FAIRSNET

IST-2001-34290

On-line Solutions for Trade Fairs

Deliverable D5

FairsNet Evaluation

Covering period 1.3.2003-1.3.2004

Report Version: Version 1

Report Preparation Date:

Classification:

Contract Start Date: 1.8.2002 Duration:

Project Co-ordinator: Ruben Riestra Cherbavaz

Partners:

Authors: Maria Francesca Costabile and Rosa Lanzilotti

Project funded by the European Community under the "Information Society Technologies" Programme (1998-2002)

FairsNet Evaluation

Table of Contents

Management summary *4*

1 Introduction *4*

2 Approach to evaluation and assessment *4*

2.1 User-Centered Design 4

2.2 Participatory Design 5

2.3 Analysis of stakeholders needs 7

2.4 Usability of interactive systems 7

2.5 Usability Evaluation in the Software Life Cycle 8

2.6 Evaluation Process 10

3 Evaluation criteria for FairsNet *11*

4 Evaluation in FairsNet life cycle *12*

5 User Evaluation of the trial *17*

5.1 FairsNet Authoring Tools 18

5.2 Plan a Trade Fair with FairsNet Authoring Tools 19

5.3 Evaluation of the trial 20

6 Discussion *24*

References *24*

Appendix A: Results of inspection of FairsNet Authoring Tools *26*

List of Figures and Tables

Figure 1: ISO 13407: Human-Centered Design process for interactive systems 5

Figure 2: The Star Life Cycle Model 5

Figure 3: Usability Definition about Nielsen. 7

Figure 4: Agrilevante trade fair: an example of query preview along Fair sector, Geographic area, and Requested services attributes. 14

Figure 5: A visualization for Agrilevante trade fair data using DaeTL. 15

Figure 6: Data set sorted according to the column the Geographic area (“AreaGeografica”) 16

Figure 7: The red square represents the data set with the companies that have more that 3 billions (ITL) turnover. 16

Figure 8: Example of the results of DaeTL on the data of Semana Verde 2002 fair 17

Table 1: Questionnaire submitted to power users involved in the trial. 23

Management summary

To be completed by Ruben, possibly taking some part of the introduction.

1 Introduction

FairsNet is a system that has the objective to offer innovative services on the web to support several activities of the organization and management of fairs.

One of the significant aspects of the project and of the development of the FairsNet system has been the use of a User-Centered methodology, which has the goal of designing and implementing a usable system focused upon the specific needs of the users. Following this methodology, the system final users and their business processes have been carefully analyzed. Moreover, during the system development, users interacted with project team according to a Participatory Design approach, as it is described in this document.

The objective of this document is to report about the work performed within WP4. Specifically, we have identified the monitoring and evaluation procedures that we have applied during the development of FairsNet, in accordance with the User-Centered design methodology that we have adopted. The report also describes the evaluation of the trial that has been performed.

2 Approach to evaluation and assessment

FairsNet has been developed by using a User-Centered Design methodology. As a consequence, the main key points have been the analysis of all possible users of the system and, in general, of all the stakeholders, as well as the evaluation process that involved all phases of the system design and development. In this section, we briefly illustrate the user-centred methodology, and we describe our approach to the evaluation of FairsNet from the point of view of the users.

2.1 User-Centered Design

User-Centered Design implies that final users are involved from the very beginning of the planning stage, and identifying user requirements becomes a fundamental and crucial phase to any development project. Early involvement of users has the potential for preventing serious mistakes and for identifying what is effectively needed for them [Pre,1994]. Poor or inadequate requirement specifications can result in interaction difficulties, including lack of facilities and usability problems.

The basic principles of User-Centered Design are:

1. analyze users and tasks
2. design and implement the system iteratively through prototypes of increasing complexity
3. evaluate design choices and prototypes with users

User-Centered approach requires understanding reality: who will use the system, where, how, and to do what. Then, the system is developed iterating a design-implementation-evaluation cycle. In this way it is possible to avoid serious mistakes and to save re-implementation time, since the first design is based on empirical knowledge of user behaviour, needs, and expectations.

In the traditional software life cycle, the standard waterfall model, which is system-centered, usability is not adequately addressed. Moreover, there are some significant drawbacks. For instance, the system is tested only at the end of the cycle, when unfortunately is too late for going through radical design modifications to cope with possible discrepancies with the requirements. Another problem is that these requirements are collected with customers, who often are different from the people who will use the system. Customers are the people who negotiate with designers the features of the intended system, while users or end users are those people that will actually use the designed systems [Cos, 2001]. A direct consequence of the restricted nature of the requirement specifications is that, usually, system testing is not only performed late in the development cycle, but is also limited to some of its functional aspects, thus neglecting system usability.

In order to create usable interactive systems, it is therefore necessary to augment the standard life cycle to explicitly address usability issues: the User-Centered Design methodology stresses the iteration of the design-implementation-evaluation cycle.

The key principles of the User Centered Design methodology, namely the focus on users, the task they perform and the context in which they work, and the iterative development through prototypes of increasing complexity that are evaluated with the users, have been captured in the standard ISO 13407 standard (Human-Centered Design process for interactive systems), that is shown in Fig. 1 [ISO, 1998].

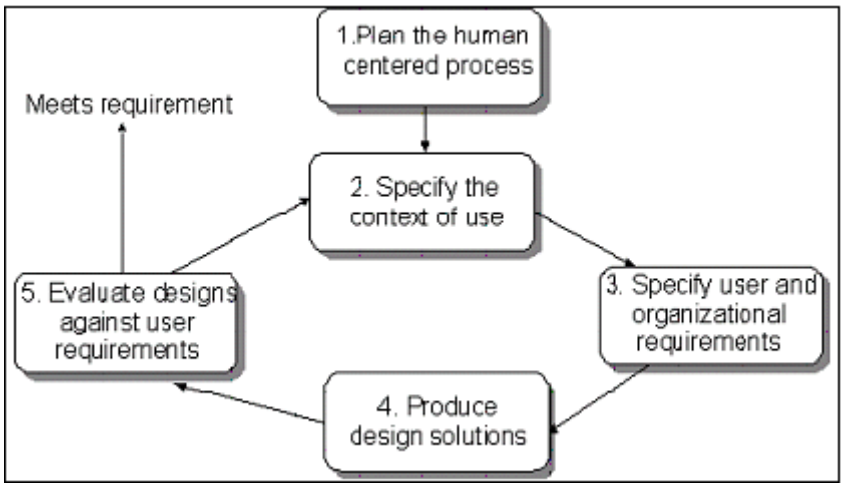

Figure 1: ISO 13407: Human-Centered Design process for interactive systems

The design solutions mentioned in block 4 of Figure 1 are implemented through prototypes that are evaluated, and if they do not meet the specified requirements, the process is iterated and goes again through a revision of the specifications and a proposal of a new prototype. The iterative process is stopped when requirements are met.

From this it follows that evaluation represents the central phase in the development cycle. For this reason, within the Human-Computer Interaction (HCI) community, Hartson and Hix have developed the star life cycle model shown in Figure 2 [Har, 1993].

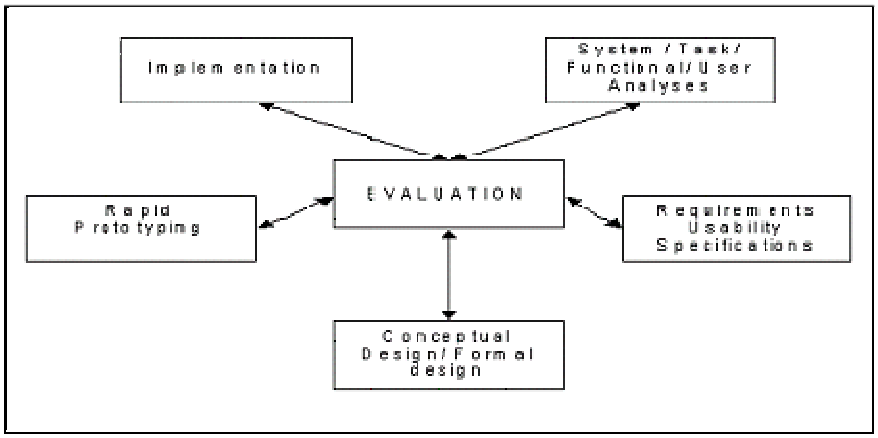

Figure 2: The Star Life Cycle Model

The star model recognizes that this approach needs to be complemented by a bottom-up (synthetic) approach, and can start from any point in the star (as shown by the entry arrows), and followed by any other stage (as shown by the double arrows).

In line with this model the FAIRSNET project has performed various type of evaluation during the project and development phases of FairsNet as illustrated in this document.

2.2 Participatory Design

Participatory Design (known as the Scandinavian approach) [Adl, 1992] acknowledges the importance of involving users in the design process and, indeed, argues that they have a right to be involved in the design of the systems which they will subsequently use. Users participate by analyzing organizational requirements and considering appropriate social and technical structures to support both individual and organizational needs.

Participatory Design is a philosophy, which encompasses the whole design cycle, and incorporates the user not only as an experimental subject but also as a member of the design team. Users are therefore active collaborators in the design process, rather than passive participants whose involvement is entirely governed by the designer. The argument is that the users are experts in the work context and a design can only be effective within that context if these experts are allowed to contribute actively to the design.

Participatory Design therefore aims to refine system requirements iteratively through a design process in which the user is actively involved, being included in the design team, in order that they contribute to every stage of the design process.

Participatory Design encompasses several methods to facilitate data exchange between the users and the designers. In FairsNet, we have adopted the following:

- Brainstorming
- Storyboarding
- Workshops
- Pencil and paper exercises

Brainstorming

Brainstorming is a technique designed to help creative thinking in initial product development. A large number of ideas are generated, many of which are discarded. In the process it is hoped that innovative ideas will arise that can then be followed up in more detail. The number of people in a brainstorming group can range from 2 - 12 people. The members of the group should have a range of experiences and stakes in the problem to be solved. Hence the group should not only consist of experts, but also lay people and a range of people in between. They should all have some familiarity of the problem preferably from different viewpoints. Brainstorming sessions are relatively easy to run, but do need to be handled carefully. The group needs to be carefully managed so that all participants can contribute without being criticized, but it is necessary to have one leader or chairperson to present the problem. It is also important to keep the conversation from straying too far and to prevent any criticism between the group members, as this will halt the flow of creativity.

Storyboarding

Storyboarding can be used as a means of describing the user's day-to-day activities as well as the potential designs and the impact they will have.

Workshops

Workshop can be used to fill in the missing knowledge of participants and provide a more focused view of the design. They may involve mutual enquiry in which all parties attempt to understand the context of the design from each other's point of view. The designer questions the user about the work environment in which the design is to be used, and the user can query the designer on the technology and capabilities that may be available. This establishes common ground between the user and designer and sets the foundation for the design that is to be produced.

Pencil and paper exercises

Pencil and paper exercises allow designs to be talked through and evaluated with very little commitment in terms of resources. Users can walk through typical tasks using paper mock-ups of the system design. This is intended to show up discrepancies between the user's requirements and the actual design as proposed. Such exercises provide a simple and cheap technique for early assessment of models.

The predominant activity in designing systems like FairsNet is that the participants in the design team teach and instruct each other; since domain experts (in this case fair experts) understand the practice and system designers know the technology. The knowledge relevant to the problem is distributed and can be opinionated - it must be acquired and formalized with a careful user requirements analysis.

2.3 Analysis of stakeholders needs

Stakeholders are persons, group actors, agents, interested parties, or institutions with interests in a project. Different group of stakeholders can be distinguish according to:

- their position in relation to the project, i.e. internal stakeholders (e.g. partners in a project) and external project (e.g. local policy makers)
- their involvement and their role in the project, i.e. primary stakeholders are role occupants (e.g. the project manager, the commissioner, the end user) and secondary stakeholders (e.g. technology providers).

Primary stakeholders are those ultimately affected. Secondary stakeholders are the intermediaries in the process. This definition of stakeholders includes those involved or excluded from decision-making processes.

Stakeholders analysis is the identification of a project's key stakeholders, an assessment of their interests, and the ways in which these interests, and the way in which these interests affect project viability. It is linked to both institutional appraisal and social analysis: drawing on the information deriving from those approaches, but also contributing to the combining of such data in a single framework. Stakeholders analysis contributes to project design through the logical framework, and by helping to identify appropriate forms of stakeholders participation.

2.4 Usability of interactive systems

User-Centered approach and Participatory design were exploited in order to design FairsNet as a usable system. It is now widely acknowledged that usability is a crucial factor of the overall quality of interactive applications. Several definitions of usability have been proposed. Nielsen defined a model in which usability is presented as one of the aspects that characterizes a global feature of a system that is acceptability by the users, reflecting whether the system is good enough to satisfy needs and requirements of the users.

The *acceptability* of a computer system is a combination of its social acceptability and its practical acceptability [Nie, 1993], as shown in Figure 3 its practical acceptability is analyzed within various categories, including traditional categories such as cost, support, reliability, compatibility with existing systems, etc., as well as the category of usefulness. Usefulness can be broken down into the two categories of utility and usability, where utility is the question of whether the functionality of the system in principle can do what is needed, and usability is the question of how well users can use the functionality.

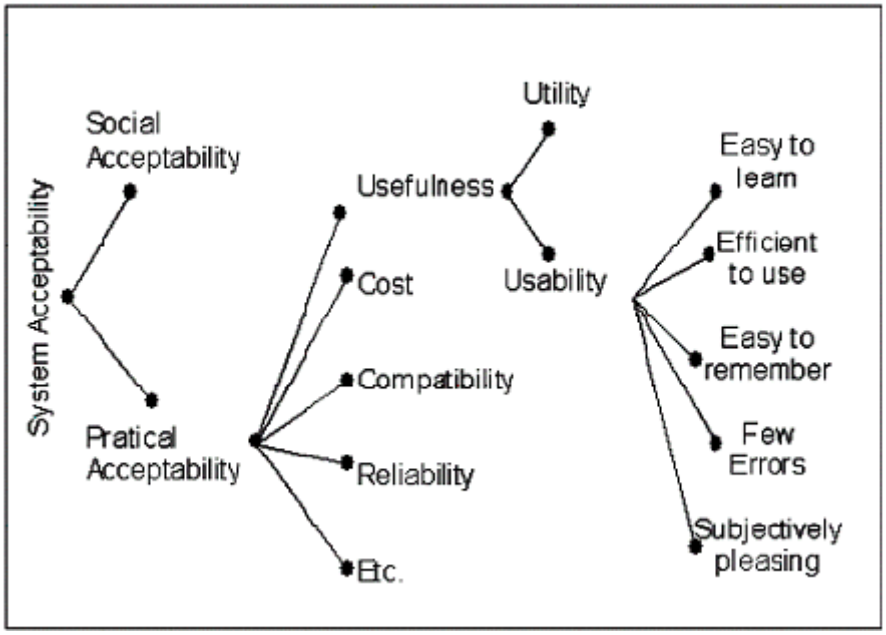

Figure 3: Usability Definition about Nielsen.

In Nielsen model, usability is not a one-dimensional property of a system, rather it has multiple components. It can be decomposed into five attributes:

- *Learnability*: the system should be easy to learn so that the user can rapidly start getting some work done with the system.
- *Efficiency*: the system should be efficient to use, so that once the user has learned the system, a high level of productivity is possible.
- *Memorability*: the system should be easy to remember, so that the casual user is able to return to the system after some period of not having used it, without having to learn everything all over again.
- *Few Errors*: the system should have a low error rate, so that users make few errors during the use of the system, and so that if they do make errors they can easily recover from them.
- *Satisfaction*: the system should be pleasant to use, so that users are subjectively satisfied when using it; they like it.

Only by defining the abstract concept of "usability" in terms of these more precise and measurable components, we can arrive at an engineering discipline where usability is not just argued about but is systematically approached, improved, and evaluated.

Different authors in the Human Computer Interaction (HCI) literature have proposed different usability principles for interactive applications. Nielsen [Nie, 1993] reports the following usability principles:

1. *Simple and natural dialogue*: Dialogue should not contain information that is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility. All information should appear in a natural and logic order.
2. *Speak the user's language*: Dialogue should be expressed clearly in words, phrases and concepts familiar to the user.
3. *Minimize use memory load*: The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be always visible or simply retrievable when it is necessary.
4. *Consistency*: Users should not have to wonder whether different words, situations, or actions should have always the same meaning.
5. *Feedback*: The system should inform the user about what he is doing by means appropriate, effective and efficiency feedback.
6. *Clearly marked exits*: Users often choose functions mistakenly and then they need simple to leave the unintentional status.
7. *Shortcut*: Accelerator, unseen by the novice user, may often speed up the interaction for expert user such that the system caters to both inexperienced and experienced users.
8. *Good error messages*: Messages should be expressed in a easy language (no codes), should indicate the problem and should suggest the solution in a constructive way.
9. *Prevent errors*: Even better than good error messages is a careful design that prevents a problem from occurring in the first page.
10. *Help and documentation*: Even better it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the users´ task, and it should specify the concrete steps to execute.

2.5 Usability Evaluation in the Software Life Cycle

In order to design usable systems, we have seen that in User-Centered design usability evaluation plays a fundamental role. The HCI research has provided several methods that can help the designers in taking their decisions during the different stage of the development of a usable system.

Many different techniques can be applied for collecting user information, among them *direct and indirect observation, interviews* and *questionnaires* [Hac, 1998, Dix, 1998, Pre, 1994]. Direct and indirect observation means observing the users while they carry out their tasks at their workplace. It is the most reliable and precise method for collecting data about users, especially valuable for identifying user classes and related tasks.

Moreover, it allows identifying critical factors, like social pressure, that can have a strong effect on user behaviour when the system will be used in the field. Unfortunately, direct observation is very expensive because it requires experimenters to observe each user individually. It can also lead to a level of 'artificial' behaviour, as those being observed react to the observation. For these this reasons, it is most useful when a reduced number of observations is enough to generalize behavioural predictions or when hypotheses have to be tested rather than generated. Interviews collect self-reported experience, opinion, and behavioural motivations. They are essential to gaining finding out procedural knowledge as well as problems with currently used tools. Interviews cost a bit less than direct observations, because they can be shorter and easier and quicker to document to code. However, they still require skilled experimenters interviewers to be effective. By contrast, self-administered questionnaires can be handed out and collected by untrained personnel allowing to the gathering from various users of a large huge quantity of data at relatively low cost. They allow statistical analyses and stronger generalizations than interviews but also lack the flexibility of questioning. Questionnaires provide an overview on the current situation as well as specific answers. They can be readily produced and distributed to reach as wide an audience as possible bur need to be carefully constructed as not to predetermine the answers given to them.

Which combination of these methods is best worth to applying depends both on requirements and budget. By elaborating the outcome of the knowledge phase, designers define a first version of the system. At this stage, design techniques (e.g., task-centered [Pat, 2001] or scenario-based [Pre, 1994]) provide satisfying solutions. The goal is to explore different design alternatives before settling on a single proposal to be further developed. Possibly, in this way designers will propose different solutions and different interaction strategies. Techniques like such as paper mock-ups and prototyping can be applied.

Paper mock-ups are the cheapest: pieces of the system interface are drawn on paper and an interviewer experimenter simulates the interaction with a user. Despite its simple trivial appearance, this technique allows for the collecting of reliable data, which can be used for parallel reviewing.

Prototyping allows testing some functionality in depth (vertical prototyping) or the whole interface (horizontal prototyping). Then, one or more solutions can be evaluated with or without users. This step, called formative evaluation, aims at checking some choices and getting hints for revising the design. Different methods can be used for evaluating systems at the different phases of their development: the most commonly adopted are *user-based methods* and *inspection methods.*

User-based methods mainly consist of user testing, in which usability properties are assessed by observing how the system, or a prototype of the system, is actually used by some representatives of real users performing real tasks [Dix, 1998, Pre, 1994, Whi, 1988]. Usability inspection methods involve expert evaluators only, who inspect the user interface in order to find out possible usability problems, provide judgements based on their knowledge, and make recommendations for fixing the problems and improving the usability of the application [Nie, 1994].

User-based evaluation provides a sounder trusty evaluation from the user perspective, because it assesses usability through samples of real users. However, it has a number of drawbacks, such as the difficulty to properly select a correct sample of the user community, and to train it to manage not only the main application features but also the most sophisticated and advanced facilities of an interactive system.

With respect to user-based evaluation, usability inspection methods are more subjective, having heavy dependence upon the inspector skills and preconceptions. Among the inspection methods, we may include:

- heuristic evaluation,
- cognitive walkthrough,
- formal usability inspection, guidelines reviews [Nie, 1994].

Heuristic evaluation is the most informal method; it involves a usability expert who analyses the dialogue elements of the user interface to check if they conform to usability principles, usually referred as heuristics, hence the name of this method. In a cognitive walkthrough, the expert uses some detailed procedures to simulate users' problem solving processes during the user-computer dialogue, in order to see if the functionalities provided by the system are efficient for users and lead the to correct actions.

Formal usability inspection is a review of users' potential task performance with a product. It was designed to help engineers to review a product and identify any find a large number of usability defects. It is very similar to the traditional 'code inspection' methods with which software developers have long been are familiar. It is carried out by the engineer designing the product and a team of peers, looking for defects. Finally, in a guidelines review, the

experts inspects the interface to check if it is conforms to a list set of usability guidelines. The method can be considered as a cross between heuristic evaluation and standard inspection, the latter is another kind of inspection to check the compliance of the interface to some interface standards. A detailed description of these and other inspection methods can be found is in [Nie, 1994].

The main advantage of inspection methods is however the cost saving: they do not involve users nor require any special equipment or lab facilities [Nie, 1993, Nie, 1994]. In addition, experts can detect a wide range of problems and possible faults of a complex system in a limited amount of time. For these reasons, inspection methods have achieved widespread use in recent the last years, especially in the industrial environments [Nie, 1994a], since industry is very much interested in effective and formalised methods, that can provide good results whilst being still cost-effective and easily operated.

Inspection methods aim at finding usability problems in an existing user interface, and to then make recommendations for fixing such these problems. Hence, they can be applied at various steps of the software development, and are appropriate certainly used for evaluating the design of the system in a prototype form, even a paper prototype, so that possible defects can be fixed as soon as possible.

When a system implementation is available, user-based evaluation is often recommended. It includes *experimental methods, observational methods,* and *survey techniques*. Among experimental methods, controlled experiments are very valuable; they provide empirical evidence to support specific hypotheses. They allow a comparative evaluation, which is very useful when alternative prototypes or versions of the same system are available. An experiment consists of the following steps: formulation of the hypotheses to be tested, definition of the experimental conditions that differ only in the values of some controlled variables, execution of the experiment, and analysis of collected data.

In order to verify the usability of a single prototype, we can also observe users working with it. A valid technique is the thinking aloud, in which users are asked to think aloud when they use the system or prototype. In this way, evaluators can detect users' misconceptions and the system elements that cause them.

Both experimental and observational methods are used for collecting data about system and user performance; they do not provide data about users' satisfaction that is a subjective measure that can be obtained by survey techniques, such as interviews and questionnaires [Dix, 1998, Pre, 1994].

By considering the industry's interest for cost effective heap but effective methods, heuristic evaluation plays an important role. It prescribes having a small set of experts analyzing the system, and evaluating its interface against a list of recognized usability principles, the heuristics. Some researches have shown that heuristic evaluation is a very efficient usability engineering technique [Jef, 1992], with a high benefit cost-ratio [Nie, 1994a], and therefore it falls within the so-called discount usability methods.

In principle, only one evaluator can conduct heuristic evaluation. However, in an analysis of various studies, it has been assessed that single evaluators are able to find only the 35% of the total number of the existent usability problems [Nie, 1993, Nie, 1994]. Different evaluators tend to find different problems. Therefore, the more experts that are involved in the evaluation, the more problems it is possible to find. The mathematical model defined in [Nie, 1994a] shows that reasonable results can be obtained by having only five evaluators.

2.6 Evaluation Process

Evaluation is not a single phase of the development process of an application, but is an iterative set of processes deployed across a set of prototypes. This has the advantages that problems can be identified as early as possible and can then be corrected easier and cost-effectively.

FairsNet has adopted evaluation techniques that are comprehensive and cost- effective. Specifically, the evaluation process is based on the use of the following techniques:

- user and task observation
- scenarios
- simplified thinking aloud
- heuristic evaluation.

User and task observation

The first step for designing a usable system is to know who will use it. There are different users; so it is important to analyse them. We can use different query techniques: *direct and indirect observation*, *interviews, questionnaires*, which have been described in the previous section.

Scenarios

Scenarios is a "story about use". Stories can be of different lengths and different levels of detail, and, indeed, the word "scenario" is used in many different ways in the literature on user and task analysis.

Scenarios can be about users, their work, their environments, how they do tasks, the tasks they need to do, and all combinations of these elements.

Scenarios can focus on the primary users- the people who will actually use what you develop- or the secondary users- the people who benefit by what the primary users do.

We distinguish four types of scenarios that vary in their level of detail and the use the development team might make of them. We include:

a. *Brief scenarios* are very brief stories that give just the facts of a real situation the primary user had to deal with, but that don't go into detail on how the user does the task.
b. *Vignettes* Brief narratives, sometimes with figures that give readers a high-level, broad brush view of a user, the user's environment, and the user's current way of doing something.
c. *Elaborated scenarios* are narratives with more details. Which details you focus on depend on what you want the team to take from the story.
d. *Complete task scenarios* are narratives that carry the story from the beginning to the end of a task or sequence of tasks.

Simplified thinking aloud

The thinking- aloud method involves having one test user at a time use the system (or a prototype) for a given set of tasks while being asked to "think out loud". By verbalizing their thoughts, users allow an observer to determine not just what they are doing with the interface, but also why they are doing it.

Traditionally, thinking-aloud studies are conducted with psychologists or user interface experts as experimenters who videotape the subjects and perform detailed protocol analysis. A major difference between simplified and traditional thinking- aloud is that data analysis can be done on the basis of the notes taken by the experimenter instead of by videotapes. Recording, watching, and analysing the videotapes is expensive and takes a lot of time.

Heuristic evaluation has been described in the previous section.

In FairsNet, we have used the above usability methods for the evaluating prototypes at various stage of the development, as described in Section 4.

3 Evaluation criteria for FairsNet

The evaluation criteria in this section are used in the overall evaluation process of FairsNet. We focused on evaluation criteria from the user's view. The criteria are divided in: General Evaluation criteria, User Interface criteria, and Evaluation criteria for the Authoring Process and the Authoring Tool.

General evaluation criteria:

G1. The Authoring Tools should support the power users in each their task.
G2. The Authoring Tools should make cost-effective development possible.
G3. The Authoring Tools should relieve authors and users from the low-level activities necessary to drive the application.
G4. The Authoring Tools should enhance interaction, navigation, positioning and orientation, as well as intuitiveness of the application.
G5. The power users have to use FairsNet Authoring Tools in a simple way. Their usage must be efficient after the team has learned to use the tools.
G6. An expert in the application field who is not familiar with programming has to be able to create the application by using the Authoring Tools.
G7. Application maintenance has to be easy.

User Interface criteria:

UI1. Creating the objects of the interface with the Authoring Tools should be fast.
UI2. The source code should be invisible for the user of the toolkit: the power user can change some parameters that are easily changeable, but does not want to change the source code.
UI3. Functions (of the same authoring tool) should be combined into one tool, so the user doesn't need to jump from one authoring tool to another all the time.
UI4. Changing component parameters must be easy.
UI5. Previews must be available.

Evaluation criteria for the authoring process and the authoring tool:

A1. The Authoring Tools should be:
  - intuitive
  - interactive
  - able to deal with navigational, orientation and positioning issues in intuitive terms, to give previews
  - they should offer intuitive and easy-to-use metaphors and interfaces to the developer and to a certain extent to the user with less expertise

A2. Customisation has to suit the needs of the power users and the application context.
A3. With the Authoring Tools, FairsNet elements can be easily customized by power users.
A4. The Authoring Tools must offer the possibility to adapt and combine the elements that have been plugged into.
A5. The Authoring Tools have to abstract the programming processes by defining the interfaces and basic behaviour of elements.
A6. Power users can select interface components from the components library to include them in a new application.
A7. Power users are able to efficiently create and organise applications with the existing components.
A8. The power users have to be able to integrate the interface components by using only the objects available in the application interface.
A9. Authoring tools should be flexible.
A10. Authoring tools must give the possibility to identify interface components, to access them, modify, adapt and combine them among each other.

4 Evaluation in FairsNet life cycle

Evaluation is not just something that happens in the delivery phase. As described above, it occurs in some form at all stages in the software cycle: analysis, design, development and delivery. A specific activity we have performed in WP4 was related to designing and implementing a monitoring system that, through the evaluation techniques described in Section 2.6 and applied during the whole life-cycle, could provide information about the quality of the developed system. The main focus has been placed upon the evaluation at an operational level, namely on issue concerning the quality of the modules developed with the FairsNet authoring tools, their usability and ease of use, the drawbacks and any problems, as well as the strong points and benefits of each module.

The evaluation criteria illustrated in Section 3 were taken into account in all the evaluation activity. The main aim of the evaluation was to identify and solve problems that could interfere with such criteria.

FairsNet system has been design according to user-centered design, as we have described above, in order to develop a system that was effective, efficient, and used by the end user with satisfaction. So, in each phase of the software process the end user has been involved in various ways and for different purposes.

In analysis phase, questionnaires, interviews, user observations, study of existing literature and documents have been used to collect information in a systematic way. This activity has been extensively documented in the first deliverable of Fairwis Project.

During the initial phases of the project, prototypes of various types and complexity have been evaluated in order to explore different design alternatives. The goal was to individuate the best interface to be further developed. Various techniques have been used: scenarios, heuristic evaluation, interviews, and simplified thinking aloud.

Some meetings have been organized among the FairsNet partners in order to evaluate prototypes and solve problems that came out in developing the FairsNet system. In particular, the members of the University of Bari,

FSGV, and University of Rome LUISS met on 3-4 February 2003 in Rome to examine some FairsNet prototypes together with user.

Moreover, various heuristic evaluations have been performed at various stages of the development. As an example, in Appendix A, we report the results of the heuristic evaluation of some FairsNet Authoring Tools performed by the University of Bari team during Spring 2003.

During 8-10 July 2003, a meeting at Fraunhofer-IPSI has been organized in order to perform usability studies of FairsNet components and discuss the results with the developers.

One of FairsNet components is DAE, the Data Analysis Engine that is exploited primarily in the post fair phase, in order to give the organisers various possibilities of analysing data of the fairs, related to exhibitors, professional visitors, etc. Such analyses will be also valuable for the organisers in order to plan next editions of the fair.

As described in D1 and D2, DAE offers various tools for analysing data. A video, which is part of the documentation provided for the project and is accessible from D2, is useful to understand all the possibilities offered by such tools. During the FairsNet project we have tested the tools with data related to different fairs. We have also evaluated these tools with users and we report here the results. These trials were not explicitly planned in the FairsNet project, but we felt they were important for the overall system evaluation activity.

One of the first data set provided by the organizer of Agrilevante, a fair on agriculture organised at Fiera del Levante in Bari, Italy, every September. We then showed to the users the possibility they have to analyse these data through the DAE tools. Users were two people of the management of Agrilevante, involved in the fair organization. As an example of use of the tool DAEQP, we presented to the users the following scenario: the organiser of Agrilevante wants to perform a segmentation of the exhibitors of the last edition of the fair. Let us suppose that the organiser wants to find out which were the most requested services at the fair, which exhibitors requested them, and so on. The objective of the organizer is to increase the fair income by selling more services. Therefore s/he is interested in selecting exhibitor segments for starting appropriate marketing campaign promoting the fair services.

In order to help the user (in this case the organiser) in his/her analysis, DAEQP is able to visualize an overview in which data are visualized along some major attributes. After accessing the system, the user is first asked to select three major attributes among those attributes considered in the database and shown on the screen. Let us suppose that the user selected Fair sector, Geographic area, and Requested services; the resulting overview is shown in Figure 4.

The user immediately gets a lot of information, for example no company comes from Central America (“America Centrale”) so that it is useless to perform any further query with Central America as Geographic area since it will return an empty data set. The user also sees that 16 exhibitors come from the Middle of Italy (“Italia Centrale”), and so on. The interface allows the user to perform data previews, for example, by clicking on the value Middle of Italy (“Italia Centrale”), only the records with this attribute value are selected and the number of retrieved data is updated consequently. More specifically, as a result of this query, the 16 records of companies coming from the middle of Italy will be selected and the bar at the bottom left of Figure 4 will be update to show 16 out of 163 records. The values of attributes Fair sector and Requested services will also be updated to indicate only the values related to the selected 16 companies.

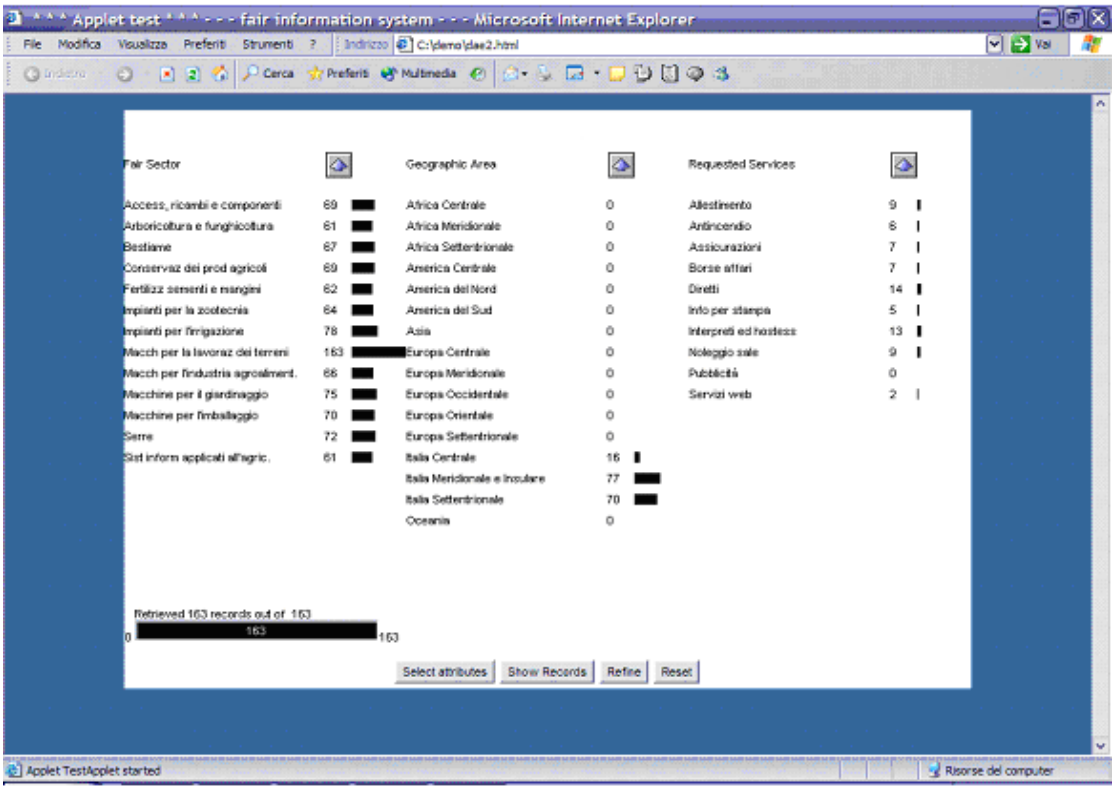

Figure 4: Agrilevante trade fair: an example of query preview along Fair sector, Geographic area, and Requested services attributes.

This visualization, called Query Preview [Shn, 1999], provides the possibility of sorting the shown elements by clicking on the icon on top of the attribute values. If the query preview shows too many records, the user may further reduce the selected data set. In this case, the system allows users to refine the query by clicking on the button “Refine”. The query refinement phase supports dynamic queries over the other relevant attributes of the database. In this way, the user can see more details of the dataset retrieved in the first phase, and get the reduced set of data he is actually interested in. For example, the user can specify further attributes, such as the company income, or s/he can zoom on the selected value of an attribute shown in the first preview.

The other tool we tested in that trial was DaeTL. It is also useful for identifying important correlations among data. A possible visualization shown by this tool is the one depicted in Figure 5. In each column, values of the attribute indicated in the column header are visualized. A global view on the data is thus provided, but the tool also permits to see some details without loosing the overall context. In the Figure 5, details about two companies are shown. The user may click on another point in any column and the detailed values will be shown.

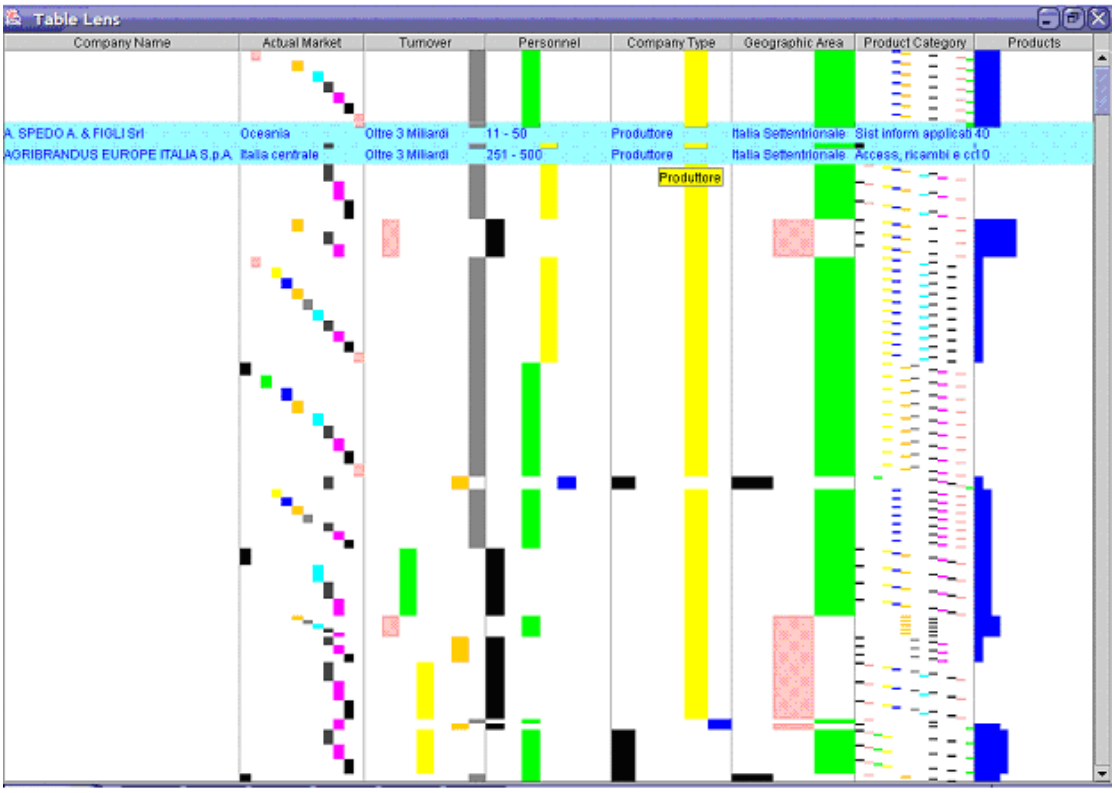

Figure 5: A visualization for Agrilevante trade fair data using DaeTL.

The details visible in Figure 5 indicate that the company “A. Spedo A. & Figli Srl” in the Figure 5 doesn’t serve the following markets (market categories are illustrated in the second column): “Africa”, represented by the black colour, “Europa Meridionale”, represented by the green colour, “Europa Centrale”, represented by the yellow colour, “Europa Settentrionale”, represented by the grey colour.

The other company highlighted in Figure 5 and named “Agribrandus Europe Italia S.p.A.” serves only the Italian market, more precisely, it serves the middle of Italy. The employee number of this company falls in the range [251-500], as shown by the yellow bar of the column related to personnel.

DaeTL permits various manipulations on the shown visualization. Data can be sorted along the values of an attribute. If the user sorts on the column geographic area, the results of this interaction are shown in the Figure 6. It is immediate to see that most companies come from “Italia Settentrionale”, showed with green colour. In such area, there are most companies with turnover more that 3 billions, while the companies coming from the middle Italy (black colour in the Geographic Area column) are either small or big, since no companies in the range [50 millions (ITL) - 1 billion (ITL)] are shown. Figure 7 shows the same data sorted by Turnover column. The dark yellow colour in this column means that the company turnover is more than 3 billions (ITL). Looking at the visualization, it is easy to see that more than a half of the companies in the fair database have a turnover of more than 3 billions (ITL).

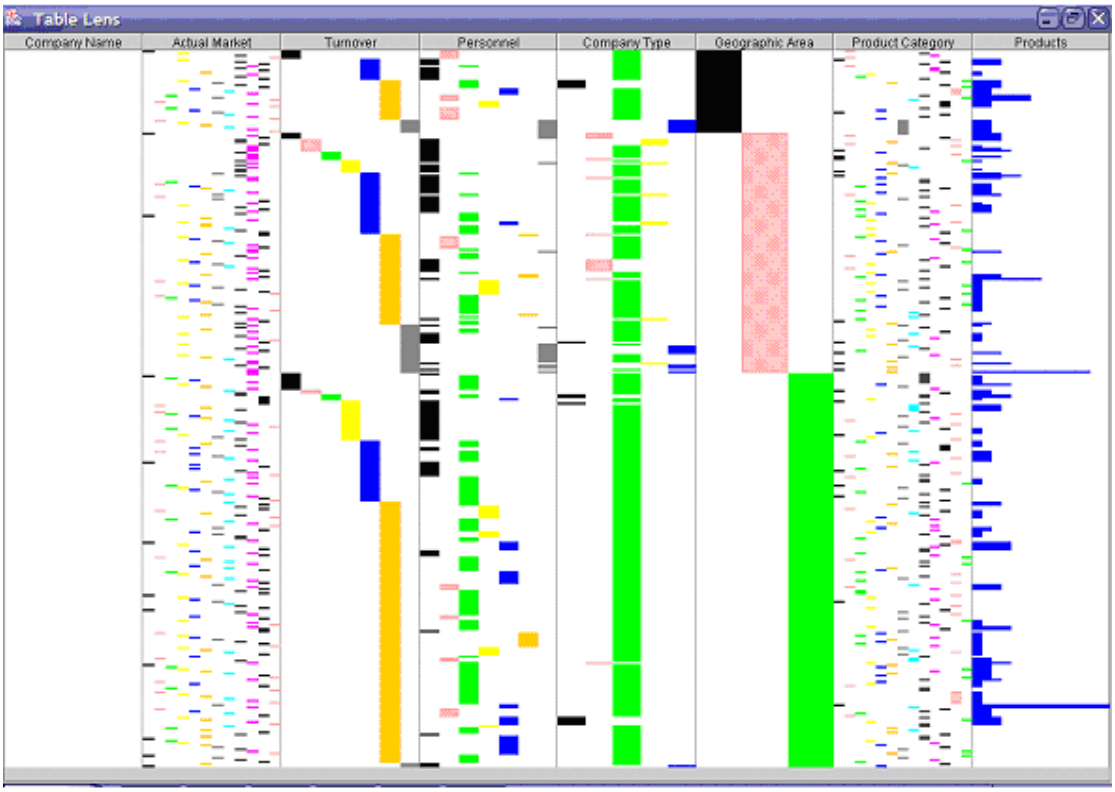

Figure 6: Data set sorted according to the column the Geographic area (“AreaGeografica”).

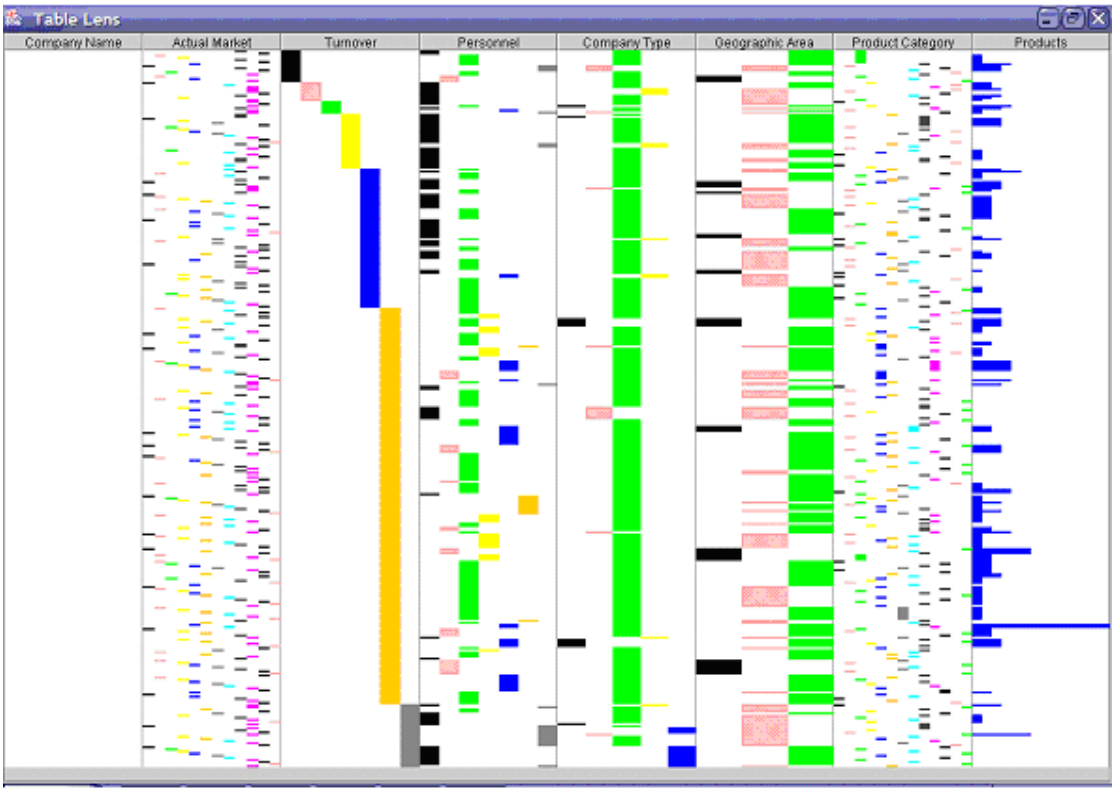

Figure 7: The red square represents the data set with the companies that have more that 3 billions (ITL) turnover.

During a meeting with Agrilevante organisers, after showing for a few minutes the use of the DAE modules, we let them interact with the modules (prototypes at that time), and observed them working with the system asking them to think aloud, according to one of the methods illustrated in Section 2. This user observation activity provides

useful indications for improving that version of the prototypes. Beside, we were very pleased to hear one of the fair organisers involved in that trial to say: “I wish I would have tools like these for directly analyse my data”. This and other comments we got emphasize the importance and utility of the tools provided by DAE, that give users the feeling of “putting their hands” on the data and allow them to explore and manipulate data as they wish.

Another data set was provided by the project partner FSVG. They were data related to the fair Semana Verde 2002. Figure 8 shows a visualization provided by DaeTL on these data. Another trial was then set up for the meeting that the FairsNet Consortium organised in Rome, at LUISS during 3-4 February 2003. We showed to the users the possibilities they have to analyse their data through the DAE tools. Users were two people of the management to FSVG, involved in fair organization. We used the same protocol of the Agrilevante trial. We first showed for a few minutes the use of the DAE modules, then we let users interact with the modules, and observed them using the thinking aloud evaluation method. As it usually occurs with user evaluation, very useful indications came from users for improving the prototypes, as well as the confirmation of the validity of such analysis tool in user daily activities.

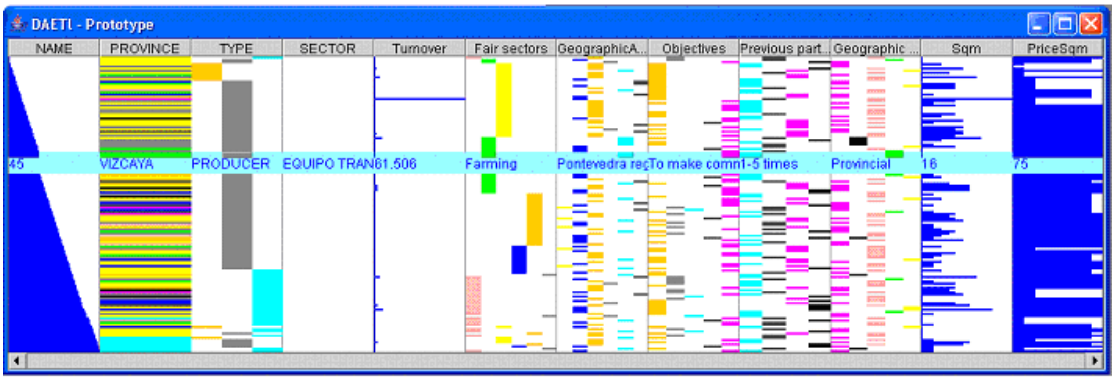

Figure 8: Example of the results of DaeTL on the data of Semana Verde 2002 fair.

5 User Evaluation of the trial

Section 4 described the overall evaluation process carried out during the FairsNet design and development cycle. This section focuses on the user evaluation of the trial mentioned in WP3 and WP4 and refers to using FairsNet for the fair ‘Semana Verde 2004’.

The evaluation plan encompassed steps to:

- identify the stakeholders of the evaluation
- define the critical success factors and evaluation criteria for each module
- prepare the appropriate questionnaires, which have been distributed to the user of the trial
- collect the user feedback
- analyse and document findings from the feedback received.

About the first step, we identify three main type of stakeholders of FairsNet systems: the users, who are represented by members of trade fair organiser enterprise Semana Verde; the developers, who are interested in the feedback of the users in order to improve the system, all FairsNet partners, who are interested in the identification of the key features of the system for dissemination and marketing purpose.

Points 2 to 5 are described in more detail in next sections. Here we only want to point out that the critical success factors identified when preparing the general evaluation plan were:

- the easy of setting up a module
- the usability of a module
- different issue for each module concerning its functionality

Indeed, the objective of the evaluation is about finding out how effective the trial has been, and how the modules being tested might be improved.

Before reporting the evaluation the FairsNet, the authoring tools used to implement the fair web site are briefly described. For setting up the trial, users had to work with such authoring tools. Moreover, the steps of the FairsNet

business process that refer to the fair planning activity are illustrated as an example. Finally, the evaluation results obtained through a questionnaire administered to the users involved in the trial are reported.

5.1 FairsNet Authoring Tools

The Authoring Tools Suite is the foundation of the flexibility of the FairsNet Web-based system and provides powerful Web application authoring support to users, empowering them to adapt the system to changing requirements. To achieve empowerment, a user must be able to make modifications that go beyond user interface customizations and be enabled to manipulate system functionalities that ultimately impact the underlying business process. To accommodate these user needs, FairsNet Authoring Tools help fair organisers shaping the system as they prefer.

In the following, instead of mentioning fair organisers, we will refer to a power user who is a member of the trade fair organiser enterprises and is empowered to be a special user of the Web application, i.e., he is allowed to create new applications and to modify them in order to adapt to changing requirements.

Each tool within the Authoring Tools Suite, referred to as a Manager, provides task-specific support for a power user to set up and customize a specific part of the Web application. The FairsNet Authoring Tools Suite is organized in two subsets of Managers:

- *System Authoring Tools* that support the power user in the creation, integration, visualization, and maintenance of user's dialogue with the FairsNet system by dynamic forms. They are described in more details in the rest of this section, in order to allow the reader to best understand the description of the work performed in setting up the trial.
- *Fair Object Management Tools* that enable the power user to create those components of the web application handling all fair data. These tools include *Taxonomy Manager, User Manager, Booking Manager*, *Fair Planning Manager*, *Data Analysis Engine, Personalization Engine.* The reader may look at D2 for details on these FairsNet components.

In System Authoring Tools, we distinguish six different managers:

- *Project Manager*: In FairsNet each fair is represented as a project, that you can create, if the fair planning and organization starts, or select to support the activities before, during, and after the fair, and finally, you can delete, if all activities in combination with this fair are concluded. The support of the fair organization is provided by the FairsNet Authoring Tools that can be accessed via the Project Manager, the workspace of the power user. The power user's role is either the project manager or an editor.
- *Publishing Wizards*: For the design, integration, and browsing the runtime modules in the Web interface, FairsNet provides a serial of Publishing Wizards, which can be accessed via the Project Manager. They contain wizards based on best-practise templates for the different trade fair business processes and enables simple set up of a trade fair web application. The power user has the possibility to create the runtime modules by the design of the header and navigation frames and the instantiation of static pages, forms and additional contents.
- *Form Dialog Manager*: allows the power user to create, edit, and manage multilingual dynamic Web forms for Trade Fair Web applications that take advantage of the current state of an application's database. When integrated into the application these forms enable on-line users to execute business workflow steps through input, inspection, update, and confirmation of data as well as associated data processing activities, e.g. in support of the identification and registration processes for users and exhibitors. Using the Form Dialogue Manager, power users can create and edit the forms for the Web application under design. This includes the following activities:
  - Selection and structuring of form elements like field elements, buttons, checkboxes, selection lists;
  - Definition of properties of such form elements;
  - Definition of labels and text in multiple languages using the language dictionary;
  - Definition of layout options for forms;
  - Coupling of form elements and business data: The form element constituting the form can be related to the business data. For example, a given form element can be assigned to the name property of the user profile entity in the trade fair domain model, so that the value of the name property is shown as content of the form element. For defining such couplings the Power User browses the associated domain view (domain views are defined using DOMM) and selects adequate domain model elements.

  Forms are assigned to fair projects and can be copied in order to reuse and modify approved forms.

- *Language Dictionary*: Besides the Taxonomy Manager the Language Dictionary is the resource of the FairsNet multilinguality. It provides the multilingual view of the dynamic forms. Each node created in the Form Dialog Manager is connected with an entry in the language dictionary, and thus it can be displayed in the fair interface in all these languages that are regarded in the dictionary.
- *Menu Manager*: is the mean for finalizing and publishing the FairsNet Web application being designed and customized. In fact, besides of the static HOME page, all publishing processes of the customized run time modules have to be finalized by creating a navigational structure of menu items for the run-time modules, and defining to which menu item a Web page managed in the Authoring Tools is linked. In this way, the power user can set up the options the end user has for navigating the Web application implemented with the FairsNet system. The Menu Manager allows the power user to create hierarchies of navigational menu and sub-menu items by defining menu profiles, search for existing menu profiles, create and classify menu profiles. A menu profile represents the basic navigational units for the FairsNet Web application being designed and customized. Each menu profile configured by the power user with the Menu Manager at design-time is used by the FairsNet system to create a corresponding item in the hierarchy of menu items presented to the end user at run-time; this hierarchy of menu items constitutes the menu frame of the run-time Web application framework and provides the set of navigational elements that the end user utilize to navigate and interact with the FairsNet Web application.
- *Domain Object Mapping Manager*: The domain model describes the business data of your application. The Domain Object Mapping Manager supports the definition of views onto the domain model, i.e. the restriction of the domain model to parts that are relevant for a specific task. Using this tool you can:
  - select elements from the domain model that are relevant for a specific task like the definition of forms containing business data. The selected elements are collected in a so-called view that is named and can be used in the definition of forms
  - provide multilanguage labels to the elements of your domain model, thus, enabling international teams to work with the domain model
  - decide which fields of your domain model are mandatory and which of them are optional.

  There are some special elements in the domain model that play a special role in the Web Application. An important example is the “current user” of a Web Application. At application runtime this is the respective user that is logged in, when e.g. a form is opened in a Web Browser. These particular domain model elements are also represented in the Domain Object Mapping Manager and you can select them as part of the view. This enables you to select things like "all booking that have been made by the current user".

In the next section, an example of the use of the above authoring tools for planning a trade fair is reported. The described steps are those used in the actual trial.

5.2 Plan a Trade Fair with FairsNet Authoring Tools

FairsNet permits adapting single steps in the workflows and business processes individually to specific needs of the trade fair and the involved parties by means of the FairsNet authoring tools. As we said, the FairsNet Authoring Tools Suite is the set of tools that support the development and evolution of online solutions in fair planning and organization. Each tool within the Authoring Tools Suite, referred to as a Manager, provides task-specific support for a power user to set up and customize her/his Fair Web application.

The trial that is described in more detail in D4, refers to the development of the Web application (also called new project) for the fair Semana Verde 2004. The main activities performed by the power user are those typical for fair planning, as summarized in this section.

As first step, the power user (in the following we call simply user) creates the new project “Semana Verde 2004” in the FairsNet system. S/he accesses the first authoring tool, Project Manager, with her/his username and password, and selects the language default. Then, s/he selects page link “Create a new project” and assigns the name to the project, a brief description of it, and chooses the editors among those shown on the page. Moreover, there is the possibility to add more editors.

At this point, the user can integrate the static web pages for the run time interface of the new project in several ways, introducing the material (text, images, etc.) that s/he wants to be shown in the fair Web application. For this the Upload Static Pages functionality of the Publishing Wizard is used, thus creating instances of various pages. The instance with the name "index" is recognized from the FairsNet system as starting (HOME) page of the fair run time interface.

A FairsNet project run time interface is always developed as three-parts Frame (header, navigation, content). The user has to take care now of the layout of header and navigation frames. For doing this, s/he uses the appropriate Publishing Wizards. For example, for the header, by calling the Publishing Wizard for the Header Frame, the user can upload the appropriate pictures, and select a background colour. Instead, for the design of the navigation frame, s/he goes to the Menu Frame Publishing Wizard, where s/he can arrange the colours for the Background and for the Menu entries. The menu structure is pre-defined as a default for the FairsNet system. She can see the results of background/menu colours combinations immediately using the preview functionality of the Wizards. Then, calling the FairsNet Pageloader, it is possible to combine the three designed frames in one single frame.

The user should now define the categories in the new project. These categories include languages, countries, thematic areas of the fair, different groups of system users and fair participants. As we said, the Taxonomy Manager permits this. In order to facilitate user-system interaction when creating a new fair, some default categories are pre-defined in FairsNet. More specifically, the user setting up a new fair does not need to update all taxonomy, but s/he can use default taxonomy. However, in the case in which the categories are insufficient for the user, because s/he develops a fair web site for a particular fair with specific requirements, s/he can customize the categories, defining new categories, modifying existing categories and organizing categories in hierarchies to start easily, the user could use the hierarchy of categories and subcategories.

The user has of course the possibility of defining the fair business tasks s/he wants to support for the specific fair. For this purpose, user activates the Menu Manager and inserts the respective tasks in the task taxonomy. This structure will be used as the basis of the navigation menu of the fair Web application. A starting point for the task list could, for example, be: Welcome, General Info, Identification, Registration, etc.

By using the various tools available in FairsNet, the user can decide for which user group (visitor, exhibitor, press etc.) the page should be visible to, in which languages the page will be displayed and to which task the page corresponds. With the Menu Manager, user creates a menu profile for each of the pages, and links them with the selected user group(s), the selected languages(s) and the selected task. Further details on the user activities in setting up a fair are reported in FairsNet user manual.

5.3 Evaluation of the trial

In this section we report the results of the evaluation of the trial to which three people of Foundation Semana Verde de Galicia worked in order to create the Web application for the fair Semana Verde 2004. The three people all worked as power user.

We have designed and administered to these power users a questionnaire for getting their feedback and identifying possible difficulties they could have during the development of the Fair web site.

The questionnaire includes a list of activities that the power user has to perform for planning a trade fair and generating the appropriate pages visible on the Web. In Table 1, this questionnaire is shown. The first column reports the number of each item in the questionnaire. In the second column, we indicated each activity in items from 1 to 26; for each one, the power user should answer Yes=Y, N=No or NA=Non Applicable to three different questions:

Q1. Was the power user able to complete the task?

Q2. Was the power user able to understand how to complete the task?

Q3. Does the power user have any comments on the activity, e.g., usability, the ease of the setting up the application, anything missing?

Each user was asked to report the answer to these three questions in the corresponding three columns of the questionnaire. Actually, we provided more space for their comments with respect to what is shown in Table 1, so that they could provide us with more information.

The last part of the questionnaire asks users to give their opinion about the documentation and to provide an overall assessment of the system. To each item, the user answers yes or not and provides an optional comment.

We asked each of three users to fill the questionnaire individually, even if they worked together to the trial, because we were interested in the opinion of each one.

All users gave positive answers to both questions Q1 and Q2 for items 1-6, with no special comments: they all agreed the tasks were easy to accomplish.

All users answered YES to Q2 and NO to Q1 for items 7-8. They also provided various comments:

User 1: *It does not indicate what kind of user it has to be in order to be an editor of the project (actually, it is not clear at all). I assumed that it was a Power user, and so I created one but it doesn't show it among the options we can choose from.*

User 2: *User problem: When you create a new project, you need to assign a project manager and you have the available option of adding a new project editor. However, we do not actually know what kind of user can adopt this function. The point is that we created "power user" but it does not appear in the selecting list of project editors.*

User 3: *I got confused. I was not able to understand who cab be an editor of the system. To me, it should be the power user, but this is not one of the options the system shows.*

The above comments indicate that user found difficult to identify who could be the editors, and therefore they were not able to accomplish that task. The system requires that, when a power user creates a new project (i.e. a Web application for a new fair), s/he must indicate the editors of the fair, but these figures are not clear. Users assumed that editors are the power users. But when a new power user is added, the name does not appear in the editor list.

All users gave positive answers to items 9-11.

Items 12-16 got different answers from the users. All answered YES to question Q2, but two users answered NO to Q1 and one user answered NO also to Q2 for item 16. They also provided comments that address the difficulties they encountered. We report here some of them:

We consider that it could be interesting to include somewhere the size of the logo in pixels.

It is impossible to delete images or logos previously uploaded.

The logo uploaded is automatically deleted when you choose a background color.

We have not found where to rename the navigation frame.

All users gave positive answers to items 17-20, but one provided a comment here reported about item 19:

Colors can not be updated individually. All colors must be indicated at the same time.

With their answers and comments to items 12-20, users pointed out some difficulties that contrast with the need of managing the objects of the web application in a simple way. They want that the power user should rename the navigation frame without problem, delete images or logos previously uploaded, update colors of the page individually, etc. One interesting option to be inserted is the size of the logo of the fair in pixel, and also it is important that the power user has the possibility to change the color of the page background without deleting the logo previously uploaded.

All users gave positive answers to items 21-26, but one provided a comment here reported about item 22:

The major inconvenience is that pages must be uploaded individually. It is impossible to upload the whole structure as in any other Web site creation software. Hence, it represents a problem when pages have to be modified in order to be adapted to this format.

All users answered YES to items 27-28 that refer to the documentation they had available, but one provided a comment here reported about item 27:

I think that the manual is written from a technical point of view. I consider that it would be more convenient to adapt it to a user level.

Indeed, user documentation has been revised to take care of the above comment. It now provides information on how to work with the system in a more constructive way, adapted to the user level.

All users answered YES to items 29-32. Two users also very positively commented on the usefulness of the Wizards. Indeed, wizards help users with no familiarity with FairsNet system to work with the authoring tools, thus improving ease of learn and ease of use.

About items 33 and 34, two users answered NO. They were those not familiar at all with programming. One of them provided the following comment:

I consider that the dependence from computer programmer is still enormous. Actually, as a system error occurs it is impossible to solve it directly from here.

The same two users who answered NO to items 33-34 also answered NO to item 35. The following comment was provided by one of them:

Without a wizard to indicate the steps to be followed, the creation of a fair could be complicated because the Authoring Tools sometimes are not very intuitive to work.

As general comments (last item in the questionnaire) users said that, after all, they were able to set up the trial with less effort they could guess in advance.

#	ACTIVITIES FOR PLANNING A TRADE FAIR	Q1 (Y/N/NA)	Q2 (Y/N/NA)	Q3 COMMENTS
	Creating a project			
1	Create of a new project			
2	Access the Project Manager			
3	Login the Project Manager			
4	Insert the project name			
5	Insert the project description			
6	Choose the editor/s from the list			
	Updating the editors list			
7	Add a new member to the editors list			
8	Delete a member from the editors list			
	Designing the layout of the header frame			
9	Select project for designing the header frame			
10	Access to the Publishing Wizards			
11	Access to Header Frame			
12	Upload logo /image			
13	Upload background image			
14	Choose Background colour			
15	Update the new header frame			
16	Design the layout of the navigation frame			
17	Access to navigation frame			
18	Choose the background colour of the menu layers			
19	Update the new menu layout			
20	Refresh the new menu layout			
	Creating the content frames			
21	Access to Upload Static Page Wizard			
22	Upload html pages and images			
23	Instantiate the uploaded HTML files			
24	Select the starting page			
25	Save the instance name			
26	Save the instantiation			
		Y/N/NA	Comments	
	Documentation			
27	It is understandable			
28	It fully describes the functionality			
	Overall Assessment			
29	The terms are easy to understand			
30	Layout of the screen is clear			
31	Functionalities meet expectation			
32	Wizards to guide the power user through the process are useful			
33	To understand where an error has occurred during the development of the web site is easy			
34	To solve a problem is easy			
35	To use Authoring Tools is easy			
36	General comments			

Table 1: Questionnaire submitted to power users involved in the trial.

6 Discussion

The results of the trial gave useful feedback to the stakeholders of the evaluation, i.e. users themselves, developers and project partners.

User appreciated the incremental approach in building the far that FairsNet adopts. They pointed out very clearly the little problems that it is necessary to fix in the system, and this is the kind of information developers need to improve functionalities, usability, and the overall quality of the system.

All partners of the project are now better aware of the key features of the system that will be stressed in their dissemination and marketing activities.

The Authoring Tools Suite is the foundation of the flexibility of the FairsNet Web-based system and provides powerful support to users, empowering them to adapt the system to changing requirements. To achieve empowerment, a user must be able to make modifications that go beyond user interface customizations and be enabled to manipulate system functionalities that ultimately impact the underlying business process. Such modifications are usually performed by skilled software developers. Trade fair organizers have the core part of the know-how in the trade fair's application domain. Thus, they are valuable co-designers and developers in system customization and evolution, but they are not assumed to possess programming skills. Trade fair organizers are professional people with a lot of knowledge and experience. They need powerful and flexible systems to support their tasks in running a fair. To accommodate these user needs, FairsNet Authoring Tools help fair organisers shaping the system as they prefer.

It is worth noting that FairsNet exploits the use of various taxonomies that are intended to support the process of managing classification hierarchies (i.e. taxonomies in computer science jargon) for classifying, browsing and retrieve content. For example, organization's members are classified in a company-chart; customers are usually classified in categories of customers, depending on their characteristics (exhibitors, visitors, male, female, registered, confirmed, countries, ages, etc.). Moreover, as a whole, content management activities need to refer to a classification schema; for instance, in the trade fair business, customers and products are classified against ad-hoc thematic groups or categories according to established criteria (thematic areas, sectors, etc.). Taxonomy-based components in FairsNet are intended to support the process of managing taxonomies. For this, another authoring tool called Taxonomy manager has been developed.

However, it would be too difficult for a fair organiser to mange taxonomy from scratch the very first time they use the system. For this, some default taxonomies have been incorporated in the system that can be used at the beginning and updated later on as soon as users get more familiarity with the system and/or feel the need of other services in their application. This pre-defined components, as well as the Publishing Wizard the system provides in order to guide in the use of the Authoring Tools, have been very much appreciated by the users.

The Authoring Tool approach used in FairsNet is a novelty that goes in the direction of empowering users by providing them with powerful environments for creating even complex applications. However, ease of use is also emphasized, so that the users shape their application and create software artefacts without writing any programming code. This has been very much appreciated by the users of the trial. Indeed, they were very sensitive to this and complained in cases in which they felt the use of the system required some programming expertise. Through the Authoring tools, users are able to perform activities of End-User Development (EUD). EUD is a new field in HCI and in software development, that has been stressed in a thematic networks sponsored by EC under project EUD-NET IST-2001-37470. EUD is defined as "a set of activities or techniques that allow people, who are non-professional software developers, at some point to create or modify a software artefact". EUD activities go from simply setting parameters, to integrating existing components, up to extending the system by developing new components. The use of authoring tool allows fair organisers to develop new applications related to different fairs, with different features and services, without any special background in Computer Science.

References

[Adl, 1992] P. S. Adler and T. A. Winograd (Eds.), Usability: Turning technologies into tools. New York: Oxford University Press, 1992.

[Cos, 2001] M.F. Costabile, Usability in the Software Life Cycle, in Handbook of Software Engineering and Knowledge Engineering, S.K. Chang Ed., World Scientific, Vol. 1, 179-192, December 2001.

[Dix, 1998] Dix, J. Finlay, G. Abowd and R. Beale, Human Computer Interaction, Prentice Hall, 1998.

[Hac, 1998] Hackos J. and Redish J.C., User and Task Analysis for Interface Design, John Wiley & Sons, 1998.

[Har, 1993] H.R. Hartson and D. Hix, Developing User Interfaces. John Wiley, New York, 1993.

[ISO, 1998] ISO (International Organization for Standardization), ISO 13407: Human-Centered Design Process for Interactive Systems, 1998.

[Jef, 1992] R. Jeffries and H.W. Desurvire, Usability Testing vs. Heuristic Evaluation: Was There a Context? ACM SIGCHI Bulletin, Vol. 24 No. 4, 39-41, 1992.

[Nie, 1993] J. Nielsen, Usability Engineering, Academic Press, Cambridge, 1993.

[Nie, 1994] J. Nielsen and R.L. Mack, Usability Inspection Methods, John Wiley and Sons, New York, 1994.

[Nie, 1994a] J. Nielsen, Guerrilla HCI: Using Discount Usability Engineering to Penetrate Intimidation Barrier, In Cost-Justifying, Usability (R.G. Bias, D.J. Mayhew, Eds) Academic Press, Also available at the URL http://www.useit.com/papers/guerilla_hci.html, 1994.

[Pat, 2001] F. Paterno', Task Models for Interactive Software Systems, in Handbook of Software Engineering and Knowledge Engineering, S.K. Chang Ed., World Scientific, Vol. 1, 817-836, December 2001.

[Pre, 1994] J. Preece, Human-computer Interaction, Addison Wesley, 1994.

[Shn, 1999] B. Shneiderman, C. Plaisant, K. Doan, and T. Bruns, Interface and Data Architecture for Query Preview in Network Information Systems, ACM Transaction on Information System, Vol. 17, 320-341, 1999.

[Whi, 1988] J. Whiteside, J. Bennet and K. Holtzblatt, Usability Enginnering Our Experience and Evolution, In Handbook of Human-Computer Interaction, (ed. M. Helander), Elsevier Science, 791-817, 1988.

Appendix A: Results of inspection of FairsNet Authoring Tools

In this appendix, we report the results of the usability inspections carried out on some prototypes during the development of FairsNet. The prototypes are: “Form Manager”, “Taxonomy Manager” and “Classification Manager”.

(it is similar to what I already sent and I did not include to limit the file size)