TAMPERE
POLYTECHNIC

FINAL THESIS REPORT

UTILISING CONSUMER-LEVEL TESTING RESULTS IN SOFTWARE PRODUCT DEVELOPMENT

Mari Kattainen

Business Information Systems
October 2005
Supervisor: Paula Hietala

TAMPERE 2005

TAMPERE
POLYTECHNIC

Author	Mari Kattainen
Degree Programme	Business Information Systems
Title	Utilising consumer-level testing results in software product development
Month and year	October 2005
Supervisor	Paula Hietala

Pages: 43

ABSTRACT

This thesis focuses on software product testing results that are received when a special consumer-level test group tests the product during its development phases. The software product in question is Nokia PC Suite 6.5 that was released in April 2005. Consumer-level test users are Nokia employees who do not work with Nokia PC Suite development, and therefore, the product is not very familiar to them. Their comments on the product can be considered to be similar to the product end-users' comments.

In this thesis, test results that consist of actual product functionalities are analysed, but the consumer-level test user considers the functionalities as product malfunction. Target for the analysis is that from these results possible needed development areas are identified either in product development, in consumer-level testing, in error report handling, or in all of them .

With a reference to Nokia PC Suite quality, some product quality and testing-related definitions are presented as reasoning for the analysis. Especially software product functionality and usability are handled in order to provide some background for reading through the analysis. Also, Nokia PC Suite product applications are presented so that they can be referred to when reading the analysis.

The analysis was done between April and June, 2005, and the analysis targets, process and results are presented separately in the thesis. Because of the product development confidentiality restrictions, only some examples of the issues and the results are given. No exact numbers are presented for the same reasons. In autumn 2005, findings from the analysis of test results have already led, and may still lead, to some improvements in the product development practises. These issues are also presented in the error analysis chapter with some suggestions and examples for handling them.

Keywords Testing Quality Functionality Usability Software product

TAMPEREEN
AMMATTIKORKEAKOULU

Tekijä	Mari Kattainen
Koulutusohjelma(t)	Tietojenkäsittely
Tutkintotyön nimi	Loppukäyttäjätason testaustulosten hyödyntäminen ohjelmistojen tuotannossa
Työn valmistumis-kuukausi ja -vuosi	Lokakuu 2005
Työn ohjaaja	Paula Hietala

Sivumäärä: 43

TIIVISTELMÄ

Tässä tutkintotyössä käsitellään testaustuloksia, jotka on saatu ohjelmistotuotteen loppukäyttäjiä vastaavan ryhmän suorittamasta tuotetestauksesta tuotteen kehitysvaiheessa. Tämä ohjelmistotuote on Nokia PC Suite 6.5, joka julkaistiin huhtikuussa 2005. Loppukäyttäjiä vastaava testaajien ryhmä koostuu Nokian työntekijöistä, jotka eivät työskentele Nokia PC Suiten tuotekehityksen parissa. Nokia PC Suite 6.5:n ominaisuudet eivät siten ole tälle ryhmälle tuttuja. Ryhmältä saatavan palautteen voidaankin siitä syystä rinnastaa vastaavan tuotteen varsinaisen loppukäyttäjän palautetta.

Testaustuloksista käsitellään sellaisia asioita, joita loppukäyttäjätasoinen testaaja pitää tuotteessa vikana, vaikka nämä ovat tuotteen senhetkistä toiminnallisuutta. Näiden asioiden analysoinnin tavoitteena on tunnistaa mahdollisia kehityskohteita joko tuotekehitykseen, loppukäyttäjätason testaukseen, vikaraporttien käsittelyyn tai kaikkiin niistä.

Nokia PC Suite -tuotteella on korkeat laatutavoitteet. Sen vuoksi tässä tutkintotyössä käsitellään myös ohjelmistotuotteen laatua ja testausta. Ohjelmistotuotteen toiminnallisuus ja käytettävyys esitellään erikseen, koska niitä tarvitaan taustaksi muun muassa analyysiosuutta varten. Myös Nokia PC Suiten eri sovellukset esitellään erikseen, jotta lukija voi palata niihin tarvittaessa esimerkiksi analyysia lukiessaan.

Analyysi tehtiin huhti - kesäkuussa 2005. Analyysin vaiheet ja tulokset esitellään omassa kappaleessaan tässä tutkintotyössä. Tuotekehitystä koskevien luottamuksellisuussäädösten vuoksi kaikkia analyysin tuloksia ei esitellä yksityiskohtaisesti, vaan kerrotaan vain joitakin esimerkkejä analyysin tuloksista. Näiden perusteella Nokia PC Suiten tuotekehityksen käytännöissä on jo tehty ja mahdollisesti tullaan vielä tekemään joitakin muutoksia syksyn 2005 aikana. Myös nämä asiat toteutusehdotuksineen ja esimerkkeineen esitellään tämän tutkintotyön analyysikappaleessa.

Avainsanat Testaus Laatu Toiminnallisuus Käytettävyys Ohjelmisto

Table of contents

1 Introduction, Scope and Structure 5
1.1 Introduction 5
1.2 Scope and Structure of the Thesis 5
2 Nokia PC Suite 7
2.1 Introduction 7
2.2 PC Suite 6.5 Requirements 8
2.3 New PC Suite Applications 9
2.3.1 Get Connected 9
2.3.2 One Touch Access 10
2.3.3 Image Store 10
2.4 Other PC Suite Applications 11
2.4.1 Existing Applications and Functions 11
2.4.2 PC Suite Start Menu 12
2.4.3 Nokia PC Sync 13
3 Software Product Quality 14
3.1 Software Quality Definitions 14
3.2 Software Quality Characteristics 16
3.3 Software Product Functionality 17
3.4 Software Product Usability 18
3.5 Functionality vs. Usability 19
3.6 PC Suite Quality 20
4 Software Product Testing 21
4.1 Testing in General 21
4.2 Consumer-level Testing 21
4.3 Nokia PC Suite Testing 21
4.4 PC Suite Error Reporting 22
4.5 Error Life Cycle and Error Functionality Status 23
4.6 Consumer-level Testing with PC Suite 6.5 24
5 Error Report Analysis 25
5.1 Introduction 25
5.2 Targets and Expectations 25
5.3 Analysis Process 25
5.4 Analysis Results 26
5.5 Conclusions and Further Suggestions 31
5.5.1 Product Functionality 31
5.5.2 Usability Errors 31
5.5.3 Small Usability Test 32
5.5.4 Change Requests 33
5.5.5 Future Actions 34
6 Thesis Conclusions 36
References 37
Other references (not publicly available) 38
Appendices 39
Appendix 1: Terms and abbreviations 39

1 Introduction, Scope and Structure

1.1 Introduction

In September 2005, Nokia is a world leader in mobile communications and the company is well known almost everywhere in the world. Being a market leader in mobile phone sales in 2005, Nokia has succeeded in bringing the mobile voice to the consumers' lives.

With mobile phone technology and properties evolving all the time, the needs for mobile phone support programs have grown, too. This is where Nokia PC Suite comes forward. Nokia PC Suite is a PC software product, which allows Nokia mobile phone users to manage their phones with a PC. Nokia PC Suite is offered to Nokia phone users to add value to their mobile phones in many different ways, which are presented in Chapter 2 in this thesis.

Nokia PC Suite is used all over the world, and its quality has a high importance in product development work. This thesis focuses on the Nokia PC Suite. Also, software quality is introduced in the thesis, especially because of one of the software quality criteria: functionality. Nokia PC Suite quality is assessed through all the product development phases. Evaluation is done, for instance, based on functional test results of the product. In this thesis, the focus, however, is not in the functional test results, but on the analysing of the consumer-level testing and its results. The analysis was made from the Nokia PC Suite consumer-level test results in the spring of 2005. The analysis and its results are introduced in detail in Chapter 5.

Due to Nokia's product development-related confidentiality restrictions, some facts and figures in this thesis are not presented in detail but percentage values are given instead. The analysed material is not attached to this thesis due to the same reason.

1.2 Scope and Structure of the Thesis

Four main areas are covered in this thesis. At first, **Nokia PC Suite, version 6.5** is introduced. New applications in PC Suite 6.5 are presented separately, older applications are not discussed in the same detail. Also, other product information is given so that the reader can refer to that information when needed, e.g., in the analysis part of the thesis. The product information also gives an overall picture of the various dependencies the product has both in its use and in its development.

Secondly, **software quality** is handled. Since the PC Suite is a widely used software product, quality-related issues have a significant role in the product development. The software quality part also gives an introduction to the third part of the thesis: **testing**. Functionality is one of the software quality criteria, and functionality is assessed especially with various test results.

In the testing presentation, the basis to the fourth issue in the thesis is covered. The fourth main area in the thesis is the **analysis of Nokia PC Suite 6.5 consumer-level testing functionality errors** (i.e. the reported errors where the issue was classified to be current product functionality in the PC Suite error handling). The process, targets and expectations, results and future actions of the analysis are introduced in Chapter 5.

Nokia PC Suite development phases and product functional testing are out of the scope of this thesis. They are referred to in a few chapters but not presented in detail, because they are not relevant to the thesis.

After all four main areas have been discussed, the conclusions chapter summarises the thesis and the thesis project from the author's point of view. Since some terms and abbreviations may be unknown to the reader in the thesis,the terms and abbreviations are explained in Appendix 1.

2 Nokia PC Suite

2.1 Introduction

Nokia PC Suite is a software product, which consists of a set of PC applications. These applications enable the PC Suite users to take full advantage of their Nokia mobile devices.

As the PC Suite 6.5 consumer-level test results were used as basis for the analysis in this thesis, also the Nokia PC Suite product descriptions in this thesis consider Nokia PC Suite 6.5 applications and features only. PC Suite 6.5 was released in April 2005.

The PC Suite product evolves continuously in order to be able to support new Nokia devices in addition to old devices. That is why some of the product applications or features may not be available as such or at all in PC Suite 6.6 and newer versions. PC Suite 6.6 was released in June 2005.

At the time of the release, the Nokia PC Suite 6.5 could be downloaded for free from the following Internet sites:

- www.nokia.com/pcsuite
- www.nokiausa.com/pcsuite and
- www.nokia-asia.com/pcsuite.

PC Suite 6.5 was also included in most of the phone model sales packages whose sales started between April and June of 2005. The phone sales package usually includes a CD (compact disk) with various software applications to give added value to the new phone. Depending on the phone model, the CD may include some applications, programs, etc. In most cases also the PC Suite is available on the CD.

PC Suite 6.5 also included connectivity cable driver installation for two connectivity cable versions. With some earlier PC Suite versions the cable driver installation had to be done separately. With PC Suite 6.5, a separate driver installation was still needed for one of the supported connectivity cable drivers. Cable drivers are needed for establishing a cable connection between the mobile device and a PC.

PC Suite 6.5 was available in 35 language package versions. A language package version is tailored to that specific geographical area and to a specific language that is most likely used in that area.

2.2 PC Suite 6.5 Requirements

In the spring of 2005, there were 100 Nokia phone models that were fully or partly compatible with Nokia PC Suite 6.5. The supported PC operating systems were Microsoft Windows 2000 and Microsoft Windows XP. The PC had to have at least 250 megabytes of free hard disk space so that the PC Suite could be installed smoothly.

Compatible Personal Information Managers (PIMs) at the time were:

- Microsoft Outlook 98, 2000, 2002 and 2003
- Microsoft Outlook Express/Windows Address book
- Lotus Notes 5.x and 6.x
- Lotus Organiser 5.x and 6.x

Connectivity medias in the PC Suite 6.5 were:

- Infrared
- USB cable
- Bluetooth

2.3 New PC Suite Applications

2.3.1 Get Connected

There were a few product improvements, as in all PC Suite versions, in the PC Suite 6.5. For instance, with PC Suite 6.5, the user was now able to connect his phone with the PC more easily than before by using a guided instruction wizard called **Get Connected.** The wizard's start dialog box is presented in Figure 1.

The Get Connected wizard appears and prompts the user to create a connection between the mobile device and PC right after the PC Suite installation. The connection can be created then or later by starting Get Connected in the PC Suite start menu by clicking the Get Connected button. The button is shown in Figure 2 below. The PC Suite start menu is introduced on page 12, in Figure 6. The connection between the devices and the PC can be established either by using a USB cable, Bluetooth or infrared.

Figure 1 Get Connected wizard

Figure 2 Get Connected start button in PC Suite start menu

2.3.2 One Touch Access

With the **One Touch Access** application and a compatible Nokia mobile device, the user is now able to establish an Internet connection more easily than before. This application was also introduced with PC Suite 6.5 for the first time. Application is presented in Figure 3 below.

Figure 3 One Touch Access application

2.3.3 Image Store

The third new application in the PC Suite 6.5 was **Image Store**, which is presented in Figure 4. With Image Store the user can transfer pictures and video clips between his phone and a PC. Image Store also makes an automatic transfer of all new images and video clips onto the PC possible. This can be made to happen right after the PC and the phone have been connected.

Figure 4 Image Store application

2.4 Other PC Suite Applications

2.4.1 Existing Applications and Functions

In addition to the new applications introduced in the previous chapters, the following functions with the other applications of the PC Suite 6.5 were, and still are, possible when writing this thesis in September, 2005. For instance, the user is able to transfer various files, manage and synchronise text messages, calendar items and contacts between a mobile device and a PC. The applications for executing the tasks are Nokia Phone Browser, Nokia PC Sync, Nokia Text Message Editor and Nokia Contacts Editor.

With the Nokia Content Copier application presented below (Figure 5) the user can easily back up his personal information and settings from the phone and restore them back to the same phone or the same phone model when needed. With some exceptions, he can also transfer the information from his phone to a PC to be used in a different phone model, e.g. when buying a newer device model or using a smaller size device during travelling.

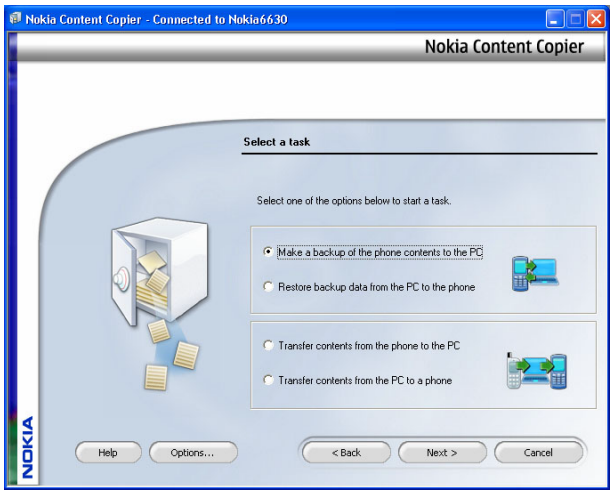

Figure 5 Content backup and restore & content transfer dialog box

It is also possible to organise and convert digital music files on the PC and transfer them to a compatible Nokia phone. That can be done with the Nokia Audio Manager. Viewing audio and video files and CD content and playlists can be performed with the Nokia Multimedia Player. Creating wallpapers and ring tones or transferring applications to the phone can also be managed with PC Suite. The applications for these tasks are Nokia Image Converter, Nokia Sound Converter and Nokia Application Installer.

2.4.2 PC Suite Start Menu

All the introduced applications can be easily started from the Nokia PC Suite start menu presented in Figure 6. When the user moves a mouse over an application icon on the start menu, he gets more information about that application in the description field. Description field is circled in Figure 6. The name and properties of the chosen application are shown to the user in a splash screen when he opens the application.

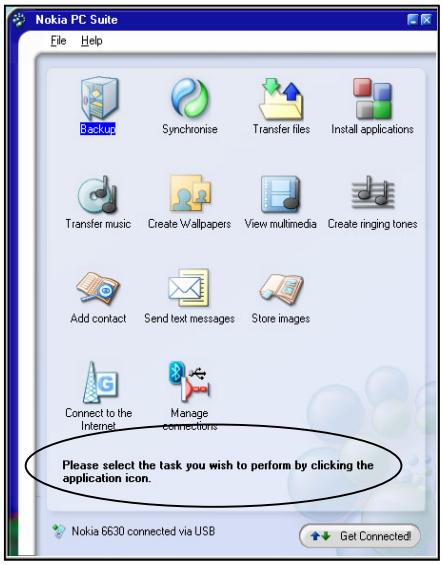

Figure 6 PC Suite's start menu

The PC Suite start menu can be opened from the PC Suite icon from the PC task bar or from the PC Start menu-> Programs ->Nokia PC Suite. In addition to that all applications can be started directly from the PC Suite icon on the PC task bar by using the 2nd mouse button. That icon is circled on the left in Figure 7. The other icon circled in Figure 7 is for the Nokia PC Sync application, which is presented in the next chapter.

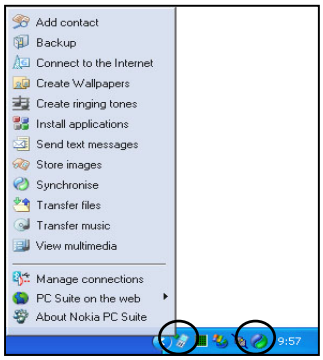

Figure 7 PC Suite and PC Sync icons and applications start menu opened from the icon on the task bar

2.4.3 Nokia PC Sync

The **Nokia PC Sync** application may be used many times a week in order to keep the calendars both in the PC and in the phone in synchronisation. PC Sync makes the use very easy for the regular user. It also copies the application icon separately to the PC task bar, and PC Sync can be started easily from there. PC Sync icon was presented in the previous figure (Figure 7).

Figure 8 below presents the synchronisation application's start menu. In order to make the synchronisation as easy as possible, it can also be set to start automatically every time the connection between the mobile phone and a PC is established.

Figure 8 PC Sync start menu

3 Software Product Quality

3.1 Software Quality Definitions

Software product quality has a significant role in all product development, and also in PC Suite software product development. The word **quality** has several meanings. For instance, the BS EN ISO (British Standard, English language version, International Organisation for Standardisation) 8402 for quality management and quality assurance – vocabulary defines quality being the totality of features and characteristics of a product or a service that bear on its ability to satisfy stated or implied needs (BS EN ISO 8402, Section 2 1995:16).

Software quality cannot be thought of only as a software product that contains no errors. A software quality specification must be more accurate and detailed. Later in this chapter and the next chapters some quality models for formalising software quality are presented.

The figure below (Figure 9) illustrates well what quality means for different people. The context or the experience that people have give various meanings to quality.

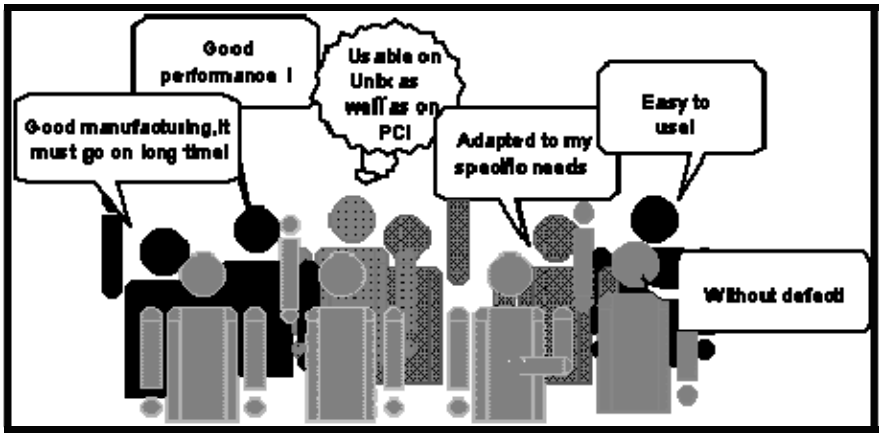

Figure 9 How quality is understood (Quality Characteristics 2005)

“Quality is meeting and exceeding the customer’s needs and expectations and then continuing to improve” is an often used expression of quality. This statement may originate from Dr. W. Edwards Deming, known as the father of the Japanese post-war industrial revival and regarded by many as a leading quality guru in the United States but the statement is also quoted in many training materials and other quality context without a reference to the originator.

In the current software development industry where quality targets are set for achieving the best customer satisfaction on the market the previous statement can be used as a guideline. In order to earn the customer trust and gratitude with the software product, the whole software development organisation must be committed to continuous quality improvement.

Another software quality authority, Tim Kasse, encapsulates software quality in being the quality of the product and the quality of the processes. The quality of the product demonstrates what people produce and the quality of processes how people do it. (Kasse Initiatives 2003.) These factors again are tied to the quality of the whole development organisation's activities, and with them a total quality management system for a development organisation can be established.

In addition to the previous definitions of software quality, a description that is often used says that software quality is a quality system that is delivered on time, to budget and to specification. These aspects require a feasible schedule, a well-planned funding, and a system that meets the specification. (Veenendaal 2002: 14.) When all these aspects are achieved a software product customer is usually satisfied.

The previous features, however, may only fulfill the expectations of an industrial customer. An industrial customer may use the quality system as a subsystem for his final software product delivery to the customer. Or he can use it as an accessory to his system, and therefore, all those quality features mentioned are extremely important and of great value to him.

On the other hand, the customer being an end-user of a software product sees the quality from a different perspective. He is of course interested in the factors mentioned earlier, but on a lot smaller scale. Being, e.g., an eager game consumer, he has a certain budget for getting a new game; he knows when it should be available and how it should work in his home computer environment. For him, software product quality appears through the actual usability and functionality of the game. He does not know how the game was specified to function, and if it has some features that do not comply with his understanding of the common game logic, in his opinion the game probably is of poor quality.

All the earlier mentioned definitions of software quality are without doubt accurate. Quality is composed of all those descriptions, one is not better than the other, and there are no false interpretations in these definitions. In the software development especially important

to an organisation is that a quality model for operations exists and quality targets have been set.

3.2 Software Quality Characteristics

BS ISO/IEC (International Electrotechnical Commission) 9126-1 standard for Software Product Quality (BS ISO/IEC 9126-1 2001:7) introduces a quality model for external and internal quality. The model details software quality characteristics and their sub-characteristics. These quality characteristics are:

- Functionality
- Reliability
- Usability
- Portability
- Efficiency
- Maintainability

The mentioned quality characteristics provide a good framework for evaluating software quality. For being able to use the provided quality model in measuring the software quality, a measurable set of internal attributes needs to be determined for every characteristic and sub-characteristic. Software quality characteristics are illustrated in the following figure (Figure 10).

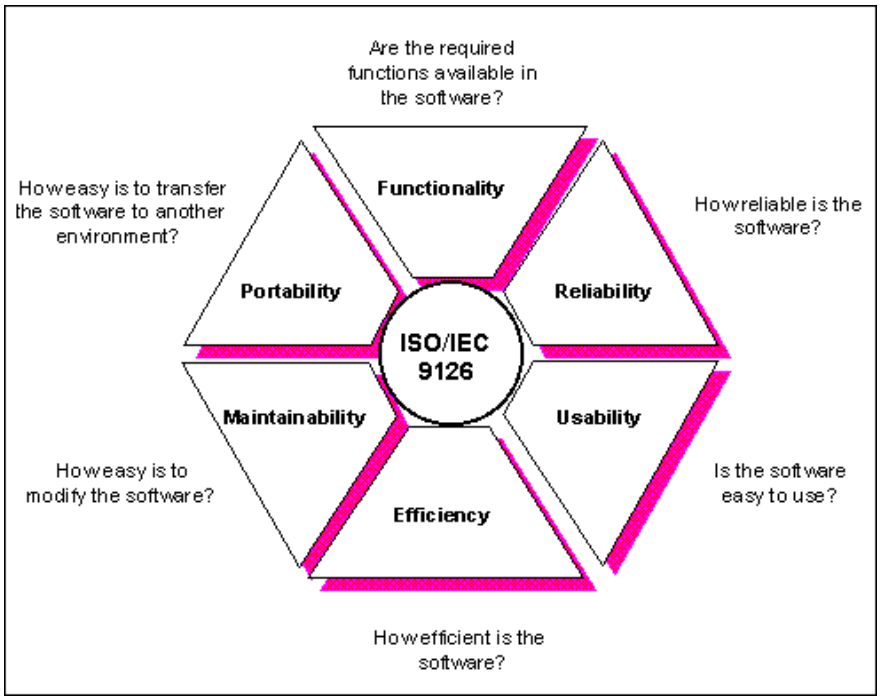

Figure 10 Quality characteristics for software (The six quality characteristics of software 2005)

3.3 Software Product Functionality

The Testing Practitioner states (according to ISO 9216-1 Standard definitions) that when a software product is capable of providing functions that are stated and implied by the product user in the environment it was specified to function, then the product functionality is as expected. The product functions as it should function. (Veenendaal: 381.)

The standard for software product quality presents that functionality is interested in what the software actually does in order to fulfill the needed functions. The other five quality characteristics mainly concentrate on when and how the software fulfills the existing needs. (BS ISO/IEC 9126-1, 2001: 7.)

The focus is on product functionality in this thesis. As introduced in the introduction of the quality characteristics, functionality is an essential part of a software product's quality. A product that functions poorly cannot be used for the purpose it was meant for. A product that functions poorly may even be completely unusable.

On the other hand, if functionality is perfect in every aspect, software product quality might still be poor. For instance, if all the tasks required by the software product user can be executed flawlessly but it takes five to six times more time than any possible user could have expected, then the quality is poor, the product is not efficient.

When defining measurable targets for product functionality the sub-characteristics presented in the next table (Table 1) provide a good environment for detailed evaluation. By setting quality targets and creating test cases for assessing every area, the results received are more precise. Such results can be used in improving exactly the right issues in product functionality.

Table 1 Functionality Sub-characteristics (BS ISO/IEC 9126-1, 2001: 7 - 8)

Software Quality Characteristic	Sub-characteristics	Definitions
Functionality	Suitability	The capability of the software product to provide an appropriate set of functions for specified tasks and user objectives
	Accuracy	The capability of the software product to provide the right or agreed results or effects with the needed degree of precision.
	Interoperability	The capability of the software product to interact with one or more specified systems.
	Security	The capability of the software product to protect information and data so that unauthorised persons or systems cannot read or modify them and authorised persons or systems are not denied to access them.
	Compliance	The capability of the software product to adhere to standards, conventions or regulations in laws and similar prescriptions relating to functionality.

3.4 Software Product Usability

Broad focus definition of usability is defined in ISO 9241-11:1998 standard: “The effectiveness, efficiency and satisfaction with which specified users can achieve specified goals in particular environments”. In a narrow focus definition of usability, it is said that software must have a capability to be understood, learned, used and liked by the user. All that must be achieved under specified conditions. Also, the narrow definition originates from the standards, this one from ISO 9126-1 Software product quality. (Veenendaal: 224 – 225.)

As the quality model’s usability sub-characteristics in Table 2 present, the product must be logical to use, and the user must be able to understand the product’s applicability for executing the tasks he demands. The product should also be easy to learn and operate and attractive to the user.

Attraction as a usability sub-characteristic has been added to the ISO 9126-1 standard in 2001. This proves the rapid development in the information technology industry, now also the attractiveness of the software product is seen as a part of product quality. In the earlier decades the development focus was inevitably only on the basic product functions and their execution so that the required tasks could be executed with the necessary level of quality.

Table 2 Usability sub-characteristics (BS ISO/IEC 9126-1, 2001: 9-10)

Software Quality Characteristic	Sub-characteristics	Definitions
Usability	Understandability	The capability of the software product to enable the user to understand whether the software is suitable, and how it can be used for particular tasks and conditions of use.
	Learnability	The capability of the software product to enable the user to learn its application.
	Operability	The capability of the software product to enable the user to operate and control it.
	Attractiveness	The capability of the software to be attractive to the user.
	Compliance	The capability of the software product to adhere to standards, conventions, style guides or regulations relating to usability.

3.5 Functionality vs. Usability

When defining usability and functionality, CSE (Center of Software Engineering) presents the definitions quite simply. According to CSE usability answers to the question: “Is the software easy to use?”. Functionality for its part answers to the question: “Are the required functions in the software available?” (The six quality characteristics of software 2005.)

In practice, these two issues are quite alike in the mind of a software product user. The user might think that he should be able to execute some functions with the new software product he had just bought, but then he does not get the results he expects. If he is a novice user, he thinks that the product is hard to use, i.e., its usability is poor although the truth is that the product was never specified to execute those certain tasks he wanted to accomplish.

When analysing the issues that had been reported as product errors by the consumer-level testers of Nokia PC Suite 6.5, the case was often contrary to the previous example. Many issues the test user had reported were qualified being current product functionality. Undeniably that is what the issue was but the cause for the report quite often was in usability. Instructions on the application’s user interface on how to use the application had confused the test user, and he had thought that the result would be something else than what happened. The software was not easy to use.

When referring to software product quality standard, note 1 in usability definitions says that “ Some aspects of functionality,

reliability and efficiency will also affect usability, but for the purposes of ISO/IEC 9126 they are not classified as usability." (BS ISO/IEC 9126-1, 2001: 8.) That interpretation also supports the previous contemplation of the fading boundaries between the functionality and usability appreciation in the user's mind.

3.6 PC Suite Quality

Software product quality plays an important role in the PC Suite development. Quality planning is a cross-functional activity of the development organisation's management team, and it aims at reaching a common understanding about quality-related activities.

The output of quality planning is the PC Suite development organisation's quality plan. The quality plan demonstrates that specific requirements for quality are being appropriately identified, defined, planned and resourced in the software development project plans and implemented accordingly.

Between autumn of 2004 and spring of 2005, four quality goals were defined for the PC Suite development organisation. Those goals concerned the following areas:

- Strategy
- Product quality
- Customer satisfaction
- Excellence in execution

All the goals consisted of many separately defined sub-targets, of which one sub-target became the subject for this thesis. That sub-target was a quantitative measurement for PC Suite 6.5 consumer-level testing. As introduced in Chapter 1.1, the measurement concerned the amount of the issues the consumer-level test users report as errors but the issue is actually current product functionality. The reports relating to current **product functionality** is the subject of this thesis, and the analysis of the reports as a whole is presented in Chapter 5. (PC Suite Quality Plan.)

4 Software Product Testing

4.1 Testing in General

In order to achieve the best software product quality on the market and at the same time diminishing the failure rates in product development, it is clear that testing is an essential part of all that. Without proper testing it may be even impossible to achieve decent software product quality, a software product many times being such a complex evolving system that needs constant evaluation of the current maturity level. In evaluating the maturity of the product, test results are of great value, since they give indisputable information of the product's current maturity state.

4.2 Consumer-level Testing

At Nokia, all product programs have their own testing organisations and appropriate testing processes. The specialised test organisations test products during all product development phases locally and globally.

In addition to each program having its own testing organisations, Nokia's employees volunteer each year to test various prototype products and provide feedback of the prototypes to the product programs. Employees are quite eager to participate in these test programs and their feedback is very valuable. With this kind of consumer-level testing, end-user needs are understood better and the right decisions for improving the products are made. Since these selected consumer-level test users are not directly involved with the product's development, their feedback and experience of the product is close to the end-user's impressions of the product. Many of the test users may not have a technical background so their ideas of the product functionality and usability can be considered equal to end-users' ones.

4.3 Nokia PC Suite Testing

The Nokia PC Suite is a product like all other Nokia products that is being tested during all the product development phases. Both functional and localisation testing is performed by the testing organisation. A comprehensive testing plan, e.g., for a calendar year is created in general PC Suite test planning. An overall test plan ensures that all testing activities are in place and resourced to meet the PC Suite release schedule, new product functionality and other relevant long-term demands. In specific PC Suite version test

plans, detailed test goals, criteria, prioritised test cases, test rounds and schedule are defined.

The overall objectives of the PC Suite testing at the time of the PC Suite 6.5 testing were defined in the PC Suite Master Test Plan. The testing objectives, among other issues, were targeted to ensure that the PC Suite would function flawlessly with compatible mobile phones and also for evaluating overall quality of PC Suite software product. (PC Suite Master Test Plan.)

4.4 PC Suite Error Reporting

The PC Suite test activities and results are followed through with various measurements gained from the testing procedures and testing tools. Defects found during testing or with a normal product use are reported to the error database according to the agreed error management process. Error management process ensures that all errors found are analysed and processed properly, and also, that corrections are verified.

Even though in software testing definitions, errors, faults and failures all get a different meaning according to the Glossary of Software Testing Terms presented in Table 3, in the PC Suite development all these anomalies in the product are reported in **error** reports and treated in this thesis as errors.

Table 3: Definitions of an error, a bug and a fault (BS 7925-1: 4 - 5)

Term	Description
Error (mistake)	Human action that procedures an incorrect result. An error can also be technical, like error in data transmission or in disk file.
Fault (bug)	An error can manifest itself in one or many faults in a phase product (diagram, document, program code). Something is wrong and the phase product does no longer fulfil its requirements
Failure (User problem)	A fault in the software manifests itself in erroneous behaviour during operation: wrong output, spurious reboots, unwieldy response times, locking machines, etc.

Every person detecting an error in the PC Suite product is responsible for creating an error report of that finding. Errors reported are handled daily by the error manager, software designers, test users and other responsible persons for the error subject. A special error meeting with all development managers and other specialists is arranged regularly. Error meeting participants prioritise the error correction work for the most

important errors, analyse the implications of certain complex error corrections, make decisions whether to ignore or postpone error correction for certain errors and evaluate software maturity.

4.5 Error Life Cycle and Error Functionality Status

Error life cycle is usually presented in similar ways in literature, in training courses and lectures, and on the Internet. Many software development manuals in various companies use the same flowchart for presenting the error life cycle. Only, e.g., some terminology and number of error stages in the life cycle description may vary. In PC Suite development, error status and steps were presented as follows in the spring of 2005 (Figure 11).

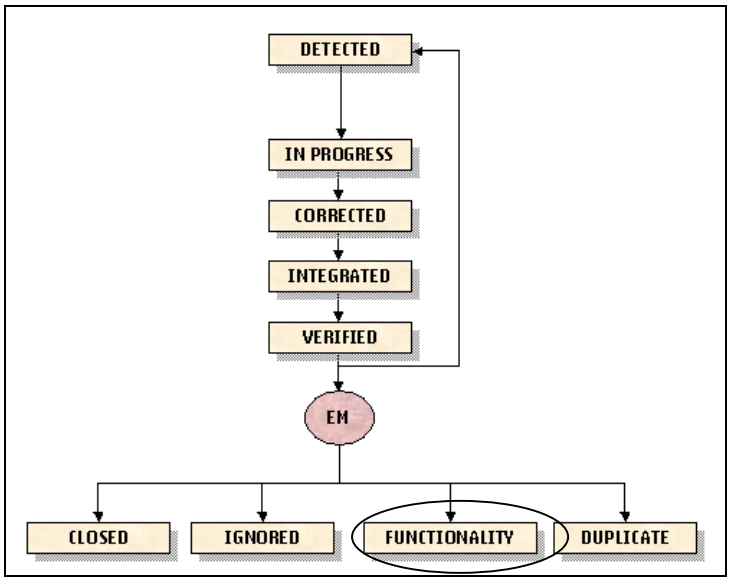

Figure 11 Error life cycle in PC Suite development, spring 2005

In Figure 11, EM stands for Error Manager. The other terms used in the flowchart are error status descriptions. Only the Error Manager and Development Managers can give the following status to the errors in the PC Suite development:

- Closed
- Ignored
- Functionality
- Duplicate

Other error status definitions than functionality are not explained in detail in this thesis, because the focus of the thesis is in the error reports where the status is functionality. **Functionality** status (Figure 11, functionality status circled) in the error life cycle means that even though the test user has thought the issue is an error and has reported it as such, the issue actually is current functionality of

the product. Development manager and/or error manager has made the decision and the issue reported may or may not be changed in the future.

Error prioritisation and classification are especially important in the later product development phases before the actual product is released to the end-users. Error correction prioritisation is crucial so that all severe defects get corrected before the product is released to the public. Some cosmetic errors that are especially hard to repeat and not common use cases may be corrected in the coming releases.

4.6 Consumer-level Testing with PC Suite 6.5

As mentioned in Chapter 4.2, many Nokia employees volunteer to test products in their development phase each year. A special system has been created where the volunteers can announce their interest areas for testing. PC Suite consumer-level test users are chosen from the employees who are especially interested in testing and giving improvement ideas for the PC Suite development.

The PC Suite consumer-level testing for the PC Suite 6.5 was arranged between February and April, 2005. A nominated test leader from the PC Suite development organisation coordinated the whole test process. All the test users were located in Finland but not all of them were Finns. The amount of the test users was less than a hundred.

Before the actual testing the test users got test phones from PC Suite organisation. In the testing they used their own PCs with the given phones. Two informal training sessions were arranged before the testing started. The error reporting tool and some features of PC Suite were presented in those training sessions. The introduction material was also delivered to every participant by e-mail.

The test users were able to download a new PC Suite 6.5 build from the PC Suite intranet page. The test leader informed the test users always by e-mail when there was a new build available.

During the testing period, errors found were reported in a special error tool. Also improvement ideas for further PC Suite versions were given by using that same tool.

When the testing ended, the best improvement idea and best reporters were awarded. Best reporters were persons who found the most of the errors that were not known earlier to the development organisation.

5 Error Report Analysis

5.1 Introduction

This paper analyses the issues that were reported as errors during the consumer-level testing of PC Suite 6.5, but that were classified as actual product functionality in the error handling by the PC Suite development organisation. In this analysis, all results are not presented in exact numbers because of the company confidentiality restrictions but in many cases percentage values are used instead.

5.2 Targets and Expectations

There were few targets for the actual analysis: first of all to understand the total amount of the issues the user, i.e., consumer-level test user understands as an error but the issue is normal product functionality. Secondly, to try to understand why the users report issues that are product functionality as errors. Thirdly, depending on the analysis results to identify development areas either in the product functionality development, consumer-level testing practises or error management process.

To summarise the targets above, the PC Suite product should be so easy and efficient to use that the user will not report its actual functionalities as errors. When the product is easy to use, the usability is good and when it is efficient to use, the functionality is good. On the whole, such a software product has good quality.

Clear expectations before the analysis were that most of the consumer-level problems relate somehow to the PC synchronisation functionality. This was anticipated because most of the feedback from the actual product end-users concerns PC synchronisation. End-users had also quite often experienced problems with the PC Suite installation. Due to some recent installation improvements, it was assumed that there will not be as many installation functionality-related errors reported as earlier.

No specific analysis was made before this study of the earlier PC Suite version functionality error report amounts. There were no expectations based on earlier studies that a certain application's functionality would be more confusing to the consumer-level test user than some other applications' functionality.

5.3 Analysis Process

After the PC Suite 6.5 consumer-level testing had ended, error reports with the functionality status were analysed. This happened

between April and June, 2005. Every error report was printed out from the error database and numbered in the handling order. Those printouts were read through and the main issues from the error reports were collected in one document.

Reviewing the error database was also needed if the actual problem description in the report printout was not understandable without seeing the possible error situation picture. The pictures and other attached files of the error situations were not printed out from the error database.

Like mentioned above, results were collected in one document and arranged per PC Suite application. For reviewing the results and for possible further detailed analysis per case, the following items were recorded in the collection document:

- Error report ID
- Error report title
- Short problem description
- Other needed comments from the error report
- Suggestions for possible further actions

Detailed analysis is not attached in this thesis, but the table below (Table 4) contains an example of the topics in one error report in the collection document. In this example, the case was clearly of product functionality, and it was also handled as such.

Table 4 Example of the error properties in the analysis error collection document

Title: Odd menu entity	Error ID: ER-ER-2005021xxxxx
Problem:	Sync results show the phone name
Solution:	At the time of the report this was new functionality related to one change request
Possible suggestions:	Functionality

5.4 Analysis Results

In the PC Suite 6.5 consumer-level testing, 14 % of all the issues reported were classified to be current functionality. The total amount of the issues reported could be thought as not being alarming, and they might correspond to the normal end-user behaviour. Further analysis of the error report amounts where status was changed to functionality should be done in the future so that a trend of the consumer-level problems can be formed, and the problems can be addressed accordingly.

During the testing phase, the consumer-level test users had reported issues of all PC Suite applications. There were no reports of One Touch Access, Nokia Contacts Editor, Nokia Image Converter and Nokia Sound Converter among the analysed reports. The feedback was good especially for the One Touch Access application development, because the application was a new application in this PC Suite version. According to these results, it could be stated that the One Touch Access's functionality does not confuse the user so that he would understand the application's functionality as an error.

When comparing One Touch Access's results to the product functional testing results, the amount of all the One Touch Access's errors where status was changed to functionality was also low, only 1 % of all the errors reported. One Touch Access, though, is an application that is probably not used that much in the networked office environment, because its purpose is to create an Internet connection with a mobile phone, e.g., in locations where no other possibilities for establishing a connection are available. That is why the amount of the errors found in normal use, not by executing defined test cases may remain low for One Touch Access. But since these report amounts are from both the product functional testing where specified test cases are in use and the consumer-level testing where the testing is also systematic the overall result is good.

PC Synchronisation seemed to cause most of the problems as was expected. The amount of the cases where the user had thought that there is something wrong with the product, but the reported issue was actually normal product functionality, was 38 % of all the errors analysed.

On the whole, there were not that many functionality issues reported as errors in the consumer-level testing . The whole distribution between the applications and some other areas (installation and USB driver) reports is presented in the following figure (Figure 12).

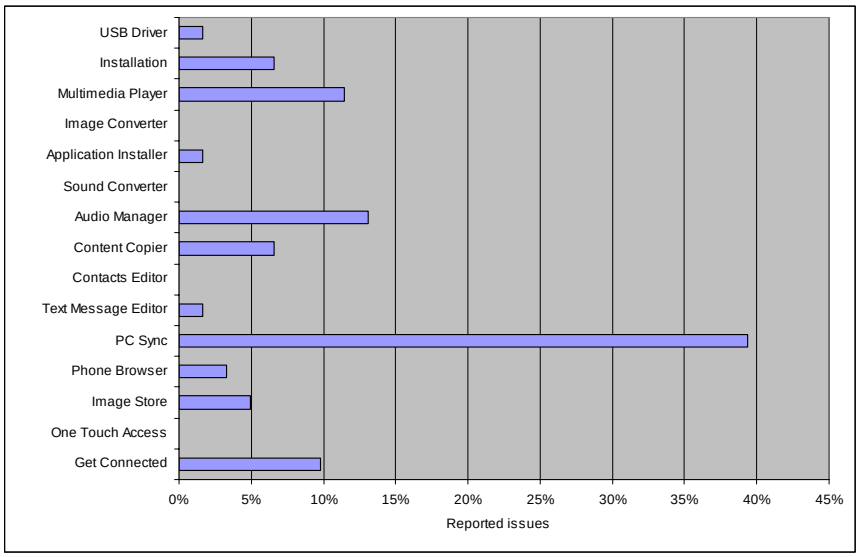

Figure 12 Applications and areas where current product functionality is reported as an error in PC Suite 6.5 consumer-level testing (Nokia name removed from those applications it belongs to due to presentation reasons)

When reviewing the reports further there was clear indication that quite many of the reported situations originated from misunderstanding of the instructions on the application user interface. When checking those cases from the application itself, it was quite understandable that the possibility to choose a wrong way to proceed was not clearly prevented on the application user interface. The issue reported related clearly to product usability. On the whole, there were 29 % of the cases where some usability improvements could have been suggested instead of closing the report because it was current product functionality.

For instance in one error report that concerned the Nokia Audio Manager application and saving CD tracks to the phone, both the testing leader and the software designer had suggested a small addition to the application help. However, this suggested usability improvement was not considered to be included in the next product development phase changes, and the error status had been changed to functionality and report closed. In these usability-related cases where some helpful information was missing from the application, the reported issue had many times received the same judgement. The error description from the example error report is presented in Figure 13.

Title

Saved Tracks from the CD are not copied to the phone

Details

Description:

1) Actual State just before error and how did you get to this state:
2) Step-by-step info how you managed to reproduce the error
3) Expected result
4) Other relevant information

I have opened Nokia Audio Manager. The select the destination for the music files as Memory card of the Phone from Options -> Preferences menu. Now I select some of the tracks from the audio cd and press "Save tracks". When the whole process completes I browser the phone just find that the folder have been on the phone but there are no audio files inside it.

Figure 13 Description of one issue found in consumer-level testing that was current product functionality

As mentioned earlier, in the solutions for the issue above, the test leader had suggested that the PC Suite's help application could explain that transferring tracks to the phone is not possible the way the user is trying to transfer them. Also, the application designer had commented the issue the same way and added that this should be corrected later. Analysis does not show how this issue was handled in the later product development phases, but at the time of this analysis this actual report was closed as current product functionality and the suggested usability improvement in this particular report was not reviewed afterwards.

Other examples of the usability-related issues are, that ,e.g., in one report the test user had reported of the issue that it is not possible to synchronise a video telephone number between the phone and PC. Another case reported that the contact middle name is not synchronised in contact synchronisation. These reports had been closed as current product functionality. The cases were current product functionality, but possibly these small usability issues could have been revised by adding some information to the user interface. In the latter case, for example, a text "Middle name synchronisation not supported" or " Please notice that middle name synchronisation is not supported by the phone, contact's middle name will not be synchronised" could be added to the Nokia PC Sync properties .

The issue above originates from the fact that in Microsoft Outlook, which is a Nokia PC Suite compatible PIM (personal information manager), it is possible to add a middle name to the contact properties but most Nokia phone models do not have a middle name in their contact properties available at the time of this analysis.

Also, a term or an expression change could have helped in some cases. An example is a case when transferring music from the PC to the phone. After the task was successfully completed, a descriptive term for the action could have been added, e.g., “Transfer OK”. The user became confused when the application had only the recently transferred music tracks highlighted on the track list and only their status was “saved”. The user reported that he was afraid that the earlier saved music tracks were now lost because they were not shown as saved anymore. This is maybe not such a usual case of misunderstanding the product behaviour, but something could have been done. If the actual result of the executed task is something else for the user than he expects it should be, he might think that the whole operation failed.

All usability-related cases were quite similar to the previous examples. The results give a clear indication that these small usability issues would need more attention in the future. A small usability issue in this thesis means that changes and corrections would not be very demanding and time-consuming to make. They should be made to the application user interface so that the user would understand the application better and he does not necessarily have to refer to the application’s help or to the PC Suite user’s guide.

Figure 14 illustrates the reasons for the reported issues where only a small usability improvement could have helped the product user. The category other in Figure 14 is used for the issues where the reason for the report was not unambiguous.

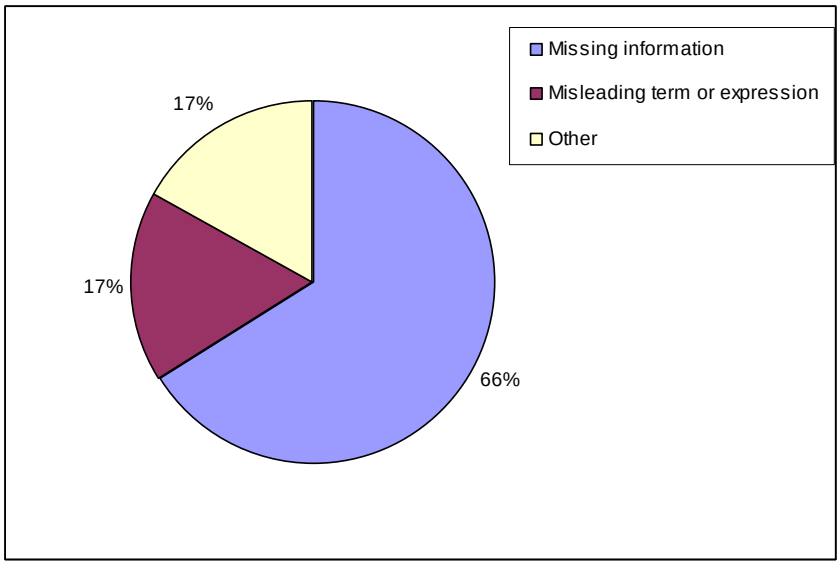

Figure 14 Usability related reasons for reporting current product functionality as an error

5.5 Conclusions and Further Suggestions

5.5.1 Product Functionality

In this quite small a scale of the analysis, it was seen that there was no functionality in the product which clearly would mislead the users, because there were no exactly similar recurring issues reported. There were not that many reports in this functionality category so the scale can be considered small. Naturally in the actual product functional testing, there usually are few recurring issues of the product behaviour anomalies reported, and they may have been corrected already in the early development phases.

It was not studied how many of the real errors had a recurring pattern in this consumer-level testing period, because it was not in the scope of the thesis. Also, errors reported in the normal product functional testing were not reviewed for the same reason. Consumer-level test users reported mostly issues that were either current product functionality or were not errors at all. In the latter cases, the user had, e.g., used the product in a way or for a purpose that it was not meant to. Therefore, those reports were ignored. In order to find some development ideas for the actual product functionality from the consumer-level testing, the results should also be collected from the later consumer-level tests and analysed accordingly.

5.5.2 Usability Errors

Some of the reported issues could have been classified as usability errors also by their status, not to be current functionality and closed. Errors with usability status could have then been handled in some other agreed way if it was not decided to handle them during the current error correction phase. Because there was no status usability or future usability in use, these issues could have been postponed to the future releases as errors and investigated and corrected in the later development phases. As mentioned earlier, in these cases only a small usability improvement could have led to a better product understanding. With small improvements, for example, in the application user interface texts, would have made the user understand the product better, and he would not assume that a current functionality is a malfunction.

In software development, the growing system size and increasing number of users contribute to the fact that the error amount also tends to increase. More resources in software maintenance work are often needed but not necessarily available. Sometimes decisions need to be made between two essential issues, e.g., whether to concentrate on creating new value for the product users or allocate the resources available to the maintenance work. (Jaaksi et

al.1999: 190 – 191.) As also mentioned in Chapter 4.5, in these cases prioritisation of development and software maintenance work is always needed. Thus, e.g., in error correction, some of the needed cosmetic or small usability-related corrections may need to be left out in the current development phase if the focus is in new product functionality. These issues will then be fixed in the later product development phases.

5.5.3 Small Usability Test

Improvements to usability may sometimes be very difficult tasks to execute if the product itself is distributed all over the world for millions of users. It may be even impossible to present certain functions so that they are understood exactly in the same way in every country, continent and culture.

In addition to professional usability tests arranged by specialised companies or universities for many companies, usability tests can be arranged also in a smaller scale. For instance, PC Suite 6.5 consumer-level test group, even with quite technical background, indicated that if the product is not that familiar to the user, or it is rarely used, some of its behaviour and functions may not be easy to understand. Usability sub-characteristics, understandability and learnability referred to in Chapter 3.5, do not come true. In the cases where the reasons for error reporting were clearly in misunderstanding the application terminology, the current product functionality or usability could have been improved by changing a word here and there.

Most offices at Nokia have employees from different countries and cultural backgrounds who have no technical background. These employees could be thought of as typical PC Suite end-users since they have the needed equipment (PC and phone) but no extensive technical expertise. Small-scale usability tests could be arranged with them, since they are easy to reach without special arrangements and can be considered end-users for the reasons given above. For instance, a designer could give new application user interface texts he has planned to use for proof-reading to such employees. Or he could even use his family or friends as test users in a small-scale usability test if such could be arranged without breaking the company’s confidentiality rules.

There are many ways to arrange small-scale usability tests. All it requires is some time, a suitable place and a few willing test users to participate and give their comments to not very demanding questions. These small-scale usability tests could help in preventing, e.g., the kind of misunderstandings that were found in the functionality error report analysis. The following table (Table 5)

gives a good example on how a small usability test can be arranged in a very simple way.

Table 5 Small Usability Test (Sinkkonen et al:133)

Steps for a Small Usability Test
1. A test user is chosen from the target user group.
2. The terms that are planned to be used in the future product, in a new application etc. are written on one paper.
3. The meanings of the earlier mentioned terms are written on another paper. They are written in a different order than they are on the other paper.
4. Papers are given to the test user who marks which terms and meanings he thinks are definitely a pair and those that he thinks may be a pair.
5. The test results are reviewed with the test user and further comments from the test user are recorded. The test user may also suggest more descriptive terms at this point.
6. The terms the test user experienced as odd will be replaced with better ones. The new terms need to be tested, too, especially if they were difficult to come by.

5.5.4 Change Requests

All the new required product functionality in PC Suite development needs a change request to the existing functionality. Change requests are analysed regularly in PC Suite development and either accepted to the next release(s) or rejected according to the current business reasons. The group analysing the change requests may postpone some of the requests into the future if they know that the suggested functionality is needed but, for instance, the environment (e.g. PC, phone, network) is not technically ready for implementation yet.

A valuable finding in the analysis was that some of the issues consumer-level test users had noticed and reported as false behaviour of the product were already known to the development organisation. Change requests had already been made and changes planned for the coming releases in 11 % of the cases. The reported issue at the moment was current product functionality, but it would be changed in the future.

On the other hand, there were also cases where the software designer commenting on the functionality error report as the responsible person had also thought that a change request should be

created. There were also some other issues found in the analysis where new features would have improved the product greatly and a change request would have been recommendable. In the last error handling phases, however, those suggestions had been ignored. Reasons for that were probably some business or development-related reasons based on the information available. Maybe it was predicted that the group analysing the change requests would need to reject them due to prioritised business reasons anyway. This is clearly a target for error management process so that from the consumer-level error reports those should be taken under study more often.

5.5.5 Future Actions

The actions from this thesis could be brought into practise smoothly. The targets for this analysis gave some indicators where to concentrate first.

For instance, the number of the reports on the current product functionality was not surprisingly large. That indicates that no remarkable large functionality improvements should be taken under study according to these results.

As the reasons for the consumer-level test users' functionality reports often seemed to be in usability - the product functions were misunderstood - a finding for the future is that small usability improvement areas relating to end-user's behaviour could be easily identified from these reports.

The third target prior to the analysis was to find some improvement issues in product development, error management or in consumer-level testing practices if the analysis results point to some or all of them. There were few improvement targets for the above found during the analysis. The reasons for the consumer-level test users' reports being usability-related issues, the issues could have been handled separately after the usability improvement need was identified in the report. In the future, there could be, e.g., a separate "future usability" status for that kind of errors just to help to differentiate them from the other errors in the error management work.

The situation in PC Suite development is good in autumn 2005. There are more people concentrating solely on product usability than there were at the time of PC Suite 6.5. Usability gets more attention in error handling than earlier. Consumer-level testing error reports are now handled separately, and a usability expert participates in evaluating them. This originates from the finding in this analysis that reasons for the reports often were usability issues.

The testing itself went quite smoothly so there is no special need to develop that process from the consumer-level testing process point of view. Instructions on reporting the errors and testing the product were understood well and the whole test period was accomplished successfully. The PC Suite development organisation must make sure that the underlying reasons for error reporting are analysed, and that the results are also used in efficient ways in future product development. After all, not much closer to actual end-user behaviour in software development can be get with such easy arrangements.

6 Thesis Conclusions

This thesis covered the analysis of the consumer-level test error reports where quite many interesting issues were found. This kind of analysis is clearly also needed in the future. Consumer-level testing, in addition to normal product functional testing, gives an indication of the end-user behaviour. This is why the issues the consumer-level test users report need to be analysed well. The set targets for the analysis were fulfilled. The feedback received through the error reports and improvement ideas for the future product development are valuable for the development organisation.

From the author's point of view, the thesis project was quite demanding. Some issues were not easy to present because the whole analysis could not be used as such in the thesis, but hopefully the subject catches the reader. A thesis project is a long-lasting project, and a software product development on the contrary can be said to be a fast moving train. This is why many issues found in the error report analysis have already been handled in the later product development phases at the time of finishing this thesis in September 2005. A new practise in the PC Suite development work in autumn 2005 is that the consumer-level testing error reports are being evaluated separately from the other error management work. They may also receive more usability-related attention than they received in the spring of 2005. That on its part makes the functionality of the product more understandable for the user. Thus, Nokia PC Suite product continues to evolve through this consumer-level feedback both functionalitywise and usabilitywise.

References

Jaaksi, Ari, Aalto, Juha-Markus, Aalto, Ari & Vättö Kimmo 1999. Tried & True Object Development: Industry-Proven Approaches with UML. UK: Cambridge University Press.

Sinkkonen, Irmeli, Kuoppala, Hannu, Parkkinen, Jarmo, Vastamäki, Raino 2002. Käytettävyyden psykologia. Helsinki: Edita Oyj.

Veenendaal, Erik van 2002. The Testing Practitioner. The Netherlands: UTN Publishers.

BS (British Standard) 7925-1:1998. Software testing – Part 1: Vocabulary.

BS EN ISO (British Standard, English language version, International Organisation for Standardisation) 8402:1995 Quality management and quality assurance – Vocabulary

BS ISO/IEC (British Standard, International Organisation for Standardisation/ International Electrotechnical Commission) 9126-1: 2001 Information technology – Software product quality- Part 1:Quality model

Quality characteristics in CSE (Center of Software Engineering) Internet [online][referred 11.9.2005].
www.cse.dcu.ie/essiscope/sm2/charact.html

The six quality characteristics of software in CSE Internet [online][referred 26.8.2005].
www.cse.dcu.ie/essiscope/sm2/9126ref.html

Other references (not publicly available)

Kasse Initiatives LLC, Software Quality Engineering, Workshop Training Material 2003, originally from “Software Quality Engineering” by Michael Deutsch & Ron Willis, no year available.

PC Suite Master Test Plan. Test Plan for PC Suite development organisation.

PC Suite Quality Plan. Quality Plan for PC Suite development organisation.

Appendices

Appendix 1: Terms and abbreviations

Application	Software which, together with the related hardware, performs a specific function for the user or, in some cases, for another application
Bluetooth	A technology that provides short-range radio links to allow mobile computers, mobile phones, digital cameras, and other portable devices to communicate with each other without cables. A device with Bluetooth support allows the user to connect wirelessly to another compatible Bluetooth device within a short range.
BS EN ISO	British Standard, English language version, International Organisation for Standardisation
Bug	Defect in a program code or data. An error in the operation of a program is not a software defect but an error due to a mistake in the programming phase. The British Standard BS7925-1 defines this as a manifestation of an error in software.
Build	Collection of programs and files operating in a certain system. The software build modules are compiled into a certain entity, with content suitable for the parameters and files used by the programs.
CD, Compact disk	Optically readable storage medium in which can be stored 700 megabytes of data
Consumer-level testing	Testing to verify that the product functions in real use by using consumer-level test users to detect errors. The consumer-level test users are volunteered Nokia employees who are interested in using proto products and giving comments about the product use and also improvement ideas to the development organisation.
Convert	Change presentation mode without changing information content
Change request	Request to implement a concept change and a new function, or to change an existing function in the software product.

CSE	Center of Software Engineering located in Dublin City University in Ireland
Defect	Nonfulfilment of an intended usage requirement or a reasonable expectation. The expectation must be reasonable under the existing circumstances.
Error	Discrepancy between a computed, observed or measured value or condition and the true, specified or theoretically correct value or condition; in testing, a human action that produces an incorrect result.
EM, Error Manager	Error Manager is a role in PC Suite development organisation. Error Manager has a central role in making sure that the daily error management duties and communication with PC Suite collaborators happen.
Failure	In testing, deviation of the software from its expected delivery or service. Failure is caused by a software defect.
Failure rate	Ratio of the number of failures to a given unit of measure. For example failures per unit of time, failures per number of transactions, or failures per number of computer runs.
Feature	Function or property of a product or system which is interesting either from the customer viewpoint or from the internal product development viewpoint.
Functionality error	Reported issue of the product malfunction that has been analysed and classified to be current product functionality, so the error status has been changed to functionality in the error management database.
Icon	Screen object; on-screen graphic representing a user interface element that a user can select or manipulate, for example an application, document, or disk.
Interface	Common boundary between two associated systems. An interface can be a physical boundary between two associated sets of equipment, across which information may flow. A named set of operations is possible across an interface.

ISO/IEC	ISO = International Organisation for Standardisation, non-governmental worldwide organisation of national standards bodies whose work results are published as international ISO standards. IEC =International Electrotechnical Commission, global organisation that prepares and publishes international standards for all electrical, electronic and related technologies.
Life cycle	Period of time which begins when the need for a system or equipment is defined and ends when the system is destroyed or possibly put to other use.
Localisation	Technical and linguistic adaptation of software or other data product for users of a certain language or culture.
Maintenance	Combination of all technical and corresponding administrative actions, including supervision actions, intended to retain an item in, or restore it to, a state in which it can perform its required function.
Megabyte, MB	Unit of information equal to 1,048,576 (220) bytes.
Multimedia	Communications, which involves simultaneous transmission of pictures, sound, and data over one connection.
Operating System, OS	Software entity that controls the start-up and execution of programs, acting as an interface between hardware and software. For example, Microsoft Windows XP, HP-UX or Sun Solaris are operating systems.
PC	Personal computer.
Peripheral equipment	Equipment, which works in conjunction with a computer but is not a part of it. For example an input-output device is a peripheral device.
PIM, Personal information management	Electronic management of personal and business information such as phone book, calendar, and note information. Personal information management functions are usually part of a PC, a personal digital assistant (PDA-device) or a mobile phone.

Release	Officially approved, frozen configuration that is approved for delivery. Often a release is described through the features and designs it contains. A release may consist of the software build, related hardware, and the related documentation.
Software, SW	Programs, procedures, rules, and any integrated documentation pertaining to the operation of a computer system.
Splash screen	Full or part-screen window that can be seen when launching an application or entering a web site on the PC .
Standard	Documented agreement containing technical specifications or other precise criteria to be used consistently as rules, guidelines, or definitions of characteristics, to ensure that materials, products, processes, and services are fit for their purpose.
Synchronisation, Sync	A process that causes something to occur or recur at the same time or in unison. Synchronisation can be used to make the contents of specific files identical on different devices. For example, a user can synchronise the phonebook or the calendar of a mobile device with the corresponding PC applications using Nokia PC Suite.
System	Set of interacting elements organised for a common purpose, where the characteristics can be modified by changing the elements or the number of elements. The elements of a system can be products or the system can be a product.
Task bar	Usually, a horizontal rectangle in a PC user interface which contains menus, tool icons, and other display entities.
Test case	Set of input values, execution preconditions, predicted outcome, and objective of the test. A test case specifies the individual steps of a test.
Track	Track formed by bits stored one after the other to a moving data or memory device, or a place for such a track, in this context meaning one song on a CD.
User interface, UI	Interface via which a user can interact with software and peripheral equipment.

Universal Serial Bus, USB	A plug-and-play interface between a computer and a compatible add-on device, such as an audio player, joystick, keyboard, phone, scanner, digital camera, or printer. With a USB, a new device can be added to a compatible computer without having to add an adapter card or even having to turn the computer off.
Use case	Collection of interactions between a system or device and its users, relating to a particular goal. A use case should describe system behaviour relevant to the users, or actors, to ensure that their goal is achieved.
Wallpaper	Pattern or picture that can be used, for example as a coloured background for a phone or PC display.
Wizard	Interactive help utility that guides the user through a potentially complex task.