Rationale
			1	2	3	
Reliability	**Integrity**	If the system crashes, critical information must not be lost.	Y			Emergency situations.
	Ease of recovery	In case the system fails, it is easy to take the system back to normal operation			Y	Not in this phase, functionality has not been decided; this type of reliability problems may be addressed later.
		The time for a system to be down must be exactly specified.		Y		Must be addressed by proper extendible system architecture. Depends on the component, i.e., it effects one user or large number of users.
	Updates	The system guarantees a proper operation during and after update procedures.		Y		Updates must be done during the trials.
	Last saved status	In case of system abort it is easy to re-establish the last saved and stable status of the system	Y			

5.1.3 Security

Table 15 shows the prioritization for security, which was rated overall as the most important quality attribute. Privacy, access / authorization and awareness are implementation dependent and must be taken into account from the beginning. Therefore these attributes are rated as very important and serve as the basis for the high importance of security overall.

Table 15: Prioritization BME - Security

		Statement	Priority			Rationale
			1	2	3	

Security	Vulnerability	The system is able to prevent possible attacks on the system in such a way that abnormal, unwanted or even critical system behavior does not occur (e.g., the system is resistant to viruses and hacker attacks).		Y		Must be addressed by proper components (Linux distro with regular security updates), and proper implementation. Problematic on the sensor network level (no state of the art today).
	Privacy	The system logs and tracks all activities which are related to possible privacy issues so that each request can be tracked back to its origin.	Y			Implementation dependent, must be taken into account from the beginning.
	Access / Authorization	The system can only be used by authorized persons.	Y			Implementation dependent, must be taken into account from the beginning.
	Awareness	The system informs the user if specific - especially privacy-related - actions are carried out (e.g., notifies me if camera or audio transmission is activated.)	Y			Implementation dependent, must be taken into account from the beginning. Ethical issue.

5.1.4 Maintainability

Table 16 shows the prioritization for maintainability. Changeability is the only attribute that is rated very important because the system developed is a prototype and supposed to change a lot.

Table 16: Prioritization BME - Maintainability

		Statement	Priority			Rationale
			1	2	3	
Maintainability	Analyzability	When modifying the system, it is easy to determine the impact of the change		Y		Clear architecture, separation of components.
		In case of changes to the software I have to find the affected elements.		Y		Clear architecture, separation of components.
		It is easy to determine deficiencies or causes of failures in the product.		Y		Clear architecture, separation of components.
		It is easy to find bugs		Y		Clear architecture,

		in the system.				separation of components.
		It is easy to understand error messages.	Y			It is a must, misleading errors messages must be avoided (GPF in module XYZ).
	Changeability	Performing changes in the software is easy	Y			Early, pilot product. The system is going to change a lot.
		It is easy to add new functionalities to the software product.	Y			Early, pilot product. The system is going to change a lot.
	Stability	It is easy to mitigate failures caused by maintenance side effects.				?
		After changes to the software the system shows few unexpected behaviors.				?
	Testability	After the system has been started I can be sure that the system runs properly.			Y	Not in this phase of the development.
		After modifications the software must be tested	Y			Minimal requirement.
		It is easy to determine whether the software is ready for operation or not.			Y	Not in this phase of the development.

5.1.5 Portability

Table 17 shows the prioritization for portability. All sub attributes are rated on a high importance what makes portability the second most important quality attribute for the system.

Table 17: Prioritization BME - Portability

		Statement	Priority			Rationale
			1	2	3	
Portability	**Adaptability**	It is easy to adapt the software to data sets in a new working environment.	Y			Early, pilot product. The system is going to change a lot.
		Adapting the software to a specified hardware environment (hardware devices,	Y			The HW changes a lot; we do not know what will be available two years by now!

		network facilities) is easy.				
		Adapting the software to a specified organizational environment (infrastructure of organization) is easy .			Y	Functionality has not been decided, it is too early to take into account these type of aspects.
		Adapting the software to a specified system software environment (operating system, network software, cooperated application software) is easy.	Y			Low level software, such OS and middleware changes a lot, and lot of cases they are very HW dependent, and HW will change!
	Installability	It is easy to physically move the system to its intended environment.		Y		Early, pilot product.
		It is easy to configure the system to a new environment (e.g., hardware devices).	Y			The HW changes a lot; we do not know what will be available two years by now!
		It is easy to configure the system to individual preferences.	Y			There are no two identical house or patient.
		Removing a feature from the product is easy;		Y		It may be left unused.
		Uninstalling the system is easy.			Y	No SW should be installed on the devices of users, other devices can be just reinstalled.
	Co-Existence	The system is capable to co-exist with other independent software in a common environment sharing common resources..	Y			These are shared resources. But no OSGi type of sharing (same address space), but min. process level separation.
	Replaceability	After replacing the software with a previous version it is easy to continue to use the same data.	Y			At least data should be transferred to the new system.
		Adding new components does not lead to unexpected behaviors when using	Y			This is a common action in such systems, sensors come and go…

		the existing user interface.				
		The system is compatible to previous versions of the system.		Y		If data can be transferred, major changes are no problem.

5.1.6 Usability

Table 18 shows the prioritization for usability. On a first view usability seem not that important over all. But looking into the prioritization in detail, there are lots of attributes that are rated very important and have to be taken into account when developing the system.

Table 18: Prioritization BME - Usability

		Statement	Priority			Rationale
			1	2	3	
Usability	**Mobility**	It is possible to take the system to different locations			Y	Dementia sufferers do not move.
	Avoid information overload	According to the specific use context, the system only provides me with appropriate and accurate information and not more.		Y		Nice feature, but somewhat functional requirement.
	Avoid lack of information	Depending on the use context and location, the system displays all the necessary information (important information is not missing).	Y			It is required to let users to dig into the details if required. It is a must for system maintenance.
	Effectiveness	Using the system allows me to achieve my specific goals and fulfill my specific tasks in particular environments completely and correctly.			Y	Generally no, we have to limit functions. Is it a functional requirement?
	Localisation	The software is adapted for different cultural regions and languages.	Y			The software needs to work in 3 countries, and users do not tend to speak foreign languages.

	Understandability	It is easy to understand the concepts of the system.	Y			On various levels, strongly depend on the user. The patient should not be required to understand anything, family and caregivers may be able to understand more. Operational staff must be in the full knowledge of system concepts.
	Appropriate feedback of system activities	The system informs the user about the current states of activities that are executed.	Y			It is the task of the system. Again, it may be a functional requirement.
	Comprehensible Icon Language	Icons used in the user interface are consistently used and easy to understand.	Y			The users have limited capabilities, it is an absolute must.
	Traceability	When using the system it is easy to trace back the navigation path.		Y		There should be minimal or no navigation path for the patient, and limited for other users.
	Navigation support	The system provides functionalities (e.g., back keys, home button) that support me in navigation activities.			Y	Minimalistic user interface…
	Consistency	Navigation elements, screen dialogs and the screen layout stay consistent during the application flow.	Y			
	Appropriate wording for each target group	For parts of the system, that are only accessed by specific user groups (e.g., caregivers, assisted persons with dementia), the wording used in the system is tailored to this specific user group;	Y			
		For system areas that all user groups need to access (e.g., login) a	Y			

		common wording that all user groups understand is used.				
	Undo / abort feasibility	It is easy to handle errors or incorrect specifications, e.g., by an undo functionality.		Y		I would ask for confirmation, and may implement a minimalistic undo queue.
	Accessibility	The system is accessible by all people, also by those people who have disabilities (e.g., low vision, tremor).			Y	Early design phase, we should concentrate on the primary functionality.
	Adaptability	The user interface can be modified according to the user's preferences (e.g. color schemes, font size).		Y		It addresses some aspects of the disabilities related questions.
	Experience / Familiarity	This system is similar in use to systems the user is familiar with.			Y	What do they use? There are millions of user interfaces on the web…
	Availability of a help system	The system offers a global help system (e.g., an online help manual).			Y	Early design phase, we should concentrate on the primary functionality.
		The system offers a context-sensitive help (help only for current screens or tasks).			Y	Early design phase, we should concentrate on the primary functionality.
	Relevant functionalities	The system provides only those functionalities that I really need and not more	Y			Early design phase, we should concentrate on the primary functionality.
	Costs	The system does not exceed a specified amount of costs.	Y			We have a project, with limited budget.
	Perceived usefulness	The system and its functionalities are useful for fulfilling my tasks and goals.	Y			
	Unobtrusiveness of system components	System components (e.g., sensors) are installed and operated in a way that there is no visible evidence of the system.	Y			The patient in a special mental state, the acceptance of system depends on the ambient nature of the system.
	Satisfaction	I feel satisfied when using the system	Y			

5.2 Prioritization of IGD

This section deals with the results of the prioritization by Fraunhofer IGD. We first provide an overview for the rating of the top level attributes. Then we show into detail, each attributes prioritization and deal with significant sub attributes to show the rational for the rating decision.
Table 19 shows the summary of the prioritization. All attributes are rated important; reliability is shown as the most important one whereas portability is the least important. Nevertheless, the attributes are rated much more similar than in the BME prioritization.

Attribute:	Priority:
Reliability	Important (1,6)
Usability	Important (1,69)
Maintainability	Important (1,75)
Security	Important (1,8)
Efficiency	Important (2)
Portability	Important (2,15)

Table 19: Prioritization of IGD

5.2.1 Efficiency

Table 20 shows the prioritization for efficiency. The only attribute that is rated very important is the response time because it is essential for usability and technical feasibility.

Table 20: Prioritization IGD - Efficiency

		Statement	Priority			Rationale
			1	2	3	
Efficiency	**Response Time**	The system reacts to a given input within a short amount of time.	X			Essential for usability and technical feasibility.
	Usage Time	For the success of the application, time is critical			X	
	Workload	(Multiuser): The system supports a number of users submitting work through individual terminals.		X		We can assume that only one care person is assigned to a patient at a given time – if not, we have to provide for transaction safety
	Transmission Time	Data is transmitted within a short amount of time.		X		

5.2.2 Reliability

Table 21 shows the prioritization for reliability. Similar to the BME prioritization, last saved status and integrity are rated as very important.

Table 21: Prioritization IGD - Reliability

		Statement	Priority			Rationale
			1	2	3	
Reliability	Integrity	If the system crashes, critical information must not be lost.	X			
	Ease of recovery	In case the system fails, it is easy to take the system back to normal operation		X		In case of failure, there should be a return to a „safe state“ if possible in order not to confuse the patient
		The time for a system to be down must be exactly specified.		X		
	Updates	The system guarantees a proper operation during and after update procedures.		X		
	Last saved status	In case of system abort it is easy to re-establish the last saved and stable status of the system	X			

5.2.3 Security

Table 22 shows the prioritization for security. The system will store highly private data which is not allowed to be accessed by unauthorized people. Therefore access / authorization is rated as very important as well as awareness which is forced by the patients need to feel “in control” of the system.

Table 22: Prioritization IGD – Security

		Statement	Priority			Rationale
			1	2	3	
Security	Vulnerability	The system is able to prevent possible attacks on the system in such a way that abnormal, unwanted or even critical system behavior does not occur (e.g., the system is resistant to viruses and hacker attacks).		X		Interfaces to the internet should be made secure/reduced to minimum necessary communication
	Privacy	The system logs and tracks all activities which are related to possible privacy issues so that each request can be tracked back to its origin.		X		

	Access / Authorization	The system can only be used by authorized persons.	X			Highly private data must not fall into the wrong hands!
	Awareness	The system informs the user if specific - especially privacy-related - actions are carried out (e.g., notifies me if camera or audio transmission is activated.)	X			Patients should feel „in control“

5.2.4 Maintainability

Table 23 shows the prioritization of maintainability. Different statements of sub attributes are rated as very important but stability is the single attribute that is very important on all statements. Rational for this decision is that every change made on the system should be tested for side effects on other parts of the system.

Table 23: Prioritization IGD – Maintainability

		Statement	Priority			Rationale
			1	2	3	
Maintainability	**Analyzability**	When modifying the system, it is easy to determine the impact of the change		X		What is meant by „modifying“ in our case? Changes in a predetermined fashion (e.g. database operations) should be predictable.
		In case of changes to the software I have to find the affected elements.		X		
		It is easy to determine deficiencies or causes of failures in the product.	X			Patient should not be confronted with errors →“routing“ to caregivers
		It is easy to find bugs in the system.			X	Maintained by professionals
		It is easy to understand error messages.		X		
	Changeability	Performing changes in the software is easy		X		
		It is easy to add new functionalities to the software product.		X		Extendibility should be built-in
	Stability	It is easy to mitigate failures caused by maintenance side effects.	X			Very important → every change should be tested for side effects; stability is paramount

		After changes to the software the system shows few unexpected behaviors.	X			
	Testability	After the system has been started I can be sure that the system runs properly.		X		
		After modifications the software must be tested		X		
		It is easy to determine whether the software is ready for operation or not.	X			The system should explain itself sufficiently to caregivers and relatives so they can decide whether maintenance is required or not

5.2.5 Portability

Table 24 shows the prioritization of portability. The system developed will co-exist with home automation systems and other AAL solutions or services. This sub attribute of portability is mandatory as has to be considered when building the system.

Table 24: Prioritization IGD – Portability

		Statement	Priority			Rationale
			1	2	3	
Portability	Adaptability	It is easy to adapt the software to data sets in a new working environment.		X		We develop the system for the „home“ environment → no specific adaptation needs, but UI adaptation to user needed
		Adapting the software to a specified hardware environment (hardware devices, network facilities) is easy.		X		
		Adapting the software to a specified organizational environment (infrastructure of organization) is easy.			X	
		Adapting the software to a specified system software environment (operating system,			X	We give specific info about the hard- and software environment and provide at least

		network software, and cooperated application software) is easy.				parts of it → adaptation by professionals
	Installability	It is easy to physically move the system to its intended environment.		X		
		It is easy to configure the system to a new environment (e.g., hardware devices).	X			Usage of existing sensors
		It is easy to configure the system to individual preferences.	X			
		Removing a feature from the product is easy;		X		
		Uninstalling the system is easy.			X	Done by professionals
	Co-Existence	The system is capable to co-exist with other independent software in a common environment sharing common resources.	X			Co-existence with home automation systems, other AAL solutions/services etc.
	Replaceability	After replacing the software with a previous version it is easy to continue to use the same data.			X	Updates done by professionals, staying true to used standards and data formats
		Adding new components does not lead to unexpected behaviors when using the existing user interface.		X		
		The system is compatible to previous versions of the system.			X	

5.2.6 Usability

Table 25 shows the prioritization of usability with several high rated sub attributes. Overall, usability is the second most important quality attribute, which can be identified here as well - 9 sub attributes are rated with “very important”.

Table 25: Prioritization IGD - Usability

		Statement	Priority			Rationale
			1	2	3	
Usability	**Mobility**	It is possible to take the system to different locations			X	Installing it in different locations should be fast, the system itself is rather

						immobile
	Avoid information overload	According to the specific use context, the system only provides me with appropriate and accurate information and not more.	X			UI focus: Timely, accurate and complete information according to user's information needs
	Avoid lack of information	Depending on the use context and location, the system displays all the necessary information (important information is not missing).		X		Dependent upon context, UI can be more or less specific at times
	Effectiveness	Using the system allows me to achieve my specific goals and fulfill my specific tasks in particular environments completely and correctly.		X		
	Localisation	The software is adapted for different cultural regions and languages.			X	
	Understandability	It is easy to understand the concepts of the system.	X			Conceptual model should be crystal clear to everyone involved
	Appropriate feedback of system activities	The system informs the user about the current states of actitvities that are executed.	X			System feedback vitally important
	Comprehensible Icon Language	Icons used in the user interface are consistently used and easy to understand.				
	Traceability	When using the system it is easy to trace back the navigation path.		X		To be determined
	Navigation support	The system provides functionalities (e.g., back keys, home button) that support me in navigation activities.		X		To be determined
	Consistency	Navigation elements, screen dialogs and the screen layout stay consistent during the application flow.		X		To be determined

	Appropriate wording for each target group	For parts of the system, that are only accessed by specific user groups (e.g., caregivers, assisted persons with dementia), the wording used in the system is tailored to this specific user group;	X			Very important → vastly different cognitive abilites
		For system areas that all user groups need to access (e.g., login) a common wording that all user groups understand is used.		X		
	Undo / abort feasability	It is easy to handle errors or incorrect specifications, e.g., by an undo functionality.	X			E.g. input of activities should be easy to correct/undo
	Accessibility	The system is accessible by all people, also by those people who have disabilities (e.g., low vision, tremor).	X			Design-for-all approach to accommodate patients (and relatives) with disabilities
	Adaptability	The user interface can be modified according to the user's preferences (e.g. color schemes, font size).		X		Not the focus of the project, but certain facilities for adaptation should be provided
	Experience / Familiarity	The system is similar in use to systems the user is familiar with.		X		Needs to be determined (metaphor-based approach vs. Direct manipulation)
	Availability of a help system	The system offers a global help system (e.g., an online help manual).	X			Necessary for caregivers and relatives, probably not used by patients
		The system offers a context-sensitive help (help only for current screens or tasks).		X		
	Relevant functionalities	The system provides only those functionalities that I really need and not more	X			
	Costs	The system does not exceed a specified amount of costs.			X	See Business Model

	Perceived usefulness	The system and its functionalities are useful for fulfilling my tasks and goals.	X			User need (patients): Coping with daily life, short-term memory augmentation, organization (daily schedule)
	Unobtrusiveness of system components	System components (e.g., sensors) are installed and operated in a way that there is no visible evidence of the system.	X			Continuum: Explicit "push" vs. implicit support ("nudge")
	Satisfaction	I feel satisfied when using the system		X		

5.3 Prioritization of DRK

This section deals with the results of the conducted interview. For each use case we first provide a description of the particular investigated use cases, as well as the mock-up that has been demonstrated to the interviewee to explain the functionality of the use case. After that, the main findings are presented for each use case (for detailed results and individual ratings, please refer to the Appendix). The sections ends with a summary of the most important quality attributes related to the Dementia Diary (see section 2.3.1).

Note: As the interview has been conducted in September 2010, the use cases that have been investigated have been adapted and changed since then. However, in this chapter we present the use cases that have been presented to the interviewed person, as also the given priorities of quality attributes refer to the use case versions as they were in September (they correspond to the use cases documented in [Del1.3]. Please refer to section 2.4 for current versions of CCE use cases!

5.3.1 Enter Activity

Table 26 illustrates the use case that has been investigated in the interview related to the activity “Enter Activity”.

Table 26: Use Case Description "Enter Activity"

Use Case Number	UC_P1.1
Use Case Name	**Enter Activity**
Actors	Caregiver, Relative, Assisted Person
Version Number + Author	V1 Özgür Ünalan
Iteration	Pre-final
Summary	This Use-Case describes the creation of a to-be activity (an activity that should be done, e.g. cooking, in a given time period). For example, the assisted person should cook once every two days.
Trigger/ intent	The caregiver wants to specify activities that should be done by the assisted person
Supported goal(s) from User Needs (if applicable)	detailed information about the planned daily activities is given
Preconditions	Caregiver should be authenticated
Flow of events: (Main Flow)	1. The caregiver chooses to create a to-be activity in the diary 2. The system provides a form for creating a to-be activity 3. The caregiver fills out the form and adds detail information about the to-be activity (time, activity type, recurrence) 4. The system saves the to-be inside the diary and shows a confirmation that the to-be activity has been saved
Alternate flows	<e.g. At step 3. ….OR 3a…>
Exceptional flows	<e.g. if user has forgotten password to log on to system>
Assumptions/ rules	-

Displayed information	Name and description of to-be activity, time frame (morning, midday, afternoon, etc.), frequency of activity (daily, weekly, other)
Postconditions	The to-be activity is saved
NFRs	- The creation of a to-be activity should take no longer than 1 minute in average
Services Required	This will be added later
Relation to other use cases	-
Open issues	-

Figure 13 illustrates an example mock-up that has been used to illustrate the functionality of the investigated use case.

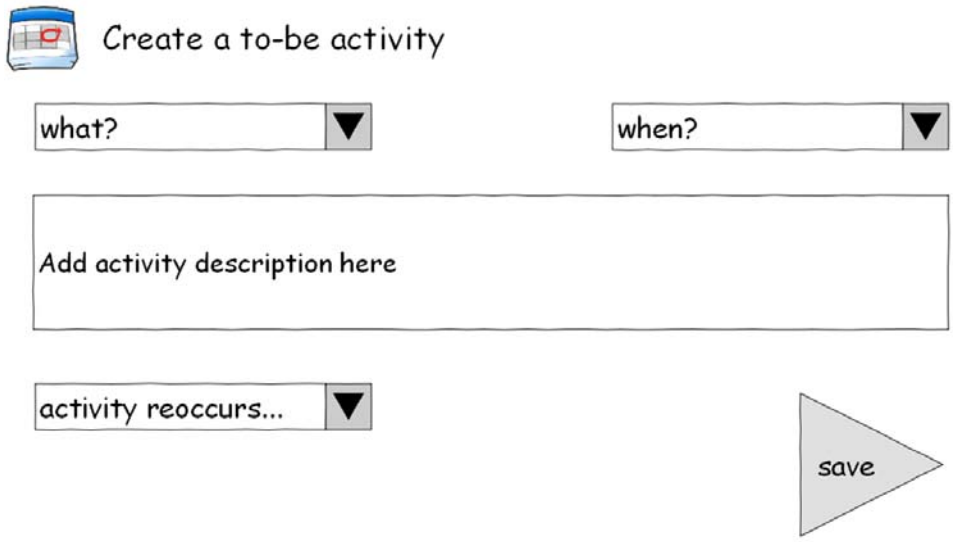

Figure 13: Mock-up "Enter Activity"

Results and Main Finding

Table 27 illustrates the results of the prioritization of the quality attributes both from the perspective of an assisted person (indicated with a "X") as well as a caregiver / relative (indicated with an "O"). To summarize the main findings based on the comment made by the interviewee we can state the following:

- Usage time should not exceed 10 minutes in case of patient / 5 minutes in case of caregivers
- understandability is currently fulfilled in mock-up for patients in an early dementia stage; for patients in later stage it might be too complex → better ask for required information stepwise
- Appropriate feedback is important to avoid memory-uncertainness
- Comprehensible Icon language is important to communicate with patient / in later stages symbols might not be useful anymore
- Consistency is very important
- Avoidance of information overload / reduction on essential information is very important (also in case of relatives as these persons are often also elderly persons)

Table 27: Prioritization of QAs related to UC "Enter Activity"
(X = Perspective of an Assisted Person, O = Perspective of Caregiver / Relative)

	Quality Attribute		Priority		
			1	2	3
UC 1: Enter Activity (Caregiver, Relative, Assisted Person)	Efficiency	Usage Time		X / O	
	Usability	Understandability	X / O		
		Appropriate Feedback	X	O	
		Comprehensible Icon Language	X	O	
		Consistency	X / O		
		Appropriate Wording for each target group	X	O	
		Undo, abort feasability	X / O		
		Avoid lack of information / avoid information overload	X / O		
	Security	Access / Authorization		X / O	

5.3.2 Confirm Activity

Table 28 illustrates the use case that has been investigated in the interview related to the activity “Confirm Activity”.

Table 28: Use Case Description "Confirm Activity"

Use Case Number	UC_P1.2
Use Case Name	**Confirm activity**
Actors	Assisted Person (dementia early and middle stage)
Version Number + Author	V1 Özgür Ünalan
Iteration	Pre-final
Summary	This Use-Case describes the confirmation of an activity (a activity that has been monitored). For example, the system has monitored that the assisted person has cooked. The system asks the assisted person if and what exactly he or she has cooked.
Trigger/ intent	A activity was monitored and the assisted person is ready to give information
Supported goal(s) from User Needs (if applicable)	- get confirmation about monitored activity - get detailed information about activities
Preconditions	An activity needs to be recognized
Flow of events: (Main Flow)	1. The system asks the user if he wants to contribute an activity description into the diary 2. The AP (assisted person) chooses to do so 3. The system displays a list of monitored activities (like cooking) and asks the AP to decribe what he or she has cooked OR displays a possibility to create a new activity that has not been monitored previously (e.g. playing chess or watching tv) 4. The AP provides detailed information (e.g. i cooked noodles) to the activity 5. The system saves the confirmed activity (cooking of noodles) in the dementia diary. The system continues to ask about other monitored activities.
Alternate flows	2. a The user does not want to add additional information to a task 3. a The system records that the AP did not want to contribute any detailed information
Exceptional flows	
Assumptions/ rules	-
Displayed information	List of monitored activities, possibility to create a new activity, confirmation message that the information has been saved
Postconditions	The user has added additional information to a monitored activity
NFRs	- There should be low cognitive effort to provide detailed information
Services Required	This will be added later

Relation to other use cases	-
Open issues	-

Figure 14 illustrates an example mock-up that has been used to illustrate the functionality of the investigated use case.

Figure 14: Mock-up "Confirm Activity"

Results and Main Findings

Table 29 illustrates the results of the prioritization of the quality attributes from the perspective of an assisted person (indicated with a "X"). To summarize the main findings based on the comment made by the interviewee we can state the following:

- This function is suitable for patients in an early / middle stage
- Transitions from one screen to another should be slightly delayed (5-10 sec.) in case of patients with dementia to avoid excessive demands
- Digital pictures are very good;

Table 29: Prioritization of QAs related to UC "Check Activity"
(X = Perspective of Assisted Person")

	Quality Attribute		Priority		
			1	2	3
UC 2: Confirm Activity (Assisted Person (dementia early and middle stage)	Efficiency	Usage Time		X	
	Usability	Understandability	X		
		Appropriate Feedback	X		
		Comprehensible Icon Language (see previous rating)	X		
		Consistency	X		
		Undo, abort feasability	X		
		Navigation Support	X		
	Security	Access / Authorization		X	

5.3.3 Check Activity

Table 30 illustrates the use case that has been investigated in the interview related to the activity “Check Activity”.

Table 30: Use Case Description "Check Activity"

Use Case Number	UC_P1.3
Use Case Name	**Check Activity**
Actors	Caregiver, relative, (assisted person)
Version Number + Author	V1, Özgür Ünalan
Iteration	Pre-final
Summary	This use case describes how the monitored and confirmed activities can be checked by caregivers, relatives or the assisted person themself
Trigger/ intent	The user wants to check the status of the monitored and confirmed activities
Supported goal(s) from User Needs (if applicable)	- get an overview about the activities of the assisted person
Preconditions	The user is authenticated to check the activities of the assisted person
Flow of events: (Main Flow)	1. The actor chooses to check the activities of the assisted person 2. The System asks the actor to provide authentication information (e.g. user name and password) 3. The actor provides their security information 4. The system authenticates the actor and offers different views for the monitored and confirmed activities of the assisted person. 5. The user views the activities by looking at the different views (monitored and confirmed activities, chronological view, as-is and to-be activities comparison, etc.) offered by the system. After a while, the user chooses to log out. 6. The system logs the user out.
Alternate flows	
Exceptional flows	2a. The system recognizes a unsuccessful login more than 3 times. 3a. The system sends a message to the person responsible for the administration of the system, that there might be a attempt of breaking the security of the system 4a. The administrator takes countermeasures 5b. If the user does not log out and there is a time-period of 5 minutes of inactivity, the system logs out the user automatically.
Assumptions/ rules	The assisted person has given permission that the caregiver or relatives can check their activities.
Displayed information	List of activities, Confirmed activities monitored activities, chronological view of activities, as-is and to-be activity

	comparison.
Postconditions	The actor has gained an overview about the activities of the assisted person
NFRs	This use case should have a high security level. The actor needs to be authenticated
Services Required	This will be added later
Relation to other use cases	-
Open issues	-

Figure 15 illustrates an example mock-up that has been used to illustrate the functionality of the investigated use case.

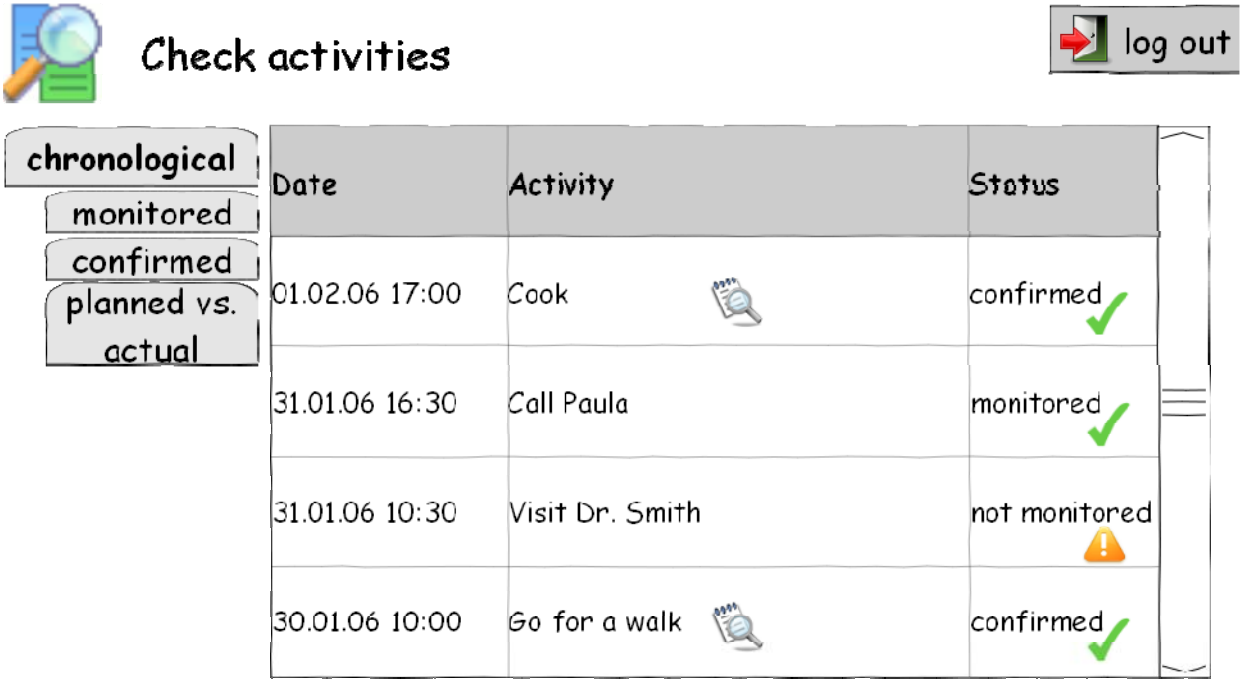

Figure 15: Mock-up “Check Activity”

Results and Main Findings

Table 31 illustrates the results of the prioritization of the quality attributes both from the perspective of an assisted person (indicated with a “X”) as well as a caregiver / relative (indicated with an “O”). To summarize the main findings based on the comment made by the interviewee we can state the following:

- Access / authorization is of importance in dependence of particular family situation
- Letter of agreements / consent forms are of importance especially in the context of assisted living

Table 31: Results of Prioritization QAs related to UC "Check Activity" (X = Perspective of an Assisted Person, O = Perspective of Caregiver / Relative)

	Quality Attribute		Priority		
			1	2	3
UC 3: Check Activity (Caregiver, Relative, (Assisted Person))	Efficiency	Usage Time	X / O		
	Usability	Understandability	X / O		
		Appropriate Feedback		X / O	
		Comprehensible Icon Language (see previous rating)	X / O		
		Appropriate Wording for each target group	X / O		
	Security	Access / Authorization	X / O		

5.3.4 Leave Message

Table 32 illustrates the use case that has been investigated in the interview related to the activity “Leave Message”.

Table 32: Use Case Description "Leave Message"

Use Case Number	UC_P1.4
Use Case Name	**Leave Message**
Actors	Caregiver, relative, assisted person
Version Number + Author	V1, Özgür Ünalan
Iteration	Pre-final
Summary	This use case describes how the actor can leave a message to another actor
Trigger/ intent	The actor wants to leave a message to another actor. Mostly this will be between caregiver and relative, since the assisted person is always present.
Supported goal(s) from User Needs (if applicable)	- provide information to another actor
Preconditions	The user is authenticated
Flow of events: (Main Flow)	1. The actor (e.g. caregiver) chooses to leave a message to another actor (e.g. the relative) 2. The system provides input fields to select the recipient of the message and for the message itself. 3. The user enters the message and chooses the recipient and confirms to save the message 4. The system saves the message and optionally sends an e-mail to the recipients e-mail address. The system shows a confirmation message

	5. The next time the recipient logs into the system, the messages that were left for him are displayed by the system.
Alternate flows	
Exceptional flows	4a. The system fails to deliver the e-mail to the recipients e-mail address. 5a. The system notifies the sender of the message, that the e-mail could not be delivered to the recipient
Assumptions/ rules	The use case is not intended to be used as a reminder
Displayed information	Recipient, message text, message status (sent, read, etc.), list of messages
Postconditions	The actor has left a message to somebody else
NFRs	-
Services Required	This will be added later
Relation to other use cases	-
Open issues	-

Figure 16 illustrates an example mock-up that has been used to illustrate the functionality of the investigated use case.

Figure 16: Mock-up "Leave Message"

Results and Main Findings

Table 33 illustrates the results of the prioritization of the quality attributes both from the perspective of an assisted person (indicated with a “X”) as well as a caregiver / relative (indicated with an “O”). To summarize the main findings based on the comment made by the interviewee we can state the following:

- Understandability: probably the patient will have problems with accessing the microphone twice to record a message
- In case of patient with dementia in an advanced stage, the number of possible recipients of a message should be restricted to 1-2 persons.
- Familiarity is currently fulfilled (e.g., typewriter)
- Pin might not be easy to understand

Table 33: Prioritization of QAs related to UC "Leave a message"
(X = Perspective of an Assisted Person, O = Perspective of Caregiver / Relative)

	Quality Attribute	Priority		
		1	2	3
UC 4: Leave Message (Caregiver, Relative, (Assisted Person))	Usage Time	X	O	
	Understandability	X / O		
	Appropriate Feedback	X	O	
	Comprehensible Icon Language (see previous rating)	X / O		
	Appropriate Wording for each target group	X / O		
	Undo, abort feasability	X / O		
	Access / Authorization		X / O	

5.3.5 Create Report

Table 34 illustrates the use case that has been investigated in the interview related to the activity “Create Report”.

Table 34: Use Case Description "Create Report"

Use Case Number	UC_P1.5
Use Case Name	**Create Report**
Actors	Caregiver, doctor
Version Number + Author	V1, Özgür Ünalan
Iteration	Pre-final
Summary	This use case describes how the system provides a report about the activities of the assisted person.
Trigger/ intent	The actor wants get an overview about the activities of the assisted person, in a given period of time
Supported goal(s) from User Needs (if applicable)	- look for unusual behavior that could indicate medical problems
Preconditions	The user is authenticated
Flow of events: (Main Flow)	1. The actor chooses to create a report 2. The system offers configuration possibilities for a new report (name of report, time frame, choice of activities, etc.) or an existing report. (e.g., amount of bathroom usage per day) 3. The actor configures the report according to his requirements and chooses, to save (if new) and generate the report. 4. The system generates the report and displays it to the actor.
Alternate flows	-
Exceptional flows	-
Assumptions/ rules	Reports should not diagnose something themselves, but merely provide raw data
Displayed information	List of available reports, input form to create an report (report name, description, time frame, list of activities)
Postconditions	A report has been created
NFRs	The report should be easy to understand
Services Required	This will be added later
Relation to other use cases	-
Open issues	-

Figure 17 illustrates an example mock-up that has been used to illustrate the functionality of the investigated use case.

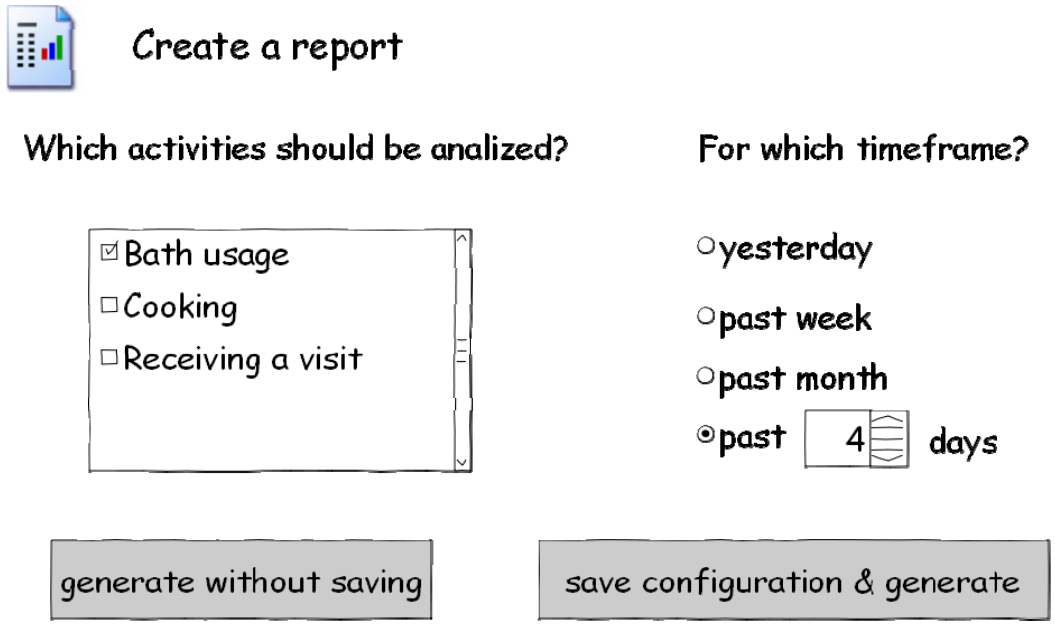

Figure 17: Mock-up "Create Report"

Results and Main Findings

Table 35 illustrates the results of the prioritization of the quality attributes from the perspective of a caregiver / relative (indicated with an “O”). To summarize the main findings based on the comment made by the interviewee we can state the following:

- Avoid lack of information: the question is, which information is relevant, might depend on particular situation and purpose of the report
- Functionality is very useful; could be supplemented with highlighting critical data
- Print-out functionality / send report per Email would be very useful

Table 35: Prioritization of QAs related to UC "Create Report" (O = Perspective of Caregiver / Relative)

	Quality Attribute		Priority		
			1	2	3
UC 5: Create Report (Caregiver, doctor)	Efficiency	Usage Time	O		
	Usability	Understandability		O	
		Comprehensible Icon Language	O		
		Appropriate Wording for each target group	O		
		Undo, abort feasability	O		
		Avoid lack of information / avoid information overload		O	
	Security	Access / Authorization	O		

5.3.6 Results (System QAs)

Table 36 illustrates the results of the prioritization of the System QAs that refer to the Dementia Diary component in general rather than on particular use cases. To summarize the main findings based on the comments made by the interviewee we can state the following:

- Adaptability (font size, contrast, etc.) is very important!
- Help system should be provided in documented form rather than integrated help function in the dementia diary

Table 36: Prioritization of QAs related to the Dementia Diary (globally)
(X = Perspective of an Assisted Person, O = Perspective of Caregiver / Relative)

	Quality Attribute	Priority 1	Priority 2	Priority 3
	Accessibility		O	X
	Adaptability / Customizability	X	O	
	Experience / Familiarity		X	O
	Availability of a help system	O	X	
		X	O	
	Relevant functionalities	X / O		
	Perceived usefulness	X / O		
	Satisfaction	X / O		

6. System Level Use Cases

In the following, we describe a set of system level use cases that describe how the system reacts to actions performed by the user.

Solution scenario Dementia diary	RQ 1.1.1 System tracks activity – record sleep/wake cycle
Solution scenario description	Peter is awaken and gets up from bed or goes to be.
Actors	End-user
Required technology (abstract)	Activity tracking, sleep and wake detection.
Proposed Technology	
HW	Pressure sensors or sensing mat in the bed and/or passive infrared or ultrasound motion sensor over the bed with a restricted view.
SW	Database, inference algorithm
Architecture	Wireless and/or wired sensors
	Health hub
Feasible	Yes
Desirable	Yes

Solution scenario Dementia diary	System tracks activity – forgotten cooking
Solution scenario description	Peter left his meal on the stove.
Actors	End-user
Required technology (abstract)	Activity tracking in the kitchen, ability to switch off the power of the stove.
Proposed Technology	
HW	Passive infrared or ultrasound motion sensors, power or gas consumption sensors, humidity and temperature sensors above the stove (e.g. on the ceiling), and/or smoke detector. Actuators: Relay to switch off the power, or remotely controlled gas tap. Future: Intelligent appliances (fridge, stove, microwave oven, etc.).
SW	Database, inference algorithm
Architecture	Wireless and/or wired sensors
	Health hub
Feasible	Yes generally, sensing gas consumption is hard to implement, also switching off a gas stove is problematic.
Desirable	Yes

Solution scenario Dementia diary	System tracks activity – falling down of patient
Solution scenario description	Peter fell down.
Actors	End-user
Required technology (abstract)	Activity tracking
Proposed Technology	
HW	Capacitive (or resistive, etc.) sensing mats on the floor, or camera (the act of falling down) or microphones (sound of falling and asking for help) with data processing system, or device on the patient, or sudden no activity in the home with no good reason.
SW	Database, inference algorithm
Architecture	Wireless or wired sensors
	Health hub server outside of the home
Feasible	Yes, effort depends on technology; various research projects investigate these alternative technologies.
Desirable	Yes

Solution scenario Dementia diary	System tracks activity – visitor arrives
Solution scenario description	Peter has been visited by someone.
Actors	End user and third party (visitor)
Required technology (abstract)	Activity tracking in front of the house, on the front door, in the anteroom, doorbell sensor.
Proposed Technology	
HW	Electronic contact sensor on the door, passive infrared or ultrasound motion sensors, sensing mats (on the floor), "door bell is pressed" sensor (simple contact sensor).
SW	Database, inference algorithm
Architecture	Wireless or wired sensors
	Health hub server outside of the home
Feasible	Yes
Desirable	Yes

Solution scenario Dementia diary	System tracks activity – visitor leaves the house
Solution scenario description	Peter's visitor has left the house.
Actors	End user and third party (visitor)

Required technology (abstract)	Same as visitor arrived.
Proposed Technology	
HW	Same as visitor arrived
SW	Database, inference algorithm
Architecture	Wireless or wired sensors Health hub server outside of the home
Feasible	Yes
Desirable	Yes

Solution scenario Dementia diary	System tracks activity – patient has left the property
Solution scenario description	Peter has left her property.
Actors	End-user
Required technology (abstract)	Same as visitor arrived.
Proposed Technology	
HW	Same as visitor arrived, plus GSM/GPS tracking device, or RFID tags on glasses, shoes, etc. (tracking may be prohibited by ethical and legal issues), RFID reader near the door. In the future we may use the smartphone of the end user because it will be typical and natural for older people also to use them.
SW	Database, inference algorithm
Architecture	Wireless or wired sensors, and/or GSM, GPS sensor, or smartphone (future) Health hub server outside of the home
Feasible	Yes
Desirable	Yes

Solution scenario Dementia diary	System tracks activity – patient has left the door of fridge opened
Solution scenario description	Peter has left the door of the fridge opened.
Actors	End-user
Required technology (abstract)	Activity tracking in the kitchen, sensors on the fridge.
Proposed Technology	
HW	Electronic contact sensor on the door of the fridge, or temperature sensor and/or brightness sensor (internal lamp), humidity sensor can also be used to enhance detection. Future: Intelligent fridge.
SW	Database, inference algorithm
Architecture	Wireless or wired sensors

	Health hub server outside of the home
Feasible	Yes
Desirable	Yes

Solution scenario Dementia diary	System tracks activity – patient did not have her/his meal
Solution scenario description	Peter has not had her meal.
Actors	End-user
Required technology (abstract)	Activity tracking in the kitchen and on the fridge.
Proposed Technology	
HW	Fridge: • Electronic contact sensor on the door, • Temperature sensor and humidity sensor (can enhance detection), • Brightness sensor (internal lamp in the fridge), Meal on the stove: 4. Passive infrared or ultrasound motion sensors, 5. Power consumption, 6. Temperature and/or humidity sensors over the stove. Practically the combination the " patient has left the door of fridge opened" and " forgotten Cooking" Future: Intelligent appliances such as fridge, stove, microwave oven, etc.
SW	Database, inference algorithm
Architecture	Wireless or wired sensors Health hub server outside of the home
Feasible	Yes, false positive/negative rate may be high due to complexity
Desirable	Yes

Solution scenario Dementia diary	System tracks activity – patient left the iron switched on
Solution scenario description	Peter has left the iron switched on.
Actors	End-user
Required technology (abstract)	Activity tracking in the room where iron is, power consumption, smoke, ability to switch off the power.
Proposed Technology	
HW	Power consumption, temperature sensors, smoke detector. Future: intelligent iron.
SW	Database, inference algorithm
Architecture	Wireless or wired sensors Health hub server outside of the home
Feasible	Yes

Desirable	Yes

Solution scenario Dementia diary	System tracks activity – patient used the toilet
Solution scenario description	Peter has used the toilet.
Actors	End-user
Required technology (abstract)	Activity tracking in the toilet and in the bathroom (optional).
Proposed Technology	
HW	Door contact sensor, passive infrared or ultrasound motion sensors, sensing mats, brightness sensor (light switched on), and water flow sensor.
SW	Database, inference algorithm
Architecture	Wireless or wired sensors Health hub server outside of the home
Feasible	Yes
Desirable	Yes

Solution scenario Dementia diary	System detects danger – heating is switched off or not operational in cold weather		
Solution scenario description	Peter switches of heating appliances in the home or the appliance shuts itself down (malfunction).		
Actors	End-user		
Required technology (abstract)	Temperature sensing in the home.		
Proposed Technology		**Partners contributing**	**Status**
HW	Temperature sensor		
SW	Database, inference algorithm		
Architecture	Wireless or wired sensors		
	Health hub server outside of the home Connection to publicly available weather information on the WEB		
Feasible	Yes		
Desirable	Yes		

Solution scenario Dementia diary	System detects danger – door or window has been left open
Solution scenario description	Peter left a door or a window open, or not vented the house for a long time.
Actors	End-user

Required technology (abstract)	Door and window contact sensors, temperature.
Proposed Technology	
HW	Door/window contact sensor, temperature sensors.
SW	Database, inference algorithm
Architecture	Wireless or wired sensors
	Health hub server outside of the home
Feasible	Yes
Desirable	Yes

Solution scenario Dementia diary	System detects danger – high carbon monoxide level
Solution scenario description	Chimney has got clogged, no venting for a long time, fatal situation.
Actors	End-user
Required technology (abstract)	Measuring carbon monoxide level.
Proposed Technology	
HW	Carbon monoxide meter and alarm device (strong light + sound).
SW	Database, (simple) inference algorithm
Architecture	Wireless or wired sensors
	Health hub server outside of the home
Feasible	Yes
Desirable	Yes

Solution scenario Dementia diary	System detects danger – smoke in the house
Solution scenario description	Chimney has got clogged or there is fire in the house.
Actors	End-user
Required technology (abstract)	Measuring smoke level.
Proposed Technology	
HW	Smoke sensor and alarm device (strong light + sound).
SW	Database, (simple) inference algorithm
Architecture	Wireless or wired sensors
	Health hub server outside of the home
Feasible	Yes
Desirable	Yes

Solution scenario	System detects danger – gas leakage in the house

Dementia diary	
Solution scenario description	A gas pipe or valve cracked, or there is a leak.
Actors	End-user
Required technology (abstract)	Measuring gas concentration.
Proposed Technology	
HW	Gas measuring or an alert device.
SW	Database, (simple) inference algorithm
Architecture	Wireless or wired sensors
	Health hub server outside of the home
Feasible	Not likely, sensors are complex and pricy
Desirable	No

Solution scenario Dementia diary	System avoids danger- motion in a room at night
Solution scenario description	Peter has forgotten switch on the light at night (e. g. she goes to toilet), the system switches lights on.
Actors	End-user
Required technology (abstract)	Activity tracking in the property, knowing the time or sensing brightness, programmatically switchable (controllable) lights.
Proposed Technology	
HW	Passive infrared or ultrasound motion sensors, sensing mats, brightness sensor, controlled light switch (as actuator)
SW	Database, (simple) inference algorithm
Architecture	Wireless or wired sensors
	Health hub server outside of the home
Feasible	Yes
Desirable	Yes, low level light is required (no blinding).

Solution scenario Dementia diary	System avoids danger- not closed tap, running water
Solution scenario description	Peter has forgotten to cut off the water tap.
Actors	End-user
Required technology (abstract)	Activity tracking in the property, sensing flush or water consumption.
Proposed Technology	
HW	Passive infrared or ultrasound motion sensors, water flow meter, flush sensor, water consumption meter
SW	Database, (simple) inference algorithm
Architecture	Wireless or wired sensors

	Health hub server outside of the home
Feasible	Yes, but flow meters are pricy, flush sensors can sense problem to late.
Desirable	Yes

Solution scenario Home Gateway	System maintenance - Home Gateway is set up by a Technician
Solution scenario description	Technician sets up the Home Gateway. The system records the event, and changes in the configuration
Actors	Technician
Required technology (abstract)	Providing a user interface, making a log file.
Proposed Technology	
HW	Home Gateway, access device (notebook, tables, iPad, etc.)
SW	Database, WEB interface to be able to set the system from far away, logging and configuration management mechanisms.
Architecture	WEB server on the Home Gateway
	Health hub server outside of the home
Feasible	Yes
Desirable	Yes

7. Integration possibilities and interface constraints of external devices

7.1 home automation

One of the main standards in the home automation area is the EIB-standard (European Installation Bus). The IGD Living Lab is currently equipped with an EIB-standard home automation control system called KNX. As a home automation system, it allows the control of lighting, heating and domotic devices.

Differences between Home Automation Systems (HAS) and Building Automation Systems (BAS)

The purpose of Home Automation Systems in general is to improve both interaction and communication between typical domotic devices. While Building Automation Systems refer to large buildings like office complexes or health care facilities, the term home automation is instead used to refer to small installations in the context of small residences. These distinct fields have different requirements and deal with a different amount of complexity. What combines both fields is the handling of the data used for device control as a core component of both kinds of systems.

Data distribution in Home Automation systems

A central challenge in home automation systems is that small amounts of (usually sensor-) data, which usually are sent rather rarely need to be sent over longer distances with a high degree of signal robustness. This central element comes into play in the control of an environment with the optional integration of other application components. Within the field of Home Automation systems, devices can be classified into the following categories:

Lighting and window blinds
Safety alarm system
Brown goods (audio/video or home theatre equipment, game consoles)
White goods (household appliances), like a washing machine or stove
Heating, Air conditioning and Ventilation systems (HVAC)
Communications equipment (intercom system, telephone)
Security and access control
Elevators and sundry special domains
Information processing and presentation equipment (tablet PCs, PDAs, PCs)

An example for automated homes

To give an example, light can be turned off in a room when there is no one present or switched on once sensors within the room detect the arrival of a human being. HVAC system functions can react to the present temperature, but also to other

factors like the opening or closing of a window. While BAS systems mainly focus on energy savings, the main goal of HAS systems is the increase of comfort and usability.
The current implementation of the KNX-standard in the AAL Lab in Darmstadt supports the key artifacts to be controlled by this protocol. Controllable devices are individually dimmable room lights, energy control, the control of the state of all electric plugs installed within the confines of the AAL Laboratory, thus providing the KNX-installation with complete control over the energy supply of all electric devices with a connection to the energy grid. The window with blinds in the AAL Lab is also connected to the KNX system and can be opened or closed and the blinds moved up and down.

Connecting PCs to the EIB control system

The connection between PCs and the EIB installation is possible through a variety of different means as the standard is meant to be compatible with standard plugs used in the field. As such the EIB system can be operated through a USB connection or via Network cable with the so-called KNXnet/IP, a Network protocol for the KNX-standard, which is a development based on the previous EIBlib/IP-protocol. The system also supports a connection via RS232 adapters, which is an older standard, but it shows that efforts have been made to also support older hardware without the now ubiquitous USB connectors.

Data Transmission possibilities

Transmitting Data between the devices is also possible through different means. There are four supported means of transmission:

- IP (standard IP protocol, device connected to the Server via network cable)
- RF – Radio Frequency (Radio connectors for wireless transmission)
- Powerline (PL110 and PL132) (allows data transmission through electrical power wires)
- Twisted Pair (TP0 and TP1)

These transmission possibilities also require the EIB-compatibility of the attached hardware for system integration. Connecting other devices to this system requires an additional abstraction layer. In the Darmstadt Living Lab this problem is solved by installing the middleware from the EU project Persona on top of the KNX installation and attach additional hardware to the Persona middleware.

Figure 18 below shows typical home automation technology that is available that can be incorporated with CCE solutions. Figure 18 Shows home automation solutions from Oasis.

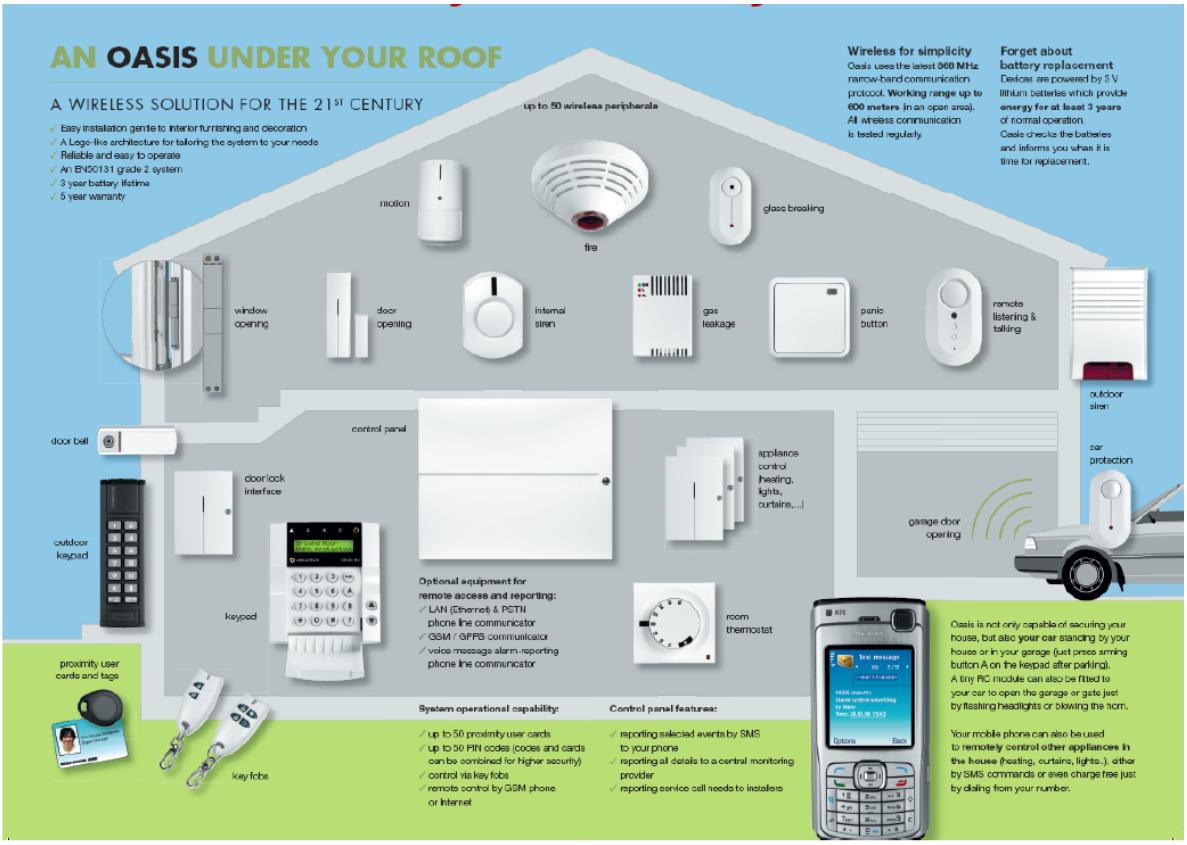

Figure 18: OASIS Home Automation Solution

7.2 Pilot Lab Issues in the UK

Pilot Lab in the UK is based at Bournemouth Council Draper Road Dementia Extra Care Homes. The facility comprises of 20 flats built specifically for dementia suffers, which will be built by April 2011. The facility has a demonstration room for residents, their carers and professional health carers, which could be used to demonstrate the CCE technology and solutions. Figure 17 below shows the technology that may be incorporated into the facility.

Figure 19 shows the technology that may be incorporated into the Draper Road Dementia Extra Care Homes

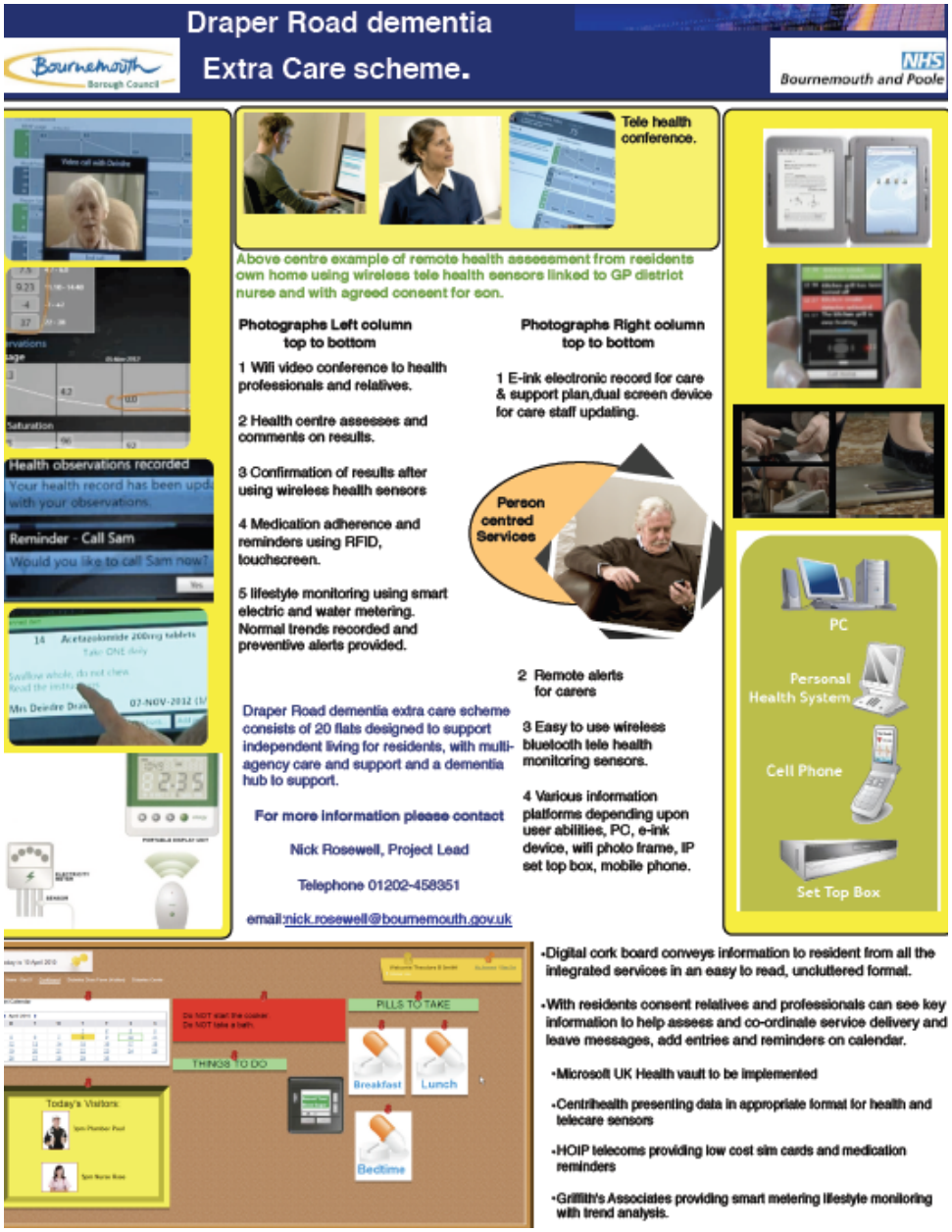

Figure 19: UK Pilot Lab Setup

Information on the other Pilot Labs in Germany and Hungary can be found in [Del5.1] and [Del1.3].

8. Conclusion

In this deliverable, we described an update of the CCE use cases in section 2. The update was a result of an internal project consolidation after the submission of deliverable 1.3. As the major focus of this deliverable was on non-functional requirements, we presented the results of an elicitation and prioritization of quality attributes / nonfunctional requirements that was performed within the consortium and with external experts. As these findings will be crucial for further system development, we will now present a short interpretation of the partner and user prioritization.

8.1 Summary and Interpretation

On a first view, the prioritizations from BME, IGD and DRK differ completely in their result. But on a deeper analysis of the "very important" rated sub attributes (see Table 37) there is a high commonality. A conclusion for the development of the system should be that developers need to have a deep analysis of important sub attributes and should take care of compliance with elicited NFR's on a sub attribute level. There are only four sub attributes that were rated as very important from one party and not important from another party (in particular: localization, navigation support, accessibility, and costs). For all other attributes rated very important from one party it is at least important for the other involved stakeholders if a prioritization of them for the specific sub attribute was available.

It turns out that efficiency is the least important quality attribute due to the fact, that the system will be used by older people and quick system responses are not necessary for them.

Table 37: Comparison of high prioritized sub attributes

Sub attribute:	**Priority BME:**	**Priority IGD:**	**Priority DRK:**
Response Time	very important	very important	n/a
Integrity	very important	very important	n/a
Last saved status	very important	very important	n/a
Privacy	very important	important	n/a
Access/ Authorization	very important	very important	very important
Awareness	very important	very important	n/a
Changeability	very important	important	n/a
Stability	n/a	very important	n/a
Adaptability	very important	important	very important
Co-Existence	very important	very important	n/a
Replaceability	very important	not important	n/a
Avoid information overload	Important	very important	n/a
Avoid lack of information	very important	important	very important
Localization	very important	not important	n/a
Understandability	very important	very important	very important
Appropriate feedback of system activities	very important	very important	very important

Comprehensible Icon Language	very important	n/a	very important
Navigation Support	not important	Important	very important
Consistency	very important	Important	very important
Appropriate wording for each target group	very important	very important	very important
Undo/ abort feasibility	important	very important	very important
Accessibility	not important	very important	not important
Relevant functionalities	very important	very important	very important
Costs	very important	not important	n/a
Perceived usefulness	very important	very important	very important
Unobtrusiveness of system components	very important	very important	n/a
Satisfaction	very important	Important	very important

9. References

[1] Chung, L., Nixon, B.A., Yu, E., Mylopoulos, J.,“NFR in Software Engineering”, Kluwer Academic Publishers, 2000

[2] Cysneiros, L.N., Leite, J.C.S.P, “Driving NFR to Use Cases and Scenarios”, XV Brazilian Symposium on Software Engineering, 2001

[3] Dzida et al., “Gebrauchstauglichkeit von Software: ErgoNorm. Ein Verfahren zur Konformitätsprüfung von Software auf der Grundlage“ DIN EN ISO 9241 parts 10 and 11. Bremerhaven: Wirtschaftverlag, 2001

[4] Russel, D, Gangemi, G.T., “Computer Security Basics”, O’Reilly, 1991.

[5] Kerkow, D., Doerr, J., Paech, B., Olsson, T., Koenig, T., “Elicitation and Documentation of Non-functional Requirements for Sociotechnical Systems” in José Luis Maté, Andrés Silva, “Requirements Engineering for Sociotechnical Systems”, Idea Group, Inc., 2004

[6] Jörg Dörr, Elicitation of a Complete Set of Non-Functional Requirements, Dissertation, Fraunhofer IESE/TU Kaiserslautern 2010.

[Del1.3] CCE Deliverable D1.3

[Del3.1] CCE Deliverable D3.1

[Del4.1] CCE Deliverable D4.1

[Del5.1] CCE Deliverable D5.1

[7] ISO/IEC 9126-1:2001 Software engineering — Product quality — Part 1: Quality model

[8] Gross A. et al: “System Requirements Report” (Deliverable D6.1, WP6, 2008), emerge-project.eu

10. Appendix