Requirements-by-Example

A Guide for the Discovery of Software Requirements

Version 1.0 – 26th November 2007

by Adriano Comai – www.analisi-disegno.com

Index

1 Introduction

“People learn most concepts by induction from examples” (Fred Brooks)

1.1 Questions and examples

Discovery and specification of requirements for a software system are both critical for the success of projects, and difficult to perform.

The definition of the so-called non-functional requirements (or quality attribute requirements) is particularly critical, because non-functional requirements drive the choice of system and software architectures; yet they are often not requested explicitly by the client and by the other project stakeholders. And they are tricky to invent by analysts and developers.

The two main elements of this requirements guide are a set of questions, and a set of examples.

Questions help to make concrete the characteristics we need to explore, overcoming the obstacle of abstract definitions. When, as an example, we think about usability requirements in abstract terms, it is hard to say anything more than generic propositions. To be able to go deeper, it is useful to ask specific questions, such as “How much the user is expected to remember from previous uses of the system? How much feedback the user needs? How shall error messages be managed?”.

Who answers the questions must be determined in the actual use of this guide, in the specific context in which it is used. The answerer may be any stakeholder of the future system, depending on the context. Anyway, the questions are, first of all, an aid for the analyst to think about requirements in the specific situation.

The examples are taken from the best engineering literature, and from projects in which I were involved. Examples are an help to focus further the characteristics to consider. The analyst may:

- use an example directly, without changing a single word, if it is adequate for the system he or she is working on;
- use the example as a starting point, then modify it;
- use the examples as source of inspiration, in order to discover analogous requirements for his or her system.

Examples are mostly useful for non-functional requirements, because it is normal to find common characteristics in different application domains.

1.2 The classification scheme

There are many classification systems for software requirements. Many requirements authority have their own. (A list of classification schemes may be found in [Wie 2003].)

There is not any standard, and not a single right way to classify requirements:

- every classification scheme reflects a particular point of view, specific experiences, peculiar sensibilities;
- sometimes requirements pertain to more than a single semantic dimension.

I use here my own classification scheme, structured in eight main categories and with an acronym easy to remember (FOCUS-TBD):

Functionality
Operativeness
Compliance
Usability
Security

Time
Budget
Documentation, Maintenance and Support

In this guide, the classification scheme is simply a mean to organize questions and examples. Take it as it is, or change it in accord with your needs and opinions.

Some of the categories of this scheme (i.e. time, budget, compliance with standards and technological choices) are considered by many requirements engineering authors as “constraints”, not “requirements”.
I prefer not to use the term “constraint”. A constraint, in my opinion, is a requirement that you can not negotiate. In my experience, hypothetic constraints, when they conflict with other requirements more important in the eyes of the client, cease to be constraints, and they may be altered or deleted.

1.3 Criteria and measures for control

The level of precision needed for requirements varies in function of the criticality of the system. An high level of precision is obviously very useful for those who have the responsibility to implement and to test the requirement.

The section “Criteria and measures for control”, where present, gives some hints about how to be more detailed in the specification, without too much effort. A far more precise way to specify requirements, not covered in this guide, is the use of Planguage, a language invented by Tom Gilb to quantify qualitative statements [Gil 2005].

1.4 Attention: examples are powerful, but they must be used wisely

Examples are easy to pick and to adapt. But requirements may be hard and expensive to implement.
This is true in particular for non functional requirements, that have a major influence on the architecture of the system. It is easy to say “availability 24 x 365”, but the system

architecture has to allow it.
Novice analysts may think that a system could reach the maximum level of quality in every requirements category. Maximum performance, and maximum reliability, and maximum portability, and maximum usability, and maximum maintainability, and so on. But this is not the case. To maximize the level in a category may result in problems in other categories.
In general, be careful not to harden requirements more than your actual needs, because it could be dangerous or too expensive, and it may lead to conflicting requirements.

1.5 How to use this guide

In a medium-to-large project, with many stakeholders involved, the most economic and effective way to use the guide is during a requirements workshop. In a workshop, the product stakeholders may communicate effectively, share their opinions about requirements and priorities, discuss their different points of view. This sharing helps to consolidate common points and to make explicit conflicting views.

In a small project, the guide may be used by the analyst as a guideline for the conversations with the client of the project, and as a reminder for the various aspects to consider.

1.6 How to improve the guide

With questions, examples, qualification criteria and measures specific to products and projects of your organization.
If you have suggestions about how to improve the guide, please email me: adriano.comai@acm.org .

1.7 Thanks

To Alan Davis, Donald Firesmith, Soren Lauesen, Dean Leffingwell, James and Suzanne Robertson, Karl Wiegers, Stephen Withall for allowing me to use their examples in this guide.

1.8 References

- [Dav 1993] Alan Davis: Software Requirements. Objects, Functions, and States - Prentice Hall 1993
- [Fir 2004] Donald Firesmith: Engineering Safety Requirements, Safety Constraints, and Safety-Critical Requirements, in Journal of Object Technology, vol. 3, no. 3, March-April 2004, pp. 27-42.
- [Gil 2005] Tom Gilb: Competitive Engineering: A Handbook for Systems Engineering, Requirements Engineering, and Software Engineering Using Planguage - Elsevier Butterworth-Heinemann 2005
- [Lau 2002] Soren Lauesen, Software Requirements. Styles and Techniques - Addison Wesley 2002
- [Lef 2003] Dean Leffingwell, Don Widrig : Managing Software Requirements. A Use

Case Approach, 2nd Edition, Addison-Wesley 2003

- [Rob 2006] Suzanne Robertson, James Robertson: Mastering the Requirements Process, 2nd Edition, Addison-Wesley 2006 See also their Volere template, http://www.volere.co.uk/
- [Sei 2006] Rob Wojcik, Felix Bachmann, Len Bass, Paul Clements, Paulo Merson, Robert Nord, Bill Wood: Attribute-Driven Design (ADD), Version 2.0, Software Engineering Institute Technical Report CMU/SEI-2006-TR-023
- [Wie 2003] Karl E. Wiegers : Software Requirements, 2nd Edition, Microsoft Press 2003
- [Wit 2007] Stephen Withall: Software Requirement Patterns, Microsoft Press 2007

2 Functionality

2.1 Use Cases

Content

Usage scenarios of the system. For interactive systems, use case descriptions and scenario analysis are an effective way to discover functional requirements.

Questions

Who will use the system? (May be a user role, or an external system)

What does he or she (it, for an external system) need to do with our system?

In which operational scenarios of the system will the user (or external system) be involved?

Examples

Here I prefer not to give examples, because they would be too complex. Use cases are too broad a topic to deal with in this guide. In particular, they may be specified both in a very concise and in a very detailed form.

It may be useful to look to the various use case guidelines and templates available in books and on the web. A good source is: Alistair Cockburn: Writing Effective Use Cases, Addison-Wesley 2000.

2.2 Detailed Functional Requirements

Content

Requirements about what the system shall do. They specify what functions a system must provide to meet stated and implied stakeholder needs.

Functional requirements may be at a broad level or at a detailed level. They may be discovered with use cases description / scenario analysis - or with other techniques. This section may be used as a complement to the Use Cases section. Or it may be used as an alternative, for systems whose functionalities are not apt to be described with use cases.

Questions

What shall the system do?

Examples

The system shall allow payments by credit card.

For every payment by credit card the user has to give as input the expiration date of the card.

[Lau 2002] The product shall record companies as customers. When an employee from the company stays at the hotel, the company shall be the “guest” and be sent the invoice, but the name of the employee shall be recorded for the stay.

[Wit 2007] It shall be possible to have a set of pricing changes automatically applied at a predetermined date and time.

2.3 Data and Accuracy Requirements

Content

Requirements about data the system shall use. Also, requirements about the required level of accuracy of data.

Questions

Are there specific data sources the system shall use? Specific data destinations?

Is there a data model that specifies entities and relationships for this system?

Are there system wide data to be defined?

How shall [objects of a type, eg orders] be identified by users?

How long shall this kind of data be retained in data stores? How long should it be visible to users?

Which level of precision do we need to display times?

What is the right level of precision for [a data type]?

Examples

The user id for this system shall be the Social Security Number.

[Lau 2002] The name field shall have 150 characters.

[Lau 2002] Sensor data shall be stored with 14 bit accuracy, expanding to 18 bits in two years.

[Rob 2006] All monetary amounts shall be accurate to two decimal places.

[Rob 2006] Accuracy of road temperature readings shall be within ±2°C.

[Wit 2007] Each order shall be uniquely identified by an order ID that is in the form of the number of the customer that placed it plus an order number allocated sequentially for that customer, starting at one for the customer’s first order.

[Wit 2007] Customer orders shall be retained online for 90 days from the date the order was shipped. Orders shall not be visible to customers after this date, but they may continue to be stored online.

[Wit 2007] It shall be possible to create an off-line archive of all data belonging to a nominated company.

[Wit 2007] It shall be possible to designate which is the system wide local currency. This value cannot be changed after the system goes live.

2.4 Interoperability

Content

Requirements about relationships and interfaces of this system with external systems.

Questions

Which other systems shall interact with the system?

What are the interoperability characteristics of these systems?

Are there existing interfaces with this external system? Are these existing interfaces useful for our system needs? Should we use specific interaction modes?

Do we need to know what we sent, and we received, through an interface?

How easy should it be to change an interacting application with another one having similar functions?

Is necessary to define or use a standard interface to manage the relationships with alternative systems of the same kind?

Examples

[Wie 2003] The Chemical Tracking System shall be able to import any valid chemical structure from the ChemiDraw (version 2.3 or earlier) and Chem-Struct (version 5 or earlier) tools.

[Wit 2007] Every email sent by and every email received by the system shall be stored persistently.

2.5 Responsibility

Content

Requirements about who has the responsibility to perform specific system functions.

Questions

Who has the responsibility to do [a function]?

Examples

The payment approval shall be given by the supervisor.

3 Operativeness

3.1 Availability

Content

Requirements about the availability of the system for operational use.

Questions

When must the system be available for use? In which days? In which hours?

Are there specific time periods for which availability is mandatory to meet business goals?

Examples

The system shall be available 24 hours per day, 365 days per year.

The system shall be available every workday between 8:00 and 20:00 .

[Lef 2003] The system shall be available for use 99 percent of the time between 8 am and midnight.

[Sei 2006] The system shall recover from a processor crash within one second.

Criteria and Measures for Control

Fraction of time that the system is working.

3.2 Performance

Content

Requirements about the performances of the system. May vary for different functions, or typology of functions.

Questions

Which are the important functions from the point of view of performance?

Which is an allowable response time for this kind of system functions?

How quickly shall the system produce reports on request?

Examples

[Dav 1993] When the pilot pushes the “launch” button, the system shall launch the missile within one tenth of a second.

[Dav 1993] The system shall generate a “launch missile” command no sooner than 5 seconds after generating a “start battery warm-up” command.

[Lau 2002] Product shall detect speed violation and take photo within 0.5 seconds.

[Lau 2002] Scrolling one page up or down in a 200 page document shall take at most 1 s. Searching for a specific keyword shall take at most 5 s.

[Lau 2002] Product shall be able to process one alarm in 1 second, 1000 alarms in 5 seconds.

[Rob 2006] Any interface between a user and the automated system shall have a maximum response time of 2 seconds.

[Rob 2006] The response shall be fast enough to avoid interrupting the user’s flow of thought. (*Quantification*: The product shall respond in less than 1 second for 90 percent of the interrogations. No response shall take longer than 2.5 seconds.)

[Sei 2006] The system shall process sensor input within one second.

[Wie 2003] Every Web page shall download in 15 seconds or less over a 50 KBps modem connection.

Criteria and Measures for Control

Response time for a transaction: average, maximum.

3.3 Capacity

Content

Requirements about volumes and peaks the system must deal with, and about the consumption of scarce resources.

Questions

How many transactions per second shall the system manage?

How many [domain objects] shall be managed by the system in [time period]?

Are there peaks the system must manage? How long will peak periods last?

Is there a minimum number of concurrent users to deal with in any moment, in order to provide an acceptable service level?

What happens after the maximum capacity level is reached? Shall the system refuse the next concurrent user, degrade the level of service, or what?

Are there memory or bandwidth limitations to take into account?

What are the conditions or events that might lead to a service degradation?

Examples

[Dav 1993] The system shall handle up to and including twenty simultaneous users performing any activities without degradation of service below that defined in Section XYZ. Other systems may make short requests of this system at a maximum rate of 50 per hour and long requests at a maximum rate of 1 per hour, also without degradation.

[Rob 2006] The product shall cater for 300 simultaneous users within the period from 9:00 am. to 11:00 am. Maximum loading at other periods will be 150 simultaneous users.

[Sei 2006] The system shall have a maximum of 50% CPU utilization.

[Wie 2003] Ten percent of the available processor capacity and 15 percent of the available system memory shall be unused at the planned peak load conditions.

Criteria and Measures for Control

Volume of concurrent users or transactions the system shall manage.

3.4 Scalability

Content

Requirements about the expected growth in volumes that the system must be able to handle.

Questions

Are there forecasts or commitments about the growth in system usage? In data volumes?

Examples

[Rob 2006] The product shall be capable of processing the existing 100,000 customers. This number is expected to grow to 500,000 customers within three years.

[Wit 2007] The system shall be scalable to accommodate unrestricted growth in the number of customers (prospectively to several hundred thousands).

3.5 Reliability

Content

Requirements about the reliability of the system, which is the probability that the software executes without failure for a specific period of time.

The following list (by Alan Davis) can be used to assess the importance of reliability for the system. What may happen in case of malfunction?

- destroy all humankind
- destroy large numbers of human beings

- kill a few people
- injure people
- cause major financial loss
- cause major embarrassment
- cause minor financial loss
- cause mild inconvenience

Questions

Which is the level of reliability needed for this system?

What services can not be allowed to fail (e.g., authentication services)?

Which is an appropriate level of MTBF (Mean time between failures) for this system?

Which is the total allowable failure rate?

Which is an appropriate level of MTTR (Mean time to repair)?

Shall the system continue to operate when abnormal events happen?

Examples

[Dav 1993] No more than five bugs per 10K lines of executable code may be detected during integration and system testing.

[Dav 1993] The system shall exhibit a mean time between failures of no less than 6 months.

[Lef 2003] 90 percent of all system failures must be repairable within 5 minutes, and 99.9 percent of all failures must be repairable within 1 hour.

[Rob 2006] The product shall achieve 99 percent uptime.

[Rob 2006] The product shall provide 10 minutes of emergency operation should it become disconnected from the electricity source.

[Wie 2003] No more than five experimental runs out of 1000 can be lost because of software failures.

[Wie 2003] If the editor fails before the user saves the file, the editor shall be able to recover all changes made in the file being edited up to one minute prior to the failure the next time the same user starts the program.

Criteria and Measures for Control

Mean time between failures (MTBF) - may be specified in minutes, hours, days, months, years or similar time units.

Mean time to repair (MTTR) - may be specified as a range.

Maximum bugs, or defect rate - may be specified as bugs / lines of code, or bugs / function-point.

Time to re-start after system failure.

3.6 Installation

Content

Requirements about system installation and data migration.

Questions

Who should install the system? What level of skills should the installer have? How long should system installation take?

Is data conversion necessary? Must conversion procedures be written?

Is a manual backup needed before installation?

How should version upgrades be managed?

Is a parallel with the previous version of the system needed?

Examples

The system shall be distributed via web and installed with an executable setup.

[Rob 2006] The product shall be able to be installed by an untrained user without recourse to separately printed instructions.

[Wit 2007] Installing an upgrade shall not modify existing configuration values. An exception is made for any values that the new version uses in different ways from the previous version.

3.7 Portability

Content

Requirements about the portability of the system to other hardware and software environments.

Questions

Is the system expected to have a long life?

Shall the system be portable on other platforms? Which?

Shall other types of client devices access the system? Which?

Which versions of common web browsers shall the system support?

Examples

The system shall run on Vista, XP, and Linux.

Criteria and Measures for Control

Inclusion criteria or list of platforms on which the system shall be available.

4 Compliance

4.1 Laws and Regulations

Content

Requirements about the laws, national and international, to comply with. Take into account the different countries in which the system may be used. In case of doubt, ask the legal department of your organization, or a legal consultant.

Questions

Are there any national or international laws relevant for this type of system?

Which are the requirements that the law mandates?

Are there any copyrights that must be protected?

Examples

[Rob 2006] Personal information shall be implemented so as to comply with the Data Protection Act.

4.2 External and Internal Standards

Content

Requirements about the standards, external or internal to the organization, to comply with.

Questions

Are there any industry standard to comply with?

Are there any internal standard or guideline to comply with?

Does a new version of the standard substitute every older version, or the system should support multiple versions of the standard at the same time?

Examples

[Lef 2003] Use our C++ coding standards.

[Rob 2006] The product shall comply with MilSpec standards.

4.3 Audit

Content

Requirements about the audit controls needed for the system.

Questions

Are there audit rules to comply with?

Who will be allowed to see logs?

Examples

The system shall maintain a log of every transaction for 10 years.

[Wit 2007] Every event recorded shall store the date and time at which it occurred. The event date and time shall always be determined by the system itself; any date and time originating externally is to be considered unreliable for this purpose.

4.4 Business Rules

Content

Policies and procedures of the organization that define or constrain some aspect of the business, and are relevant for the system. Business rules typically exist before and independently from the system we have to develop.

Questions

Are there internal policies to comply with?

Examples

[Wie 2003] A user may request a chemical on the Level 1 hazard list only if he has had hazardous-chemical training within the past 12 months.

[Wie 2003] Commercial airline flight crews must receive at least eight hours of continuous rest in every 24-hour period.

[Wie 2003] The unit price is reduced by 10% for orders of 6 to 10 units, by 20% for orders of 11 to 20 units, and by 35% for orders of more than 20 units.

4.5 Technologies

Content

Requirements about technologies to use, or not to use. May be asked for by the client, or be a consequence of previous investments, or derive from strategic choices.

Questions

Do we need to constrain technological choices in order to be compatible with existing systems?

Do we need to use (or not to use) specific technologies?

Examples

[Lef 2003] The application must run on both our new and old platform.

[Lef 2003] The application must run on Windows XP.

[Wit 2007] The system shall use the Oracle database. For each installation, data shall reside in the same Oracle instance used by the company's other systems.

[Wit 2007] The user interface shall be Web based and all functions shall operate fully with all popular Web browser/operating system family combinations (up to a maximum of eight such combinations) as nominated by a designated person (who is expected to be the marketing manager).All versions of any such Web browser that have been the latest version at any time in the past two years shall be supported, except for any version that was superseded within a month of its release.

4.6 Cultural and Political Requirements

Content

Requirements about social factors that may influence the acceptability of the system. These requirements may be related to the user world (e.g. local cultural conventions) or to the internal politics of the organization.

Questions

Have intended users different cultural conventions from our own? Are there colors, icons, or words that have different meanings for them?

Are there internal conflicts that may affect the system characteristics?

Examples

[Rob 2006] The product shall not be offensive to religious or ethnic groups.

[Rob 2006] The product shall be installed using only American-made components.

5 Usability

5.1 Physical Environment

Content

Requirements about the physical environment in which the system will operate.

Questions

Which will be the normal usage environment?

Should the system operate also in other environments? Which ones?

Examples

[Rob 2006] The product shall be used by a worker, standing up, outside in cold, rainy conditions.

[Rob 2006] The product shall be able to fit in a pocket or purse.

5.2 Appearance and Style

Content

Requirements about the style and appearance of the system to its users. Style requirements are mostly marketing-driven. Detailed appearance requirements may be discovered through prototyping.

Questions

How shall the system appear to its users? How should it be perceived?

Must the system be compliant with corporate style guides, logos, etc.?

Which kind of packaging (if any) shall be used?

Which kind of advertising (if any) shall be used?

Examples

[Lau 2002] System shall follow style guide zz. Menus shall have at most three levels

[Rob 2006] The product shall be attractive to a teenage audience. (*Quantification*: A sampling of representative teenagers shall, without prompting or enticement, start using the product within four minutes of their first encounter with it.)

[Rob 2006] The product shall appear authoritative. (*Quantification*: After their first encounter with the product, 70 percent of representative potential customers shall agree they feel they can trust the product.)

[Wit 2007] All reports shall adhere to a standard layout, which includes headings and trailers (footers). This layout shall allow for branding by the company (logo, company name, and system name in headings).

[Wit 2007] All reports shall show on each page the page number. All reports shall show on each page the total number of pages in the report.

5.3 Ease of Use

Content

Requirements about the ergonomic properties of the system, that is the ease with which the system can be used by the intended users. It is useful to analyze these requirements from the perspective of all types of users, because the usability requirements may be different.

Questions

Which are the most complex functions to use? Of these, which are the most frequent?

How much the user is expected to remember from previous uses of the system?

How much feedback the user needs?

How shall error messages be managed?

Does the system allow to correct previous mistakes?

Shall the system allow to click technical terms to obtain an explanation?

Examples

The system shall allow the scaling of texts within the user's Web browser.

[Lau 2002] Novice users shall perform tasks Q and R in 15 minutes. Experienced users complete tasks Q, R, S in 2 minutes.

[Lau 2002] Three prototype versions shall be made and usability-tested during design.

[Lau 2002] The system must give understandable messages in text form in the event of errors, and instruct the user on what to do

[Lau 2002] It must be possible to review – and possibly edit – prints on the screen before they are printed on paper.

[Lef 2003] The new system shall be judged by 90 percent of the user community to be at least as usable as the existing XYZ system.

[Rob 2006] The product shall help the user to avoid making mistakes. (*Quantification*: One month's use of the product shall result in a total error rate of less than 1 percent.)

[Wie 2003] All functions on the File menu shall have shortcut keys defined that use the Control key pressed simultaneously with one other key. Menu commands that also appear on the Microsoft Word XP File menu shall use the same shortcut keys that Word uses.

[Wit 2007] At any point from which the user can initiate an operation that would take longer than 20 seconds (when using a 1Mbps Internet connection), a warning to this effect shall be displayed.

Criteria and Measures for Control

Average number of errors made by users in a given time period.

5.4 Personalization

Content

Requirements about the adaptability of the system to the personal preferences of each user, or, at a collective level, to the needs of specific organizations.

Questions

Shall the system be configurable by each user according to personal preferences?

Shall the user interface of the system be configurable?

Shall the system be configurable by each adopting organization?

Shall the notation for decimal points be configurable independently from the language?

Shall the duration of transaction logs be configurable?

Examples

The content of the personal area of the portal shall be configurable by the user.

[Rob 2006] The product shall retain the buyer's buying preferences.

5.5 Internationalization

Content

Requirements about the language support of the system. Internationalization is a specific and demanding form of personalization.

Questions

Which languages shall be used by the system?

Shall diverse character sets be used (e.g. Chinese, Arab, Cyrillic)?

Shall more than a language be displayed in each user interface?

Which format should be used to display dates? Shall the time zone be managed and displayed?

Examples

The system shall use in each user interface both German and Italian.

[Rob 2006] The system shall allow the user to select a language from this list [List of languages].

[Rob 2006] The user may switch language at any time during the normal usage of the system.

[Rob 2006] The product might eventually be sold in the Japanese market.

[Wit 2007] Whenever a time (or date-and-time) value is shown to a user, it shall be according to the user's designated time zone. This requirement does not apply to:

- Times obtained from an external source (for example, a data feed) from which the time zone cannot be discerned.
- Times for events that clearly occurred (or will occur) at a remote place, for which its local time zone may be used (for example, when inquiring on times of concerts in a selected city).

In circumstances where there is no known user, the system local time zone shall be used instead of the user's time zone. (For example, this may occur when the system automatically runs reports.)

A user's "designated time zone" is the one they have specified as a personal preference or, in the absence of that, the system's local time zone.

5.6 Learning Time

Content

Requirements about the time needed to learn to use the system. May vary from "zero time" (web sites) to months. It is useful to analyze these requirements from the perspective of all types of users, because the learning requirements may be different.

Questions

How much time is allowable before a user of type [specify user typology] can begin to use the system, performing simple functions?

How much time is allowable before a user of type [specify user typology] can master the system, performing complex functions?

Examples

[Rob 2006] The product shall be easy for an engineer to learn. (*Quantification*: An engineer shall produce a [specified result] within [specified time] of beginning to use the product, without needing to use the manual.)

Criteria and Measures for Control

Average time for novices to execute successfully a defined set of tasks.

5.7 *Accessibility*

Content

Requirements about the accessibility of the system for people with physical or cognitive disabilities. In some countries, may be regulated by laws.

Questions

Is there any law about accessibility to comply with, for this type of system?

Is there any standard about accessibility to comply with, for this type of system?

Which kind of disabilities shall be managed? May we try to list a priority sequence?

Examples

The system shall be usable by deaf people.

[Rob 2006] The product shall conform to the Americans with Disabilities Act.

[Wit 2007] The system's user interface shall avoid visual constructs that are apt to cause eye strain after several hours of continuous use. Such constructs include the flashing of visual objects, low contrast between adjacent objects (such as text and its background) and bright colors.

[Wit 2007] Color coding shall never be the only means of conveying any information, indicating an action, prompting a response, or distinguishing a visual element. That is, if all color were removed, the system could be used equally as well.

[Wit 2007] It shall be possible to perform all input to the system (including all navigation and initiation of operations) using the keyboard alone. An exception is granted for purely graphical operations in situations where using the keyboard is inherently impractical.

6 Safety and Security

6.1 Safety

Content

Requirements about the likely safety effects (loss, damage, or harm) of an improper usage of the system. Take into account the different countries in which the system may be used. In case of doubt, ask the legal and security departments of your organization, or a consultant.

Questions

Is this system safety-critical? May there be dangers to humans, or to properties, or to the environment?

Are there safety standards and laws relating to safety to comply with for this kind of systems?

Are there dangerous actions that may be prevented?

Examples

[Fir 2004] The petroleum refinery control system shall keep the pressures within reactant tanks at least 30% below their maximum pressure ratings at all times.

[Fir 2004] The automated airport subway system shall identify a combination of a train moving with its doors open with a probability of at least 99.99%.

[Rob 2006] The product shall not emit noxious gases that damage people’s health. (*Qualification*: The product shall be certified to comply with the Health Department’s standard E110-98. It is to be certified by qualified testing engineers.)

6.2 Access Protection

Content

Requirements about restrictions of access to the system. In case of doubt, ask the security department of your organization, or a security consultant.

Questions

Who has authorized access to the system?

In which situations the access is granted?

Is every function accessible to everybody?

Should we manage different user profiles?

Shall user sessions end after a given period of inactivity?

Examples

[Rob 2006] Only direct managers can see the personnel records of their staff.

[Wie 2003] Every user must change his initially assigned login password immediately after his first login. The initial password cannot be reused.

[Wit 2007] Places from which functions are normally selected (for example in menus and as buttons or hyperlinks) shall be display in a disabled manner those functions to which the current user does not have access.

[Wit 2007] The privileges a user has shall be determined by the roles to which the user is assigned. Privileges shall not be given directly to each user.

Criteria and Measures for Control

Time required to break into the system.

6.3 Integrity

Content

Requirements about the expected integrity level of the system. Also, requirements about protection of the system from viruses, spyware, trojans and similar threats.

Questions

Is it possible that a wrong behavior can compromise the integrity of the system?

What kind of external events could compromise the integrity? Shall a restore to previous state be available in every situation?

Which essential services could be significantly impacted by an attack?

What sensitive information must be protected?

Which measures shall be taken to prevent attacks from malicious software?

Examples

The system must be integrated with an updated antivirus software, to verify every attachment.

[Rob 2006] The product shall prevent incorrect data from being introduced.

6.4 *Privacy*

Content

Requirements about privacy of personal and organizational data. Take into account the different countries in which the system may be used. In case of doubt, ask the legal department of your organization, or a legal consultant.

Questions

Do we need to notify users about our management of their data?

Do we need to put cookies in the computers of our users?

Are there sensitive data?

May our customers view the personal data we manage about them? May they correct their personal data?

Do we need a particular care to protect some kinds of data from malicious access?

Examples

The system shall store information for only as long as it is needed for the stated purposes.

Usage data must be erased immediately after each session.

Users must be able to withdraw their consent to the processing of traffic and location data at any time.

The system shall require each individual's explicit, informed consent to collect or share his or her personal information.

The system shall provide a readily accessible mechanism for individuals to cause prompt cessation, including, when appropriate, the deletion of any personal information.

The system shall allow the user to inspect and make corrections to her or his stored personal information.

[Rob 2006] The product shall notify customers of changes to its information policy.

[Wie 2003] Only users who have Auditor access privileges shall be able to view customer transaction histories.

[Wit 2007] Passwords shall never be displayed on the screen when they are entered or at any other time.

7 Project Time Requirements

Content

Requirements about milestones and durations for the project.

Questions

When should the project end?

Do other business events depend on the delivery of the system in a particular date?

Is it possible an incremental delivery of system functions?

Are there opportunities to show a demo of system capabilities in advance? When?

Examples

The system must be completed in [date].

A demo of the system must be available in [date]

The software interfaces must be stabilized in [date].

8 Project Budget Requirements

Content

Requirements about economic resources available for the project.

Questions

Which is the budget for this project? Is it fixed?

Are there opportunities of incremental funding?

Examples

The development of the system may not cost more than [budget].

9 Documentation, Maintenance and Support

9.1 *Documentation*

Content

Requirements about the user documentation. It is useful to think from the perspective of all types of users, because the documentation requirements may be different.

Questions

Which kind of documentation would be better for the intended users? Which kind of documentation would be less expensive to produce?

Which kind of user documentation must at a minimum be provided?

In which languages must the user documentation be provided?

Who shall produce the documentation?

Who will be responsible for future updates to the documentation?

Examples

The installation package for the system shall include a user manual.

[Wit 2007] There shall be operating instructions that explain to an operator (system administrator) how to run the system. It shall cover at least startup, shutdown and the monitoring of the system’s health.

9.2 *Maintenance*

Content

Requirements about the maintenance needs of the system.

Questions

Who shall maintain the system? Will different parts of the system be maintained by different people?

Which kind of documentation will be needed for the maintenance?

Which are the likely types of upgrades or modifications to the system?

How easy should it be to correct a defect or modify the software?

Examples

[Lef 2003] Modifications to the system for a new set of withholding tax rates shall be accomplished by the team within 1 day of notification by the tax regulation authority.

[Rob 2006] New MIS reports must be available within one working week of the date when the requirements are agreed upon.

[Wie 2003] A maintenance programmer who has at least six months of experience supporting this product shall be able to make a new hardcopy output device available to the product, including code modifications and testing, with no more than one hour of labor.

[Wit 2007] Whenever the credit reference agency releases a new version of the interface to its credit reference service, the system shall be amended to use the new version. Amendments of this nature shall be installed within three months of the new version being released.

Criteria and Measures for Control

Average time from report to closure of defects.

Average time from an update request to the availability of an updated version of the system.

9.3 Support

Content

Requirements about the kind and the level of support needed for the system.

Questions

Which kind of support is required?

Shall different levels of support be available?

Which documentation will be available for the support group?

Examples

Support shall be provided via help desk.

9.4 Training

Content

Requirements about the user training needed in order to use effectively the system. It is useful to look from the perspective of all types of users, because the training needs may be different.

Questions

What training will be necessary?

Who will design the training?

Who will provide the training?

Examples

The initial training shall be designed and provided by the developers to the support group.

The support group will train end users.