



Remote Accessibility to Diabetes Management and Therapy in Operational Healthcare Networks

REACTION (FP7 248590)

# D2-7 Validation Framework

Date 2011-02-28

Version 1.1

Dissemination Level: Public

# Table of Contents

# Document control page

| | |
|---|---|
| **Code** | D2-7_Validation_Framework_v1.1_FORTH.doc |
| **Version** | 1.0 |
| **Date** | |
| **Dissemination level** | PU |
| **Category** | R |
| **Participant Partner(s)** | FORTH, ATOS, BTS, CHC, DELTA, FHG-SIT, IMM, IN-JET, MUG, SOLIANIS |
| **Author(s)** | F. Chiarugi (FORTH), V. Kontogiannis (FORTH), D. Manousos (FORTH), G. Zacharioudakis (FORTH), J. Thestrup (IN-JET), H. Udsen (IN-JET), T. Klotzbuecher (IMM), C. Cavero Barca (ATOS), M. Enzmann (FHG-SIT), F. Franke (FHG-SIT), A. Megej (SOLIANIS), R. Gronbek Haahr (DELTA), K. Neubauer (MUG), L. Schaupp (MUG), Th. Eissing (BTS), S. Schaller (BTS), J. Fursse (CHC) |
| **Verified and approved by** | |
| **Work Package** | WP2 |
| **Fragment** | No |
| **Distribution List** | All |
| **Abstract** | This deliverable contains the definition of the verification (internal) and validation (in the clinical sites) procedures to be performed in each prototype cycle. |
| **Comments and modifications** | |
| **Status** | ☐ Draft<br>☒ Task leader accepted<br>☒ WP leader accepted<br>☐ Technical supervisor accepted<br>☐ Medical Engineering supervisor accepted<br>☐ Medical supervisor accepted<br>☒ Quality manager checked<br>☐ Project Coordinator accepted |
| **Action requested** | ☐ to be revised by partners involved in the preparation of the deliverable<br>☐ for approval of the task leader<br>☐ for approval of the WP leader<br>☐ for approval of the Technical Manager<br>☐ for approval of the Medical Engineering Manager<br>☐ for approval of the Medical Manager<br>☐ for approval of the Quality Manager<br>☐ for approval of the Project Coordinator<br>**Deadline for action**: N/A |
| **Keywords** | |
| **References** | |
| **Previous Versions** | |

**Version Notes**

| Version | Author(s) | Date | Changes made |
|---|---|---|---|
| 0.1 | F. Chiarugi, G. Zacharioudakis | 2010-12-28 | TOC and task assignment. |
| 0.2 | F. Chiarugi, G. Zacharioudakis, H. Udsen | 2011-01-07 | TOC and task assignment. |
| 0.3 | F. Chiarugi, T. Klotzbuecher, L. Schaupp, H. Udsen | 2011-01-16 | Consolidation of TOC and task assignment. |
| 0.4 | F. Chiarugi, C. Cavero Barca, M. Enzmann, F. Franke | 2011-01-31 | Provision of FORTH's contents and contributions (security) from ATOS and FHG-SIT. |

| | | | | |
|---|---|---|---|---|
| | 0.5 | F. Chiarugi, D. Manousos, G. Zacharioudakis, A. Megej, R.Gronbek Haahr | 2011-02-02 | Further upgrade of FORTH's contents, insert Solianis and Delta contents. |
| | 0.6 | F. Chiarugi, V. Kontogiannis, K. Neubauer, L. Schaupp, Th. Eissing, S. Schaller, J. Fursse, J. Thestrup, H. Udsen | 2011-02-06 | Further upgrade of FORTH's contents, insert IN-JET (framework and user validation plans added, structure changed), CHC, MUG and BTS contributions. |
| | 1.0 | F. Chiarugi, J. Thestrup, H. Udsen | 2011-02-25 | Corrections and elaboration of Jesper's and Lukas's comments. Incorporation of reviewers' comments. |
| | 1.1 | F. Chiarugi, J. Thestrup | 2011-02-28 | Final version submitted to the EC. |

| **Internal review History** | Reviewed by | Date | Comments made |
|---|---|---|---|
| | Matts Ahlsen (CNET) | 2011-02-15 | Approved with comments. |
| | Stephan Spat (MSG) | 2011-02-15 | Approved with suggestions for improvement. |

# 1. Executive Summary

This deliverable is the result of task T2.4 in WP2 carried out in the three subtasks of validation planning, organization of the application field trials and organization of the deployment preparation.

This document provides some hints about the system development life cycle adopted in the REACTION project. In such context it describes the validation framework illustrating the different procedures implemented in order to assure that not only the REACTION platform is built right but also that we are building the right platform, i.e. the one matching the actual end user requirements.

An accurate validation framework is a major objective of the REACTION project, since it provides the skeleton for all the verification and validation procedures that have to be performed during the entire lifecycle of the platform. Such skeleton includes also some constraints which impacts in the global structure of the design and development process which must be undertaken taking into consideration some new rules recently emerged in the medical software scenario (e.g. the recent amendment of the medical device directive).

The validation framework provides a well-described methodology to serve as a baseline on how, when and by whom validation is going to take place. The validation framework provides guidance for carrying out the validation activities and for driving the decisions about redesign, error correction, start of implementation etc., on the basis of the validation results.

Considering that the REACTION project envisages four iteration cycles of requirement, design, development, verification and validation, a proper operation of the tests at all levels will provide the necessary retrofits to the subsequent iteration, thus driving the project towards a better match with the actual user requirements.

The present validation framework has been placed in the adopted system development life cycle and describes a well-defined structure for the software testing and user validation already in the beginning of the project according to the verification and validation (V&V) methodology, which is agreed by all partners.

The overall validation activities in REACTION consist of three distinctly different elements:

1. Verification, i.e. a quality control (QC) process that is used to evaluate whether or not an artefact, product, service, or system complies with regulations, specifications, or conditions imposed at the start of a development phase
2. User validation, i.e. a quality assurance (QA) process of providing a high degree of assurance that a product, service, or system accomplishes its intended requirements namely the expectations and requirements of its intended users
3. Usability testing, i.e. tests in order to assess the quality of use of the applications.

While verification has to be performed by the technical partners and will mainly involve the test of the software at its different stages of integration, the user validation including the usability testing have to be conducted by the clinical partners with real end users.

It should be noted that traditional clinical research and validation of the clinical protocols is outside the scope of the project.

Guidelines for organizing the various tests have been described as well as guidelines for properly reporting the results of the tests so that they can be useful for the next design phase.

The process followed is similar in all validation cycles and foresees fixed steps to follow: an initial preparation part, an internal verification activity and/or a validation activity with (expert) end users, the collection and analysis of the outcomes and feedback of the results into the loop for the next step in the process.

# 2. Definitions and Abbreviations

| | |
|---|---|
| A&E | Accident and Emergency Department |
| API | Application Programming Interface |
| BPM | Blood Pressure Measurement |
| BTS | Bayer Technology Services GmbH |
| CASE | Computer Aided Software Engineering |
| CE | Conformité Européenne |
| CGM | Continuous Glucose Monitor |
| CHC | Chorleywood Health Centre |
| DELTA | Delta Dansk Elektronik |
| EC | European Community |
| ECG | Electrocardiogram |
| eDSS | Electronic Decision Support System |
| EEC | European Economic Community |
| ePatch | Electronic Plaster |
| EPR | Electronic Patient Record |
| EU | European Union |
| GP | General Practitioner |
| HIS | Hospital Information System |
| ICT | Information and Communication Technologies |
| IEC | International Electrotechnical Commission |
| IEEE | Institute of Electrical and Electronics Engineers |
| IMM | Institut fuer Mikrotechnik Mainz GmbH |
| ISO | International Organization for Standardization |
| JIRA | Issue and Project Tracking Tool by Atlassian |
| MDD | Medical Device Directive |
| MS | Multisensor |
| MSG | JOANNEUM RESEARCH Forschungsgesellschaft MBH |
| MUG | Medizinische Universitaet Graz |
| N.A. | Not Applicable |
| PC | Personal Computer |
| PDA | Personal Device Assistant |
| PDF | Portable Document Format |
| PR | Property Rights |
| QA | Quality Assurance |
| QC | Quality Control |
| RPM | Remote Patient Monitoring |
| SDLC | System Development Life Cycle |
| SoA | Service-oriented Architecture |

| | |
|---|---|
| SOAP | Simple Object Access Protocol |
| SSL | Secure Sockets Layer |
| SUMI | Software Usability Measurement Inventory |
| SVN | Subversion Software Versioning System |
| TC | Test Case |
| TLS | Transport Layer Security |
| UI | User Interface |
| V&V | Verification and Validation |
| WP | Workpackage |
| WS | Web Service |
| WSDL | Web Service Description Language |

# 3. Introduction

## 3.1 Purpose, Context and Scope of This Deliverable

In this section we discuss the background and context of this deliverable. We also describe the target audience, the purpose and the scope of this document.

### 3.1.1 Background and Context

The background and context of the work performed and described in this deliverable follow the first phase of the project design and aims at identifying a uniform framework for the verification and validation of the various subsystems and prototypes which will be built during the whole duration of the project.

This deliverable is the result of the activities performed till the 12th month in T2.4 “Validation of platform and services” of WP2 and it is more specifically related to the subtask 2.4.1 “Validation planning”.

### 3.1.2 Target Audience

The target audience is mainly the technical and clinical people of the consortium partners which will be involved in the verification and validation procedures of the REACTION prototypes.

### 3.1.3 Purpose

The purpose of this deliverable is to describe the approach that will be adopted for the testing and validation phase of the subsystems and prototypes of the REACTION platform and to describe the procedures and the documentation structure that will be produced during these phases.

### 3.1.4 Scope

The scope of this deliverable is limited to the definition of the validation structure and not to report results from testing or validation procedures. The validation procedures are described taking into account the evolutionary design methodology used in this project with production of a prototype, validation, review of the requirements based also on the results of the validation and design of a next prototype. Furthermore, also the need of verification and validation at level of the subsystems and then of the prototype has been considered as well. Finally the different aspects of the internal test and the test to be performed at the clinical sites have been analyzed and reported including the differences in the validation of the different envisaged prototypes.

## 3.2 Outline

In section 4 of this deliverable the work performed in order to organize the validation activities is accurately described, starting from the main phases of the REACTION user centred development and the description of the activities performed in Task 2.4 of WP2 and more specifically in subtask 2.4.1. Finally a planning of the verification and validation activities with their main goals is presented for all the prototypes envisaged in the REACTION project.

In Section 5 the internal verification procedures are described. These procedures are focused on the verification of the subsystems and prototypes to be performed at the technical partner premises and based on tests against the requirements available at the beginning of each iteration. The correspondence between each test and the design phase is explained as well. The limitations of these tests are discussed together with the use of simulators or additional tools where necessary. Then the test environment is described and the different types of tests are illustrated. The tests of units, subsystems and prototypes are described testing for the software are illustrated and special attention is dedicated to the procedures necessary for the adherence to the relevant standards. An approach based on test cases is shown for the verification of the prototypes and finally the structure of the “Internal Test Report” is presented.

In Section 6 the organization of the user validation activities is reported with its main purpose and the implemented approach. The validation process is described with the planning and preparation of the validation activities. User needs are extracted from the JIRA requirements and used to prepare validation templates and questionnaires to be used during the validation activities. The structure of a user validation report and the timelines for the validation process are finally illustrated.

In Section 6 the organization of the field trial usability testing activities is described with the purpose of validating the prototypes at the clinical sites with real end-users. The different types of procedures to be applied in the different environments and for the different trials are reported with an accurate analysis of the parameters, stakeholders and the metrics for user satisfaction. The safety, usability and performance tests are presented together with the structure of the “End User Validation Report” to be used as an input for the next design iteration.

In Section 8 the organization of the deployment preparation activities is shown with the objective of providing an overview of the potential validity of the clinical applications and validating their economic benefits.

# 4. Validation Framework

Validation is part of the implementation of a user-centred development process. The main aim is to assure that the REACTION services developed adheres to the necessary quality standards for professional services, meets the needs and requirements of users and customers, and can be recommended for adoption.

The specific objectives of the validation work will be to obtain feedback of the applied technologies from all stakeholders involved in order to evaluate the potential clinical value and validate the impact on clinical workflows from the REACTION applications with special focus on validating feedback and sensor performance as well as potential for interoperability and scalability.

Validation activities assume different forms in the phases of the project:

- In the requirements phase the focus is on the analysis of user needs and requirements, and the context in which the new service will be used. In this phase the focus will be on contextual design methods (Beyer1998).
- In the design and implementation phase the early and efficient detection of quality shortcomings and flaws in the design is the main aim of user validation. Methods used can include verification, expert evaluation, cognitive walkthroughs and usability tests of prototypes. Users should be involved as soon as a sufficiently stable and capable prototype is available.
- Usability measurements are the domain of the field trials in WP8, where controlled conditions are needed to assure that valid and interpretable results are generated. Other methods used should include subjective assessment of system quality by users, and the collection of data, which can serve as performance benchmarks for customers.
- In deployment phase, the project has strong focus on the customers' (i.e. healthcare provisioning bodies) point of view in form of development of sustainable business models. Proposals are made as to how both qualities for the user and for the customer are to be measured.

One of the objectives of the user validation approach is to use the synergy between the two applications (in-hospital and outpatient) as much as possible by using common methods, and by looking for complementary results. A set of methods is proposed for the two application scenarios and initial user validation plans are drawn up for each application and will be updated as needed.

## 4.1 The Main Phases of REACTION User Centred Development

There are three main phases of user centred design, which partly correspond to the project phases. In all phases the objective is to generate information by user analysis, which guides the design and development activities.

- Analysis of system requirements, user needs, and application context – involving all stakeholders.
- Evaluation of design concepts early (UI specifications, design ideas, and early prototypes).
- Test working prototypes with real users (as early as possible) and feed results back to the development team.

The REACTION project contains four iterative cycles which will result in four prototypes:

- End of year 1: Rapid prototype of closed-loop system to carers to be used in general ward (including some software mock-ups).
- End of year 2: Prototype of outpatient closed-loop system to clinicians and patients and improved closed-loop system to carers used in general ward (including sensor prototypes).
- End of year 3: Partly/fully functional prototypes of in-hospital and outpatient prototypes including relevant features such as multi-parametric monitoring, risk analysis and full backend interoperability (depending on the domain).

- End of year 4: Automatic glycaemic control with closed-loop feedback directly to insulin dosage pumps and field trials with final prototypes.

During platform development, validation is carried out to detect possible deviations from the original objectives and to provide feedback to the development team and to the Project Board for early corrective action. To this end, project progress is assessed in yearly intervals corresponding to the four iterative cycles to allow tight, results-oriented monitoring of project status.

Annual user validation will be eventually concluded with the usability testing during field trials defined in WP8, which will demonstrate the benefit provided for individual users and healthcare organisations in terms of efficiency of closed loop healthcare provisioning in diabetes management. The field trials will also be used to evaluate the potential validity of the clinical applications, and the benefit for the healthcare domain, acceptance by patients and other users, and to assess the impact on the organizational level. Fields trials in the in-hospital environment can only be carried out (concerning interaction with patients) if the requirements of the MDD are met. User tests with clinicians will be performed without this precondition.

It is essential that the results of user validation are addressed towards the individuals and groups who are able to use and implement them to improve design quality. In this respect, design refers to the entire software platform and other relevant features, which determine the user experiences when interacting with the applications developed, i.e. functionality, graphical and navigation design, and also quality factors such as performance and productivity, security, added value, etc.

The validation work is the subject of task T2.4 in WP2. It will be carried out in three subtasks:

#### Subtask 2.4.1 – Validation Planning

The validation framework provides a well-described methodology to serve as a baseline on how, when and by whom validation is going to take place. The framework guides the collection of information about the project specific objectives, requirements and constraints on user validation (different methods measure different quality dimensions). The validation framework will provide guidance for carrying out the validation activities and for making the decisions about redesign, error correction, start of implementation etc., on the basis of the validation results. This task will be performed in close cooperation with the developers (technical partners) of the prototypes.

#### Subtask 2.4.2 – Application Field Trials

Usability will be tested in the field trials in WP8 with a small number of users to detect user problems and deficiencies of the prototypes early in the development process and to feed these back to the development teams. This subtask reports the conclusions of the field trials in order that common assessment criteria are adopted to allow aggregation and analysis of data.

#### Subtask 2.4.3 – Deployment Preparation

Knowledge gained through the field trials will be used to develop a road map for describing key elements to the process of Public Health Systems implementation including user requirements, operational and technical requirements as well as how to address safety, regulatory and socio-economic requirements. In this way the project will promote the use of the REACTION platform to a wide range of stakeholders addressing the specific needs of diabetes management and therapy.

## 4.2 Validation Planning

The present validation framework describes a well-defined structure for the software testing and user validation already in the beginning of the project, which is agreed by all partners. The validation framework also includes definition of appropriate metrics and guidelines for usability testing, refinement of the initially defined success criteria, and measurement.

The overall validation activities in REACTION consist of three distinctly different elements:

1. Verification; to test if the software is free of bugs.
2. User validation; to evaluate if the services meet the expectations and requirements of its intended users.

3. Usability testing; to assess the quality of use of the applications.

The general approach for the project is described here, whereas the selection of specific methods for each verification and validation activity will be described in subsequent sections.

The *software verification* (debugging and testing) is a quality control (QC) process that is used to evaluate whether or not a system component complies with regulations, specifications, or conditions imposed at the start of a development phase. It is always performed at the laboratory level by the technical partner(s) responsible for the component.

Verification is the answer to the following question: Have we built the system right? (i.e., does it match the requirement specification?). Thus, verification is the process of evaluating a sub-system or system in order to check whether the products of a given development phase satisfy the conditions imposed at the start of that phase.

The *user validation* element is partly done at laboratory level, with internal technical partners analysing each software module and verifying its consistency alone and inside the overall architecture. Then the assessment of performance measurements is done with user partners and technical partners who have not contributed to the implementation, so that there is an evaluation of the (stable) components and prototypes from different point of views.

User validation is the answer to a different question: Have we built the right system? (i.e., is this what the end users need and want?). Thus, validation is the process of evaluating a sub-system or system at the end of the development process in order to establish whether it satisfies specified user needs.

The third element, *usability testing*, assesses the quality of use in the field trials made with end users, where controlled conditions are needed to assure that valid and interpretable results are obtained, useful as comparable benchmarks for customers.

### 4.2.1 Purpose of Software Verification

The software verifications performed at the technical partner premises have the main purpose of verifying the correctness of the implementation of the subsystems and systems respect to the requirements available at the beginning of each development phase. Verification is also an important condition for the MDD, and consequently for the field trials.

Moore's law holds that the number of transistors in an integrated circuit doubles every couple of years. Thanks to this trend, the amount of software embedded in vehicles, consumer electronics, and many other types of systems continues to grow (Murphy2008). In any system the software constituent represents a very relevant part and the specific techniques for software testing become more and more important.

According to the available and reference standards (IEEE2004-1) the verification process provides objective evidence whether the software and its associated products and processes

- Conform to requirements for all life cycle activities during each life cycle process.
- Satisfy standards, practices, and conventions during life cycle processes.
- Successfully complete each life cycle activity and satisfy all the criteria for initiating succeeding life cycle activities.

Analyzing the various different definition of testing provided by different standardization bodies we can argue that testing is a process both dynamic and static and can be done on any work product or product (Hass2008).Thus the test can be applied to products or work artefacts (e.g. subsystems, intermediate prototypes, etc.) that will not necessarily be released as product.

- Static testing is a form of software testing where the software isn't actually executed. It checks mainly for the correctness of the code, algorithm, or document. It is primarily syntax checking of the code and/or manually reviewing the code or document to find errors, using code inspections and walkthroughs. This type of testing can be used by the developer who wrote the code or by an expert independent testing team usually also during the development phase.
- Dynamic testing is the testing of the dynamic behaviour of the software. In dynamic testing the software must actually be compiled and run. Dynamic testing is used to test software through

executing it. Actually Dynamic Testing involves working with the software, giving input values and checking if the output is as expected.

An important part of the verification activities is the verification of adherence to mandatory standards, in particular the group of EU directives around medical devices. These directives are the Medical Devices Directive 93/42/EEC, the In-Vitro Diagnostic Medical Devices Directive 98/79/EC and The Active Implantable Medical Device Directive 90/385/EEC (in as far as the potential use of insulin pumps is concerned). Fields trials in the in-hospital environment can only be carried out (concerning interaction with patients) if the requirements of the MDD are met (documentation, testing, risk analysis).

The complete integrated platform will not be available in the verification phase and the use of simulators or ad-hoc population of the data repository will be necessary in order to allow a comprehensive verification of all the implemented workflows.

Both approaches have strengthens and weaknesses, so should be seen as complementary. But dynamic testing will remain the predominate approach. Theoretically, dynamic testing can be used to show the presence of defects, but never their absence! (Dijkstra1969) Therefore, the correctness of software should depend only on the code itself and should be demonstrated using formal verification methods. However, formal verification methods are very expensive or not effectively applicable in many cases in order to assure absolute “correctness”, thus dynamic testing remains the most cost effective approach to building confidence within most software systems.

### 4.2.2 Purpose of User Validation

The purpose of user validation is to assure that the results of the development project - i.e. the implemented result - is in agreement with the needs and requirements of users, and is accepted by these in the end..

According to the available and reference standards (IEEE2004-1) the validation process provides evidence whether the software and its associated products and processes

- Satisfy system requirements allocated to software at the end of each life cycle activity.
- Solve the right problem.
- Satisfy intended use and user needs.

User validation, including such topics as analysis of user needs, contextual inquiry, ethnographic analysis, usability testing, or user satisfaction measurement, is a mature approach now, based on scientific knowledge, and proven and tested methods.

However, there is only one method, which has produced consistently positive results in the development of successful new products and services; it is continuous collaboration with users in the analysis and evaluation of the technical concepts and results during the entire development project.

The history of new technologies and projects is densely populated with examples where the initial expectations for the technology were not met, and where, even when the technical development effort was successful, the final product was not accepted by users and the market. The opposite case - that new solutions were much more widely and rapidly accepted than expected - also exists, but is much less frequently encountered.

User-centred development has the main aim to avoid the disappointment of development projects which do not meet their large expectations, and attempts to achieve this in two ways: Firstly by making sure that all is done to make the product which is developed as close as possible to user needs, and secondly to provide sound and reliable information about the value and applicability of technology, also in comparison to competing solutions.

Independent of the type of product, service, or industry considered, there is only one reliable approach to assure that at the end of the development process the result is accepted by users: To involve users from the start of the development of technology and of applications in an effective manner. Effective means that valid - correct and relevant - information is collected, and that it is used to improve the solution under development. It is important to distinguish between two types of stakeholders, users and customers.

- Users are the individual healthcare workers and patients who in the course of their work or disease interact directly with the product or service, which is developed. The acceptance criterion of users is that they are able to carry out the intended tasks efficiently and successful, and without undue problems or stress, and that their subjective assessment of the innovation is positive.
- Customers are the healthcare provisioning organisation who makes the buying decision, and who must be persuaded of the value of a new service or technology. The acceptance criterion of customers is the total cost/benefit advantage obtained by the introduction of innovative technology. This includes non-monetary factors, such as evidence for clinical results as well as the consequences for human resource management and the positive acceptance by the personnel affected.

Validation activities will focus on impact on patients, their relatives, healthcare personal and other individual users as well as on organizational processes (e.g. in primary and secondary care as well as nursing care), with appropriate weight given to either aspect according to the phase of project progress. Hence, the validation will mostly centre on organisational workflows and stakeholder interaction as observed during the field trials. Traditional clinical research and validation of the clinical protocols is outside the scope of the project. However, the REACTION platform will be available for one full year after the end of the project thus allowing the clinical partners to carry out outcome studies in that time.

### 4.2.3 Purpose of Usability Testing

Usability testing assesses the quality of use in the field trials made with end users, where controlled conditions are needed to assure that valid and interpretable results are obtained, useful as comparable benchmarks for customers.

Usability will be tested in the two field trials in WP8 (in-hospital and outpatient) with a small number of users to detect user problems and deficiencies of the prototypes and to feed these back to the development teams. Each field trial reports the conclusions of the trials in order that common assessment criteria are adopted to allow aggregation and analysis of data.

The field trials can be seen as an advanced acceptance test with users. It provides the information for subsequent management decisions about the performance and features of the REACTION platform and allows for the clinical protocol for insulin dosing to be validated with patients in the in-hospital trial. Careful planning helps considerably to obtain interpretable and valid results at the end of the field trials. The test conditions, instructions of users, data analysis procedures and benchmarks for comparison have been defined.

The synergy between the two applications will be used as much as possible to create common methodologies, and by looking for complementary results. A set of methods is proposed for each of the two application scenarios and initial user validation plans are drawn up and will be updated as needed.

# 5. Organization of Internal Verification Activities

The most common approach in order to control and assure the required quality in any product life cycle is through the use of the verification and validation (V&V) process.

V&V main objectives are to prove that requirements are correct, complete, consistent, accurate and testable; to facilitate early detection and correction of software errors; to enhance management insight into process and product risk; and to support the software life cycle processes to ensure compliance with program performance, schedule, and cost requirements. V&V activities and tasks are interrelated and complementary.

Verification is a quality control (QC) process that is used to evaluate whether or not an artefact, product, service, or system complies with regulations, specifications, or conditions imposed at the start of a development phase. Verification can be performed at different phases and it is often an internal process (ISO2005). Validation is a quality assurance (QA) process of establishing evidence that provides a high degree of assurance that a product, service, or system accomplishes its intended requirements. This often involves acceptance of fitness for purpose with end users and other stakeholders.

The different impact of verification and validation in the system development life cycle (SDLC) (Kulyamin2008) is illustrated in Figure 1.

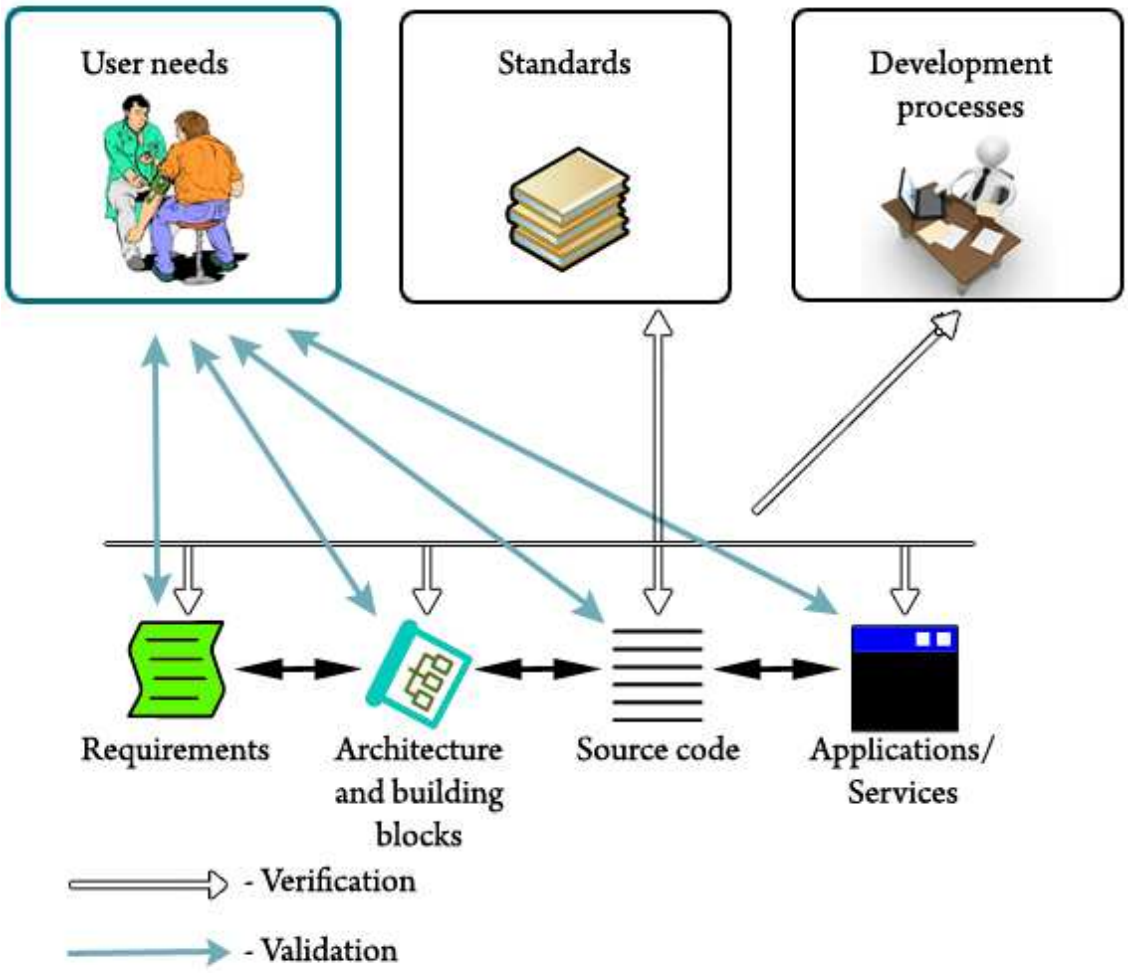


**Figure 1: Different impact of verification and validation in the system development life cycle.**

In the REACTION project, the internal verification procedures will be held at the technical partners' premises and will have the main purpose, not involving real end users and stakeholders, of performing the necessary tests in order to check whether the products of a given development phase satisfy the conditions imposed at the start of that phase or that the starting specifications have been correctly implemented. These procedures will be part of WP10.

The validation procedures will be performed at the clinical sites with the involvement of real end users and stakeholders after interfacing interoperable third-party systems.

In each iteration of the REACTION project, the lifecycle of the software runs along a path (see Figure 2) composed by the software requirement definition, the architectural design, the detailed design and the coding. Once the code is available the testing phases aim at verifying the correct behaviour in correspondence of each integration phase. The unit tests aim at verifying the modules/components identified and built in the detailed design, the integration tests assemblies the units in order to verify the architecture while the system tests aim at putting together the subsystems and obtaining the assembled system putting together all the tested subsystems obtained after the integration test (ESA1995).

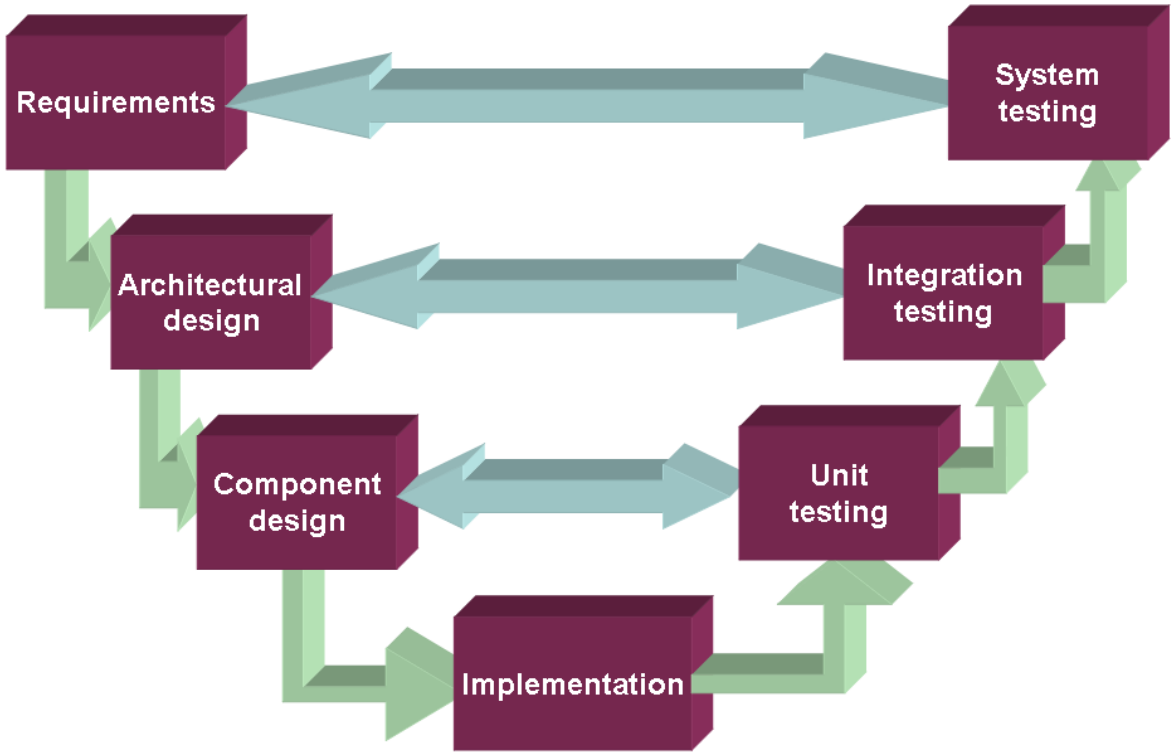


**Figure 2: The V model of software lifecycle.**

These verification activities demonstrate compliance to the requirements available at the beginning of each design phase. This may be done by showing that the product performs as specified and contains no defects that prevent it performing as specified.

In some cases the use of Computer Aided Software Engineering (CASE) tools may enhance the software verification process reducing the effort needed for mechanical tasks, increasing the amount of software verification work that can be done especially at the unit test level and improving the accuracy of software verification.

## 5.1 Purposes and Limitations of the Internal Verification

The internal tests performed at the technical partner premises have the main purpose of verifying the correctness of the implementation of the subsystems and systems respect to the requirements

available at the beginning of each development phase. The very likely lack of real end users in this phase hampers the performance of tests aiming at confirming that the product, as provided, fulfils its intended use like effective interoperability with third party systems, ergonomics, usability, performance and user satisfaction. That might be partly smoothed with the preparation of a detailed specification document. The main goal of such document will be to improve the requirement elicitation and engineering phase but cannot eliminate all problems e.g. related to correct reporting of requirements, completeness of requirements and correct interpretation of requirements.

Furthermore, it is necessary to consider the possibility that the complete integrated platform will not be available in the verification phase. In fact, some significant parts of the integrated system will not be available (an HIS from which patient data and other relevant information can be collected for the purpose of the REACTION platform) or properly usable (a sensor detecting measurements out of the range in order to check alerts and alarms) and thus the full integrated platform will be obtained only with the use of simulators or ad-hoc population of the data repository. These auxiliary tools will be necessary in order to allow a comprehensive verification of all the implemented workflows.

### 5.1.1 Use of Simulators and Support Tools

Simulators preferably implemented by software will be realized in order to allow the dynamic test of specific workflows. Such simulators will be mainly used in order to simulate the behaviour of sensor/patient in the initial phases of the project when the sensor is not available yet and there is the need of testing specific workflows where the data acquisition is part of the workflow itself. Simulators can also be used during the integration of components and subsystems in order to verify the designed interfaces even before the availability of the subsystem or component. That is also particularly useful when there is the need of testing a workflow which includes the interactions with subsystems or components not yet available at the time of the test.

In order to perform functional test in some cases some ad-hoc population of the data repository will be necessary and in such cases there will be some support tools (e.g. specific scripts) able to insert the required data in the data repository before the start of each specific test. In such case the support tool will be used not at the test running time (as done with simulators) but just before the test in order to reset the test environment and then to initialize it properly.

## 5.2 Software Test Environment

The testing of the software is a process which should be integrated as much as possible with the production of the software itself. As the software development follows an iterative cycle, testing procedures should follow closely the development in order to give precise feedback to the developers as soon as possible, and to help in identifying and repairing problems early in the design and implementation phase.

In order for the testing process to be efficient and for the results to be objective and repeatable, the testing should be based, as much as possible, in automated methods and tools, or at least to methods that minimise the human factor into the testing process. For this reason, various tools and frameworks have been developed to assist most of the software development phases, including requirement analysis, bug tracking, versioning control, unit testing, performance analysis, usability testing, etc.

In the REACTION project, such tools have already been selected and are being used, as it is the JIRA tool (Atlassian2011) for the requirements and issue tracking and the SVN tools (server side: VisualSVN2011, Subversion2011) (client side: TortoiseSVN2011) for versioning control.

Specifically for the in-hospital environment the following tools have been used: TestNG for unit testing on the back-end site and Android JUnit Testing (automated UI testing) at the front-end site. Also Maven 2 has been used in the test environment.

## 5.3 Structure of the Internal Tests

Tests will start with the test of the units (components) and once these tests will be passed then the tests will move to the next level of hierarchy where multiple components or subsystems will be connected in order to exchange information and realize a specific function or a workflow or part of it. Then, further components or subsystems will be added till the last level where the prototype with some support tools, as described in the previous clause, will be integrated in order to verify all the functionalities and workflows implemented by the applications.

The testing of the REACTION platform is an iterative process which involves multiple steps and targets at all layers of the platform. The testing procedure is performed in parallel with the development of the platform and also evolves in parallel with it. Its goal is to guarantee that the platform meets the requirements that have been specified by the end users, in a stable and seamless way, to test that the platform adheres to standards wherever this is required and that the system is able to cope in exceptional cases without crashes or unrecoverable problems. The following kinds of tests are expected to be performed in the REACTION platform:

- **Parameter testing to the Web Services (WS)**. This testing serves multiple purposes:
    - It can verify, as a proof-of-concept, that the parameter set for each WS is complete and can support the required functionality of the WS.
    - It can verify the status of the WS implementation.
    - It can serve as a monitoring tool, as the project evolves, to ensure that the WS are up-and-running and are responding to requests.
    - It can be used for stress-testing the WS components, to check possible performance issues.
- **Unit tests to the back-end components**. Using automated unit tests it can be verified through various test cases that the functionality of the system meets the user requirements. These unit tests involve also the low-level components of the system, e.g. databases, using unit tests or automated scripts, to verify the data model and to check that the data is maintained in a consistent status.
- **Integration tests**. Several high level tests will be performed, to verify that the various system components interact and integrate in a seamless way, and that there is a coherent and concrete semantic model.
- **System tests**. The final tests will be performed on the integration of all physical and/or logical subsystems in order to realize the envisaged prototypes of the REACTION platform. These tests will involve not only the integration of the internal components of the REACTION platform, but also the integration of the platform with external components or methodologies (ward, HIS, medical devices). The prototypes of the platform will be tested against the functionalities specified in the requirements.
- **Adherence to standards**. There is the requirement for some components to adhere to specific international scientific standards, e.g. for data exchange. In those cases, specific targeted tests will be performed, following the specified protocols or by using appropriate external tools for each case, to verify that the components adhere to the standards.

It is expected that until the first prototype of the platform is released, the testing procedure will focus mainly on design and implementation problems due to shortage of available time, so every developer or involved partner is expected to conduct testing on the components which he develops. After the first prototype, when comments from the users and conclusions from the usability tests are available, the testing procedures will be specified in more detail.

Specifically for the user interfaces, we expect that usability tests will be accurately performed during the validation phase by user groups, in order to provide feedback back to developers. Based on our experience, it is an expensive and challenging task to organise a review of the software by experts in software usability through techniques like usability inspection. So, our proposed approach is to organise usability evaluation sessions using Hallway testing (Wikipedia2011-1) and test with real end user at the clinical partners' premises. The usability tests should also consider (and not be limited to) the tests of attributes as menu format and access, screen colours, fonts and font colours and anomaly management (ASCEM2010).

In order to be able to compile a descriptive and objective usability evaluation, the criteria and the feedback should be, as much as possible, in a quantitative form of specific metrics, such as efficiency (measured in time) and accuracy (measured in number of errors or successes) to complete a task, scenario or workflow. It is suggested also that the subjective and qualitative criteria be measured in a user satisfaction scale and not in the form of free text commenting. The results of the evaluation sessions, as well as the overall comments from the testers regarding the REACTION platform, will be included in the validation report with also suggestions for the next versions of the platform.

### 5.3.1 Parameter Testing and Unit Tests

There are various approaches into software testing and a common categorisation is the distinction between “black box” testing and “white box” testing. In “black box” testing, we treat the tested component as a solid piece of software, without any knowledge or interest to its internal functioning. Based on the requirements specification of the component, we test if its behaviour adheres to its specification. In “black box” testing we are mainly interested into what the component does, but not how it does it. In “white box” testing, we examine also the internal functioning of the component, data structures, algorithms, side-effects etc. In such a case, we are interested not only to the compliance of the software to its specification (what it does), but also how it does it. Static testing better applies to the “white box” testing while dynamic testing is usually performed in the “black box” testing.

In the REACTION architecture, as in most Service-oriented Architectures (SoA), the separation of software into distinct, collaborating services and modules, directs us towards a testing procedure which favours the “black box” testing procedure. Based on the requirements set for each of the different services, we will set up a testing procedure which examines if the services deliver the functionality that they “promise” through their Application Programming Interface (API). If we want to proceed into a more fine-grained testing, we can further set up a refined testing procedure for each service respectively, which will test the correct functioning of the internal structure of the service. The latter testing could follow the “white box” paradigm, based not only on the API and the requirements of the service but also on the design of each service distinctively.

We are not aware of a tool or framework which focuses on unit testing of Web Services based directly on their Web Service Description Language (WSDL). Nevertheless, based on the WSDL description of each Web Service we can build clients into whatever environment or language we prefer in order to conduct our testing. This way, not only we achieve the same functionality but also we have the option to select whatever environment or programming language we prefer for the testing process (e.g. soapUI (free version) can be used for functional testing of web services or WS-I for black box testing of web services (see deliverable D10.2)).

The proposed unit testing process (in case of Java will be selected for the implementation of a specific component) might be:

- Test of the WSDL description for each web service, in order to check if the API corresponds to the functionality defined for that service.
- Building of Java (or any other preferred language or environment) clients, based on the WSDL description of each Web Service, by using the Axis tool (Apache2011).
- Each Java client will be used as a wrapper (structural pattern) for the delegate underlying web service, in order to set up the testing process. The Java client will be just forwarding the method calls to the respective functions of the underlying web service.
- For each service, a rapid mock-up implementation will be built, simulating the functioning of the service as specified in the REACTION architecture, in order to verify the integration and spot possible flaws on the completeness of the services.
- The JUnit (JUnit2011) framework or another framework from the xUnit (Wikipedia2011-2) family of tools or other tools like TestNG will be used in order to apply tests to those services and to verify the stability of the system. This testing will be gradually enhanced, as the implementation progresses, to test in more deep the correct functioning of the REACTION platform based on the test cases presented below. This will serve not only for testing and verification purposes, but also for monitoring the stability of the platform.

The tests to be performed will be drawn from the test cases outlined below. We stress out of course that unit testing can examine specific functional characteristics of the software and indicate possible problems, but there are complex issues on software that require more sophisticated analysis or different approach, such as security flaws, adherence to standards and integration issues. Consequently, the whole testing process will be separated into a step-by-step procedure, to ensure that the testing process is helping and not just burdening the software development, and certain complex functions will be checked manually and not automatically.

### 5.3.2 Integration Tests

Integration tests will be based on the seamless integration of the released and verified components or subsystems which will be available at each integration cycle. They are iterative and involve multiple steps and targets higher layers of the platform. The testing procedure may ask for some retrofit at the component or subsystem level and in such case generate some overlap and parallel evolvement with the development of the platform. Its goal is to guarantee that the units are properly integrated and interfaced and the relevant subsystems meet the requirements that have been specified at the beginning of the iteration cycle. It will mainly follow the bottom-up and function-based approaches. In these tests the interface among the module will be tested verifying there is no interface misuse or misunderstanding and the functionalities implemented by each subsystem are well performing.

Several high level tests will be performed in order to verify that the various physical or logical subsystems, then to verify the system components interact and integrate in a seamless way, and that there is a coherent and concrete semantic model.

#### Test of Physical or Logical Subsystems

The most relevant physical or logical subsystems will be accurately tested in order to verify if their behaviour is conforming to the requirements relevant to the specific subsystem and available at the beginning of the iteration cycle.

Among the subsystems we can identify the sensors and the security framework.

The test of physical and logical subsystems will therefore include these subsystems but not necessarily be limited to them.

#### Sensor Performance and Reliability

The sensors developed within REACTION are tested for performance and reliability prior to integration into the REACTION platform. The test plan and integration with the REACTION platform is described in deliverable D10.2.

The test methods applied in the performance testing for the sensors developed within REACTION are based on in-vitro tests, approved in-vivo tests, and stability tests.

- **In-vitro tests** are the first approach in testing the functionality of the sensors. The in-vitro tests are based on either classified samples, samples measured and compared with the golden standard, or simulated data.
- **In-vivo tests** represent the final step in testing a sensor. The in-vivo tests are designed based on biostatistics principles. The approach is to apply and test the sensor with healthy subjects before proceeding to patients.
- **Stability tests** are performed both with in-vitro and in-vivo test. The tests make sure the sensor function as specified under the conditions that it will be exposed to during use such as high and low temperatures, high humidity, etc.

These tests serve to ensure basic functionalities of the prototype sensor systems and documentation of their performance and reliability.

Following test and approval of the sensors' performance and reliability, the sensors are incorporated into the ePatch or other wearable wireless devices. These devices support primarily the Continua Alliance supported standards ISO/IEEE 11073 (IEEE2004-2) and can thus be integrated into the REACTION demonstrator platform for functional testing of the software components and data communication.

The use of the sensor devices in clinical studies requires compliance with the Medical Device Directive (MDD), the most relevant for REACTION is the Directive 93/42/EEC (MDD1993). It was reviewed and amended by the Directive 2007/47/EC (DIR2007).

Concerning the IMM glucose monitoring sensor, in the first approach it will be connected to medically approved micro-dialysis needles, so that the sensor can be applied early in the project within clinical studies based on intravascular micro-dialysis at MUG. For that a risk analysis will be performed, following the DIR2007, carefully documented. The same procedure will apply for the second approach, where the fibre based optical cell is going to be implemented at the front end of the micro-dialysis

catheter. Whether the final approach of the IMM sensor is going to be applied within clinical studies is questionable due to the requirements set forth in DIR2007.

The DELTA ePatch for ECG monitoring is at present in part compliant with the essential requirements of MDD1993 and the relevant parts of IEC 60601 (IEC2005) standard for medical electrical equipment and will become fully compliant with the essential requirements. This includes risk analysis and tests such as test for biocompatibility and toxicity. It will therefore be possible to use the DELTA ePatch in clinical studies.

SOLIANIS Continuous Glucose Monitor (CGM) is described in the deliverable D3.3. It consists of the Multisensor (MS) and Tool Suite (software) running on a laptop computer. The system is not certified according to any standards at the moment. However, the MS was already used within several experimental clinical and outpatient trials approved by Swissmedic and Swiss Ethic Committee (e.g. Caduff2009). The system can only be characterised in-vivo vs. glucose excursion.

In-vitro measurements can only be conducted on single sensors of the multisensor system (e.g. GHz spectroscopy, optical reflection, etc). However, the corresponding test procedures and internal measurement results will not be made available due to intellectual property issues.

SOLIANIS will perform release tests according to the internal test procedures for Software (Solianis2010) and MS hardware (Solianis2009) and certify conformance of the delivered system to the internal specifications and safety norms in the same way as for the experimental trials approved by Swissmedic.

SOLIANIS CGM system can be tested as a “black box” within the REACTION platform. Measurable specifications of this system will be summarised in deliverable D3.3.

**Security Framework**

The testing of the security framework will use a similar approach as outlined in Section 5.2, though probably at a different level. While the Web services will primarily deal with application data, i.e., the SOAP payload, the security framework will have to deal with security extensions from the SOAP header or even lower level protocols, e.g., SSL/TLS. Individual components from the security framework will be tested offline, with unit tests, as well as in concert using Web service clients.

The purpose of the security framework will be largely focused on providing access control, authenticity of identities and messages, integrity, and confidentiality. Thus, test cases will focus on testing whether ‘bad’ messages will be detected and properly dealt with. For this, a number of attack messages will be handcrafted to see how the system responds to such simulated attacks. As attacks are conceivable from the client and the server side, attack messages will have to be formed for either side.

Test cases will include, but are not limited to:

- Unauthorised access, e.g., a properly identified entity tries to access a resource in a way that is not permitted to her (violation in access control).
- Changed message content, e.g., the message was inadvertently or maliciously modified (violation in integrity).
- Invalid identities, e.g., an identity was presented that is not known to the target system (violation in authenticity).
- Spoofed user identities, e.g., an attacker attempts to take on the identity of a user known to the system (violation in access control / authenticity).
- Spoofed server identity, e.g., for carrying out a man-in-the-middle attack, in which the legitimate participants believe that they are interacting with each other, but are in fact interacting with the attacker (violation in access control / authenticity / confidentiality).

### 5.3.3 System Test

The system tests imply the final integration of all subsystems in order to set-up the prototype. Of course, the test of the prototypes will be performed after the test of the subsystems has been performed and it will be the last logical step of the internal verification procedures.

The system tests involve not only the integration of the internal components of the REACTION platform, but also the integration of the platform with external components or methodologies (ward, HIS, medical devices) whose behaviour can be simulated dynamically (using simulators) or statically (using support tools able to populate the data repository before running the test).

The prototypes of the platform will be tested against the functionalities specified in the requirements and related to the applicative environment of each prototype. Volume and stress testing, and configuration testing (the process of testing a system with each of the configurations of software and hardware that are supported) will be parts of the system testing.

### 5.3.4 Adherence to Standards

Rules relating to the safety and performance of medical devices are harmonised in the EU and consists of 3 directives:

- Directive 90/385/EEC (MDD1990) regarding active implantable medical devices;
- Directive 93/42/EEC (MDD1993) regarding medical devices;
- Directive 98/79/EC (MDD1998) regarding in vitro diagnostic medical devices.

These 3 main directives have been supplemented over time by several modifying and implementing directives. For REACTION the most relevant directive is the Directive 93/42/EEC (MDD1993). It was reviewed and amended by the Directive 2007/47/EC (DIR2007) and a number of changes were made (e.g. software for medical applications has become a medical device).

The Directive 2007/47/EC (DIR2007) defines a medical device as: *"any instrument, apparatus, appliance, software, material or other article, whether used alone or in combination, including the software intended by its manufacturer to be used specifically for diagnostic and/or therapeutic purposes and necessary for its proper application, intended by the manufacturer to be used for human beings. Devices are to be used for the purpose of:*

- *Diagnosis, prevention, monitoring, treatment or alleviation of disease.*
- *Diagnosis, monitoring, treatment, alleviation of or compensation for an injury or handicap.*
- *Investigation, replacement or modification of the anatomy or of a physiological process.*
- *Control of conception."*

The government of each Member State has been required to transpose the Medical Device Directive 2007/47/EC (DIR2007) into National Law by March 21, 2010 (Wikipedia2011-3).

This means that beside medical hardware also medical software is now classified as a medical product by the Medical Device Directive. A new regime is in force governing all medical device software development for all classes of device.

Previous software safety standards were best suited to medical devices with low levels of risk, as opposed to products where software failure could be extremely serious and result in death. As more electronic products have become dependent on embedded software, the focus has shifted to the reliability of software systems within the devices and the associated risks at all levels of usage. As a result, the new IEC 62304 (IEC2006) standard has emerged as a global benchmark for management of the software development lifecycle.

IEC 62304 (IEC2006) is a harmonised standard for software design in medical products adopted by the European Union and the United States. Because the standard is "harmonised," medical device manufacturers adopting it will satisfy the essential requirements contained in Medical Devices Directive 93/42/EEC (MDD1993) with amendment 2007/47/EC (DIR2007) as related to software development. This is the least onerous route to ensuring compliance with the MDD (MDD1993 & DIR2007).

Designing according to IEC 62304 (IEC2006) ensures that quality software is produced by means of a defined and controlled process of software development. This standard provides a framework of life cycle processes with activities and tasks necessary for the safe design and maintenance of medical device software. IEC 62304 (IEC2006) is a well considered, logical standard for developing safety critical and high reliability software for medical devices. Now that this standard has been adopted it would be very difficult for a medical device software developer to justify any equivalent approach that

meets the requirements of the MDD (MDD1993 & DIR2007), without effectively complying with this standard.

The REACTION in-hospital glucose management system, which will assist professionals (physicians and nurses) in the glucose management for patients at general wards in the hospital, must be considered as a medical device. Same considerations apply to the outpatient environment. At the current status of the project, the REACTION remote monitoring client and BAN&PAN subsystem should be considered medical devices. However, these issues related to the outpatient environment will be clarified in the next phases of the project. Therefore, the system, which consists of software as well as hardware, must fulfil the essential requirements set out in the Medical Device Directive (MDD1993 & DIR2007). In order to prove its compliance with the MDD (for the Ethics committee and the legal authorities) the software development process will be based on IEC 62304 (IEC2006).

The hardware development process (for the REACTION in hospital glucose management system but also for all medical devices being developed within REACTION) must also fulfil the essential requirements set out in the Medical Device Directive (MDD1993 & DIR2007). In order to prove its compliance adherence to standards is strongly recommended. Due to the fact that there are different developments it will also be mandatory to fulfil several different standards. The most prominent standards are IEC 60601-x (for all medical electrical devices) (IEC2005) and the ISO 10993-x (for all issues related to biocompatibility) (ISO2009).

#### Risk Analysis for Hardware and Software Design

Medical product designers have used risk management techniques to help reduce the risks associated with device hardware. ISO 14971 (ISO2007) has traditionally been adopted as the base standard for risk management for medical devices. The 2007 version of this standard is considerably extended from its previous version, and the techniques described are now intended to be applied to both software and hardware systems.

The approach that should be taken is to consider the risks posed by the medical device as a whole, before the software/hardware split has been decided. Hardware risk analysis can then run alongside software risk analysis to define the required safety systems for the device.

The compliance process for Medical Device Software and its relationship to standards are illustrated in Figure 3 (Hall2011).

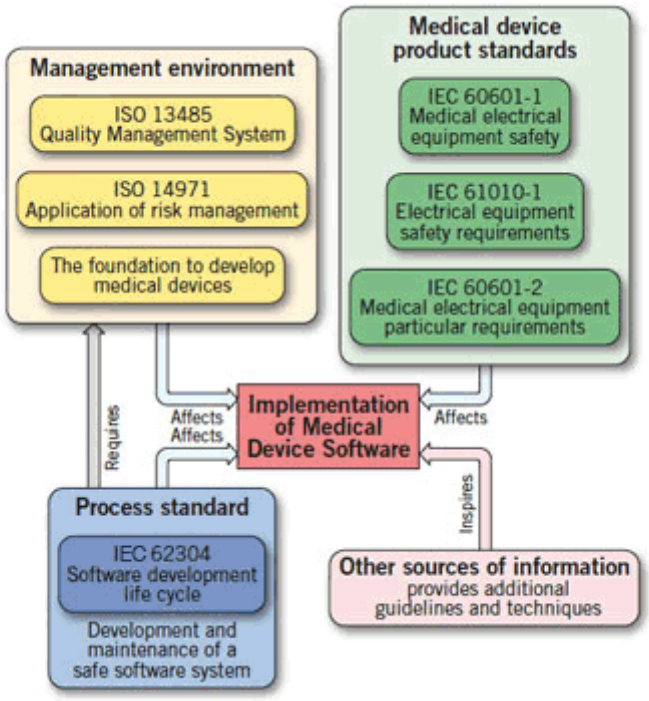


**Figure 3: Compliance process for Medical Device Software and its relationship to standards.**

**Risk Management Process**

The manufacturer shall establish, document and maintain throughout the life-cycle an ongoing process for identifying hazards associated with a medical device, estimating and evaluating the associated risks, controlling these risks, and monitoring the effectiveness of the controls. This process shall include the following elements (ISO2007):

- risk analysis;
- risk evaluation;
- risk control;
- production and post-production information.

The risk analysis process is started with a description of the intended use and characteristics related to the safety of the medical device. In the next step potential hazards are identified and its risks are estimated for hazardous situations.

Both components of a risk, probability and consequence, are analysed separately for the estimation of a hazard. For risk control there will be a stepwise approach to reduce risk:

1. inherent safety by design;
2. protective measures in the medical device itself or in the manufacturing process;
3. information for safety

This means that if practicable, the medical device should be designed to be inherently safe. If this is not practicable, then protective measures such as barriers or alarms are appropriate. The least preferred protective measure is a written warning or contra-indication. It is recognised that one possible results of the risk control option analysis could be that there is no practicable way of reducing the risk to acceptable levels. In this case, a risk/benefit analysis can be carried out to determine whether the benefit of the medical device outweighs the residual risk.

## 5.4 Test Cases

In order to make the testing procedure as focused and productive as possible, a test suite of specific test cases will be written, in accordance with the corresponding use cases or user requirements. In some cases, tests are generated automatically based on the services and methods modelled with the MDA CASE tools.

Exhaustive testing of almost any non-trivial system is impractical due to the fact that the range of input data values to most practical software systems is either extremely large or infinite. The logical approach is to design an optional test suite that is of reasonable size and can reveal as many errors existing in the system as possible. Actually, if test cases are selected randomly, many of these randomly selected test cases do not contribute to the significance of the test suite, and thus, the number of random test cases in a test suite is, in general, not an indication of the effectiveness of the testing.

The test cases can have impact on units, subsystems or the entire prototype depending on the specific requirement(s) they address.

In order to ensure that the user requirements are met, for each requirement there should be a detailed set of conditions which verify with certainty when a requirement has or has not been fulfilled (positive and negative test cases). A formal test case should include at least the following information:

- Preconditions. A set of input parameters and/or the state of the tested component before a test is conducted.
- Post conditions. The expected result or effect of the test, in order for the tested component to pass or fail the test.

At this phase of the project a full and exhaustive list of test case is unthinkable. A more reasonable approach is that each partner is expected to keep track of his own test cases, according the components he develops or assemblies. Below are some examples of what a test case could include, so that all developers keep the same structured format for their test cases.

| | |
|---|---|
| **Test Case ID** | TC-001 |
| **Version** | V1 |
| **Author** | |
| **Test Case Description** | User login |
| **Reference Environment** | In-hospital application |
| **Unit to Test** | User management |
| **Assumptions** | |
| **Test Input** | User credentials |
| **Expected Result** | Successful login or an appropriate error message |
| **Addressed Requirements** | Corresponds to user requirement … |
| **Comments** | |

| | |
|---|---|
| **Test Case ID** | TC-051 |
| **Version** | V1 |
| **Author** | |
| **Test Case Description** | Search for a patient |
| **Reference Environment** | In-hospital application |
| **Unit to Test** | Patient management |
| **Assumptions** | |
| **Test Input** | Part of patient's demographic data or patient's ID |
| **Expected result** | Listing of the patient's stored information |
| **Addressed Requirements** | |
| **Comments** | |

| | |
|---|---|
| **Test Case ID** | TC-100 |
| **Version** | V1 |
| **Author** | |
| **Test Case Description** | List the patients of a ward |
| **Reference Environment** | In-hospital application |
| **Unit to Test** | Patient management |
| **Assumptions** | |
| **Test Input** | The ward in question |

| | |
|---|---|
| **Expected Result** | The list of patients or an appropriate error message |
| **Addressed Requirements** | |
| **Comments** | |

| | |
|---|---|
| **Test Case ID** | TC-200 |
| **Version** | V1 |
| **Author** | |
| **Test Case Description** | List the enrolled patients in a ward |
| **Reference Environment** | In-hospital application |
| **Unit to Test** | Patient management |
| **Assumptions** | |
| **Test Input** | The patients in the ward in question and the active enrolments |
| **Expected Result** | The list of enrolled patients or an appropriate error message |
| **Addressed Requirements** | |
| **Comments** | |

| | |
|---|---|
| **Test Case ID** | TC-250 |
| **Version** | V1 |
| **Author** | |
| **Test Case Description** | List the open tasks |
| **Reference Environment** | In-hospital application |
| **Unit to Test** | Task management |
| **Assumptions** | |
| **Test Input** | All tasks either performed or open |
| **Expected Result** | The list of the open tasks or an appropriate error message |
| **Addressed Requirements** | |
| **Comments** | |

As the project evolves and several technical details will be decided and documented, the test case can be further analysed and may also contain specific technical information:

| | |
|---|---|
| **Test Case ID** | TC-001 |
| **Version** | V2 |
| **Author** | FORTH |

| | |
|---|---|
| **Test Case Description** | User login |
| **Reference Environment** | In-hospital application |
| **Unit to Test** | User management |
| **Assumptions** | |
| **Test Input** | User credentials. The credentials should be two alphanumeric strings, the username and the password of the user. The username cannot contain spaces or special characters. The password should … and may … |
| **Expected Result** | Successful login or an appropriate error message.<br>If the user provides correct credentials, the system will keep log of the user login and show the screen …<br>If the user does not exist or provides incorrect credentials, the system will show the message … and will keep showing the login screen. |
| **Addressed Requirements** | Corresponds to user requirement … |
| **Comments** | |

It is expected that it will be impractical to keep track of all the test cases if documented inside this document. Therefore, the full list of applied test cases will be kept and updated in the internal test report.

## 5.5 Internal Test Report

The internal test report should contain a description of the tests performed in the internal test sites mainly focused on assuring that the subsystems or system match the specification. The tests at unit level will typically be performed at the site of the technical partner who completed the unit implementation. Each unit has to be delivered for the next test phase with its own test report which will be included in the internal test report. The main goal is to check whether the subsystems or system of a given development phase satisfy the conditions imposed at the start of that phase. Specific problems, inconsistencies or bugs at any level should be reported in order to be properly addressed in the next release. In some cases also retrofits to the requirements imposed at the beginning of the phase can be proposed.

More specifically the internal test report should contain a traceability analysis describing the inter-relations between the source code components and the requirements. The identified relationships should be analyzed for correctness, consistency, and completeness. Any observed anomalies have to be reported including also suggestions for their solution.

Another section of the internal test report should contain an evaluation report of the source code components (source code and source code documentation) for correctness, consistency, completeness, accuracy, readability, and testability. Any observed anomalies have to be reported including also suggestions for their solution.

The interfaces between the software source code and hardware, users, operators, and other third-party systems must be analyzed to verify correctness, consistency, completeness, accuracy, and testability. Anomalies have to be properly reported.

The internal test report should contain the full list of the tests performed on units, subsystems and prototypes, the description of all test cases and procedures including which parts they were applied to and the results of the tests. Version of each unit has to be reported.

All test procedures at unit level, subsystem level and prototype level have to be described and test cases have to be catalogued per unit, subsystem and prototype including also the requirement(s) they address.

Finally also hazard analysis, security analysis and risk analysis have to be reported, each one with any observed anomalies and suggestions for their solutions. In the hazard analysis it has to be verified that the implementation correctly implements the critical requirements and introduces no new hazards. In the security analysis it has to be verified that the implementation is completed in accordance with the system design, that it addresses the identified security risks and does not introduce new security risks through coding flaws. The verification is done against the requirements, thus using test cases addressing specifically the critical requirements and the security requirements. Furthermore, security has to be verified also at subsystem level. In the risk analysis any observed or anticipated technical risks have to be identified and recommendations provided in order to eliminate, reduce or mitigate the risks.

# 6. Organisation of User Validation Activities

The purpose of user validation is to assure that the implemented result is in agreement with the needs and requirements of users. The user validation activities will focus on impact on patients, their relatives, healthcare personal and other individual users as well as on organizational processes (e.g. in primary and secondary care as well as nursing care), with appropriate weight given to either aspect according to the phase of project progress. Hence, the validation will mostly centre on organisational workflows and stakeholder interaction as observed during the field trials. Traditional clinical research and validation of the clinical protocols is outside the scope of the project.

User validation is the answer to the question: Have we built the right system? (i.e., is this what the end users need and want?). Thus, validation is the process of evaluating a sub-system or system at the end of the development process in order to establish whether it satisfies specified user needs.

Then the assessment of performance measurements is done with user partners and technical partners who have not contributed to the implementation, so that there is an evaluation of the (stable) components and prototypes from different point of views. These activities will be performed yearly in the context of WP2.

## 6.1 Approach to User Validation

The user validation process consists of three steps:

1. Planning the validation at the end of each iterative cycle.
2. Carrying out validation activities according to this plan before and after the prototype demonstrator is available.
3. Making decisions on the basis of the validation results (e.g. redesign, error correction, start of implementation, release) as part of the requirements re-engineering work.

The present validation framework guides the collection of information about the project specific objectives, requirements and constraints that may limit to a certain extent the choice of appropriate methods for user validation (different methods measure different quality dimensions).

## 6.2 The Validation Process

The process followed is similar in all validation cycles and foresees fixed steps to follow: an initial preparation part, an internal verification activity and/or a validation activity with (expert) end users, the collection and analysis of the outcomes and feedback of the results into the loop for the next step in the process.

### 6.2.1 Plan and Prepare the Validation Activities

The first part of this step is to define and briefly describe the subject of the validation, including platform and components to be validated. Also the specific requirements (in the Volere template) to be validated are identified.

A clear and precise *User Validation Plan* has several advantages:

- It is possible to compare different ways of performing user validation activities, as they will occur later in the project. We can evaluate and compare these different ways easily, until the most effective and efficient approach is found.
- Formal schedules help us to identify critical factors (e.g. time, cost, personnel skills and qualifications), which will need a concentration of effort and the commitment of project partners and users.
- The plan is a powerful persuader in engendering a commitment amongst the development team by demonstrating visually that the plan has been well reflected and discussed. The plan makes clear that and how state-of-the-art user validation is achievable.

Ensure that the person who will be leading the evaluation has sufficient skills and experience in the methods used. If necessary, bring in some outside expertise.

- Sound method and conduct are essential. It is all too easy to get misleading results from small or unrepresentative samples, or from evaluator bias, or by over-generalising from single instances.
- Liaise with the design & development and technical teams over timing and status of what is to be evaluated.
- The readiness of designs & prototypes for testing is critical, so be realistic about timing.
- Make sure you understand the technical constraints of the product, and the time and budget constraints for making changes. Discover what changes are most easily possible and what is more difficult.
- If users are to be involved, make arrangements sufficiently in advance, and try to keep to those arrangements.
- It is not easy to find suitable users who have the time available to participate.
- There are PR aspects for the project when involving outside people in testing.

The validation templates, to be prepared before the evaluation activities take place, identify the actors, i.e. the test persons. It is useful also to draft the corresponding user scenarios, use cases or test cases that the actors need to go through as part of the validation. This allows customisation of the validation procedure, selecting from already existing methods that are considered appropriate.

### 6.2.2 Analysis of User Needs, Requirements and Preferences

The requirements and needs of groups of users are analysed by detailed studies of the demonstrator application context, based on the involvement of expert users from the application domain. In addition to the factors, which are part of a systematic requirements analysis in traditional terms, security needs and realistic business models are investigated.

The JIRA tool and the Volere template is an excellent tool for managing the user requirements. An appropriate validation form using the same basis could look as the example shown below in Table1.

| ID | Req. No.: R-217 | Test case No.: | | Use case No.: | | Scenario Ref.: | |
|---|---|---|---|---|---|---|---|
| Type | Functional -Outpatient pilot application | Priority | Major | | | | |
| Summary | Acquired values in the alarm range | | | | | | |
| Rationale | When the acquired values are in the alarm range, an alarm has to be sent to the clinicians in charge (call centre). If the alarm is confirmed by them, then either the patient has to be sent to the hospital in case of serious episode or the treatment and the RPM schema have to be adequately changed. | | | | | | |
| Fit criterion | Check the overall procedure in case of acquired measurements in the alarm range. | | | | | | |
| | 1st cycle | 2nd cycle | | 3rd cycle | | 4th cycle | |
| Result | N.A. | Partly supported | | Supported | | | |
| Comment | Second year prototype | Tested in lab | | | | | |

**Table 1: Example of Validation Form.**

The fields ‘Type’, ‘Priority’, ‘Summary’, ‘Rationale’ and ‘Fit criterion’ are the same as in the Volere template. The ‘ID’ fields allow references to Requirements, Test cases, Use cases and Scenarios.

In case the fit criterion has to be measured with means of a laboratory test, the validation template must clearly indicate information such as testing method, statistical processes to be applied, the

number of trials necessary for a trustworthy the result, boundary conditions and any other data necessary to conduct a reliable experiment.

In case the fit criterion refers to a quality dimension which is not related to a numerical measurement (e.g. user satisfaction or user acceptance) a questionnaire is to be prepared which is aimed at measuring the specific aspect. The questionnaire should investigate provided functionality, added value and other related topics. This type of questionnaire will add information for the depiction a quality space through performance, productivity and added value dimensions from a user point-of-view.

The needs of users are not fixed in the sense that precise requirements, constraints, and preferences are maintained under all conditions, but there is a certain amount of "elasticity" such that one attribute may be traded for another attribute.

Users who have some experience with a service are quite capable to answer questions, which allow the analysis of the user preferences in terms of tradeoffs. A meaningful (quantitative) analysis demands that a substantial amount of data is available, and is outside the scope of the project. But simple interview and rating techniques may allow the collection of data, which give an indication of the tradeoffs which users consider when selecting services or products for use and purchase.

The result would allow estimates of the value of adding specific quality features to the services, and would indicate which main quality features users would like to see integrated into application packages.

### 6.2.3 Conduct Validation Activities

Once the validation template and questionnaire are completed, the test person(s), assisted by the working group who prepared the evaluation activities, has to follow the instructions and perform the validation, which can be a laboratory test or a trial of the application before answering the questions.

Involve members of the design & development team and other stakeholders in observing the evaluation activities. This is the single most effective way of promoting feedback into design, and getting people to act on results.

In the initial development phase user interfaces guidelines for the design of information presentation and navigation structures, and samples which illustrate these, are highly effective. When used in a more rigorous form, style guides must be complemented by a review process, which tests for the adherence to the guidelines. This is usually done by expert review - but not to be carried out by the developers themselves.

Design reviews during the early development phase should be carried out by the project’s experts, which are not involved in the development effort. They use checklists and test the system according to the defined use cases, assuming the role of a user. They report the results directly to the developers, and possibly involve the developers directly in the design review.

Effective inspection methods such as heuristic evaluation and cognitive walkthrough, where experts systematically follow scenarios and use cases to test a system, can be done with paper prototypes and specifications. The number of defects found is initially quite large. Experience shows that the more immature an implementation is, the faster will defects be found. Different expert evaluators do not find the same defects, and not in the same order. It is therefore advisable to use two or three experts. In later development stages longer test sessions should be foreseen.

Users who are presented with incomplete and defective software become frustrated and can not provide much constructive feedback. Users should only be involved in tests as soon as the development team is confident of the quality of the result of development. User tests should be well planned and correspond to minimum methodological constraints. Experts should help with the planning of test sessions. In order to demonstrate shortcomings of the application and problems of users and to convince developers, it can be useful to videotape relevant episodes of test sessions for later review and presentation to the developers.

There must be an understanding that the tests are carried out in order to identify as many problems as possible, and to find a better solution immediately. The number of subjects to be used for testing can be small initially, but really conclusive tests require in the range of 8 users as a minimum.

### 6.2.4 Analyse Data

After completing the test trials and questionnaire submission results must be analysed and evaluated. Obviously data analysis will be performed with different approaches for the laboratory measurements and the questionnaire responses. While the first will hopefully result in immediate numbers, the examination of questionnaires will be made with both quantitative (statistical calculations on multiple choice questions) and qualitative analysis (comments and observations emerging from open questions).

Be impartial when analysing the data, aim for speed with sufficient rigour, and focus on the things that matter for the product's success. 'Too-late' results are no use to the project. Same-day analysis is the norm in much commercial evaluation work.

### 6.2.5 Feed Back Results into the Loop

In line with the iterative approach the validation results will contribute to the success of the project because all the user feedback will be shared with the system developers.

The data emerging from the previous analysis will be distributed to the consortium partners, and they will serve in refining the user requirements and improving the system characteristics. This is the case also in the last iteration, where the assessment result provides the basis for a list of further recommendations.

Discuss the findings with the development team. Share with them the implications for how the product (and project) will achieve its quality goals. Do this as early as possible - discuss interim findings and work-in-progress - don't wait until you deliver the formal report! The biggest value of user validation is how it improves design.

### 6.2.6 User Validation Report

The user validation report should contain a description of the experience with the use of the platform at the clinical site, report the results of the usability test, the clinical workflow validation and the performance evaluation. Specific problems, inconsistencies or bugs at any level should be reported in order to be properly addressed in the next release and also new functionalities addressing specific user needs not yet included in the current requirement specifications should be clearly listed.

Finally the user satisfaction should be quantitatively evaluated and reported and specific suggestions should be retrofit to the technical team.

Frame the report so that it is meaningful to all relevant stakeholder groups. Remember that readers will have varying levels of technical, business and ergonomic understanding. Acknowledge all contributions to the work.

A sample structure for a user validation report could be:

| **User validation report** |
|---|
| Executive summary |
| Description of the development project |
| The “service”<br>Objectives, requirements and constraints of the development project<br>Phase in which the application was validated and development status<br>Objectives of the user validation<br>Critical success factors<br>Constraints for user validation |
| The quality strategy |
| Validation questions agreed with the users of the validation results<br>The validation scenario<br>Focus of the assessment<br>Quality dimensions and assessment criteria |

| |
|---|
| Users, tasks, and context of use |
| Description of user groups<br>Description of the tasks users intend to perform with the application<br>Description of the context of use |
| Methods for user validation |
| The user validation plan<br>Description of the validation procedure |
| Analysis of the validation results |
| Recommendations and conclusions |

### 6.3 Time Line for the Validation Process

Time lines should be defined for each iteration cycle, correlated with the anticipated progress and deadlines for the development work. The length of the four different steps may of course differ from one cycle to the next and from one project to another. Figure 4 shows a possible time line, exemplified for the first iteration cycle:

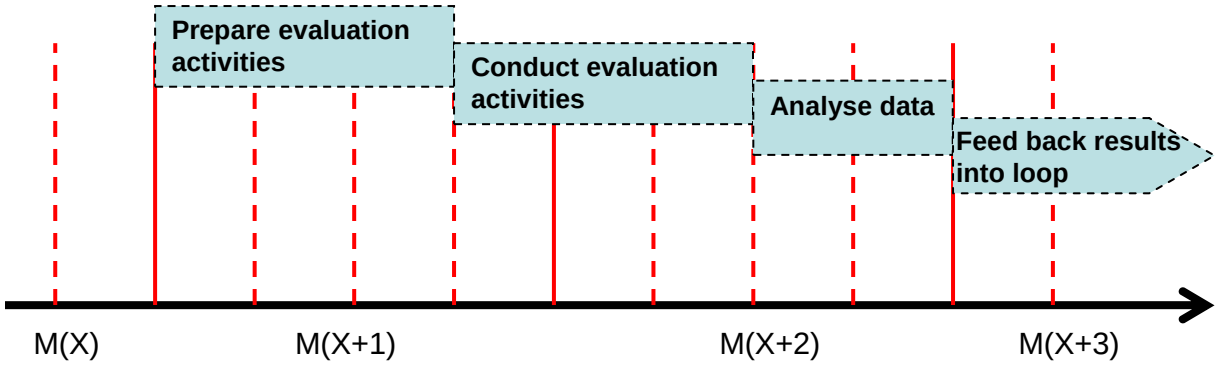


**Figure 4: Time line – First iteration.**

# 7. Organisation of Field Trial Usability Testing Activities

The purpose of the Field Trial Usability Testing is to perform usability tests of prototypes used in the field trials. *Usability testing* is thus the assessment of the quality of use of the REACTION applications.

The overall aim of the field trials is to assess the effectiveness of the REACTION platform (i) within a hospital environment, (ii) with outpatients under therapeutic control and (iii) for patients who are self-managing their disease. The goal is to conclusively prove the validity of the applications, demonstrate the benefit for healthcare providers and provisioning authorities, gain acceptance by patients and other users and to assess the impact at the organizational level.

In order to evaluate the overall user experience in using the REACTION platform, usability tests will be conducted once or twice in each envisaged environment, in order to collect feedback from the users of the platform, and give retrofit to the developers for redesigning, if necessary, user interfaces or functionality of the REACTION platform. These testing activities will be part of WP8.

The field trials provide the information for subsequent management decisions re the performance and features of the REACTION platform. Careful planning helps considerably to obtain interpretable and valid results at the end of the field trials. The test conditions, instructions of users, data analysis procedures and benchmarks for comparison have to be defined.

Usability will be tested in two field trials (in-hospital and outpatient) in WP8 with a small number of users to detect user problems and deficiencies of the prototypes and to feed these back to the development teams. Based on our experience, it is an expensive and challenging task to organise a review of the software by experts in software usability through techniques like usability inspection. So, our approach is to organise at the clinical partners' premises usability evaluation sessions using

- Randomly selected users (Hallway testing)
- Real users, as doctors and nurses

The users will be given a small introductory training regarding the system and its functionalities and then each user of the evaluation group will be asked to perform specific sequence of tasks based on a predefined scenario, and he/she will be asked to complete a questionnaire with observations. The evaluation will be based on specific usability metrics, in order to have objective and quantitative data for analysis of the usability test.

Each field trial reports the conclusions of the trials in order that common assessment criteria are adopted to allow aggregation and analysis of data. The results of the evaluation sessions, as well as the overall comments from the testers regarding the REACTION platform, will be included in the validation report.

The synergy between the two applications (in-hospital and outpatient) will be used as much as possible to create common methodologies, and by looking for complementary results. A set of methods is proposed for each of the two application scenarios and initial user validation plans are drawn up and will be updated as needed.

## 7.1 Establish Quality Criteria Which the Users Will Apply

Quality will decide the success of the REACTION platform in the market. Quality is a combination of all features and properties of the REACTION services, which determine their attractiveness and value for the users and customers.

However, quality means different things to different people. The purpose and the tasks for which the REACTION platform will be used play an important role in defining the preferences for the service features and properties, and subjective factors define the preferences of users for style and aesthetics of the user interface. Quality of use is an issue from the viewpoint of users; healthcare authorities are more concerned about the total cost of ownership of a service.

User requirements analysis should result in a list of features and properties of the REACTION services including quality criteria, which are considered relevant by the users and customers. User usability validation carried out in the project's field trials will test if these quality requirements are fulfilled.

Quality does not come for free. A value must be attached to the cost and benefit of quality-oriented actions. We have to determine which of the features and properties, requested by users and customers, to implement, how this is done, and what the optimal investment is.

Quality dimensions which users typically apply to assess the value of ICT applications are quite well known, but of course not the specific quantitative values of these in a particular context. These can be summarized as:

1. performance (effectiveness - the ability to actually carry out tasks successfully, and efficiency - the cost in terms of time and other factors for carrying out the task);
2. subjective assessment (affect) of the quality of an application;
3. learning effort required using a system;
4. cognitive workload;
5. added value.

## 7.2 In-hospital Usability Testing Procedures

In the general ward, a REACTION application will monitor blood glucoses. Mathematical algorithms will be used to calculate the required insulin doses.

### 7.2.1 Parameters of the In-Hospital Trial

#### Clinical Objectives and Rationale for the Trial

The objective of the in-hospital trial is to validate - in an inpatient environment - a suite of multi-parametric monitoring services designed to facilitate the close monitoring of diabetic patients by remote dedicated diabetes experts and so enable more widespread use of Safe Glucose Control.

#### Critical Success Factors

The following success factors have been established for the in-patient trial:

- The fulfilment of legal requirements for security, safety and privacy are mandatory. Conformance with standards. The service must be in line with major government policies. Government initiated programs and strict requirements for documenting improvements are in force.
- The regulatory framework for safety and security for the healthcare sector must be taken into account. Suitability, trust, privacy and security are essential requirements for all types of users.
- The service must be 100 % reliable. Erroneous user actions must not lead to critical situations.
- All actors rely on remote communication via tablet PC's, mobile phones and PDA's.
- The process of iterative design and evaluation cycles should help the technical WPs to generate and test new ideas.

At the end of the project the summary evaluation of total quality for users and the cost / benefit of the entire application should prove the added value of REACTION platform.

#### Constraints for the Validation

Meaningful intermediate release dates for feedback to developers remain to be defined. The development partners interact closely in the validation and receive feedback immediately. There are immediate opportunities to modify the design and implementation by the development team. Monitoring of progress should take account of the need to have sufficient time and resources remaining to make meaningful adaptations to the application.

Available personnel and resources for validation: The main testing will be carried out by members of the development team in cooperation with doctors and nurse practitioners from the user partner, who have limited experience in usability engineering, with some help from the usability partner.

Motivated users are expected to be available flexibly from the user partner MUG. Tests are carried out by way of integration into the normal work procedures and environment.

#### Who Needs and Who Will Use the Validation Results?

Designers and developers of the REACTION platform for in-hospital use need feedback from usability experts and users about the quality of use of prototypes in order to refine the prototypes before the final application is implemented. Feedback is needed about what works well and is considered advantageous. Concrete recommendations of how to improve the prototypes should be provided as a list of items starting with the most critical problems, which "must be improved", and ending with not too critical issues, which are "nice to have", would be very helpful.

Validation results are also needed for dissemination and exploitation activities.

The project manager needs validation results to assess the progress made in the REACTION project and to report to the European Commission.

### 7.2.2 Stakeholders in the In-Hospital Trial

The indented initial application is to support glucose management in the hospital of the University of Graz at two wards namely Endocrinology and in a further later step Cardiology. The system will assist glucose management for patients for whom it was decided by the physicians to perform glucose management.

Users of the prototype will be the medical staff (physicians and nurses) and technicians to maintain the system. Therefore the users can be considered as professional users. At the beginning a dedicated study team, which is well trained to glucose management and the new system, will test the first prototype.

To facilitate the implementation and to improve the acceptance of the platform at the general ward, the clinical staff will need a training programme. Glucose management training is important, because the clinical staff must have high competence and skills to guarantee high quality and safe patient care by using the electronic decision support system. The mentioned training will comprehend basic knowledge about diabetes mellitus, hypoglycaemia, hyperglycaemia, nutrition, measurement, target ranges, medication and case studies. The use of the algorithm as well as the eDSS system will be trained. Additional, experts will support the clinical staff in the first period using the prototype.

### 7.2.3 In-Hospital Metrics for User Satisfaction

Relevant quality dimensions ordered according to their importance are:

- Security and privacy
- Added value
- Cost / benefit of the service
- Learning effort
- User acceptance

The following Table 2 indicates assessment criteria for the above quality dimensions (example):

| Quality dimension | Measure | Unit of Measurement | Critical Value | Required Value | Optimal Value | Methods |
|---|---|---|---|---|---|---|
| Security and Privacy | Rating by users and experts | To be determined | Below average | Above average | Highest values | Questionnaire, Positioning, Conjoint Measurements |
| Added Value | Rating by users | To be determined | Below average | Better than the average | Above average | Questionnaire, Positioning, Conjoint Measurements |
| Accuracy | Error/success by user in operating the system | Number of errors and number of successes in using the platform | >2 per patient session | <=1 per patient session | 0 | Counting |
| Efficiency | Time to complete a task or scenario | Seconds | >30s | <=15s | 10s | Measure time from start to completion |
| Acceptance of users | User satisfaction | Affect Helpfulness Learnability Efficiency Control | Below average | Above average | Highest values | SUMI standardised user satisfaction questionnaire |
| Learning effort | Time to learn | Minutes | To be determined | | | Learning time measurements |

**Table 2: Assessment criteria for in-hospital scenario.**

### 7.2.4 Safety, Usability and Performance Test for the In-Hospital Prototypes

The goal of the tests is to assess safety, usability and performance of the system. After finishing the safety test, usability and performance tests will be investigated. Performance and usability testing can be performed in parallel. In order to meet safety, usability and performance of the system, the clinical staff will be trained using an educational program as mentioned above.

The aim of the safety investigations is to:

- identify potential risks (e.g. wrong dosing of insulin, wrong measurement, …),
- quantify the probability of occurrence (e.g. daily, weekly, monthly, …) and
- assess their impact (e.g. hypoglycaemia, hyperglycaemia, …).

These safety investigations will be part of a *risk analysis* and *risk management process* which will be performed according to the ISO 14971:2007 standard (ISO2007).

After finishing the safety tests, a usability study (including interviews in clinical environment) will be performed.

The aim of the usability study is to qualitatively and quantitatively collect data regarding:

- Functionality:
    - Does the system offer all necessary functionalities?
    - Are the functionalities well implemented?
    - Do the functionalities support the usual workflow?
    - What additional functionalities should be implemented?
- Layout:
    - Is the information clearly arranged?
    - Are the buttons well named?
    - Are the diagrams well named?
    - Do the diagrams show what they should?

- Are the input forms clearly perspicuous?

The following parameters have been defined, which are essential for the system quality and usability and will be tested through the first prototype:

| | |
|---|---|
| Time: | Time is precious in clinical environment; therefore operating with the application must not slow down the workflow. |
| Clarity: | The application has to clearly present important data to the user. |
| Workflow: | The application has to support the usual workflow of physicians and nurses. |
| Facilitation: | The application should make things easier, not more complicated. |
| Functionality: | All functionalities of the application must be robust and well implemented. |

There are lots of usability techniques, which can be performed. Possible methods are heuristic evaluation, followed by Thinking Aloud tests. The testing of usability will be based on the standard IEC 62366 (IEC2007) related to the application of usability engineering to medical devices.

After finishing the usability tests, performance trials in clinical environment will be performed.

The following parameters have been defined, which are essential for the system quality and performance and should be tested through the first prototype:

- Glycaemic control (e.g. reachability of the target range, time within/outside a certain target range, progress of glycaemia during a patient's stay, …)
- Risk of hypoglycaemia (e.g. number of hypoglycaemic events, time in hypoglycaemia, …)
- Risk of hyperglycaemia (e.g. number of hyperglycaemic events, time in hyperglycaemia, …)

Safety, usability and performance investigations will be performed in different phases during the REACTION project:

**Phase 1**: 50 inpatients with established diabetes mellitus or with newly diagnosed elevated blood glucose will be analysed to obtain information about the actual number of glucose measurements, the mode of diabetes treatment, the use of insulin, the overall quality of diabetes control etc. These data represent the status of glycaemic management at the beginning of the REACTION project and can be used as a baseline for the assessment of the progress of the project.

**Phase 2**: A clinical training program of general in-hospital glycaemic control and the clinical protocol (algorithm) which will be used in the in-hospital Glucose Management System will be performed for medical and nursing staff.

**Phase 3**: The clinical protocol (modified RABBIT 2 algorithm) will be tested in the clinical environment for safety, usability and performance.

**Phase 4**: The electronic decision support system of the in-hospital Glucose Management System which is based on the modified RABBIT 2 algorithm will be tested in clinical trails for safety, usability and performance.

**Phase 5**: Devices or algorithms, which potentially will support the in-hospital Glucose Management System and which are developed during the REACTION project, will be tested during the development phase at the clinical research centre.

**Phase 6**: Tested devices and algorithms will be integrated into the in-hospital Glucose Management System and will be investigated in a clinical environment.

### 7.2.5 Usability Testing Report for the In-Hospital Environment

The end user usability testing report will describe the experiences with the use of the platform at the clinical site. It will also include the performance evaluation. Specific problems, inconsistencies or errors at any level will be reported in order to be properly addressed in the next release. New functionalities addressing specific user needs not yet included in the current requirement specifications will be clearly listed. The results of the decision support (protocol for insulin dosing) tests will be part of safety validation.

The user satisfaction will be quantitatively evaluated and reported and specific suggestions will be reported to the technical team.

Field trials will be finally used to evaluate the effectiveness and efficacy of the platform. The outcomes of field trials including the clinical workflow validation will report the process and outcome quality of the prototype.

## 7.3 Outpatient Usability Testing Procedures

In the outpatient trial, the REACTION platform will support medication compliance, adherence to clinical pathways, education, and self management health services for diabetes related conditions. Furthermore, clinical intervention for patients will be targeted to those with need; those that are well controlled will have less need for routine check up, and those above guidance levels will receive pro-active timely intervention.

Careful monitoring of multiple parameters may represent a useful integrated basis for achievement of strict and sustained glucose control that will provide a better opportunity to reduce diabetic complications and improve patients' quality of life. Devices for glucose and physical activity monitoring will be used to determine whether multi-parametric monitoring provide a reliable measurements as compared to classical mono-parametric monitoring. The trial will adopt a multi-dimensional approach, including: impact of education to improve compliance and modify lifestyle; enhancing primary care management with monitoring; and improving risk assessment to determine those likely to develop disease and complications.

### 7.3.1 Parameters of the Outpatient Trial

#### Clinical Objectives and Rationale for the Trial

The objectives of the outpatient trials are to specify and validate a suite of services aiming at simultaneous monitoring of blood glucose, blood pressure and physical activity to achieve comprehensive protection against diabetic complications and promote pre-active disease management. A small clinical development program will be initiated and performed in two main sections: Data monitoring, validation and interactive algorithm identification and clinical assessment of the capacity of multi-parametric monitoring.

#### Critical Success Factors

The following success factors have been established for the in-patient trial:

- Impact of education to improve compliance and modify lifestyle through monitoring of multiple parameters.
- The fulfilment of legal requirements for security, safety and privacy are mandatory. Conformance with standards. The service must be in line with major government policies. Government initiated programs and strict requirements for documenting improvements are in force.
- The regulatory framework for safety and security for the healthcare sector must be taken into account. Suitability, trust, privacy and security are essential requirements for all users.
- Application should be intuitive and easy to use (easy to learn), leading to high acceptance in initial phase of use of each user. The service must be 100 % reliable. Erroneous user actions must not lead to critical situations.
- Enhancing primary care management.
- Patients use specific medical devices (e.g. for measuring blood pressure), which can upload measurements to an online database.
- The process of iterative design and evaluation cycles should help the technical WPs to generate and test new ideas.

At the end of the project the summary evaluation of total quality for users and the cost / benefit of the entire application should prove the added value of REACTION platform.

#### Constraints for the Validation

Meaningful intermediate release dates for feedback to developers remain to be defined. The development partners interact closely in the validation and receive feedback immediately. There are immediate opportunities to modify the design and implementation by the development team. Monitoring of progress should take account of the need to have sufficient time and resources remaining to make meaningful adaptations to the application. Available personnel and resources for validation: The main testing will be carried out by members of the development team in cooperation with doctors and nurse practitioners from the user partner, who have limited experience in usability engineering, with some help from the usability partner.

Motivated users are expected to be available flexibly from the user partner CHC and by offering incentives to patients where needed. Tests are carried out by way of integration into the normal work procedures and environment.

#### Who Needs and Who Will Use the Validation Results?

Designers and developers of the REACTION platform for in-hospital use need feedback from usability experts and users about the quality of use of prototypes in order to refine the prototypes before the final application is implemented. Feedback is needed about what works well and is considered advantageous. Concrete recommendations of how to improve the prototypes should be provided as a list of items starting with the most critical problems, which “must be improved”, and ending with not too critical issues, which are “nice to have”, would be very helpful. Validation results are also needed for dissemination and exploitation activities. The project manager needs validation results to assess the progress made in the REACTION project and to report to the European Commission.

### 7.3.2 Stakeholders in the Outpatient Trial

In Table 3 below there is a list of the main stakeholders within the REACTION Platform. The list names the stakeholders and how they will be seen to interact with the platform.

| Stakeholder | Description |
|---|---|
| Patient | The person who will have the monitoring equipment in their home, take measurements, input data into monitoring system and receive data |
| Carer / Family | The person nominated by the patient who may also view patient data / assist patient in taking measurements |
| Clinician - Nurse | First line clinician responsible for patient - will liaise with Clinician – General Practitioner (GP) |
| Clinician - GP | Second line clinician responsible for patient - will liaise with Clinician – Nurse |
| Installer | Will be responsible for the installation, de-installation of equipment and patient training - could be a Clinician – Nurse |
| Admin Support | Will be responsible for providing support / answering calls from patients and support Clinician - Nurse, Installer and Clinician - GP |
| Community Team | A person responsible for providing community support to the patient |
| Other Health Provider | Hospital / Accident and Emergency department (A&E) / Social Services / Specialist |

**Table 3: List of the main outpatient stakeholders.**

### 7.3.3 Outpatient Metrics for User Satisfaction

A number of components of usability were developed so that they could be tested were developed by Bennett (Bennet1984). These include:

1. Learnability – the time and effort required to reach a specified level of use performance (ease of learning).
2. Throughput – the tasks accomplished by experienced users, the speed of task execution and the errors made (ease of use).
3. Flexibility – the extent to which the system can accommodate changes to the tasks and environments beyond those first specified.
4. Attitude – the positive attitude engendered in users by the system.

A one day workshop will be conducted at Chorleywood Health Centre (CHC) to understand the requirements and satisfaction measures for the clinical user interfaces and for the patient interfaces and feedback mechanisms. These requirements are being collected from both the clinical teams and the patients themselves. The patients will be a mixture of those who participated in the initial scenario thinking workshops as well as from those who have previous experience of using the Telehealth devices.

This workshop will take the form of a brainstorming activity where the users can describe clearly how they would to interact with the system and how the system should interact with them.

We then propose to follow two different methodologies in order to develop and validate user satisfaction with the REACTION platform. The main reason for this is the ability to get access to more frequent feedback from the clinicians and other stakeholders at CHC as this can be captured during the routine operation of the platform. It is anticipated that it would be harder to arrange for patients to repeatedly volunteer to take part in focus groups or answer questionnaire, etc.

#### Patient / Carer User Satisfaction

We envisage it will be harder to capture continual patient feedback from the patient and so we propose to capture feedback on two separate occasions during the project lifecycle.

#### Clinical User Feedback

To obtain feedback and validate the REACTION platform from the clinical user we propose to use the spiral methodology. This will enable us to make updates and improvements to the system in more incremental steps. The spiral model (Boehm1988) uses the main processes of the more traditional waterfall method, requirements gathering, analysis, design and implementation, but all introduces the notion of an incremental process (see Figure 6). Designs and prototypes would be generated for the clinicians to use, validate and feedback on. An example of such a process is as follows:

1. The new system requirements are defined in as much detail as possible. This usually involves interviewing a number of users representing all the external or internal users and other aspects of the existing system.
2. A preliminary design is created for the new system.
3. A first prototype of the new system is constructed from the preliminary design. This is usually a scaled-down system, and represents an approximation of the characteristics of the final product.
4. A second prototype is evolved using four steps:
    a. Evaluate the first prototype and identify its strengths, weaknesses, and risks.
    b. Define the requirements of the second prototype.
    c. Plan and design the second prototype.
    d. Construct and test the second prototype.
5. At the project sponsor's option, the entire project can be aborted if the risk is deemed too great. Risk factors might involve development cost overruns, operating-cost miscalculation, or any other factor that could result in a less-than-satisfactory final product.

6. The existing prototype is evaluated in the same manner as was the previous prototype, and, if necessary, another prototype is developed from it according to the fourfold procedure outlined above.
7. The preceding steps are iterated until the customer is satisfied that the refined prototype represents the final product desired.
8. The final system is constructed, based on the refined prototype.

We imagine that such a process would take place possibly 6 times as the REACTION platform grows to include different devices and intelligent feedback systems.

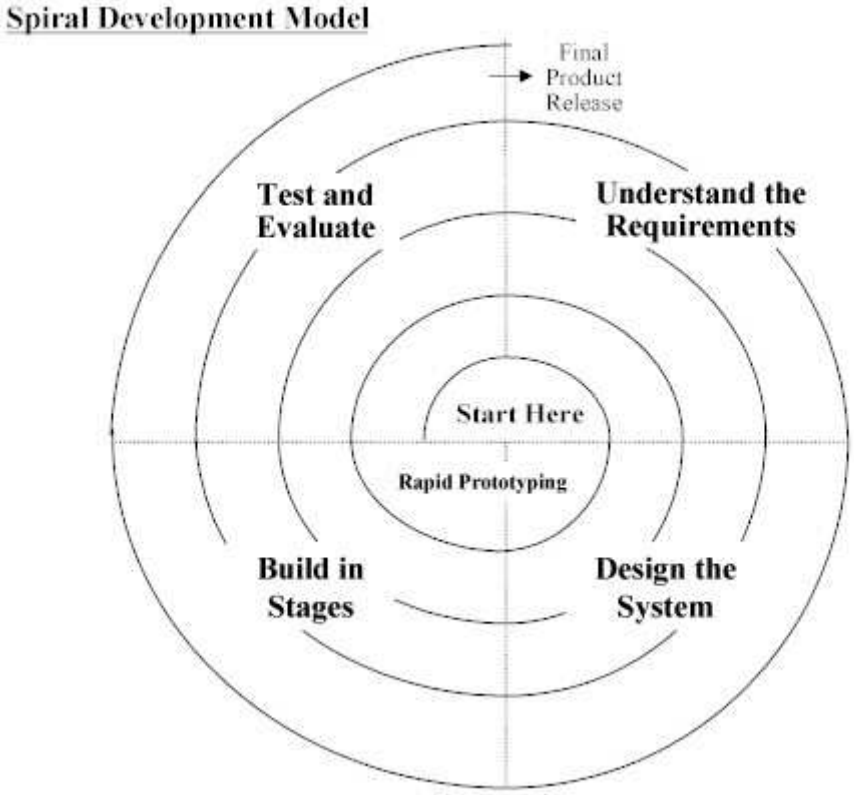


**Figure 5: Spiral Development Model (SDM2011).**

While we as yet are not able to state the exact measurement criteria we will use to measure user satisfaction the list may include:

| # | **Measurement Criteria** |
|---|---|
| 1 | Time to complete task |
| 2 | Percentage of task completed |
| 3 | Percentage of task completed per unit time (speed metric) |
| 4 | Ratio of success to failures |
| 5 | Time spent on errors |
| 6 | Percentage number of errors |
| 7 | Number of commands used |
| 8 | Frequency of help or documentation use |
| 9 | Time spent using documentation |
| 10 | Percentage of favourable / unfavourable comments |
| 11 | Number of repetitions of failed commands |

| | |
|---|---|
| 12 | Number of times the interface misleads the user |
| 13 | Number of good and bad features recalled by the user |
| 14 | Number of available commands not invoked |
| 15 | Number of regressive behaviours |
| 16 | Numbers of times users need to work around a problem |
| 17 | Number of times the user is disrupted from a work task |
| 18 | Number of times the user loses control of the system |
| 19 | Number of times the user expresses frustration or satisfaction |

### 6.2.2 Safety, Usability and Performance Test for the Outpatient Prototypes

The following provides an example of Safety, Usability and Performance criteria that should be tested on the Outpatient Prototype. These have been split into Device / Hub Measures and Clinical User Interface Measures.

**Device / Hub Measures**

| Measure | Description | Success |
|---|---|---|
| Clinical Monitoring Devices for patients home<br>Devices will be wireless<br>Devices are CE Marked<br>Devices should be battery powered | The devices will be able to measure the following:<br>• Pulse Rate - BPM<br>• Oxygen Saturation level - %<br>• Blood Pressure – mmHg<br>• Blood Sugar – mmol/L | |
| **Monitoring Hub**<br>Hub/s are CE Marked | The devices should wirelessly link to a central monitoring hub. This hub should:<br>• Provide an interface that is large enough for an elderly person to view the screen clearly.<br>o Large display<br>o Font size<br>o Icons<br>o Easily readable<br>• Clear instructions to the patient<br>• Provide feedback that communication has been successful<br>• Use a unique identifier for each hub to ensure a secure audit trail<br>• Re-programming of the hub can be undertaken remotely<br>• If interrupted during transmission, the hub will store the data until it is able to send data<br>• Be able to display simple list of questions for the chosen disease categories.<br>• Be able to display health monitoring feedback to the | |

| Measure | Description | Success |
|---|---|---|
| | patient<br>• Be able to provide a means for manual input of measurement data / responses to questions<br>• Provide audible feedback for people with visual impairments<br>• Multi-user support for residential / nursing care settings | |
| Communications | The device should transmit data via telephone, the internet or mobile communications. | |
| Power | • The hub should operate on both mains power and battery power<br>• In the event of a power outage the hub should automatically reset without the need for user intervention | |

**Clinical User Interface Test**

| Requirement | Description | Success |
|---|---|---|
| Secure Web Based Access for clinicians / Admin staff | Ability to access the monitoring system via the internet. | |
| Secure Log in / Access levels | There should be different levels of role based access. | |
| Prioritized Alerting Screen based on pre-defined rules | There should be a single alerting screen for monitoring data. Data should be sorted by:<br>• Alert Status<br>o Above limits<br>o Missing Data<br>o Technical / communications problem<br>o No data received<br>o Date | |
| Tabular Trend Data | Patients data should be able to be viewed in a tabular trend chart and be sorted by:<br>• Date<br>• Measurement<br>• Time period e.g. 7, 14, 21, 28, all days | |
| Graphical Trend Data | Patients data should be able to viewed in a graphical chart with the ability to be sorted by:<br>• Date<br>• Measurement<br>• Time period e.g. 7, 14, 21, 28, all days<br>• Show more than one measurement on the chart<br>• Show trend line on chart<br>• Show target / alert parameters on chart | |

| Requirement | Description | Success |
|---|---|---|
| Individual Patient Data screen | Each patient should have an individual record on the monitoring software that provides an overview of their:<br>• Demographic Details<br>• Devices assigned to them<br>• Disease categories<br>• Alert limits and question sets assigned to them<br>• Monitoring data | |
| Ability to set up user defined patient questions | Create a library of disease specific questions that utilize branch logic which can be selected and customized based on individual patients. | |
| Ability to store / edit Patient Demographic Data | The system should hold patient demographic data. | |
| Ability to set / edit and remove alert parameters around patient data | • Upper – Lower limits<br>• Change in Weight over 1 or 3 days<br>• End target | |
| Intelligent Algorithm to respond to data | To utilize developed / developing intelligent algorithms to model data. | |
| Ability to set up to 4 monitoring sessions per day | Set reminders for patients to take their readings at pre-determined time-frame. | |
| Ability to add / edit / patient monitoring data | • Manually enter in Clinical Monitoring data<br>• Mark existing data as invalid | |
| Ability to assign equipment to individual patients | • Assign equipment to patient by device serial number<br>• Enter in the location of devices e.g. which room the sensors are in | |
| Ability to view a list of equipment and who it is assigned to | Complete list of monitoring equipment which can be sorted by:<br>• Device Type<br>• Serial Number<br>• Assigned<br>• Unassigned<br>• Faulty | |
| Ability to search a list of equipment and who it is assigned to | Sort by:<br>• Patient name<br>• Device Serial Number | |
| Ability to remotely edit monitoring data | Remotely edit monitoring data:<br>• Devices<br>• Questions<br>• Monitoring session times | |

| Requirement | Description | Success |
|---|---|---|
| Display a list of all patients on the system | A screen which shows a list of all Patients who are enrolled onto the monitoring system. To be searched and sorted by:<br>• Name<br>• Condition<br>• Active<br>• Inactive (patients that do not have equipment in their home at the moment) | |
| Ability for the clinical team to add responses to incoming monitoring data | • No action taken<br>• Close watch<br>• Call Patient<br>• Escalate to GP<br>• Emergency Response Required<br>• Technical Issue<br>• Other: free text option | |
| Ability to review monitoring protocols dependent | A link should be available for to display the monitoring protocol dependent on the patient data:<br>• No data received<br>• Within Limits<br>• Above Limits<br>• Technical Issue | |
| Provide a database of resources that the clinician can select | • Online resources<br>• Document Library<br>• People Directory | |
| Ability to Video Conference | Video conference directly from the People Directory | |
| Provide a complete audit trail | Maintain a complete audit trail of all actions taken on the system and by whom. No data should be deleted permanently:<br>• Action taken<br>• Who took the action<br>• Time of action | |
| Ability to Export Data manually: | • Word<br>• Excel<br>• PDF<br>• Print<br>• Print screen<br>• Save<br>• Email | |

| Requirement | Description | Success |
|---|---|---|
| Ability to automatically export data: | • Excel file<br>• Electronic Patient record<br>  o Most recent monitoring data<br>  o Chart<br>  o Outcome of monitoring | |
| Ability to link to the Monitoring system directly from within the EPR | Select an icon from within the EPR to open the patients monitoring data | |
| Reports | The system should be able to provide user defined reports including:<br>• All Patient List<br>  o Summary<br>  o Alert Patterns<br>  o Compliance<br>• Individual patient report<br>  o Summary<br>  o Alert limits<br>  o Monitoring data<br>  • Graphical<br>  • Tabular<br>  o Question responses<br>  o Patient compliance<br>  o Alert history<br>  o Equipment<br>  o Free text notes<br>• Disease group<br>  o Names<br>  o Monitoring data<br>  o Question responses<br>  o Patient compliance<br>  o Alert history<br>  o Equipment<br>  o Free text notes<br>• Equipment Lists<br>• Audit trail lists | |

### 6.2.3 End User Usability Testing Report for the Outpatient Environment

The end user usability testing report will contain a description of the experience with the use of the platform at the Chorleywood Health Centre, including the performance evaluation. Specific problems, inconsistencies or bugs at any level should be reported in order to be properly addressed in the next release and also new functionalities addressing specific user needs not yet included in the current requirement specifications should be clearly listed.

Finally the user satisfaction data will be quantitatively evaluated and reported and feedback into the development lifecycles as described above.

A mixture of Interviews, Questionnaires and Focus Groups will be used to elicit user feedback about the system. In addition, data collection systems will be in place to capture usability and clinical workflow validation.

Table 4, 5, 6 provide an overview of the proposed evaluation protocol for the project with the aim of capturing data to identify if the project outcomes have been met. Each table provides a description of what data which will be captured by Chorleywood Health Centre, the purpose of the data and how it will be collected.

In addition, Chorleywood Health Centre has a responsibility to feed a summary of this information back to each participant. This will include a review of the clinical data captured as well as an overview of the outcomes of the project as a whole.

#### Clinical Monitoring Data

Table 4 describes the types of data we would like to collect via the Monitoring system. As well as data being stored on the monitoring system, data will be exported into the patients EPR. Data will be used to evaluate clinical management, participant compliance, Alert Rules and Technical issues.

| Evaluation Matrix | | | | | | |
|---|---|---|---|---|---|---|
| **Indicator(s)** | **Methods** | **Data Source/ collection method** | **Measure** | **Outcome** | **Timeframe** | **Resp. Party** |
| Participant Physiological Data | System | Patient / System | Clinical Management | Effectiveness of Clinical Management | Daily | CHC |
| Participant Habits Data | System | Patient / System | Habits Monitoring Data | Effectiveness of Social Care | Continuous / Daily | CHC |
| Participant Question Data | System | Patient / System | Clinical Management | Effectiveness of Clinical Management | | CHC |
| No of Alerts | System | Patient / System | Above Physiological Parameters | Effectiveness of Clinical Management | Daily | CHC |
| Missing Data | System | Patient / System | No Physiological data received | Patient Compliance | Daily | CHC |
| Partial Missing Data | System | Patient / System | Partial Physiological data received | Patient Compliance | Daily | CHC |

**Table 4: Data collected via the monitoring system.**

#### Clinician / CHC Staff

Table 5 describes the data we will collect from the Clinical / Admin team at CHC in order to evaluate their satisfaction and perception of using the system, and the impact on work load. Frequency and types of support requirements will be recorded formally to provide an indication of satisfaction. Outcomes of alerts will be used to monitor impact on clinical workload.

| Evaluation Matrix | | | | | | |
|---|---|---|---|---|---|---|
| **Indicator(s)** | **Methods** | **Data Source/ collection method** | **Measure** | **Outcome** | **Timeframe** | **Resp. Party** |
| Outcome of Alerts | CHC Clinical Team | System | No Contact<br>Patient Contact<br>Emergency Referral<br>Referred to GP<br>Other | Effectiveness of Solution / Workload | Daily | CHC |
| Clinician reported Web User Interface Issues | CHC Clinical Team | Manual | No of Support Calls Required | Effectiveness of Solution / Workload | Daily | CHC |
| Clinician Reported Training Issues | CHC Clinical Team | Manual | No of support calls required | Effectiveness of Solution / Workload | Daily | CHC |
| Clinician reported Device Issues | CHC Clinical Team / CHC Admin Team | System / Manual | No of support calls required | Effectiveness of Solution / Workload | Daily | CHC |
| Clinician perception of system Web User interface | CHC Clinical Team | Manual - Questionnaire | Satisfaction | Satisfaction | Pre-pilot<br>1 month after start of project / end of project | CHC |
| Clinician perception of devices | CHC Clinical Team | Manual - Questionnaire | Satisfaction | Satisfaction | Pre-pilot<br>1 month after start of project / end of project | CHC |
| Clinician perception of “whole” system | CHC Clinical Team/ CHC Admin Team | Manual - Questionnaire | Satisfaction | Satisfaction | Pre-pilot<br>1 month after start of project / end of project | CHC |

**Table 5: Data collected from the Clinical / Admin team at CHC.**

**Patient**

Table 6 describes the data we will collect from the patients. Comments and user observations will be recorded ad hoc during installation, de-installation and during any clinical / admin contacts. Frequency and types of support requirements will be recorded formally to provide an indication of satisfaction.

| Evaluation Matrix | | | | | | |
|---|---|---|---|---|---|---|
| **Indicator(s)** | **Methods** | **Data Source/ collection method** | **Measure** | **Outcome** | **Timeframe** | **Resp. Party** |
| Patient Reported Device Fault | Phone Call / In Person / Other | Manual | No of Support Calls | Patient Satisfaction | Daily | CHC |
| Patient perception of devices | Phone Call / In Person / Other | Ad Hoc / Questionnaire / Focus Group | Questionnaire | Patient Satisfaction | Daily / End of Demo | CHC |
| Patient perception "service model" | Phone Call / In Person / Other | Ad Hoc / Questionnaire / Focus Group | Questionnaire | Patient Satisfaction | Daily / End of Demo | CHC |
| Patient perception of privacy / security | Phone Call / In Person / Other | Ad Hoc / Questionnaire / Focus Group | Questionnaire | Trust | Daily / End of Demo | CHC |

**Table 6: Data collected from the patients.**

## 7.4 Closed-Loop Control System Usability Testing Procedures

The closed loop control system consisting of a mechanistic model kernel and a control algorithm is a core component of Automatic Glycaemic Control. It translates glucose measurements together with other information such as environmental conditions, food uptake and activity status into optimal insulin delivery rates.

To overcome known hurdles like lag-times, intra- and inter-individual patient variability as well as shortcomings of state-of-the-art closed-loop concepts, several known control algorithms will be combined with mechanistic physiologically-based models of glucose-uptake, insulin-delivery and glucose-insulin. Thereby, an initial evaluation of control strategies indicated the difficulty of fully addressing the issue without a suitable model kernel at hand. Consequently the focus during the first year was placed on model kernel development and a more detailed evaluation of control algorithms will enter the focus during the second year.

In accordance with the work plan, the validation of the model based closed loop control system for clinical application is structured as follows:

**Part 1 (off-line model validation)**

- For a first internal validation of the predictive performance of the developed model kernels, BTS will use data from literature that has not been used for the development of the model.
- The validation will initially focus on type 1 diabetes, where there is more data available, and will be extended to type 2 diabetes upon availability of suitable data.
- The literature-data based model validation is followed by an additional retrospective validation on available clinical data from MUG for type 1 diabetics and additional data generated within REACTION during the technical feasibility study for type 2 diabetics.

- One of the main objectives of the internal validation procedure is to test the extrapolation capabilities of the model for individual patients using independent datasets.
    - Validate chosen parameter sets used for model individualization
    - Assess the predictive quality of the developed model; the prediction error (residuals) will be assessed using a number of methods (R2, auto- and cross correlations, e.g. (Continuous) Error Grid Analysis).

**Part 2 (off-line controller validation)**

- A number of control algorithms (as described in Subtask 3.3.2) will be implemented on retrospective data/experiments and tested/evaluated with respect to their performance taking the following criteria into consideration:
    - Robustness/Safety
        - Number and severity of hypoglycaemic events
        - Handling of unknown disturbances/events (e.g. uninformed food intake)
    - Optimality/Best Performance
        - Reachability of the target range
        - Time within/outside a certain target range
        - Time needed to reach the target range after glucose intake in a safe manner
    - Usability/Simplicity
        - Computational power needed for the algorithm

**Part 3 (1st round of on-line (quasi-) closed loop validation)**

In the course of the project it will be clarified to what extend the glucose control system is defined as a medical device, as it would then have to be validated following the MDD (MDD1993 & DIR2007).

After successful internal integration of the glucose control system (control algorithm + mechanistic model kernel) and a successful validation in a simulated environment using real patient data, the control system will be tested with inpatients in a clinical environment using the same set of safety, performance and usability criteria as in Part 2.

Depending on the results of the internal validation a number of closed loop setups will be chosen for on-line validation. BTS and MUG will jointly perform the implementation, testing and refinement of the chosen closed loop systems during new clinical experiments. The test cases will include at least patients with type 1 diabetes, if feasible both, type 1 and type 2 diabetes.

- This first round of experimentation and algorithm evaluation will use invasive glucose measurement systems established at MUG and closed-loop will either be implemented in a prototype system or as a quasi-closed-loop with a physician being the interface between glucose measurement, algorithm and insulin delivery system. MUG has already demonstrated the power of this quasi-closed-loop approach in the CLINICIP project (CLINICIP2011). The closed-loop approaches will be benchmarked against each other and the potential for further optimization of the algorithm and the mechanistic model kernel will be evaluated.
- Test cases should include/consider e.g. changes in health status, calorific intake and treatment (medication).
- When available, a reference implementation will be made with continuous sensor systems developed within REACTION. Otherwise the controller performance will be validated using “artificial” continuous blood glucose sensor data generated from data measured invasively to maximize the applicability of the results from this initial testing of the algorithm in humans. For this, the expected (lower) accuracy of the minimal invasive REACTION continuous sensor system will be simulated by adding a virtual error (noise and drift) to the measured data. The resulting insulin delivery rates calculated from the simulated REACTION continuous sensor signal are then compared to the original insulin delivery rates calculated from the invasive sensor signals.

**Part 4 (2nd round of on-line (quasi-) closed loop validation)**

In year 4 of REACTION the closed-loop control system will be validated in a second iteration using a setup similar to the first round of closed-loop validation described in Part 3. However, if feasible, the invasive measurement systems established at MUG will be replaced by the newly developed minimal invasive patch system (using the continuous blood glucose measuring sensor developed within the REACTION project (Task 3.3.1)

Again, BTS and MUG will jointly perform the implementation, testing and refinement of the chosen closed loop systems within new clinical experiments with the support from the sensor developers (Delta, IMM and Solianis). The test cases will again include at least patients with type 1 diabetes, if feasible both, type 1 and type 2.

### 7.4.1 Stakeholders

Different stakeholders will be involved in different evaluations. More specifically:

- the potential clinical value:
    - Clinical personnel (MUG)
- impact on clinical workflows:
    - Clinical personnel as well as technical personnel (MUG and MSG)
- validation of feedback/control performance:
    - Physicians and (algorithm) developers (MUG and BTS)
- interoperability and scalability:
    - Physicians, technical personnel and developers (MUG, MSG and BTS)

### 7.4.2 Metrics for User Satisfaction

The metrics for user satisfaction will be based on the following minimal set:

- Patient individualization capabilities (see Part 1)
- Predictive error (see Part 1)
- Average glycaemic control (see Part 2)
- Risk of hypoglycaemia (see Part 2)

Eventual additional metrics to be considered are:

- Concomitant diseases
- Overall inpatient mortality and morbidity
- Workload for medical staff
- Acceptance of patients and staff
- Costs/effectiveness

### 7.4.3 Safety, Usability and Performance Test for the Controller Prototypes

All model-based control algorithms will be first validated using different sources of off-line data before entering two iterations of on-line testing and model/controller refinement, see above.

### 7.4.4 End User Usability Testing Report for the Control Algorithm

The end user usability testing report should contain a description of the experience with the use of the algorithm at the clinical site, including the performance evaluation. Specific problems, inconsistencies or bugs at any level should be reported in order to be properly addressed in the next release and also new functionalities or patient scenarios addressing specific test cases not yet included in the current test scenario should be clearly listed. Finally the user satisfaction should be quantitatively evaluated and reported.

# 8. Organisation of Deployment Preparation Activities

The objective of these activities is to provide an overview of the potential validity of the clinical applications, validate the economic benefit for the healthcare domain and assess the impact at the organisational level arising from the deployment of the REACTION platform in real-life healthcare systems. The aim is to provide the best possible framework for the successful deployment of REACTION applications in future diabetes management and therapy.

## 8.1 Business Benefits

Assessment of business benefit is the attempt to quantify the benefit for the customer in relationship to the total cost of ownership. The results should provide basic parameters to explain the value of the introduction of the REACTION service platform. This is a standard approach in market research and product strategic planning. It will be applied in a selective manner to the most critical aspects of REACTION technology and the two applications. The results should be among the main arguments presented to users and customers for the uptake of the REACTION technology.

This work will be undertaken as part of WP9 Socio-economic Framework which deals with the ethical, social, legal, regulatory, and economic aspects of the REACTION platform. Special emphasis will be placed on how to share proprietary information across organisational barriers, involve and transform the patient from a passive health information provider to an active information user, and safe handling of the massive flow of information and intellectual property rights to healthcare information.

New models of business constellations will be explored including private public partnerships, collaboration pharmaceutical companies as innovation drivers and bringing together payers, providers and patients in new constellations. An ontological perspective on the exploration of service concepts and for quantifying value creation will be adopted. The chosen approach is based on the analysis of economic value creation, distribution (Thestrup2008).

An important objective of the field trials with real users is to gather information about the quality and the competitiveness of the solutions, which have been developed. This information is needed by customers, investors, and by decision makers who are involved in the implementation of new solutions. The data to be provided should inform about:

- Efficiency and effectiveness of the work procedures.
- Quality (not clinical evidence) of the clinical results obtained by using the REACTION service platform and its applications.
- Motivating effect on users, both patients and healthcare professionals.
- Direct and indirect value for the owner of the applications.

The field tests can provide initial data, which may be analyzed and discussed with decision makers. As a result some of the important tradeoffs will be recognized: A question raised may be "How much more efficient must the work procedure be in order to justify an investment of XXXX Euros?"

## 8.2 Analysis of Business Benefits, Drivers and Inhibitors

In order to shed light on the possible business models in healthcare and secure the widest possible foundation for the business cases, the REACTION project will conduct a one-day, high-level workshops in each domain with a group of experts in the field

The project's application experts for the two application areas (inpatient and outpatient) will be invited together with external experts. The participants will represent diverse viewpoints such as clinicians, solution providers, system integrators, component manufacturers, and healthcare economists. A set of healthcare scenarios will be presented as lead-in to the discussions.

Simple procedures will be used to investigate these relations such as rating scales or "positioning", where experts are directly asked to assess the scenarios and quantity the tradeoffs.

This work will also be undertaken as part of WP9 Socio-economic Framework.

# 9. Figures and Tables

# 10. References


(Apache2011) Apache Axis2 – Apache Axis2/Java – Next Generation Web Services: http://axis.apache.org/axis2/java/core/.

(ASCEM2010) ASCEM, "Software Verification and Validation Plan – Basic Phase", United States Department of Energy, 2010.

(Atlassian2011) Bug, Issue and Project Tracking for Software Development - JIRA: http://www.atlassian.com/software/jira/.

(Bennet1984) Bennet J, "Managing to meet usability requirements: establishing and meeting software development goals", in Bennet J, Case D, Sandelin J, Smith M (Eds.), Visual Display Terminals. Prentice-Hall, Englewood Cliffs, NJ, pp. 161–184, 1984.

(Beyer1998) Beyer H, Holtzblatt K, "Contextual Design: Defining Customer-Centered Systems", San Francisco: Morgan Kaufmann, 1998.

(Boehm1988) Boehm BW, "A Spiral Model of Software Development and Enhancement", Cornputer, pp. 61-72, May 1988,

(Caduff2009) Caduff A, Talary MS, Mueller M, Dewarrat F, Klisic J, Donath M, Heinemann L, Stahel WA, "Non-Invasive Glucose Monitoring in Patients with Type 1 Diabetes: A multisensor System Combining Sensors for Dielectric and Optical Characterisation of Skin", Biosens. Bioelectron. 24, 2778-2784, 2009.

(CLINICIP2011) CLINICIP - closed loop insulin infusion for critically ill patients: http://www.clinicip.org/.

(Dijkstra1969) Dijkstra EW, "Structured Programming" in Classics in Software Engineering ed. Edward Nash Yourdon, 1979.

(DIR2007) "DIRECTIVE 2007/47/EC OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 5 September 2007", in the Official Journal of the European Union, 2007.

(Drusinsky2008) Drusinsky D, Michael JB, Shing MT, "A Visual Tradeoff Space for Formal Verification and Validation Techniques", IEEE System Journal, vol. 2 issue 4, pp. 513-519, Dec 2008.

(ESA1995) ESA Board for Software Standardization and Control, "Guide to Software Verification and Validation", European Space Agency, 1995.

(Hall2011) Hall K, "Developing Medical Device Software to IEC 62304", European Medical Device Technology, vol. 1, no. 6, 2011 (http://www.emdt.co.uk/article/developing-medical-device-software-iso-62304).

(Hass2008) Hass AMJ, "Guide to advanced software testing", Artech House, Inc., 2008.

(IEC2005) IEC 60601, "Medical Electrical Equipment Standard", International Electrotechnical Commission, 2005.

(IEC2006) IEC 62304, "Medical device software – Software life cycle processes", International Electrotechnical Commission, 2006.

(IEC2007) IEC 62366, "Medical Device Usability Engineering", International Electrotechnical Commission, 2007.

(IEEE2004-1) IEEE Std 1012-2004, "1012 IEEE Standard for Software Verification and Validation", IEEE Computer Society, 2004.

(IEEE2004-2) ISO/IEEE 11073 "Health informatics - Medical / health device communication standards", IEEE Computer Society, 2004.

(ISO2005) ISO 9000:2005, "Quality management systems -- Fundamentals and vocabulary", International Standard Organization, 2005

(ISO2007) ISO 14971:2007, “Medical devices – Application of risk management to medical devices”, International Standard Organization, 2007.

(ISO2009) ISO 10993, “Biological evaluation of medical devices”, International Standard Organization, 2009.

(Junit2011) Welcome to JUnit.org! | JUnit.org: http://www.junit.org/.

(Kulyamin2008) Kulyamin VV, "Software verification methods", RAS Institute for System Programming, 2008.

(MDD1990) “Council Directive 90/385/EEC of 20 June 1990 on the approximation of the laws of the Member States relating to active implantable medical devices”, Official Journal L 189, 1990.

(MDD1993) “COUNCIL DIRECTIVE 93/42/EEC of 14 June 1993 concerning medical devices”, Official Journal L 169, 1993.

(MDD1998) “Directive 98/79/EC of the European Parliament and of the Council of 27 October 1998 on in vitro diagnostic medical devices”, Official Journal L 331/1, 1998.

(Murphy2008) Murphy B, Wakefield A, Friedman J, “Best Practices for Verification, Validation, and Test in Model-Based Design”, The Mathworks, Inc., 2008.

(SDM2011) Software Development Methodology: http://www.hyperthot.com/pm_sdm.htm.

(Solianis2010) Solianis, “MGMS Integration & System Test Plan”, internal technical report v00, Feb 2010.

(Solianis2009) Solianis, “Multisensor Test Specification”, internal test report, v000, Mar 2009.

(Subversion2011) subversion.tigris.org: http://subversion.tigris.org/.

(Thestrup2008) Thestrup J, Guarise A, Gräfe G, “Business Modelling of Emerging Service Opportunities in the IoTS Ecosystem, The Internet of Things and Services”, 1st International Research Workshop, Sophia-Antipolis, France, September 2008.

(TortoiseSVN2011) tortoisesvn.tigris.org: http://tortoisesvn.tigris.org/.

(VisualSVN2011) VisualSVN Server | Subversion Server for Windows: http://www.visualsvn.com/server/.

(Wikipedia2011-1) Usability testing – Wikipedia, the free encyclopedia: http://en.wikipedia.org/wiki/Usability_testing.

(Wikipedia2011-2) List of unit testing frameworks – Wikipedia, the free encyclopedia: http://en.wikipedia.org/wiki/List_of_unit_testing_frameworks#Java.

(Wikipedia2011-3) Medical Device Directive – Wikipedia, the free encyclopedia: http://en.wikipedia.org/wiki/Medical_Devices_Directive.