

Developing Systems with NDT and NDT-Tool

M.J. Escalona, M. Mejías, J. Torres

Department of Computer Languages and Systems University of Seville

ETS Ingeniería Informática. AV. Reina Mercedes S/N 41012 Sevilla, Spain

Ph. +34954553867

{escalona, risoto, jtorres}@lsi.us.es

Abstract. The rising interest for developing systems in web environments has produced that the research community poses the necessity of offering methodological proposals in order to give a suitable reference when a web system is produced. In the last years, several methodological proposals have appeared: OOHDM (Object Oriented Hypermedia Design Method, UWE (UML Based Web Engineering), OOH (Object-Oriented Hypermedia Method), WSDM (Web Site Design Method) or WebML (Web Modelling Language) are only some examples. However, there is not a standard and world wide accepted methodology. Each methodology proposes its own techniques and models. This fact has produced that some studies have been made in order to compare them. These comparative studies show that there are some gaps in web engineering. This paper presents these gaps and introduces NDT (Navigational Development Techniques). NDT is an approach which offers a different development process that starts with requirements treatment and allows to get design models using a systematic process. It tries to solve some of these gaps detected in web engineering. In order to introduce how NDT confronts these gaps, the paper presents NDT development process, a global vision of its objectives and its tool case, NDT-Tool. Also, this paper enumerated some real projects developed by NDT and NDT-Tool in real companies.

Key words: Web Engineering, Web Methodologies, Tool Cases

1 Introduction

In the last years, several methodological proposals have been developed by the research community: OOHDM [22], UWE[16], WebML[4], UWA project[23] or OOH[5] are only some examples. However, after studying them [8] and analyzing several comparative studies [2][12][15], we can deduce some ideas.

- A. **Web proposals are mainly focused on the design and implementation phases.** They often do not offer new techniques or models to capture and deal with web requirements. These methodologies often propose use cases as the only technique to treat requirements and only some of them, like UWE, propose techniques to describe or validate requirements. Others, like UWA, offer a complete procedure to capture and describe prototypes, but their proposals do not make easy the communication with the user.
- B. **Most of them are based on the conceptual model.** The conceptual model represents the static structure of the web system and it is often represented using a class diagram in UML. In most of approach, there are other system models, like navigation model or the abstract interface model, which are

designed using the conceptual model. It causes that these models depend on the conceptual model. Thus, a small change in the conceptual model can produce a big change in the other ones. Also, a critical question in web system development is to sure that a designed conceptual model is the most suitable one for a system and that it does not have any mistakes.

- C. **There is a gap in tool cases.** Although the research community in web engineering was mainly focus in the design phase, recently, some groups are more interested in other phases like the requirements treatment [2][8][12][15]. They are upholding for offering an easier way to go from requirements to the conceptual model [5][7][16]. In this way, the researchers are demanding the necessity of giving tool cases which guide the team in the development of web information systems [2][8][12][15].

This paper offers a global vision of NDT[10]. NDT is a web methodological proposal that is focused on the first phases of the life cycle. For that, in the second section, a global vision of the most referenced web methodologies is shown in order to present the motivations to develop NDT. In the third section, a global vision of NDT is presented. After this, in the fourth section, it presents a case tool, NDT-Tool[11], which guides in NDT application and gets NDT results automatically. In the fifth section, some applications of NDT and NDT-Tool in real projects are shown. Finally, some future works and conclusions are given.

2 Current web methodologies

Nowadays, there are a lot of methodological proposals oriented to the web environment. The huge number of methods, models and techniques that has appeared has produced that some comparative studies have been made in order to analyse similarities and differences between them [2][8][12][15]. In figure 1 a short schema with the most famous proposed web methodologies is shown¹. In this figure, fixes lines represent that one proposal is based in other previous, dotted lines represent that these methodologies are related.

We can observe in this figure that there are a lot of relations. Thus, really, in web environment, there is not a consensus in the used terminology or in the used development process. However, if the comparative studies are analysed, some common ideas can be found. The first common point is that most web methodologies are mainly focussed on the design and the implementation phases. Design has been the most worked phase and most methodologies propose to separate in design three basic aspects, at least:

- The conceptual aspect of the system, which is often represented using a class diagram.

¹ All of these methodologies are referenced and hardly studied in [8][12][15].

- The navigational aspect of the system, which represents how the user can navigate in the system.
- The abstract interface aspect of the system, which indicates how the user can interact with the system.

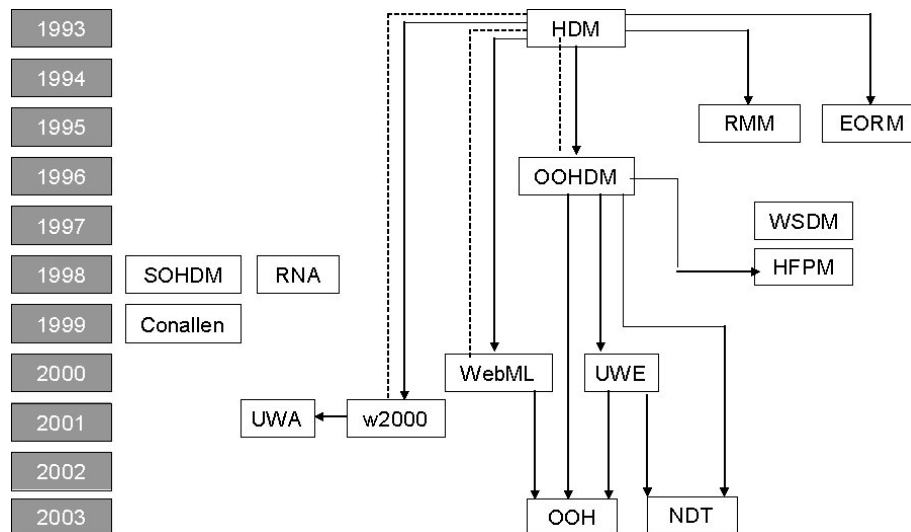


Fig. 1. Web methodologies relations

The idea of separating these aspects in web systems was first presented in OOHDM, but it has been accepted in following important proposals like SOHDM [18], UWE or OOH. Moreover, other proposals, like UWA, W2000[1] or WebML, besides propose to design other aspects like the internal functionality or the personalization with independent models.

Although there are different proposals, the most interesting aspect in our study is to analyse the development process used by these methodologies. The work of these methodologies is focussed on design which is mainly based on the first enumerated model: the conceptual model. This general characteristic is presented in figure 2. As it is shown, the process starts with the requirements treatment. The group of analysts receives information from the users and customers that allows them to define the requirements catalogue. Some methodologies do not cover this phase. Perhaps OOHDM, SOHDM, Conallen's proposal [6], UWE, W2000 and UWA are the most interested proposals in requirements treatment. All of them, except for UWA and SOHDM, propose to achieve the requirements capture and definition using the use cases [14], which is the UML proposal [3][17]. Starting in requirements definition, the conceptual model of the system must be designed. When this model is developed, other models like navigational model, abstract

interface models, etc. are made. These other models are based on the conceptual one. Afterwards, the system is implemented using all these models. The implementation is valued by users who detect errors and bugs. This whole process is repeated until the suitable final system is obtained.

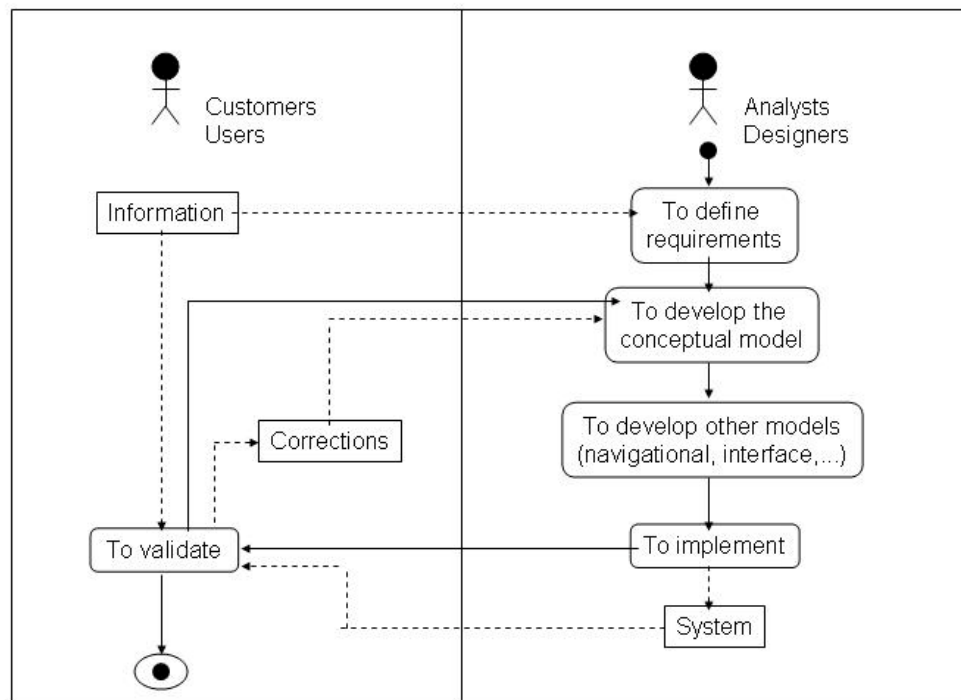


Fig. 2. The general development process in web methodologies

However, although this process has been accepted by the research community, nowadays, there are some important points which are being questioned: *are use cases a suitable technique to treat requirements in web environment?*, *is it easy enough to get a good conceptual model from use cases?*, *is it necessary to develop other system models from the conceptual model?*, etc.

The answers depend on the research group. Use cases are very easy to be understood by the user but sometimes, it is not easy to get a conceptual model from them because they are very little concrete [12]. Moreover, it could be possible to make other models like navigation or abstract interface models independent of the conceptual model. It will assure a cheaper maintenance and more possibilities of reusing.

3 NDT (Navigational Development Techniques)

In order to deal with these questions, since 2001 a new proposal named NDT is being developed. NDT is a methodology focussed on the first phases of the life cycle. Its development process can be defined as a bottom-up process. It is focussed on a very detail requirements definition guided by objectives [19], which covers three subphases: requirements capture, requirements definition and requirements validation. In figure 3, the NDT process is presented. As it can be observed, it is quite different from the process in figure 2. The NDT process only covers the first phases in the life cycle. Its proposed models (conceptual, navigational, abstract interface models) are independent and it is necessary to stick out that workflows in NDT, which are represented with continuous arrows in figure 3, are systematic workflows. These workflows can be even automatic if the development team uses NDT-Tool, which is a case tool that let apply the NDT process.

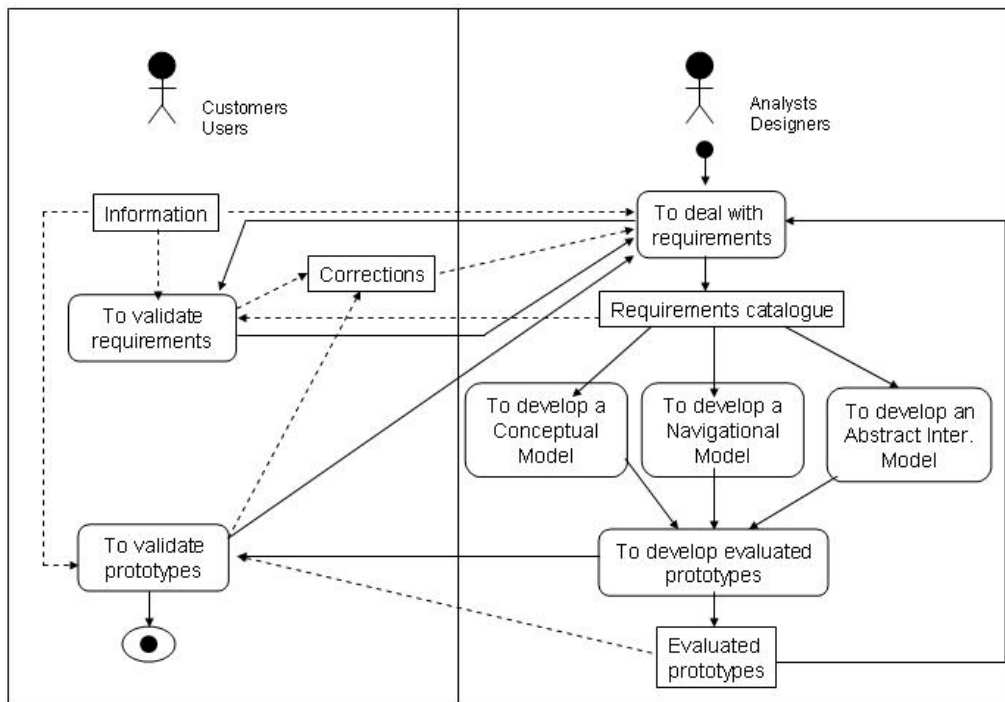


Fig.3. NDT development process

The process starts by defining objectives. Using a described procedure, requirements are captured and defined. Requirements are classified and dealt with depending on whether they are: information storage requirements, interaction requirements, etc. When requirements are validated, the NDT process continues

defining three models: the conceptual model, the navigational model and the abstract interface model.

The most important characteristic of the NDT process is that the way to go from requirements to models is systematic and makes each model independent. It is systematic because NDT offers algorithms which indicate how each model has to be obtained from the requirements definition. And they are independent because, although different models are related between them, because all of them represent an aspect of the same system, each of them can be obtained independently from the requirements. It is not necessary to model the conceptual schema to design the navigational or the abstract interface model.

From these models, some evaluated prototypes can be obtained. These prototypes can be valued by users and customers. In this section, the NDT process is finished. Starting with the developed models and making sure that they are correct, the development team can apply other web methodologies like UWE or OOHDm in order to continue the development process.

The main goal of NDT is to offer a systematic process to get models which other accepted web methodologies use as the base to their development process and to guarantee the quality of these models given techniques and metrics to validate the results. NDT is a methodological procedure to get design models from the users' requirements. In the next sections, NDT development process is going to be presented in a more concrete way.

3.1 Requirements treatment

The first workflow in the NDT process is the requirements treatment. In this workflow, system requirements have to be described. The process starts by studying the environment and defining system objectives [20]. When they are defined, information storage necessities have to be studied. In this second activity, the analyst has to determine what kind of information the system has to store and how it is structured.

A web information system can be very different depending on the actor who interacts with it. Thus, the third activity in NDT is to study possible actors. In this activity actors are classified, possible roles are defined and relations and incompatibilities between them are detected.

After that, functional internal necessities are studied. Using use cases, functional options that the system has to offer to each role are described.

The fifth activity is one of the most important for the rest of the system. In this activity, interaction requirements have to be studied. To get this, in a first task phrases must be described. They indicate which information is important for each actor to retrieve the storage information. In a second task, visualization prototypes are described. These represent who can interact with the system, what information will be shown and what use cases described in the fourth activity can be executed

in each moment, and also, how users can navigate in the system. These visualization prototypes are the base of the NDT process.

The last activity in requirements treatment is to define non-functional requirements. Non-functional requirements are those ones that are not directly related to system functionality and describe constraints to system's capabilities. They usually affect to the performance and semantics of the system.

To finish this short description, it is necessary to indicate that the requirements treatment in NDT is oriented in two different ways. On the one hand, it is easy enough to be understood by users and customers. But, on the other hand, it has to offer concrete and structured results in order to be useful to the development team. In order to get these aims, to describe objectives and requirements, several patterns are proposed in NDT. A pattern is a table with concrete fields which will be completed by the user and the analyst. Each kind of requirements has its own pattern. This proposal based on patterns of NDT has been applied to several real problems giving very good results [9].

3.2 The conceptual model development

The concise and structured description of requirements given by patterns lets design models systematically. In this way, the conceptual model can be obtained from patterns defined in the second activity in the requirements treatment.

To continue with the present research community tendency [15], the conceptual model is represented using a class diagram in UML [3]. According to the UML authors, when a static model of a system is developed using a class diagram, it is used to get one of these aims: to represent the vocabulary and the limits of the system, to model simple collaborations between elements in the system or to model the logical scheme of the database. In NDT, like in other web methodologies, the class model gets the first and the second aims, and although the conceptual model can be used as a base to develop the database, this step is not included in the NDT process.

3.3 The navigational model development

The navigational model represents how the user is going to visualize the storage information in the system and how he will be able to navigate through it. This description has been accepted in most of web methodologies. However, there is not a standard representation of this model [2][8][12][15]. The most widespread idea in the research community is to use a class model with special classes like nodes, links, menus, etc. After doing an exhaustive comparative study [8], NDT has assumed the UWE approach [16] because it seems the most suitable for us. This proposal offers a class diagram extension based on UML where the navigational model is a special class diagram with special classes: nodes, links, queries, menus, guided routes and indexes. This navigational model is quite easy to be understood because it is based on UML and offers enough semantic to represent all the system

navigational structure. Starting with interaction requirements patterns, and mainly with visualization prototypes patterns, a definition of nodes, indexes, routes guides, menus and links is obtained in a systematic way, in an automatic way if NDT-Tool is used. From phrases patterns more links and queries can be determined.

To conclude, it is necessary to indicate that NDT navigational model is really a group of navigational models. Using the study about actors made in the requirements phase, each actor's navigation necessities are detected. Sometimes each actor has a very different navigational model and it is necessary to develop several navigational models.

3.4 The abstract interface model development

The abstract interface model lets represent how the user will interact with the system. Again, interaction requirements are the base to obtain it. Nowadays, the conceptual and the navigation model is completely defined in NDT and also algorithms that allow to get them. However, the abstract interface model is still being developed. There are a lot of tendencies to represent this model and NDT research group is evaluating them to decide which is the best one and if it could be included in the NDT process or if it is necessary to introduce a new one. Nowadays, we find very interesting the UWE proposal [16] and the Conallen's proposal [6] but we want to apply them to our real examples and some new ones in order to evaluate the results.

Because this phase has not been developed completely yet, it could be possible to find the necessity of extending any available proposals, like UWE or Conallen's, to get a more suitable approach.

4 NDT-Tool

NDT-Tool is a case tool which allows to apply NDT algorithms and techniques and to generate results, documents and models automatically. Nowadays, only requirements phase is completely codified, but some algorithms and models of the next phases are already implemented.

The first form in NDT offers a global menu with several options:

- ✓ Management project: this option allows to add, modify or delete web projects into the tool. Each project has a short description, which will be automatically included in the output documents.
- ✓ Requirements treatment: this option lets apply all techniques and models proposed by NDT requirements phase. It allows to complete objectives and requirements patterns and to check any inconsistency. It also offers the possibility of making the traceability matrix, which allows to value requirements. When this phase is finished, NDT-Tool can generate the requirements catalogue. It generates a word document with a defined structure

that includes all pattern definitions, the participants in the system development, an index, a cover, etc.

- ✓ Conceptual model option: this option takes patterns definition in the requirements phase and automatically generates a class diagram from them. It also generates a dictionary which describes classes in class diagram in a more concrete way.
- ✓ Navigational model option: this option generates a navigational model using the definition in patterns. To represent this model, the proposal of UWE has been used. Also, a dictionary which describes the navigational model can be generated.
- ✓ Interface model option: this option generates the interface model and a document which describes its elements. This last option must generate some evaluated prototypes. Our idea is to generate these prototypes in JAVA, but, nowadays this option has not been developed yet.

Models in conceptual model, navigational model and abstract interface model can be visualized in Rational Rose environment. Nowadays, algorithms which generate these models are implemented but we are still working in the connection with the Rational Rose tool.

In next figures, some screens of NDT-Tool are presented².

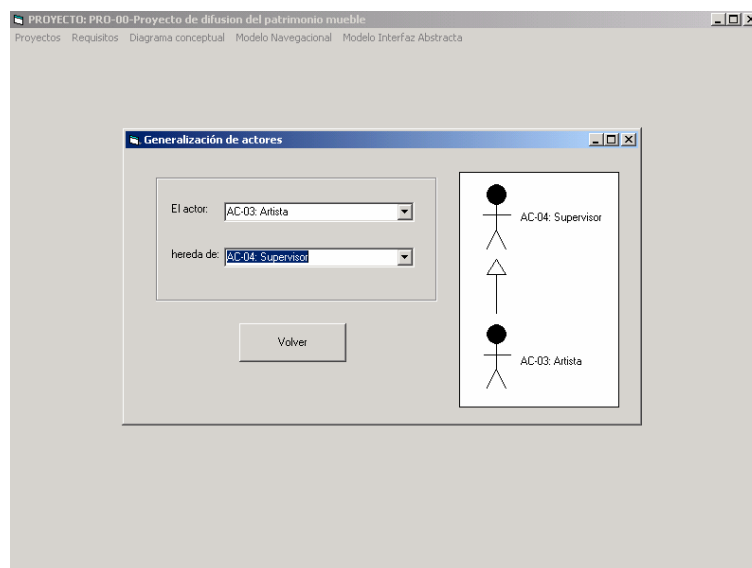


Fig. 4. Actor's generalization option

In figure 4, the graphical option to work with actors and roles in requirements definition is offered. In the figure 5, the matrix of traceability which makes possible to evaluate if all the objectives have been covered in the specification is

² Nowadays, only the Spanish version of NDT-Tool is available

presented and in figure 6, an example of a requirements pattern is shown. Data in this screen has been obtained from a real system developed with NDT and NDT-Tool. NDT has been applied to several real problems giving very good results. Now, we are applying our proposal of NDT-Tool to these examples to evaluate the use of the tool.

	OBJ-01: Gestionar los bienes	OBJ-02: Gestionar los perfiles	OBJ-03: Gestionar los perfiles de usuarios	OBJ-04: Gestionar los museos
RA-01: Bien Mueble				
RA-02: Bien Inmueble				
RA-03: Autores	X			
RA-04: Tesoro				
RA-05: Situaciones de protección			X	
RA-06: Estado de conservación				
RA-07: Intervenciones y restauraciones				
RA-08: Museos				
RA-09: Autores bien				
RA-10: Imagen bien	X			
RA-11: Resolución				
RA-12: Alteración				
RA-13: Usuarios				
NA-01: Código			X	
NA-02: Código inmueble				
NA-03: Cronología				
AC-01: Arqueólogo				
AC-02: Etnólogo				
AC-03: Artista				
AC-04: Supervisor				
AC-05: Catalogador				

Fig. 5. Matrix of traceability option

Código	Nombre descriptivo
PV-01	Datos de identificación y localización del bien
PV-02	Datos de descripción del bien
PV-03	Datos de análisis del bien
PV-04	Datos del estado de protección del bien
PV-05	Datos del estado de conservación
PV-06	Datos de las intervenciones que se realizan sobre las piezas
PV-07	Datos de los museos

Datos del prototipo

código: nombre descriptivo:
 [PV-01] [Datos de identificación y localización del bien] [Nuevo]

descripción:
 El sistema deberá permitir la visualización de los datos concretos que se muestran a continuación y la navegación expresada y que representan los [Recoger cambios]

precondición:
 Tras ejecutar una de las frases anteriores se elegirá un bien mueble que será para el que se muestren los datos de identificación. [Eliminar]

comentarios:
 []

Datos básicos Actores Frases Funcionalidad asociada
 Información visualizada Prototipos relacionados Autores Fuentes Avanzadas

Fig. 6. An example of the visualization prototype pattern

5 Real projects developed with NDT and NDT-Tool

Since 2001, when NDT started to be developed it has been applied to several real projects. These applications to the real environment in public and private companies are a good feed back for us. The real applications with real customers and users let validate the NDT techniques, patterns, models, results and models.

The first real project developed with NDT was the system to Manage Monuments in Andalusia [13]. This system was developed in 2001 in a collaboration between the department of Computer Languages and Systems and the Historical Patrimony Andalusian Institute [13]. In this case, only the requirements phase of NDT was applied but the results was very interesting because we could value that patterns are a good technique to make easy the communication with costumers and users. The collaboration was so good, that NDT has been applied to others systems in this institution: the web system to Manage the Thesaurus of Historical Patrimony [13], the web system to Generate Automatically Cultural Routes in Andalusia [13], etc.

NDT also has been applied in private companies. For instances, it was applied in several projects with the company Sadiel S.A. [21] developed during 2002 and 2003.

The most recent application of NDT was really interesting. It has been used to develop a system to Generate Medical Reports to People who have Physical Handicaps [12]. It was really difficult because the medical terminology used by customers was very complex, but patterns and prototypes generated automatically with NDT were again a good technique to make easier this communication.

Apart of these applications, NDT and NDT-Tool are being also applied by our student in Computer Sciences. Nowadays, about 35 final projects in Computer Sciences in our Department have been developed with NDT.

6 Conclusions and Future Works

This paper has presented a global vision of NDT, a new web proposal to specify and analyze web system. For that, it has first given a global vision of the current web methodologies to introduce which are the motivations of NDT. After that, it has presented a global description of the NDT development process and its case tool, NDT-Tool. Finally, it has enumerated the application of NDT in real projects with public and private companies.

As conclusion, we can say that NDT is giving very good results in real companies. We though that NDT is getting its goals and solving questions made in section 2. It offers a good way to deal with requirements in web environment and it does not focus the development process in the conceptual model. It offers a systematic way, which is even automatic with NDT-Tool, to model web aspects in analysis easier and cheaper.

As a future work we are focussed on two lines. The first one is oriented to companies. We think that it is very important to continue applying NDT and NDT-Tool in real projects to validate its results, models and techniques. But, in other way, we are working with several research groups to compare NDT with other web methodologies and have expert points of view which let us learning new possibilities to NDT.

7 References

- 1 Baresi L., Garzotto F., Paolini P (2001). *Extending UML for Modelling Web Applications*. 34th Annual Hawaii International Conference on System Science. IEEE Computer Society.
- 2 Barry, C. & Lang, M. (2001) *A Survey of Multimedia and Web Development Techniques and Methodology Usage*. IEEE Multimedia. April-June 2001, 52-60.
- 3 Booch G., Rumbaugh, J., Jacobson, I. (1999). *Unified Modeling Language User Guide*. Addison-Wesley.
- 4 Ceri, S. Fraternali, P., Bongio, A., Brambilla M., Comai S., Matera M. (2003). *Designing Data-Intensive Web Applications*. Morgan Kaufman.
- 5 Cachero, C (2003). *OO-H: una extensión a los métodos OO para el modelado y generación automática de interfaces hipermediales*. P.h. Tesis. Universidad de Alicante, 2003.
- 6 Conallen, J. (1999). *Building Web Applications with UML*. Addison Wesley 1999
- 7 De Troyer, O., Leune, C. (1997). *WSDM: A User Centered Design Method for Web Sites*. Technical Report of Tilburg University, Infolab. Belgium.
- 8 Escalona, M.J., Mejías, M., Torres, J. (2002). *Methodologies to develop Web Information Systems and Comparative Analysis*. Informatik/Informatique. núm. 2/2002 de I/I.
- 9 Escalona, M.J., Torres, J., Mejías, M. (2002). *Requirements Capture Workflow in Global Information Systems*. Proceedings of OOIS. LNCS 2425. Pág. 267-269. ISSN. 0302-9743 Springer-Verlag. Montpellier, France.
- 10 Escalona, M.J., Mejías, M., Torres, J., Reina, A.M. (2003). *The NDT Development Process*. The International Conference on Web Engineering. (ICWE 2003). Springer Verlag. LNCS 2722. Pág.463-467. ISSN: 0302-9743. Oviedo, España.
- 11 Escalona, M.J., Mejías, M., Torres, J., Reina, A.M. (2003). *NDT-Tool: A case tool to deal with requirements in web information systems*. The International Conference on Web Engineering. (ICWE 2003). Springer Verlag. LNCS 2722. Pág.212-213. ISSN: 0302-9743, Oviedo, España.
- 12 Escalona, M.J., Koch, N. (2004). *Requeriments Engineering for Web Applications: A Comparative Study*. Journal on Web Engineering, Rinton Press, 2004, Vol2. N°3. pp 193-212.
- 13 Escalona, M.J., Villadiego, D., Torres, J., Mejías, M (2004). *Aplicación de NDT al sistema para el reconocimiento, declaración y calificación del grado de minusvalía*. Report Interno. Departamento de Lenguajes y Sistemas Informáticos. Universidad de Sevilla.
- 14 Instituto Andaluz de Patrimonio Histórico. www.juntadeandalucia.es/cultura/iaph/
- 15 Jacobson, I. (1995). *Modeling with Use Cases: Formalizing Use Case Modelling*. Journal of Object-Oriented Programming,
- 16 Koch, N. (1999). *A Comparative Study of Methods for Hypermedia Development*. Technical Report 9905. Ludwig-Maximilian-University, Munich, Germany.
- 17 Koch, N. (2001). *Software Engineering for Adaptive Hypermedia Applications*. Ph. Thesis, FAST Reihe Softwaretechnik Vol(12), Uni-Druck, Munich, Germany
- 18 Kruchten, P. (1998). *The Rational Unified Process*. Addison Wesley
- 19 Lee, H., Lee, C., Yoo, C. (1998). *A Scenario-based Object-oriented Methodology for Developing Hypermedia Information Systems*. 31st Annual Conference on Systems Science. Sprague R.
- 20 Liu, L., Yu, E. (2001). *From Requirements to Architectural Design using Goals and Scenarios* Proceedings of the 6th Micon Workshop. Canada.
- 21 SADIEL. Tecnologías de la Información, Telecomunicaciones e Ingeniería. www.sadiel.es
- 22 Schwabe D., Rossi G. (1998). *Developing Hypermedia Applications using OOHDM*. Workshop on Hypermedia Development Process, Methods and Models, Hypertext '98, Pittsburg, USA.
- 23 UWA (2001), *UWA Requirements Elicitation: Model, Notation, and Tool Architecture*. www.uwaproject.org