Legal Requirements Reuse: A Critical Success Factor for Requirements Quality and Personal Data Protection [1]

Ambrosio Toval
Department of Informatics and Systems. University of Murcia
Murcia, 30071, Spain
atoval@um.es

Alfonso Olmos
Department of Informatics and Systems. University of Murcia.
Murcia, 30071, Spain
aom2@alu.um.es

Mario Piattini
Department of Informatics.
University of Castilla-La Mancha.
Paseo de la Universidad, 4 -
Ciudad Real, 13071, Spain
mpiattini@inf-cr.uclm.es

Abstract

Information Technologies misuse has increased the vulnerability of personal data, which has lead to growing concern about issues of personal privacy among political leaders, IT managers, information security consultants and the millions of people currently online. Many countries have developed, or are preparing, Laws and Regulations to combat the related threats and to guarantee Personal Data Protection. Despite efforts to construct secure systems, few papers have, as yet, focused on security from the very outset of the system development life-cycle. This paper presents a pragmatic proposal to incorporate the legal and regulatory measures to guarantee Personal Data Protection as a part of the requirements engineering process, instead of an addendum to system deployment. The authors investigate how recent efforts in the Requirements Engineering field can contribute to improving security issues in Information Systems, in particular those dealing with Personal Data. A reusable collection of security requirements and, as a novelty, Personal Data Protection requirements (including information on related software components links) are provided. The pre-defined requirements, together with a simple process model based on requirements reuse, provide a strategy that organizations can use to become privacy-compliant.

1. Introduction

People concerned with automatically managed personal data - i.e. most of us, the hundreds of millions of people currently on line- are beginning to realize just how important these data are, and how careless or malicious use, on behalf of the organizations hosting them, can harm our interests. While not yet afraid of "Big Brother" as portrayed by Orwell in his novel "1984" [25], we do, however, detect many "little brothers" who are threatening our privacy in the form of Internet or local unauthorized access to the Information System (IS), illegal dissemination for commercial purposes, unauthorized disclosure, modification, or loss of use, etc. This situation can be made even worse by sharing and mining knowledge about people, often in a negotiated business relationship. These issues also constitute a barrier to the widespread of e-commerce, and other e-initiatives.

In countries which have personal data protection (PDP) laws, software dealing with personal data has to fulfill the regulations in force. For instance, many countries in the EU, among them Spain, UK and Italy, have developed a number of legislative initiatives to protect citizens' privacy from misuse and to prosecute and punish offenders. Naturally, these regulations also take into account the necessary measures to allow access to personal data in the interest of the community or to preserve national security issues. For these laws to be put into practice we need databases, data warehouses, IS and software to comply with a basic set of security requirements regarding the regulations in effect.

A software development method that includes these regulations, within its software development life-cycle, will favor the resulting information systems accomplishing the required measures to ensure law compliance. Therefore, likely liability of and penalties to the organizations responsible for their protection would be minimized or avoided.

As tangible products resulting from this research, we obtain: 1) a process model to guide software developers, which is based on requirements reuse; 2) a reusable requirements document template, constructed taking into account the most prominent related international standards such as IEEE 830-1998 [5] and IEEE 610.12 [6]; and 3) a reusable set of specific requirements for PDP and the

[1] Partially granted by the Spanish Ministry of Science and Technology, project TIC2000-1673-C06-02 SIRENrm.

related security aspects, which is compliant with our national legislation and related European directives.

The information needed to identify and write the requirements mentioned in point (3) has been obtained, mainly, from the PDP Spanish legislation in use, namely the Constitutional Law 15/1999, (LOPD) [2], the Spanish personal data privacy law and related regulations. LOPD is an adaptation of the EU Directive 95/46/CE [3], on PDP. As a consequence of this Directive, all the European Union states, have had to issue similar regulation provisions.

After this introduction, section 2 provides some reasons for the work carried out. Section 3 describes the structure and main features of the *PDP Reusable Requirements Catalog*, and is illustrated by several examples. In section 4 the SIREN *Process Model* is briefly outlined together with the guidelines on applying the Catalog in IS projects. Finally, the main conclusions and an outline of the work to be done in the future, are given.

2. Motivation

Considering Security from the very outset of the system development has of late begun to be appreciated, in particular in the system requirements specification phase: Chung [28] describes a proposal to include security at design and Lutz [27] studies the role of software requirements in safety-critical embedded systems. There is also an increasing concern in taking the security constraints into account from the beginning of the project in order to avoid further security patches and to get things right from the start [33]. Worthy of mention too are some public IS methods which have been integrated with security-related ones, e.g. SSADM with CRAMM, and MÉTRICA3[2] with MAGERIT. However, we believe they lack sufficiently accurate guidance to carry out the PDP security-related RE activities. This shortage was experienced by the authors themselves in a one-year Risk Analysis Project in our Regional Administration [30], which involved MAGERIT and METRICA3 developed IS projects.

As for Personal Data Protection, apart from a preliminary discussion on this approach by the authors [29], results related to the systematic production of requirements to protect personal data have not, to the best of our knowledge, been published yet. It is worth mentioning again that the inclusion of these requirements from the first stages of the system life cycle makes the system law-compliant from the beginning, and not as a subsequent addendum, and thus it increases productivity and security aspects. Moreover, reusing these requirements helps to increase their quality: inconsistency errors, ambiguity and other problems can be detected and corrected for an improved use in subsequent projects. As Fitzpatrick [12] remarks, quality has been limited to usability excellence and technical excellence, however, a number of developments are forcing a change in perspectives. Information technology has had to be subjected to the rigors of the law and new legislation which software products must comply with.

The results of the research described in this paper are offered in the hope that they may interest, mainly, software, requirements and quality engineers, in charge of developing IS projects involving personal data, and practitioners in the area of IS security requirements. In countries where Personal Data Protection is enforced by law, IT managers and IS executives will also be interested. For example, Spanish law is particularly strict, with fines of up to $700.000, or strict disciplinary methods for not compliance.

3. The Personal Data Protection Requirements Catalog

In this section we will explain the structure proposed to organize the requirements documents, and how the first version of the Security and PDP requirements Catalog is filled with reusable requirements.

3.1. Requirements Engineering and security

For a given project, a project requirements document (PRD) is, in a wide sense, the formal statement of the system and software requirements of the project. Tipically, the PRD is divided into a hierarchical collection of related sub documents. Each document in this hierarchy should correspond to a different specification level and, therefore, it must have different objectives and users [19]. The PRD may contain a huge variety of kinds of requirements. Although there exist many taxonomies of the different types of requirements [4], [5], [20], [21], there is a general agreement in making a rough functional/ non-functional division of the requirements in a project. Functional (or behavioral) requirements define what the system does, while non-functional requirements define the quality attributes of the system as it performs its job [20].

In the RE community, security requirements have been traditionally considered as non-functional. Moreover, security is too frequently considered as a vague goal to be satisfied [18], and a precise description and enumeration of specific security properties and behavior is often missing. On the other hand, in the Security practicing community, security issues entail both kinds of functional

[2] MÉTRICA3 is the standard IS development method of the Spanish public administration, which is similar to SSADM. MAGERIT is the risk analysis and management method of the Spanish public administration, which conforms to the ISO/IEC 15408 (Evaluation Criteria for Information Technology Security Standard).

and non-functional features. For instance, part 2 of the ISO/IEC 15408 (Evaluation Criteria for Information Technology Security Standard) is wholly devoted to security functional requirements. However, there is a current trend towards integration of these issues concerning both communities, as manifested for instance by the annual SREIS (Symposium on Requirements Engineering for Information Security) workshops (http://www.sreis.org/), and a series of talks and papers in recent issues of REJ and the last RE conference [30], [31], [32]. Consequently, our PDP Catalog contains both kinds of functional and non-functional security requirements (the list of requirements in section 3.3 contains examples of both types). In the research reported in this paper, we focus on those security requirements that are considered in, or are derived from, the PDP regulations mentioned above. In this paper, we will use the names PDP requirements or Security requirements, indistinctly, to refer to the requirements in the PDP Catalog. It is worth mentioning that the set of general Information Systems Security requirements (such as those referred to in ISO/IEC 15408) is bigger, and has already been considered by the authors in another reusable catalog [30], with a different purpose.

Requirements, in the PRD, are accompanied by its attributes, which provide meta-information on the particular requirement. For example, the IEEE 1233 standard [4], recommends the following: identification (unique), priority, criticality, viability, risk, source, and other project dependent ones, such as maintainability, performance and reliability. A particular type of interesting information on the requirements comes from the notion of traceability. In RE, traceability refers to the clarity in determining the origin of each requirement and to the facility of referencing each requirement in future development or enhancement documentation [5]. Traceability is also defined as the capacity to describe and track the life of a requirement in two directions, forwards and backwards. Thus, the life of a requirement can be understood from its origin, through its development and specification until its deployment and use, including its periods of iteration and refinement [26]. Figure 3 provides some examples of traces between requirements.

On the other hand, software reuse is a well known technique in Software Engineering for developing quality systems in a productive way. The benefits of reuse are greater when the abstraction level is increased, and not only code, but also designs and specifications, are reused [23], [24]. Reuse of requirements models has been identified as a major trend in RE [22].

The description and classification of requirements for reuse can be performed in several ways. Furthermore, requirements can be expressed in many different forms [21], according to their formal representation level (e.g. natural language, use cases, scenarios, mathematical notation), writing style (e.g. declarative versus procedural), granularity, and so on. In previous research we selected a formal representation for requirements [14], [16], [17] because the main goal then was the rigorous detection of inconsistencies. However, for the production and reuse of a requirements specification document in the context of the IS projects involving PDP issues, we have selected natural language (which is also used by some of the main standards in the field [4], [5]) to express requirements. Many non-technical stakeholders[3] involved in these projects prefer this "digestible" way of writing requirements, for their reading, analysis and discussion, and this, in turn, encourages participation. In order to write good requirements and to avoid the ambiguity inherent in natural language**,** we apply and recommend the rules given in [4], [5], [9] and [11].

3.2. The PDP Catalog hierarchy

In order to establish a systematic RE process in an organization, a definition of the kind and number of RE documents together with the definition of inter-relationships between them, the so-called hierarchy of RE documents should be previously adopted. While it is preferable that this definition becomes standard in the organization, the configuration of documents can vary from one project to another, but the arrangement of RE documents chosen for a particular project must be clear from the beginning.

In SIREN (SImple REuse of requiremeNts), the process model used in this approach, both the PDP Catalog and the PRD in use share a common hierarchy of sub documents. Initially, a hierarchy of empty specification templates is established. These templates are filled with generic requirements, coming, in the general case, from different application domains. For Security and PDP realms, a number of official related documents have been taken into account as sources (as described in section 3.3), obtaining the PDP Catalog as a result. From this catalog, the analyst will complete the PRD according to the RE process model described in section 4. The main templates (some of them are optional) are listed below:

- **SyRS**. System Requirement Specification, according to the IEEE Std. 1233-98 standard. System requirements should be included in this template, dealing -in high abstraction level- with aspects of administrative requirements, organization, human resources, physical devices and other elements related to protection of data [4]. It is a mandatory document.

[3] In this context, system stakeholders include clients, end-users, developers, the person in charge of security or external bodies such as regulators and certification authorities [20] [24].

- **SRS.** Software Requirement Specification, according to the IEEE Std. 830-98 standard. Software requirements should be included in this template [5]. It is also mandatory.
- **STS.** Software Test Specification. A way of checking the fulfillment of each requirement on the final product should be defined for all requirements identified in the SRS. It is optional, although strongly recommended.
- **SyTS.** System Test Specification. Each requirement identified in the SyRS should be checked on the final product. This specification is optional, but recommended.
- **IRS.** Interface Requirement Specification, according to the standard DoD MIL-STD-498. It deals with communication interfaces, graphic user interfaces and interfaces between software devices. It is optional.

In PDP, technical security measures, as well as administrative and contextual ones, are necessary. This is particularly true in the Security field. Many disasters have their origin in insignificant hardware components, a small piece of code, untrained staff, lack of emergency guidelines, or any other element of the IS. As mentioned above, the hierarchy can be adapted by the project leader or analyst to the particularities of each project. In this case, it is sufficient to select and export the set of single reusable requirements, and their relationships, from the PDP Catalog to the new location. The hierarchy of documents is shown graphically below (Figure 1):

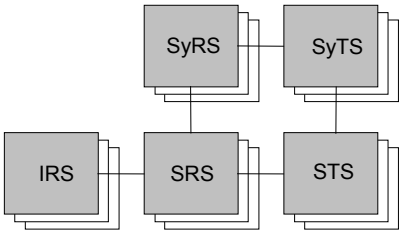

Figure 1. Requirements Catalog structure

3.3. Security and PDP reusable requirements

The PDP Catalog contains generic requirements related to PDP regulations in use, which impose security rules and constraints. The set of requirements concerning Personal Data Protection shown in this paper, and included in the aforementioned filled template, has been mainly obtained from the so-called "Security Measures Regulations of Automated Files" -[1]- and the Constitutional Law 15/1999, (LOPD), the Spanish personal data privacy law [2]. As mentioned above, LOPD is an adaptation of the European Union Directive 95/46/CE [3].

LOPD, regulates the automated management of personal data, and establishes a taxonomy of personal data at three security levels, depending on the need to guarantee the respective categories of protection, namely: integrity, availability and confidentiality. These levels are: basic, medium and high. Each level implies the legal obligation of applying a set of security measures aimed at guaranteeing the categories of protection required. These sets of security measures are incremental in the sense that, e.g., measures to be applied to protect personal data at "medium" level include these to be applied to protect data at "basic" level, plus a subset of new measures. These measures entail not only IT requirements (firmware, hardware or software implemented), but also administrative measures such as organizational, personnel, physical, and procedural controls.

The security measures regulations [1] is a subsidiary legal document that refines the LOPD and makes it effective by specifying the technical measures to be implemented by all organizations managing this kind of private information. We have considered the three levels of security measures and have rephrased the contents of both documents (law and rules), to convert them into software and system requirements format in a traceable way and so that they are ready to be included in any IS project. As an immediate result, we have obtained a PDP Catalog filled with the security requirements necessary to protect personal data, according to the legislation or regulations in use. The catalog, and the requirements inside, have been defined with reuse in mind.

In addition, these initial sets of reusable requirements have been complemented with a collection of security requirements, in line with the adaptation of the ISO/IEC 15408 Evaluation Criteria for Information Technology Security Standard, provided by the MAGERIT method [7]. The ISO/IEC 15408 standard (also known as the *Common Criteria Framework* -CCF), is applicable to IT security measures implemented in hardware, firmware or software, but, unlike LOPD, it does not contain security evaluation criteria pertaining to administrative security measures not directly related to the IT security measures. Nevertheless, it is recognized that a significant part of the security of an IS can often be achieved through administrative measures. The consideration of all these sources (LOPD, the related security measures regulations and MAGERIT -which conforms to CCF) contributes to achieve a fairly complete set of security requirements to be applied in the protection of personal data.

Hereinafter we will use the term PDP requirements to refer to both PDP and all the related security requirements (see for instance requirements labeled SRS3531L53 and SRS3531L55 below). As mentioned above, the catalog contains both functional and non-functional requirements for this domain.

Requirements in the catalog can be *parameterized* in favor of reuse (these contain some parts that have to be adapted to the application being developed at the time; e.g. see SRS3531L52 below), or *non-parameterized* (can be applied directly to any project concerning the profiles and/or domains in the repository; e.g. see SyRS331L1).

From our point of view the reuse of legal requirements may become critical for success mainly because of:

1. The difficulty of understanding and extracting requirements directly from legal documents. Legal language is often difficult for non juristic experts to understand, and thus prone to misinterpretations.
2. The possibility of ensuring, beforehand, that the IS to be built - based upon the given set of reusable requirements- will fulfill the regulations in question.
3. High efficiency in the reuse of these requirements because of the similar PDP needs of the applications managing personal data.

Each requirement is identified by its label, according to the type it belongs to (data protection, security, databases, etc.). This label allows us to relate independent requirements through traceability relationships. In SIREN, traceability relationships are extended to establish links among requirements at the same or different levels and, by extension, to link requirements with subsequent software artifacts related to them [19]. Requirements included in the catalog have been established generically by using parameter-based mechanisms, according to the analysis carried out in [10].

Each requirement in SIREN is accompanied by a series of attributes which provide additional information (see Figure 2): PUID (requirement unique identifier inside the project), parents' PUIDs, children's PUIDs, current status, compliance, source, risk, verification method, verification documents, type, priority, who has carried it out, and software components covering it. These attributes have been identified taking into account some of the recommendations given in [4], [5] and [10].

Attributes can be filled manually using the associated template with each catalog requirement, or assisted by a CARE (Computer-Aided Requirements Engineering) tool such as RequisitePro©[8], which has been used for our purposes.

Regarding the "Software components links" attribute, note that, as an additional benefit, the PDP Catalog becomes a centralized (requirement-driven) repository on components links to help in the implementation of secure and solid IS dealing with Personal Data. This can be a starting point for the design of software development processes focused on the achievement of correctness in each step through the use of COTS components [34]. Some examples of requirements, taken directly from the catalog, together with the filled attributes template for one of them, are shown below.

The following are examples of software requirements (the term "file master" is coined by the LOPD to denote the person or entity responsible for the file):

SRS3421L46. *The file master shall choose a [physical device] in order to make data backups*

SRS3421L47. *The file master shall obtain the [storage unit] according to the chosen [physical device].*

SRS3531L51. *The file master shall take charge of obtaining the appropriate software in order to make data backups in the chosen [storage unit].*

SRS3531L52. *The data backup shall be made with the [X] software if we have the Operating System [Y].*

SRS3531L53. *The application shall have a software subsystem in order to implement an [identification procedure] and an [authentication procedure] to avoid unauthorized access to system.*

SRS3531L55. *The application shall use an [encryption algorithm] in order to ensure that passwords are stored in an unreadable way.*

SRS3533L58.*The application shall use an [encryption algorithm] in order to encrypt the data in the hardware devices that are going to be moved.*

There are others requirements in the Catalog, included in the SRS document, closest to the PDP such as:

SRS3531S102. *The application shall allow the retrieval of the personal data gathered and its source as well as the real or scheduled transference to third parties, in order to guarantee that the right of the interested party to access his/her personal data is upheld.*

SRS3531S103. *The application shall allow the cancellation of the personal data gathered (within the 10 days following the request of the interested party). Thereafter, the data will be only accessible by Public Administrations, Judges and Courts. Hence the personal data cancellation is merely a block of the data.*

SRS3531S105. *The application shall allow personal data rectification within the 10 days following the request of the interested party.*

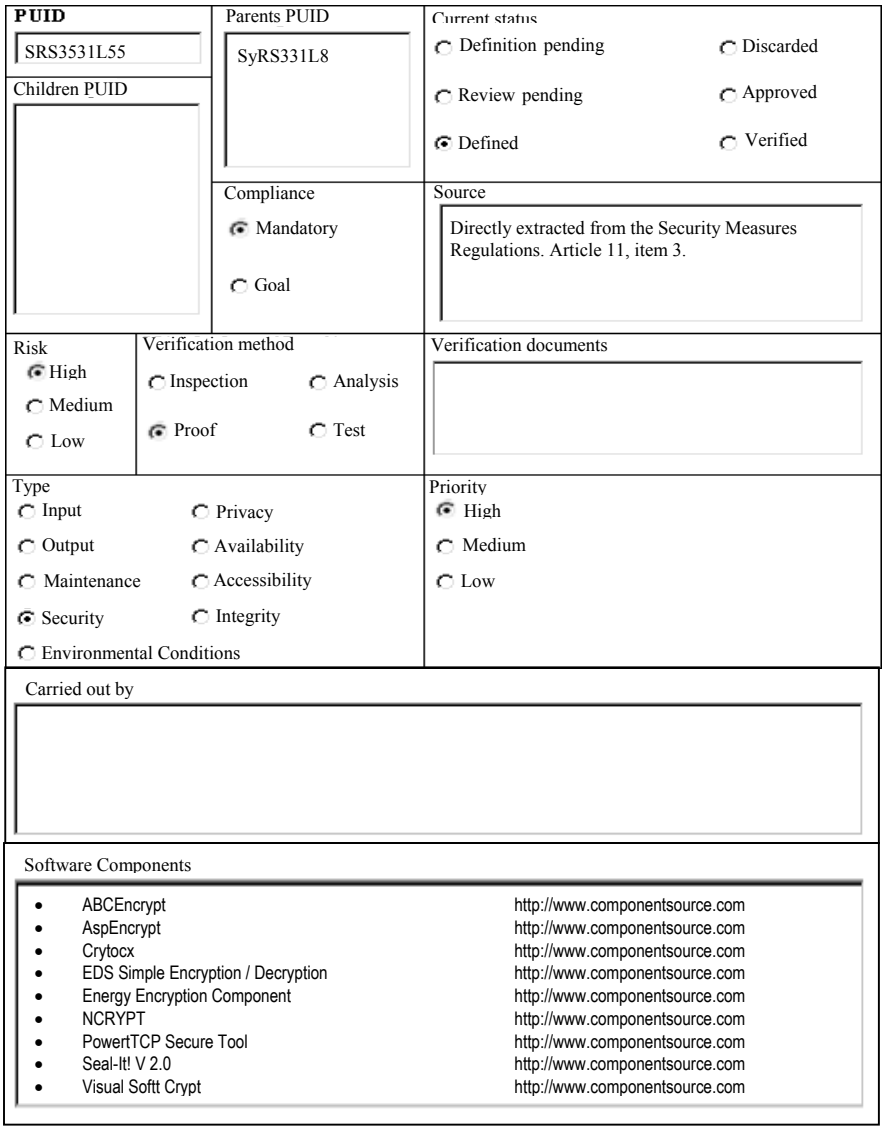

Figure 2. Example of a requirement (SRS3531L55) from the PDP Requirements Catalog

SRS3531S109.*The application shall allow the notification to the grantee of the cancellation/rectification of those data transferred to other entities.*

SRS3531S111. *The application shall keep the personal data gathered during the [deadlines as specified in the applicable laws, or in accordance with the contractual relationship between the person responsible for the data management and the interested party].*

SRS3531S113. *The application shall allow the notification to the interested party of the transfer to a third party of his/her personal data, the object of the file, the type of data transferred, and the name and address of the grantee.*

Some System Requirements examples:

SyRS331L1. *The file master shall draw up a security document by means of which the security regulation will be implemented. This document is mandatory for all personnel who can access the automated personal data. The document shall contain as a minimum the following:*

a) Document scope with a detailed specification of all protected resources.
b) Tasks, rules, procedures and standards aimed at ensuring the required security level.
c) Personnel duties and obligations.
d) The structure of the files containing personal data and the description of the information systems that manage them.
e) The procedure for notifying, managing and responding to incidences.
f) The necessary procedures for making data backups and the later restoring of the data.

SyRS331L2. *The file master shall state mechanisms to avoid users' accessing data with privilege and rights other than was granted.*

SyRS331L6. *The file master shall state an [identification procedure] and an [authentication procedure] in order to access the system.*

SyRS331L9. *The stated procedures for making data backups and restoring data shall ensure that the data can be restored in the same state as it was before the loss or destruction took place.*

SyRS332L28. *The [identification procedure] and the [authentication procedure] shall limit the possibility of repeated attempts of unauthorized access to the system.*

SyRS332L37. *Tests preceding the modification or implantation of the information systems which are going to manage files that contain personal data shall not be made with real data, unless the security level is guaranteed according to the type of the file managed.*

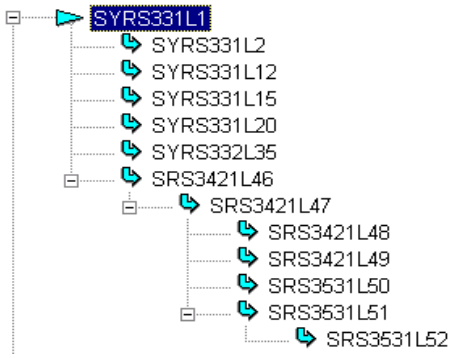

Figure 3. Example of a traceability tree

Figure 3 shows a partial traceability tree, as an example of traceability relationships existing in the PDP Catalog.

4. The SIREN Requirements Engineering process model

The requirements reuse process model provided, SIREN, is a general purpose one, and can be applied to a variety of domains and fields, for instance Security, Databases, Electronic Commerce, etc, provided that the corresponding filled catalogs are available. More details on this RE approach can be found in a related paper [30]. A novel feature of SIREN when applied to the Personal Data Protection domain, is that it systematically incorporates the security aspects in the IS concerned from the very beginning of their development, and not as a result of *a posteriori* Security or Risk Analysis.

This process has already been tested in the context of two Security and IS development-related projects in our regional government: a one-year project in the IS and Communications Government Department [30] and the other in the Ministry of Labor and Welfare of the Autonomous Region of Murcia.

The SIREN process is based on the following two elements:

- A catalog of requirements, which is reusable in any software project (initially empty or containing requirements of different domains, coming from previous developments or studies). This catalog is made of a hierarchy of specification documents, according to standards of the software industry (i.e. the IEEE standards, as described before).
- The specific requirements related to the specific project we are dealing with.

Starting from that initial information, the Requirements Engineer will develop a specification of requirements with the same template structure as the Catalog of Reusable Requirements. Basically, the specification for the project will consist both of specific requirements directly imported from the project and requirements coming from the catalog. The latter are used in two main ways: either by instantiating a generic requirement of the catalog of requirements according to the necessities of the project or by incorporating them directly. The requirements definition is mainly textual, but the addition of use case models [13] and other alternative graphic notations is allowed. Developers merely have to choose the suitable set of requirements from the catalog and instantiate them to the current project.

SIREN is concerned with how to make the work done in a phase (requirements specification phase) reusable, and how to reuse it in other development phases [18]. To summarize, the process consists of the following steps:

1. The Catalog is initially filled with a collection of requirements for reuse by using the hierarchy, standards recommendations and the security and PDP sources mentioned in section 3. This step produces a first version of the PDP Catalog.
2. The PDP Catalog is used to select and instantiate the generic requirements that are suitable for the project under study. They are put together with the new project specific requirements, after analyzing and resolving possible conflicts. A first version of the current PRD, including both reused (and possibly adapted) requirements and the new ones is obtained.

3. The PRD is submitted for the stakeholders approval. If changes are required, a new iteration begins. This iteration ends when the PRD is formally approved. Note that this step is necessary in order to continue with the rest of software development activities. However, the PRD is not "frozen", but subject to further changes.
4. Once validated, the PRD is taken as the basis for subsequent development phases, depending on a particular method or approach.
5. In order to enhance the quality of the PDP Catalog, we propose an explicit task of "improvement", to introduce new requirements or to correct ambiguities, inconsistencies and errors detected in the generic requirements during the steps above. The Catalog is thus gradually improved for future reuse.

It is worth noting that once the PRD has been validated and accepted, SIREN does not impose any particular software development strategy. In the examples above, we have assumed that a Component Based Development (CBD) approach is followed as software development strategy.

5. Conclusions and further work

This paper has presented a requirements process model, based upon reuse, together with a reusable template to organize the requirements document of any organization and a catalog filled with reusable personal data security requirements. All these elements are compliant with the Spanish personal data privacy law. The strategy presented can be extrapolated to other States' legislation, particularly EU member states, as these share a common source (the European Union Directive 95/46/CE). Therefore, any information system including these PDP requirements must pass an audit aimed at verifying compliance with the legislation. This is an excellent area for reuse because of the presence of a widely accepted PDP principles core in spite of the particular considerations of each law and standard.

This approach can substantially improve the productivity of software development teams, with regard to system managing personal data, as they can start from a set of predefined requirements in a software and system language, rather than from legal jargon. Quality of the security requirements specification is also enhanced because the generic requirements included in the catalog are analyzed and potentially improved after each project. In addition, the extension of the traceability notion to link directly security requirements with software components enables the inclusion, in the reusable requirements Catalog, of the information on related security software components that totally or partially implement the requirements.

The Requirements Catalog is also a useful source to define: 1) the security documents necessary (those defining particular security policies in the organization, as required by many personal data privacy laws); 2) similar security documents as required by other regulations or standards (like the MAGERIT method, or ISO 15408, *Common Criteria Framework,* on general security and safety requirements); 3) security audits policies and realization [20]. In particular, the PDP Requirements Catalog described in this paper has been used to develop the security document established by the Spanish personal data privacy law in a High School in our region [15] (which manages confidential personal data on teachers, students and administrative staff, such as medical data, political affiliation -trade unions-, family status, and so on), in addition to its use as a source of system and software reusable personal data security requirements.

To date, two other projects following this strategy (already mentioned in section 4) have been completed: the first one in the IS and Communications Government Department and the second in the Ministry of Labor and Welfare of the Autonomous Region of Murcia. We are currently undertaking a new project, involving personal data, aimed at the harmonization of existent and new applications and databases for the regional Ministry of Agriculture.

As future work, we plan to extend the SIREN process to deal with the UML representation of the requirements, in order to integrate it with a more general Validation and Verification Process Model and consider automated inconsistencies detection. This will include previous research on formal verification of UML models [14], [17]. Finally, we are working on the development of a framework aimed at obtaining software components that fulfill the reusable requirements of the Catalog and procedures to prove the degree of suitability of those components.

6. Acknowledgements

The authors would like to thank the anonymous reviewers for their helpful comments on this paper and the Spanish Ministry of Science and Technology which is partially supporting this research.

7. References

[1] Spanish Royal Decree 994/1999, June 11th, by means of which the Security Measures Regulations of Automated Files which contain personal data is approved. BOE no. 151, 25/06/1999, page 24241 (In Spanish)

[2] Spanish Constitutional Law 15/1999, December 13th, on Personal Data Protection. BOE no. 298, 14/12/1999 (In Spanish)

[3] Directive 95/46/CE of the European Parliament and Council, dated October 24th: about People protection regarding the personal data management and the free circulation of these data. DOCE no. L281, 23/11/1995, P.0031-0050

[4] Software Engineering Standards Committee of the IEEE Computer Society: IEEE Guide for Developing System Requirements Specifications. IEEE Std. 1233-1998

[5] Software Engineering Standards Committee of the IEEE Computer Society: IEEE Recommended Practice for Software Requirements Specifications. IEEE Std 830-1998

[6] Software Engineering Standards Committee of the IEEE Computer Society: IEEE Standard Glossary of Software Engineering Terminology. IEEE Std 610.12 – 1990

[7] Spanish Ministry of Public Administration: MAGERIT Version 1.0. Information System Risk Analysis and Management Methodology. 1996 (In Spanish)

[8] Requisite Pro. Rational Software. http://www.rational.com

[9] Pradip Kar, Michelle Bailey: Characteristics of Good Requirements. INCOSE 1996, prepared by Requirements Working Group of the Int. Council on Systems Engineering

[10] W.Lam, J.A. McDermid, A.J.Vickers: Dep. of Computer Science, University of Hertfordshire and Rolls-Royce University Technology Center, Dep. of Computer Science, The University of York. Ten Steps Towards Systematic Requirements Reuse. Requirements Eng (1997) 2: 102-113

[11] William M. Wilson: Writing Effective Natural Language Requirements Specifications. Feb 99. *Crosstalk*. The Journal of Defense Software Engineering. STSC. http://www.stsc.hill.af.mil/crossTalk/1999/feb/wilson.asp

[12] Fitzpatrick, R. Strategic Drivers of Software Quality: Beyond External and Internal Software Quality. Proc. of the AQAPS Int. Conf.. IEEE Comp. Society, 65-72, 2001.

[13] García, J., Ortín M. J., Moros, B. Nicolás, J. and Toval, A.: Towards Use Case and Conceptual Models through Business Modeling. Laender, A.H.F., Liddle, S.W., Storey, V.C. (eds.): Conceptual Modeling - ER 2000. Lec. Notes in Comp. Sci., Vol. 1920,. Springer-Verlag (2000) 281-294

[14] Toval, A. and Fernández, J.L.: Improving System Reliability via Rigorous Software Modeling: The UML Case. Proc. of the 2001 IEEE Aerospace Conf. (Track 10: Software and Computing), Montana, USA IEEE Computer Society March 10-17, 2001

[15] Olmos, A.; Toval, A.: Reusing Personal Data Protection Requirements with SIREN. Tech. Rep. Dep. of Informatics and Systems. University of Murcia. Spain. 2001

[16] Nicolás, J.; Toval, A.; Arenas, A. and Alcalde, J., Formal Validation and Verification of Atomic Resolution Microscope Control and Topography, Cybernetics and Systems, Volume 32, Number 8, December 2001

[17] Fernández, J. L. and Toval, A.: Can Intuition Become Rigorous? Foundations for UML Model Verification Tools. Proc. of the Int. Sym. on Software Reliability Eng. (ISSRE 2000), IEEE Computer Society Press, Oct. 8-11, 2000

[18] Zave, P.: Classification of Research Efforts in Requirements Engineering, ACM Computing Surveys, vol. 29, n. 4, pp. 315-321, 1997

[19] Gabb A.: The Requirements Spectrum. In: INCOSE (International Council on Systems Engineering) Eds. First Regional Symposium of the Systems Engineering Society of Australia, SE'98. Australia. 1998

[20] Robertson S & J.: Mastering the Requirements Process. Addison–Wesley. 1999

[21] Pohl, K.: Requirements engineering: An overview. Technical Report TR 96/2, CREWS, 1996

[22] Nuseibeh, B. y Easterbrook, S.: Requirements Engineering: A Roadmap.Proc. of 22nd Int. Conf. on Software Engineering (ICSE'00), Limerick, Ireland, IEEE Computer Society Press (2000)

[23] Cybulsky JL, Reed K.: Requirements Classification and Reuse: Crossing Domain Boundaries. In: 6th Int. Conf. on Software Reuse, ICSR'2000. Springer, Lecture Notes in Computer Science. Viena. 2000

[24] Kotonya, G. y Sommerville, I.: Requirements Engineering. Processes and Techniques, John Wiley & Sons, 1998.

[25] Orwell, G.: Nineteen Eighty-Four. Martin Secker & Warburg Ltd. 1949

[26] Gotel, O. and Finkelstein, A.W.: An Analysis of the requirements traceability problem. Proceedings of the Int. Conf. on Requirements Engineering, Colorado Springs Co, 1994, IEEE Computer Society Press, pp. 94-102

[27] Lutz R.: Analyzing software requirements errors in safety-critical embedded systems. Proceedings of RE'93, San Diego, CA, pp. 126-33

[28] Chung L.: Dealing with Security Requirements during the development of Information Systems. In: Rolland C, Bodat F. and Cauvet C. (eds.). Advanced Information Systems Eng., Proc., 5th Int. Conf. CAiSE '93. Berlin: Springer Verlag. Paris. 1993. pp. 234-251

[29] Toval, A., Olmos, A., Rodero, J.A. "The Role of Requirement Reuse in Protecting Personal Data", Proc. of the 5th World Multiconference on Systemics, Cybernetics and Informatics (SCI 2001), Vol. VII, Computer Science and Engineering: Part I, ("Database Security invited session") Orlando, Florida (USA), July 2001.

[30] Toval, A., Nicolás, J., Moros, B., García F., Requirements Reuse for Improving Information Systems Security: A Practicioner's Approach, Requirements Engineering Journal, vol. 6, n. 4, pp. 205-219, 2002 Springer-Verlag.

[31] Gene Spafford.The Hidden Meta-Requirements of Security and Privacy (keynote talk). Procs. of the Fifth IEEE Int. Sym. on Requirements Engineering, August, 27-31, 2001, Toronto, Canada.

[32] Annie I. Antón, Julia B. Earp, Colin Potts and Thomas A. Alspaugh. The Role of Policy and Stakeholder Privacy Values in Requirements Engineering. Proc. of the Fifth IEEE Int. Sym. on Requirements Engineering, August, 27-31, 2001, Toronto, Canada.

[33] Hall, Anthony and Chapman, Roderick. Correctness by Construction: Developing a Commercial Secure System. IEEE Software Jan/Feb 2002, pp. 18-25

[34] Ghosh, Anup K. Howell, Chuck. Whittaker, James A. Building Software Securely from the Ground Up. IEEE Software Jan/Feb 2002, pp. 14-16