15 May, Trento (Italy)

SREP

Security Requirements Engineering Process

Speaker:

David Garcia Rosado
GSyA Research Group, University of Castilla La-Mancha (Spain)

Authors:

Dr. Daniel Mellado Fernández
GSyA Research Group, University of Castilla La-Mancha (Spain)

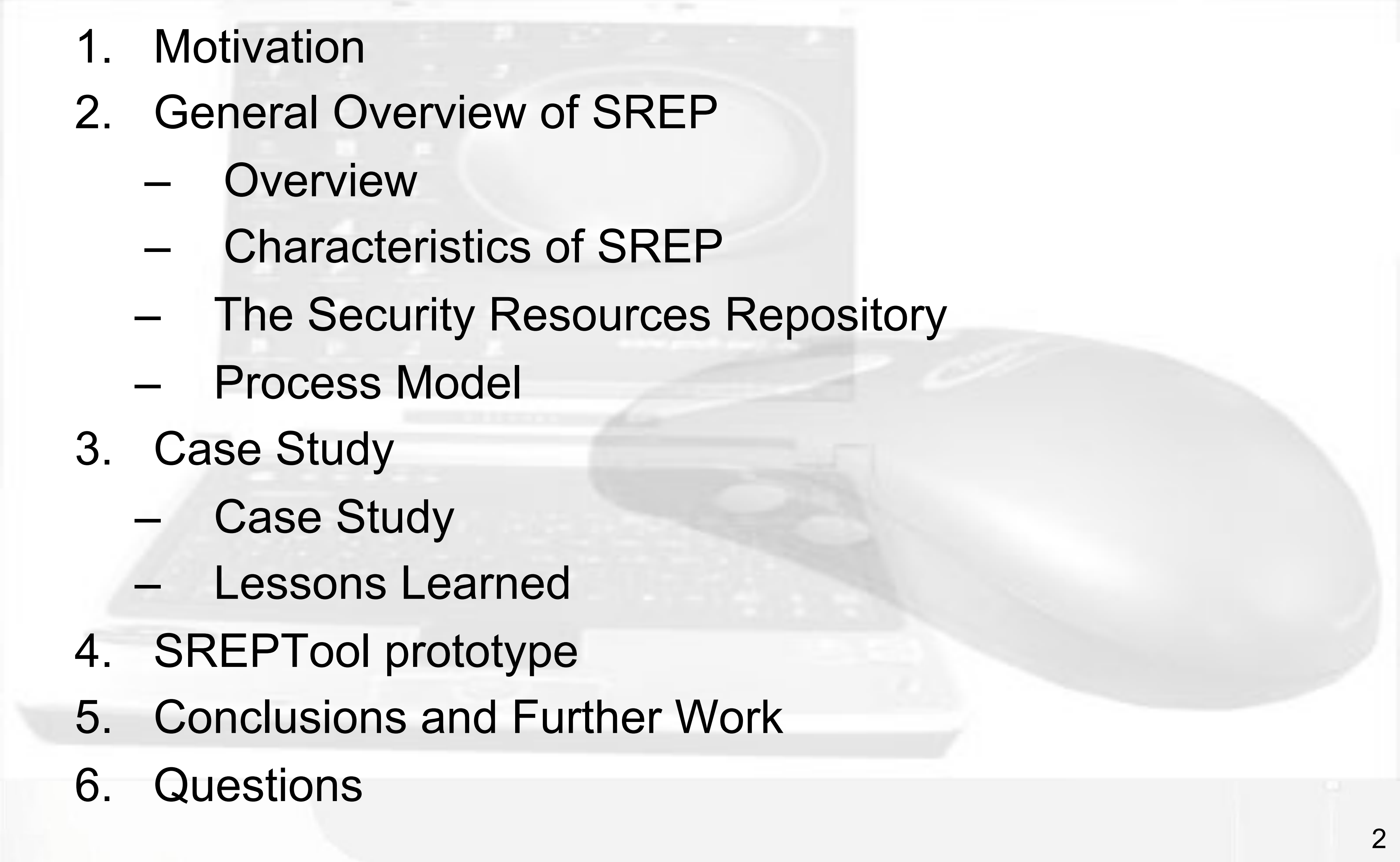
Dr. Eduardo Fernández-Medina
GSyA Research Group, University of Castilla La-Mancha

(Spain)

and

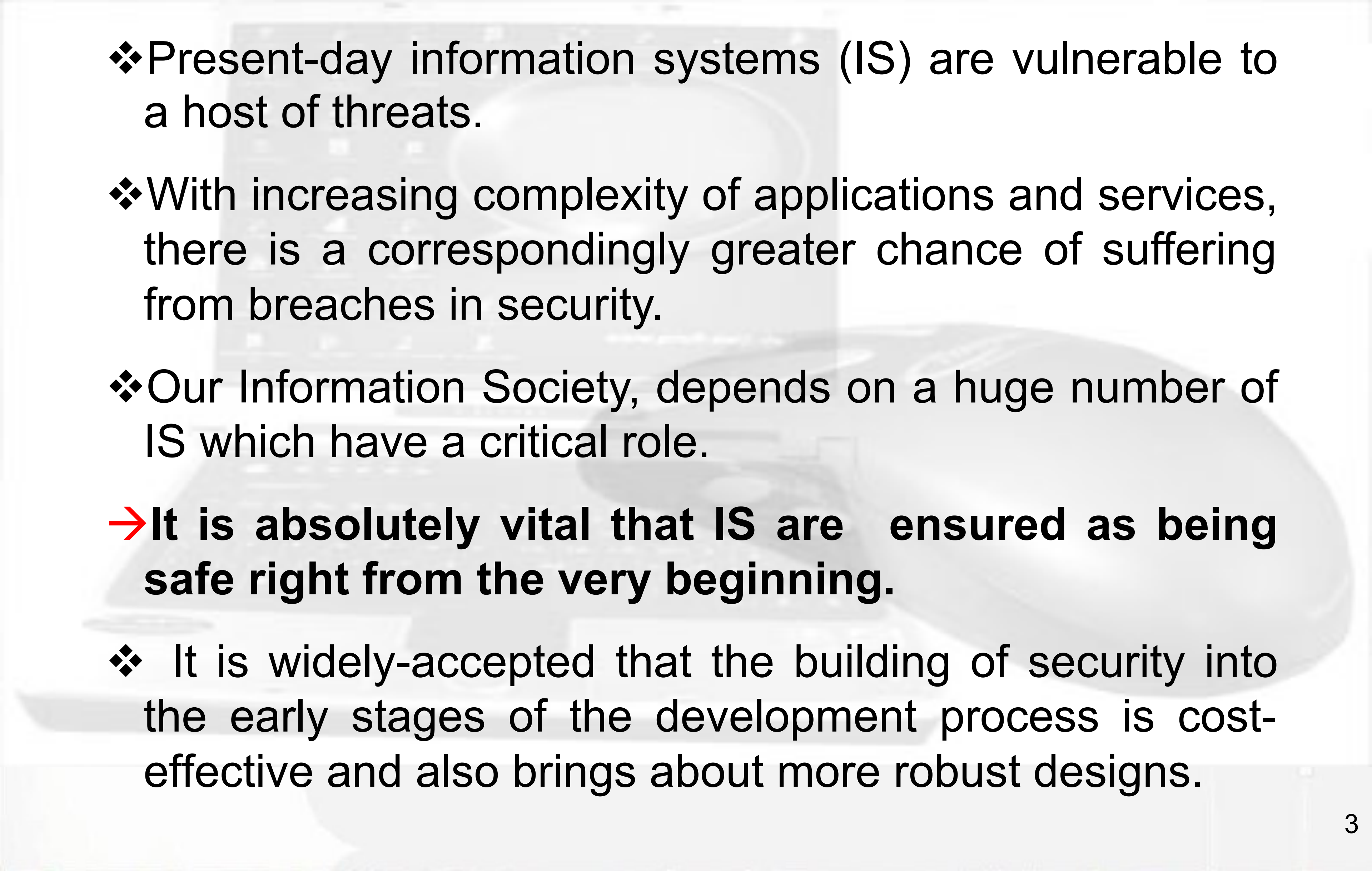
Dr. Mario Piattini
Alarcos Research Group, University of Castilla La-Mancha (Spain)

CONTENTS



1. Motivation
2. General Overview of SREP
   - Overview
   - Characteristics of SREP
   - The Security Resources Repository
   - Process Model
3. Case Study
   - Case Study
   - Lessons Learned
4. SREPTool prototype
5. Conclusions and Further Work
6. Questions

Motivation



❖Present-day information systems (IS) are vulnerable to a host of threats.

❖With increasing complexity of applications and services, there is a correspondingly greater chance of suffering from breaches in security.

❖Our Information Society, depends on a huge number of IS which have a critical role.

→**It is absolutely vital that IS are ensured as being safe right from the very beginning.**

❖ It is widely-accepted that the building of security into the early stages of the development process is cost-effective and also brings about more robust designs.

Motivation

- ☹ However, in the majority of software projects security is dealt with when the system has already been designed and put into operation.

- ☹ The requirements specification phase is often carried out with the aid of just a few descriptions, or the specification of objectives are put down on a few sheets of paper.

- ☹ Many developers tend to describe design solutions in terms of protection mechanisms instead of making declarative propositions regarding the level of protection required.

Motivation

- ❖As it is common that security requirements are undervalued and not well understood

→A very important part of the achieving of secure software systems in the software development process is that known as **Security Requirements Engineering**,

- →which provides techniques, methods and norms for tackling this task in the IS development cycle

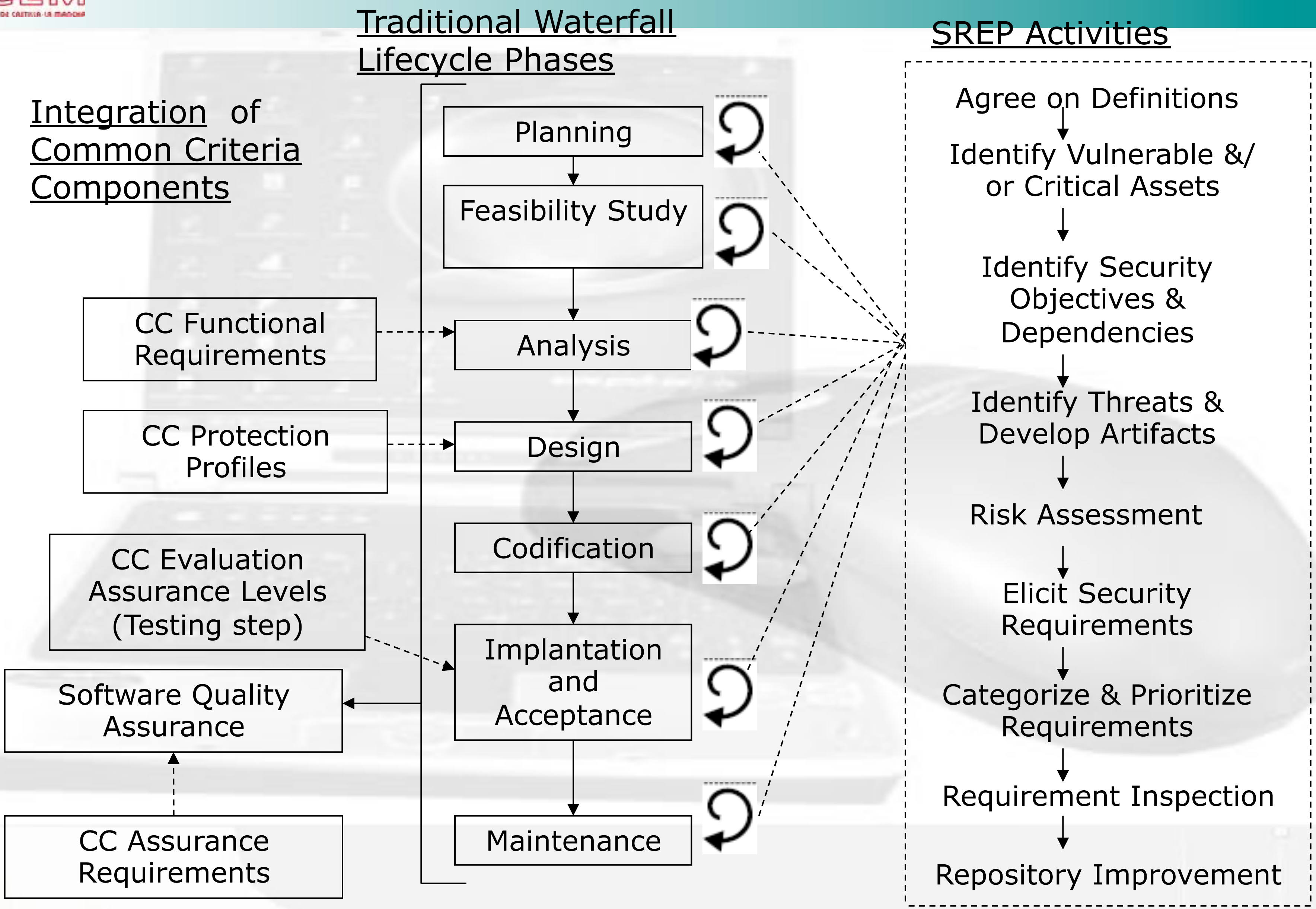
Motivation

❖After having performed a comparative analysis of several relevant proposals of IS security requirements

❖Conclusion: those proposals did not reach the desired level of integration into the development of IS, nor are specific enough for a systematic and intuitive treatment of IS security requirements at the early stages of software development.

SREP: deals with the security requirements at the early stages of software development in a systematic and intuitive way, it is based on the reuse of security requirements, together with the integration of the Common Criteria and the use of specific techniques within the scope of Security Requirement Engineering

A General Overview of SREP



Integration of Common Criteria Components

Traditional Waterfall Lifecycle Phases

SREP Activities

CC Functional Requirements

CC Protection Profiles

CC Evaluation Assurance Levels (Testing step)

Software Quality Assurance

CC Assurance Requirements

Planning

Feasibility Study

Analysis

Design

Codification

Implantation and Acceptance

Maintenance

Agree on Definitions

Identify Vulnerable &/ or Critical Assets

Identify Security Objectives & Dependencies

Identify Threats & Develop Artifacts

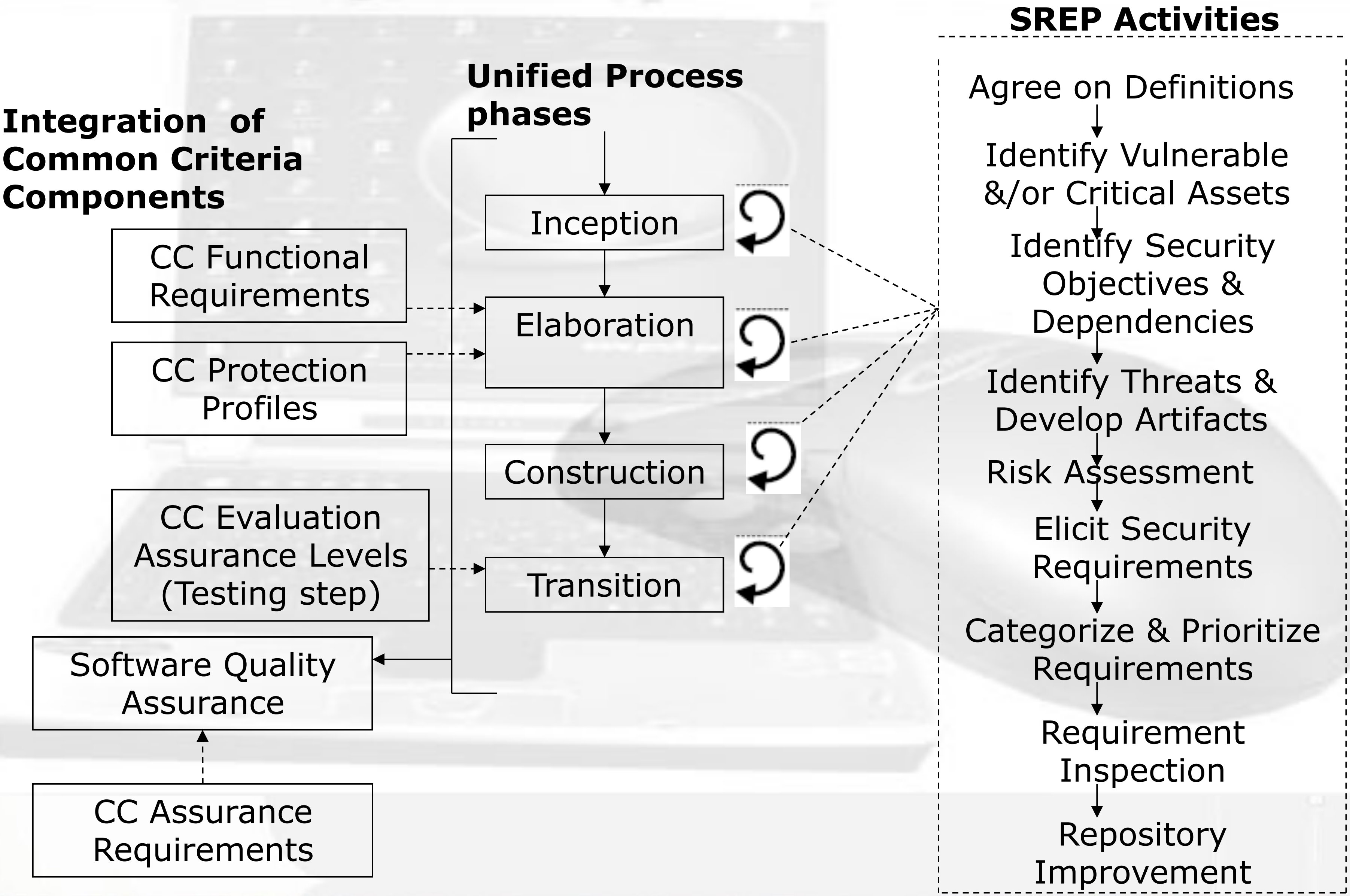
Risk Assessment

Elicit Security Requirements

Categorize & Prioritize Requirements

Requirement Inspection

Repository Improvement



SREP Activities

Agree on Definitions

Identify Vulnerable &/or Critical Assets

Identify Security Objectives & Dependencies

Identify Threats & Develop Artifacts

Risk Assessment

Elicit Security Requirements

Categorize & Prioritize Requirements

Requirement Inspection

Repository Improvement

Unified Process phases

Inception

Elaboration

Construction

Transition

Integration of Common Criteria Components

CC Functional Requirements

CC Protection Profiles

CC Evaluation Assurance Levels (Testing step)

Software Quality Assurance

CC Assurance Requirements

General Overview of SREP

- Asset-based and risk-driven method.
- It describes how to integrate the CC into the software lifecycle model.
- Reuse of security requirements, assets, threats, security objectives, countermeasures → security resources repository
- The core of SREP is a micro-process, made up of nine steps which are repeatedly performed at each stage of the lifecycle.
- It uses different CC Components according to the phase of the lifecycle and the activity of SREP, although the Software Quality Assurance (SQA) activities are performed along all the phases of the software development lifecycle

Characteristics

- **Iterative and incremental**
- The security requirements evolve along the lifecycle
  - for instance, during the design, the specification could be enriched with requirements related to the technological environment and its associated countermeasures.
- The core concept is the use of a micro-process for the security requirements analysis, made up of nine steps, which are repeatedly performed at each level of abstraction throughout the incremental development.
- Each iteration accomplishes all the steps defined within SREP, and each output from a complete iteration improves and refines the Security Requirements Specification by adding, correcting or specifying/detailing security requirements.

Characteristics

- **It facilitates the reusability**
- We proposed a security resources repository and a meta-model for it.
- The purpose of development with requirements reuse is to identify descriptions of systems that could be used (either totally or partially) with a minimal number of modifications, thus reducing the total effort of development
- Moreover, reusing security requirements helps us increase their quality: inconsistency, errors, ambiguity and other problems can be detected and corrected for an improved use in subsequent projects.
- Thereby, it will guarantee us the fastest possible development cycles based on proven solutions.

Characteristics

- **It facilitates the traceability of the security requirements.**
- The focus of this methodology seeks to build security concepts at the early stages of the software development
- It supports and includes concepts and techniques within the scope of Security Requirement Engineering and Risk Management and Analysis → UMLSec, security use cases, misuse cases, threat/attack trees.
- It conforms to several standards within the scope of Requirement Engineering and Security Management → ISO/IEC 17799:2005 (current ISO/IEC 27002) and ISO/IEC 15408

Characteristics

- **SREP Compliance with Standards**.
  - It conforms to ISO/IEC 17799:2005 (current ISO/IEC 27002) with regard to security requirements (sections: 0.3, 0.4, 0.6 and 12.1)
    - It says that "*Security requirements should be identified and agreed prior to the development and/or implementation of information systems. All security requirements should be identified at the requirements phase of a project and justified, agreed, and documented as part of the overall business case for an information system*".
  - And this is exactly what SREP proposes to do.

- **SREP Compliance with Standards**.
  - We take into account the IEEE 830-1998 standard (Requirements Inspection).
    - the step of “Requirements Inspection” of our micro-process for the security requirements analysis verifies whether the security requirements conform to this standard.
    - Because according to the IEEE 830-1998 standard, a requirement of quality has to be *correct, unambiguous, complete, consistent, ranked for importance and/or stability, verifiable, modifiable, and traceable*.
  - Therefore all these factors are verified at the end of each iteration of the micro-process, just before the “Repository Improvement” step.

- **SREP Compliance with Standards**.
  - Common Criteria (ISO/IEC 15408) is the standard requirements catalogue for the evaluation of security critical systems.
    - Using the CC, a large number of security requirements within the system itself and in the system development can be defined.
    - And the CC scheme can be introduced into the software lifecycle of new and existing applications to meet stricter security requirements. So, we propose to introduce it.
      - integrating CC functional requirements into the Software Requirements Specification;
      - integrating CC assurance requirements into Software Quality Assurance (SQA) activities;
      - introducing EALs (Evaluation Assurance Levels) into the software test plan;
      - and introducing CC Protection Profiles into architectural design

Characteristics

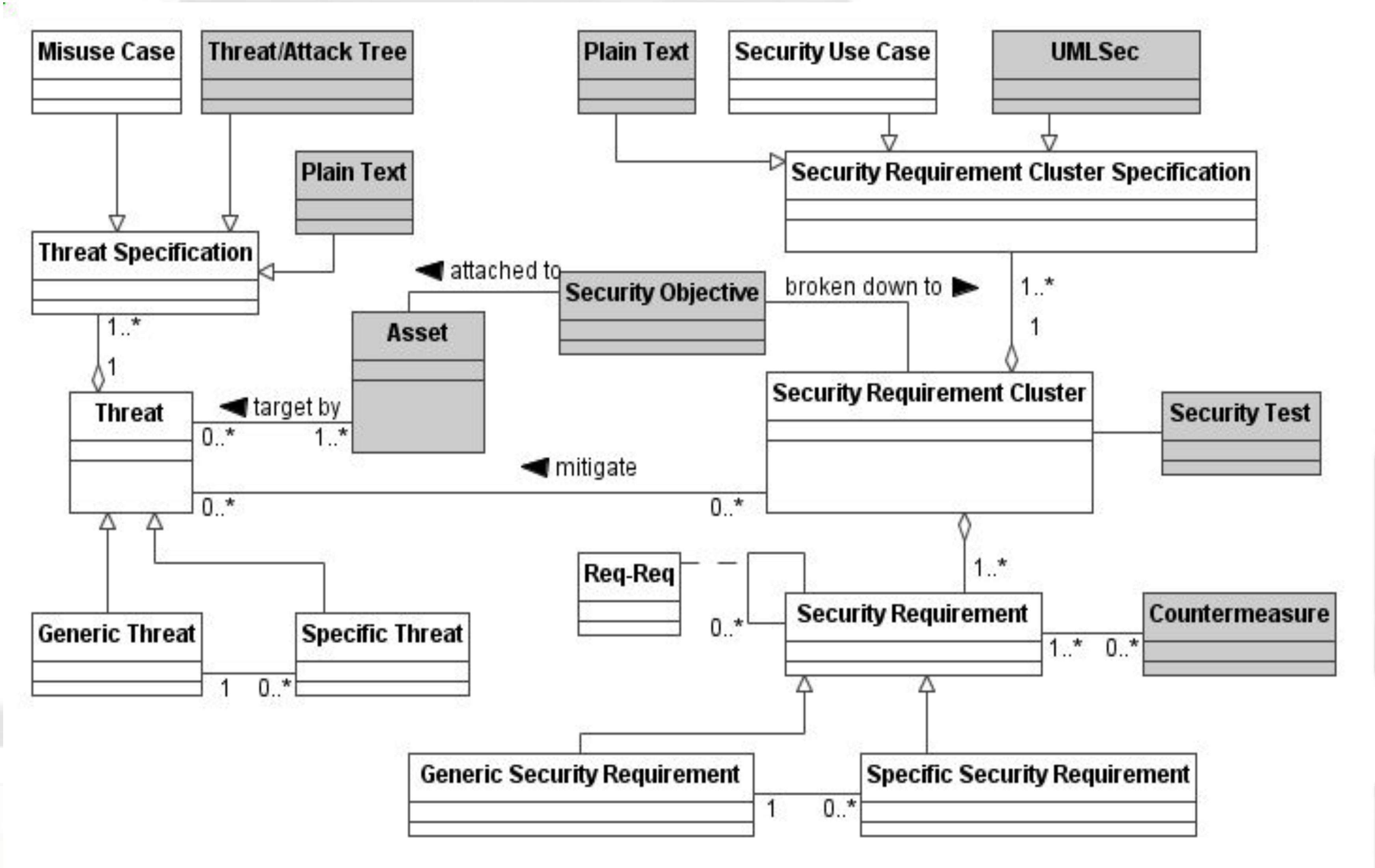
- SREP is based on several current techniques which deal with security requirements, in order to make it easy the task of dealing with security requirements in the first stages of software development in a systematic and intuitive way:
  - UMLSec allows us to express security-related information within the diagrams in a UML system specification
  - Security use cases are a technique that we used in order to specify the security requirements that the application must fulfil to be able to successfully protect itself from its relevant security threats
  - Misuse cases are a specialized kind of use cases that are used to analyze and specify security threats

Security Resources Repository

- *Security Resources Repository* (SRR), which stores all the reusable elements.
- The repository supports the concepts of domains and profiles.
  - The domains consists of belonging to a specific application field or functional application areas, such as e-commerce.
  - The profiles consists of a homogeneous set of requirements which can be applied to different domains, as for example personal data privacy legislation.
- We propose to implement the domains and profiles by taking advantage of the CC concepts of packages and Protection Profiles (PP).

- A meta-model, which is an extension of the meta-model for repository proposed by Sindre, G., D.G. Firesmith, and A.L. Opdahl, showing the organization of the SRR is exposed in the figure.
    - The dark background in the objects represents our contribution to the meta-model

Security Resources Repository



- ‘Generic Threat’ and ‘Generic Security Requirement’ describe independently of particular domains. And they can be represented as different specifications, thanks to the elements ‘Threat Specification’ and ‘Security Requirement Cluster Specification’.
- ‘Security Requirement Cluster’ is a set of requirements that work together in satisfying the same security objective and mitigating the same threat. We agree with Sindre, G., D.G. Firesmith, and A.L. Opdahl that, in many cases, it is a bigger and more effective unit of reuse.
- The ‘Req-Req’ relationship allows an inclusive or exclusive trace between requirements. An exclusive trace between requirements means that they are mutually alternative, as for example that they are in conflict or overlapping. Whereas, an inclusive trace between requirements means that to satisfy one, another/other/s is/are needed to be satisfied.

- CC does not give methodological support, nor contain security evaluation criteria pertaining to administrative security measures not directly related to the IS security measures
- However, it is known that an important part of the security of an IS can be often achieved through administrative measures

→ We propose to include legal, statutory, regulatory, and contractual requirements that the organization, its trading partners, contractors, and service providers have to satisfy, and their socio-cultural environment (ISO/IEC 17799 – current ISO/IEC 27002).

→ After converting these requirements into software and system requirements format, these would be the initial subset of security requirements of the SRR.

Process Model

- *Step 1: Agree on definitions*
- *Step 2: Identify vulnerable and/or critical assets*
- *Step 3: Identify security objectives and dependencies*
- *Step 4: Identify threats and develop artefacts.*
- *Step 5: Risk assessment*
- *Step 6: Elicit security requirements*
- *Step 7: Categorize and prioritize requirements*
- *Step 8: Requirements inspections*
- *Step 9: Repository improvement*

Process Model

- At the same time, as we integrate into these nine steps the CC security functional requirements, we propose to outline the EALs in the software test plan and then verify them during test execution.

- And parallely, we proposed to introduce the CC security assurance requirements into SQA activities, like quality control, defect prevention and defect removal activities. →the configuration management plan is the first activity that is explicitly required to fulfil the CC security assurance requirements

Process Model

- ***Activity 1: Agree on definitions.***
- The first task for the organization is to define the stakeholders and to agree upon a common set of security definitions, along with the definition of the organizational security policies and the security vision of the IS.
- It is in this activity when the *Vision Document* artefact is created and it must contain the general vision of the IS with a special focus on security aspects.
- In addition the stakeholders will participate in these latter tasks, and the candidate definitions will be mainly taken from ISO/IEC and IEEE standards, such as ISO/IEC 13335, ISO/IEC 17799:2005 (current 27002), ISO/IEC 27001:2005, ISO/IEC 9126, IEEE Std. 830:1998, or IEEE Std. 1061-1992

Process Model

- ***Activity 2: Identify vulnerable and/or critical assets***.
- This is where the SRR is used for the first time. It consists of the identification of the different kinds of valuable or critical assets as well as vulnerable assets by the requirements engineer, who can be helped by using:
  - Lists of assets of the SRR, where the assets can be searched by domains, even it can be selected a similar profile.
  - Functional requirements.
  - Interviews with stakeholders.

- ***Activity 3: Identify security objectives and dependencies***.
- In this activity the SRR can be also used. Otherwise we will take into account the security policy of the Organization as well as legal requirements and other constraints in order to determine the security objectives.
- For each asset identified in the previous activity, the appropriate security objectives for the asset are selected and the dependencies between them are identified.
- Moreover the security objectives for the environment are retrieved and the assumptions about the environment are made in this activity.
- Security objectives are expressed by specifying the necessary security level as a probability, and they are also specified in terms of likely attacker types.
- The *Security Objectives Document* is developed in this activity and it may be refined in subsequent iterations (within the Inception and Elaboration phases).

Process Model

- ***Activity 4: Identify threats and develop artifacts***.
- Each asset is targeted by threat/s that can prevent the security objective from being achieved.
- First of all, it is necessary to find all the threats that target these assets with the help of the SRR. In addition, it could be necessary to develop artifacts (such as *misuse cases* or *attack trees diagrams* or *UMLSec use cases* and *classes* or *sequence/state diagrams*) to develop new specific or generic threats or requirements.
- Also it is necessary to look for threats that are not linked/related to the assets of the repository, therefore according to CC assurance requirements we could search in public domain sources to identify potential vulnerabilities in the IS, or we could instantiate the business use cases into misuse cases or instantiate the threat-attack trees associated to the business and application pattern. At this point it may be possible to take one or several existing *Protection Profiles* or *packages* and adapt them to meet modified requirements.
- Finally, it is also defined the security problem and the conformance claims, thereby it is generated the *Security Problem Definition Document* which must contain the threats, assumptions, and conformance claims. In addition, this document may be refined in subsequent iterations.

Process Model

- ***Activity 5: Risk assessment*.**
- Risk must be normally determined from application to application. The final goal to achieve is the 100% risk acceptance.
- Firstly, it is necessary to assess whether the threats are relevant according to the security level specified by the security objectives.
- Then we have to estimate the security risks based on the relevant threats, their likelihood and their potential negative impacts. All of this is captured in the *Risk Assessment Document*, which is refined in subsequent iterations (within the Inception and Elaboration phases).
- Several methodologies can be used to carry out the risk assessment.
  - The ISO/IEC 13335 (GMITS), provides guidance on the use of the risk management process.
  - In Spain it might be used MAGERIT (the Spanish public administration risk analysis and management method) or CRAMM (CCTA Risk Analysis and Management Method) in the UK.
- Thereby, this assessment allows us to discover how the organization’s risk tolerance is affected with regard to each threat. The stakeholders will take part in this activity.

- ***Activity 6: Elicit security requirements***.
- Here, the SRR is used again. For each threat retrieved from the repository, one or more associated clusters of security requirements may be found.
- The suitable security requirements or the suitable cluster of security requirements that mitigate the threats at the necessary levels with regard to the risk assessment must be selected.
- However, additional requirements or clusters of requirements may be found by other means. Moreover, it might be specified the security test for each security requirement cluster, as well as an outline of the countermeasures for each security requirement, although they are refined at the design stage.
- Nevertheless, we agree with Firesmith in the fact that care should be taken to avoid unnecessarily and prematurely architectural mechanisms specification.
- Thus, at the end of this activity and according to ISO/IEC 17799:2005 (current ISO/IEC 27002) it must have been specified the functional, assurance, and organizational security requirements, along with the security requirements for the IT development and operational environment.
- Thereby, the *Security Requirements Specification Document* is created and refined in subsequent iterations

Process Model

- ***Activity 7: Categorize and prioritize requirements***.
- Each requirement is categorized and prioritized in a qualitative ranking in a way that the most important requirements (in terms of impact and likelihood) are handled first.

- ***Activity 8: Requirements inspection*.**
- Requirements inspection is carried out in order to validate all the generated artifacts (all the documents, requirements, the modified model elements and the new generated model elements) and it is generated a *Validation Report*.
- Its aim is to review the quality of the team's work and deliverables as well as assesses the security requirements engineering process. So, it is used as a sanity check.
- Moreover, it is verified whether the security requirements conform to the IEEE 830-1998 standard, because according to this standard, a requirement of quality has to be correct, unambiguous, complete, consistent, ranked for importance and/or stability, verifiable, modifiable, and traceable.
- After all, the security requirements documentation is written, so that a *Security Requirements Rationale Document* is provided, showing that if all the security organizational, functional and assurance requirements are satisfied and all security objectives are achieved, the defined security problem is solved:
  - all the threats are countered, the organizational security policies are enforced and all assumptions are upheld.

- ***Activity 8: Requirements inspection*.**
- Furthermore, it is performed within the Test workflow of the UP and with the help of the CC assurance requirements and EALs (Evaluation Assurance Level) and the SSE-CMM (ISO/IEC 21827).
- Thereby, we propose to evaluate the security of the IS along with the security engineering process by using the CC assurance requirements and the SSE-CMM at the same time with the help of CC_SSE-CMM .
- Thus referring to CC_SSE-CMM Part 3, the Process Area (PA) in association with CC EAL can be selected and based on the PA selected it can be determined the current level of SSE-CMM operation capability and extract the path for the better operation capability level .
- Thus, it can be assured that a security IS with a high reliability will be developed by conducting the CC evaluation and the SSE-CMM evaluation at the same time.
- Additionally, this activity is carried out by the quality assurer and by the inspection team at the last phase (Transition phase), with the participation of the stakeholders and security requirements engineers mainly

- ***Activity 9: Repository improvement*.**
- The new model elements (threats, requirements, etc…) found throughout the development of the previous activities and which are considered as likely to be used in forthcoming applications and with enough quality, according to the Validation Report, are introduced into the SRR.
- Furthermore, the model elements already in the repository could be modified in order to improve their quality.
- Thereby, all these new or modified model elements / artifacts, which have been introduced into the SRR, altogether constitute a baseline.
- After that the *Security Target* or *Protection Profile documents* of the CC are written.
- This activity will be performed coinciding with the milestone at the end of each phase of the UP.

SREP Roles

X, has responsibility *, supports O, does not participate	Business modeller	Security requirement engineer	Risk expert	Security expert	Security developer	Quality assurer	Inspection team
Agree on definitions	*	X	O	*	O	*	O
Identify vulnerable and/or critical assets	*	X	O	*	O	*	O
Identify security objectives and dependencies	*	X	O	*	O	*	O
Identify threats and develop artifacts	*	X	O	*	*	*	O
Risk assessment	O	O	X	*	O	O	O
Elicit security requirements	O	X	*	*	O	*	O
Categorize and prioritize requirements	*	X	O	*	O	*	O
Requirements inspection	*	*	*	*	*	X	X
Repository improvement	O	X	O	*	O	*	O

SREP Roles

→ These roles are a supplement to the roles in software engineering, but are especially focused on security and also require special training

- ***Business modeller***.
  - He/she describes the business processes, the roles involved and the artifacts produced or used in the process.
  - He/she helps develop artifacts in SREP (like misuse cases, etc.) and construct the processes in a security-enhanced way, which fit in the business model of the IS.

SREP Roles

- ***Security requirement engineer***.
- This is the key role and it participates and leads most activities.
- It is in charge of the security vision of the IS, it also identifies the assets, the security objectives and its dependencies and the threats, and elicits and specifies the requirements, as well as categorizes and prioritizes the requirements with the help of other kind of specialists (if needed).
- Depending on the size of the project more than one person can be assigned to this role.
- Furthermore, this role must not necessarily have a thorough technical understanding of security, although a sound security management is required.

SREP Roles

- ***Security expert***.
- The main task of the security expert is to improve the overall security of the IS.
- This role is the technical expert in security so that he/she acts as a consultant, and helps us find security relevant information, estimate the degree to which IS meets its security claims and define the security vision of the IS and the organizational security policies and measures.
- ***Security developer***.
- The role of the security developer is to support the construction of tests to help the Requirements Inspection activity during the Test workflow of the UP.

SREP Roles

- ***Quality assurer***.
- This is the role responsible for the Requirements Inspection activity within the Test workflow of the UP and it could take advantage of the use of the CC assurance classes.
- In addition, this role can help us with informal reviews of the quality of the most important artifacts in each activity.
- ***Inspection team***.
- It is a group external to the IS development team whose aim is to review the quality of the development team's work and deliverables as well as evaluate the security engineering process by using the CC assurance requirements and the SSE-CMM, with the help of CC_SSE-CMM [12].
- Besides it is the role responsible for the Requirements Inspection activity within the Transition phase of the UP. Additionally, this team is in charge of the assurance that the IS meets its security claims with the help of the EALs.

Model Process

- **Iterations**

  The integration of SREP, with the CC and with the phases of the UP is presented below:

- ***Inception*.**
- It is the first phase and it is focused on the earlier activities of SREP.
- The security vision document is produced, and around the 50% of the first order requirements are defined, therefore a similar percentage of the assets, security objectives and threats.
- In addition, the security problem definition is carried out and an overall risk outline is performed.
- Moreover, the main focus with regard to the CC assurance classes is on the following classes: Composition, Lifecycle Support and Vulnerability Assessment.
- Also, at this point, it may be possible to take an existing or several Protection Profiles or packages and adapt them to meet modified requirements.
- Nevertheless, it is difficult to conduct everything in one iteration, so it might be necessary another iteration with more mature understanding of the IS.

Model Process

- ***Elaboration***.
- More than one iteration may be normally made at this phase depending on the size and complexity of the project.
- The goal of this phase, and according to ISO/IEC 17799:2005, is to identify around 98% of the critical/vulnerable assets, security objectives, threats and first ordered requirements and around 90% of second ordered requirements.
- Moreover a refinement of the risk assessment and the security problem definition is carried out.
- In addition, this phase is also focused on the requirements categorization and prioritization, and on the requirements inspection as well as on the security requirements rationale.
- Therefore, the most important CC assurance classes for this phase are: Security Target Evaluation, Protection Profile Evaluation, Guidance Documents, Development, and Vulnerability Assessment.

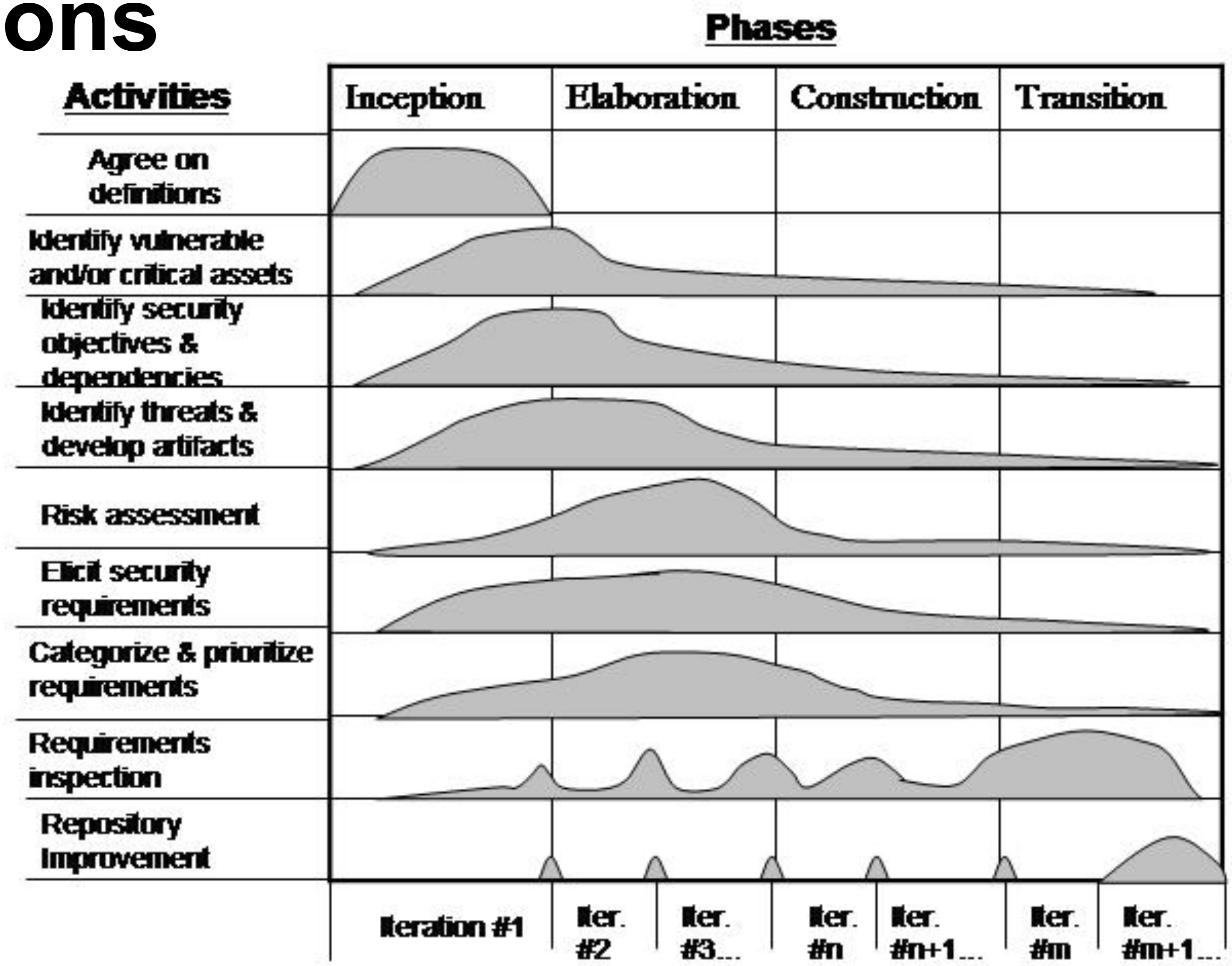
Model Process

- **Construction**.
- At this phase, the remaining requirements are defined along with the final design and the implementation of the security countermeasures.
- The Requirements Inspection activity is emphasized at this phase.
- The main focus with regard to the CC assurance classes is on the following classes: Security Target or PP Evaluation, Development, Composition and Vulnerability Assessment.

Model Process

- **Transition.**
- It is the last phase and when the IS is put into productive use.
- The danger is, however, that other requirements can emerge, thus security risks must be considered and therefore they must be dealt with carefully and in a pragmatic way.
- This phase is focused on the Requirements Inspection and Repository Improvement activities.
- So, the most important CC assurance classes for this phase are: Security Target or PP Evaluation, Tests, Guidance Documents, Composition, and Vulnerability Assessment

Iterations


Phases
Activities
Inception
Elaboration
Construction
Transition
Agree on definitions
Identify vulnerable and/or critical assets
Identify security objectives & dependencies
Identify threats & develop artifacts
Risk assessment
Elicit security requirements
Categorize & prioritize requirements
Requirements inspection
Repository Improvement
Iteration #1
Iter. #2
Iter. #3...
Iter. #n
Iter. #n+1...
Iter. #m
Iter. #m+1...

Case Study

SECRETARIA DE ESTADO
DE LA SEGURIDAD SOCIAL

INSTITUTO NACIONAL DE LA
SEGURIDAD SOCIAL

- It is a representative case of a security critical IS in which security requirements have to be correctly treated in order to achieve a robust IS

- It will be analysed the case of an administrative unit of the National Social Security Institute (of Spain)
  - which has the porpoise of providing citizens e-government services

- It will be studied the case of an e-government service
  - which consists of an application (called Pension-App) that basically allows to provide information about the pension/s of a concrete citizen.

Case Study

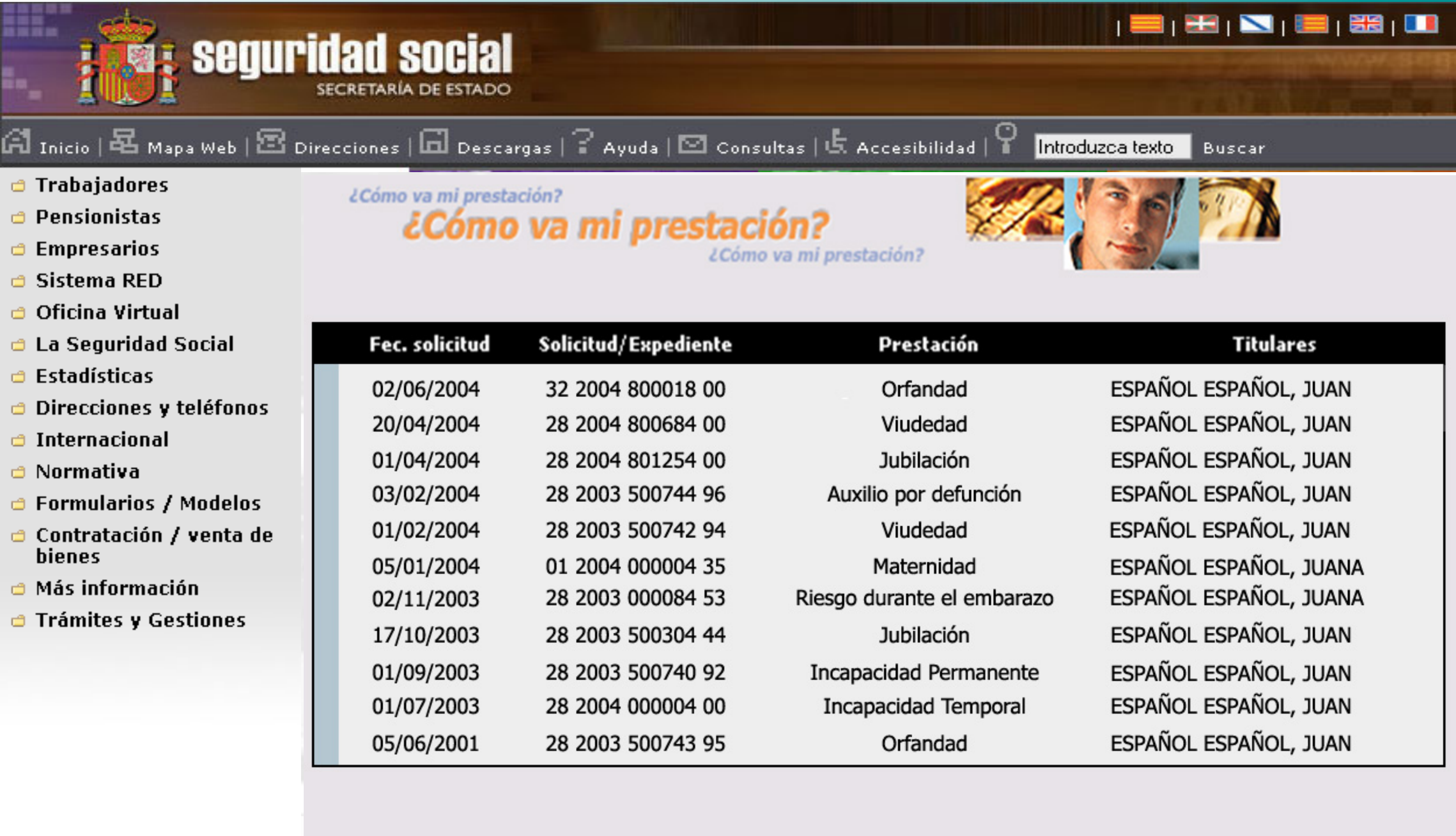
- PensionApp is an application that allows citizens to obtain an official document which reflects the current amount and the status of their pension/s
  - whether it is being processed and the stage where it is at the moment of the request,
  - or whether it has been successfully granted or rejected
  - it also allows citizens to update some personal data, such as their address and bank account number.
- One of the main design goals was maximum ease of use.
- Thus, citizens have online access to PensionApp through the Internet
  - or they can go to an office of the National Social Security Institute
- They can only obtain information about his/her own pension and update his/her personal information

Case Study

- We assume that initial functional requirements have been elicited and that there is only two functional requirements:

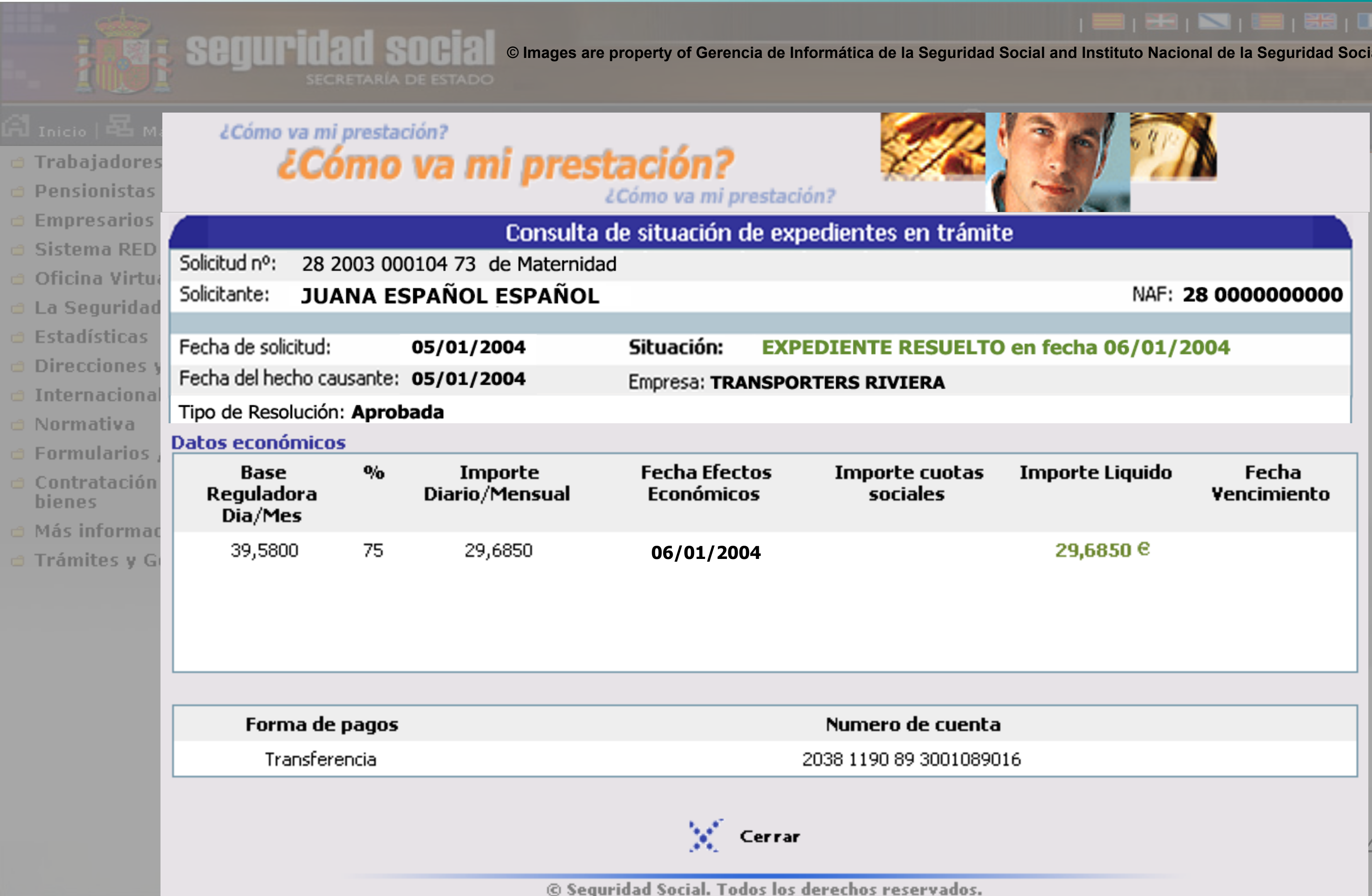
- Initial functional requirements :
  – ***Req 1***: On request-1 from an EndUser, the system shall display information about his/her pension/s. This request shall include the social security number of the EndUser.
  – ***Req 2***: On request-2 from an EndUser, the system shall update the personal information of the pensioner. This request shall include the social security number of the EndUser and changed personal data.

1 2 **Case Study** 4 5 6



Fec. solicitud	Solicitud/Expediente	Prestación	Titulares
02/06/2004	32 2004 800018 00	Orfandad	ESPAÑOL ESPAÑOL, JUAN
20/04/2004	28 2004 800684 00	Viudedad	ESPAÑOL ESPAÑOL, JUAN
01/04/2004	28 2004 801254 00	Jubilación	ESPAÑOL ESPAÑOL, JUAN
03/02/2004	28 2003 500744 96	Auxilio por defunción	ESPAÑOL ESPAÑOL, JUAN
01/02/2004	28 2003 500742 94	Viudedad	ESPAÑOL ESPAÑOL, JUAN
05/01/2004	01 2004 000004 35	Maternidad	ESPAÑOL ESPAÑOL, JUANA
02/11/2003	28 2003 000084 53	Riesgo durante el embarazo	ESPAÑOL ESPAÑOL, JUANA
17/10/2003	28 2003 500304 44	Jubilación	ESPAÑOL ESPAÑOL, JUAN
01/09/2003	28 2003 500740 92	Incapacidad Permanente	ESPAÑOL ESPAÑOL, JUAN
01/07/2003	28 2004 000004 00	Incapacidad Temporal	ESPAÑOL ESPAÑOL, JUAN
05/06/2001	28 2003 500743 95	Orfandad	ESPAÑOL ESPAÑOL, JUAN

© Images are property of Gerencia de Informática de la Seguridad Social and Instituto Nacional de la Seguridad Social



Consulta de situación de expedientes en trámite

Solicitud nº: 28 2003 000104 73 de Maternidad

Solicitante: **JUANA ESPAÑOL ESPAÑOL** NAF: **28 0000000000**

Fecha de solicitud: **05/01/2004** **Situación:** **EXPEDIENTE RESUELTO en fecha 06/01/2004**

Fecha del hecho causante: **05/01/2004** Empresa: **TRANSPORTERS RIVIERA**

Tipo de Resolución: **Aprobada**

Datos económicos

Base Reguladora Dia/Mes	%	Importe Diario/Mensual	Fecha Efectos Económicos	Importe cuotas sociales	Importe Liquido	Fecha Vencimiento
39,5800	75	29,6850	06/01/2004		29,6850 €	

Forma de pagos	Numero de cuenta
Transferencia	2038 1190 89 3001089016

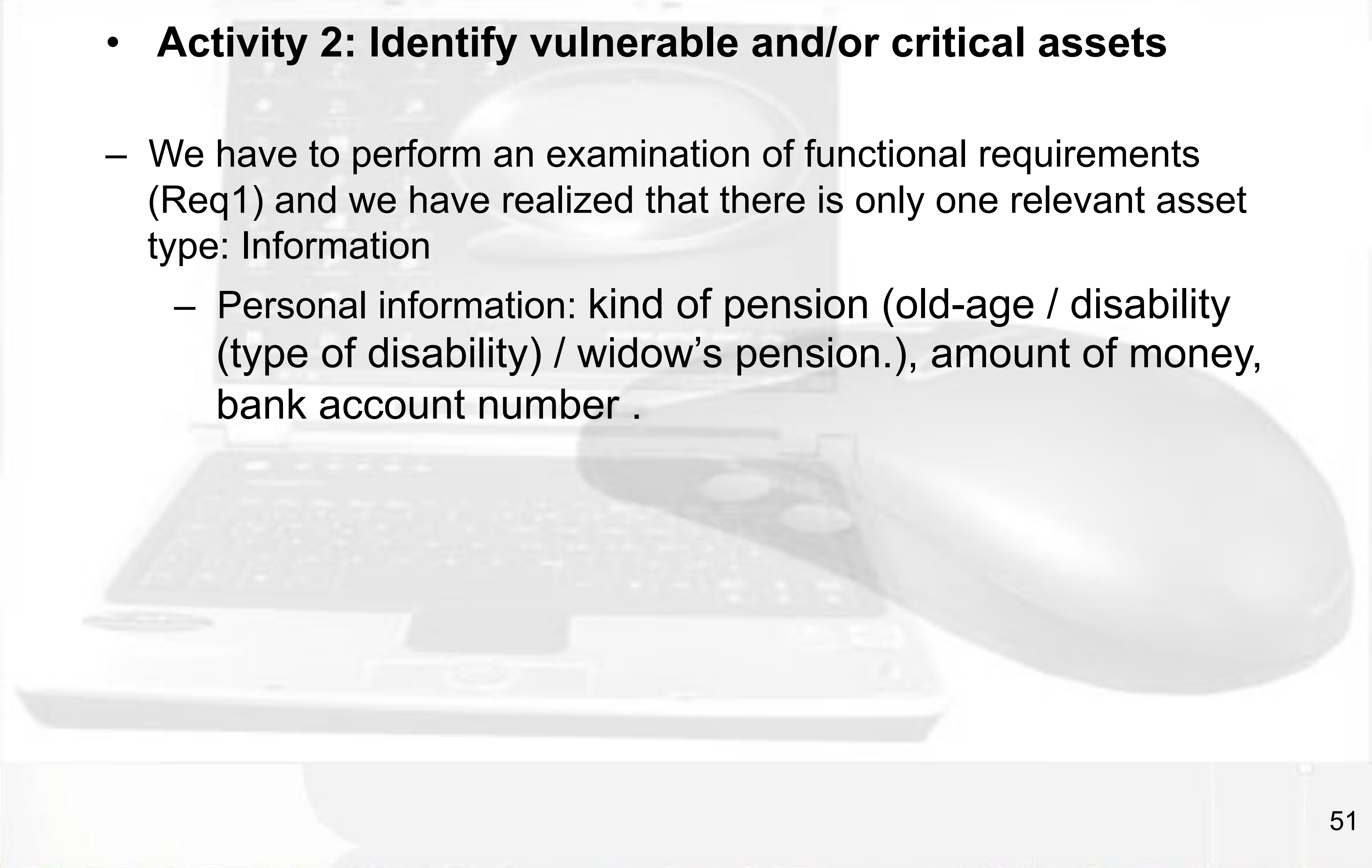
- SREP defines nine activities to be carried out as well as several iterations through the software development lifecycle

- Each iteration will generate internal or external releases of various artefacts which altogether constitute a baseline

- We will only describe one iteration at the early stages of the software development lifecycle.

Case Study

- **Activity 1: Agree on definitions**
- In this activity we have to agree upon a common set of security definitions, along with the definition of the organizational security policies and the security vision of the IS.
- Definitions that should be agreed:
  - Information security: preservation of confidentiality, integrity and availability of information; [ISO/IEC 17799:2005]
  - Threat: a potential cause of an unwanted incident, which may result in harm to a system or organization [ISO/IEC 13335-1:2004].
  - Availability: the property of being accessible and usable upon demand by an authorized entity [ISO/IEC 13335-1:2004].
  - Confidentiality: the property that information is not made available or disclosed to unauthorized individuals, entities, or processes [ISO/IEC 13335- 1:2004].
  - Integrity: the property of safeguarding the accuracy and completeness of assets [ISO/IEC 13335-1:2004].
  - Asset: anything that has value to the organization [ISO/IEC 13335-1:2004].
- The **Security Vision Document** will be written, in which it will be outlined the security vision of the IS.

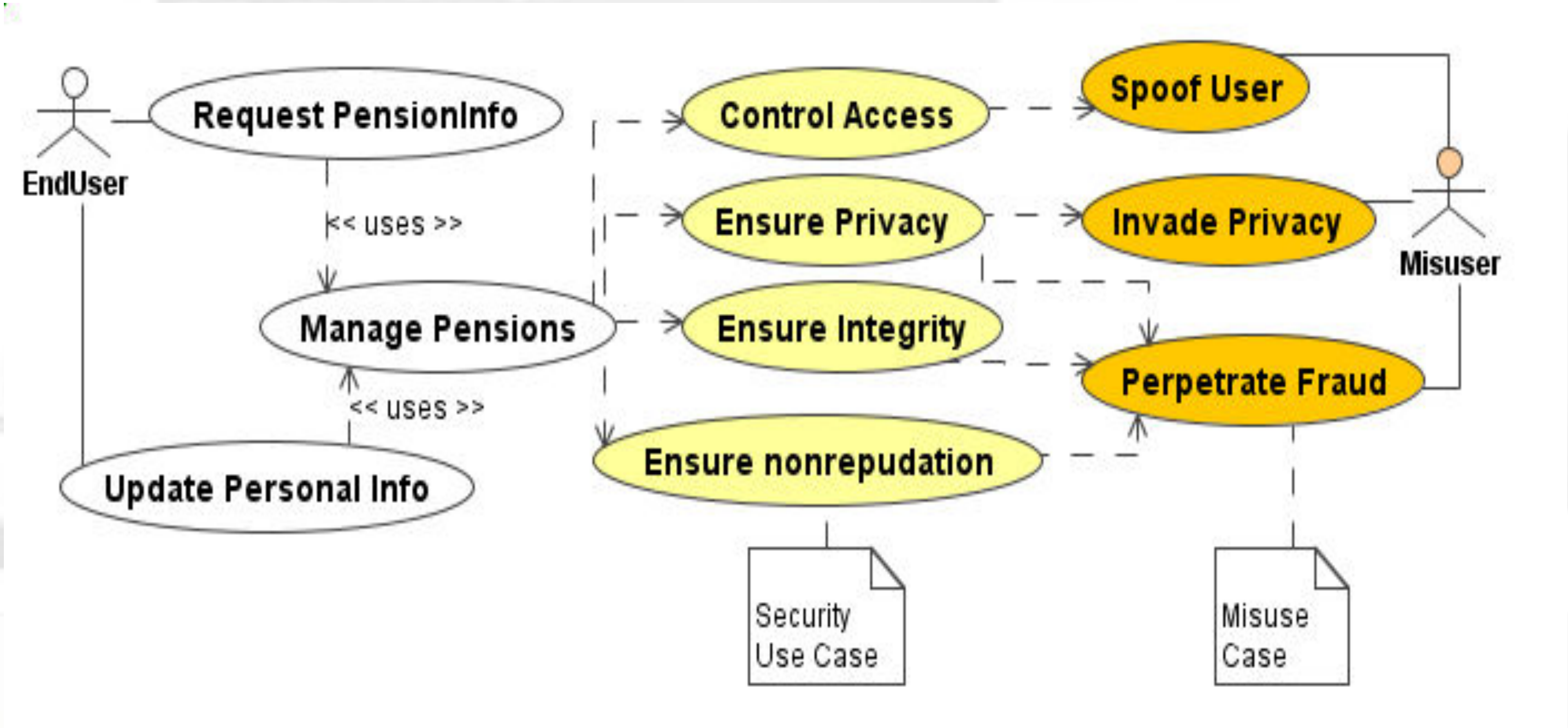


- **Activity 2: Identify vulnerable and/or critical assets**
  - We have to perform an examination of functional requirements (Req1) and we have realized that there is only one relevant asset type: Information
    - Personal information: kind of pension (old-age / disability (type of disability) / widow’s pension.), amount of money, bank account number .

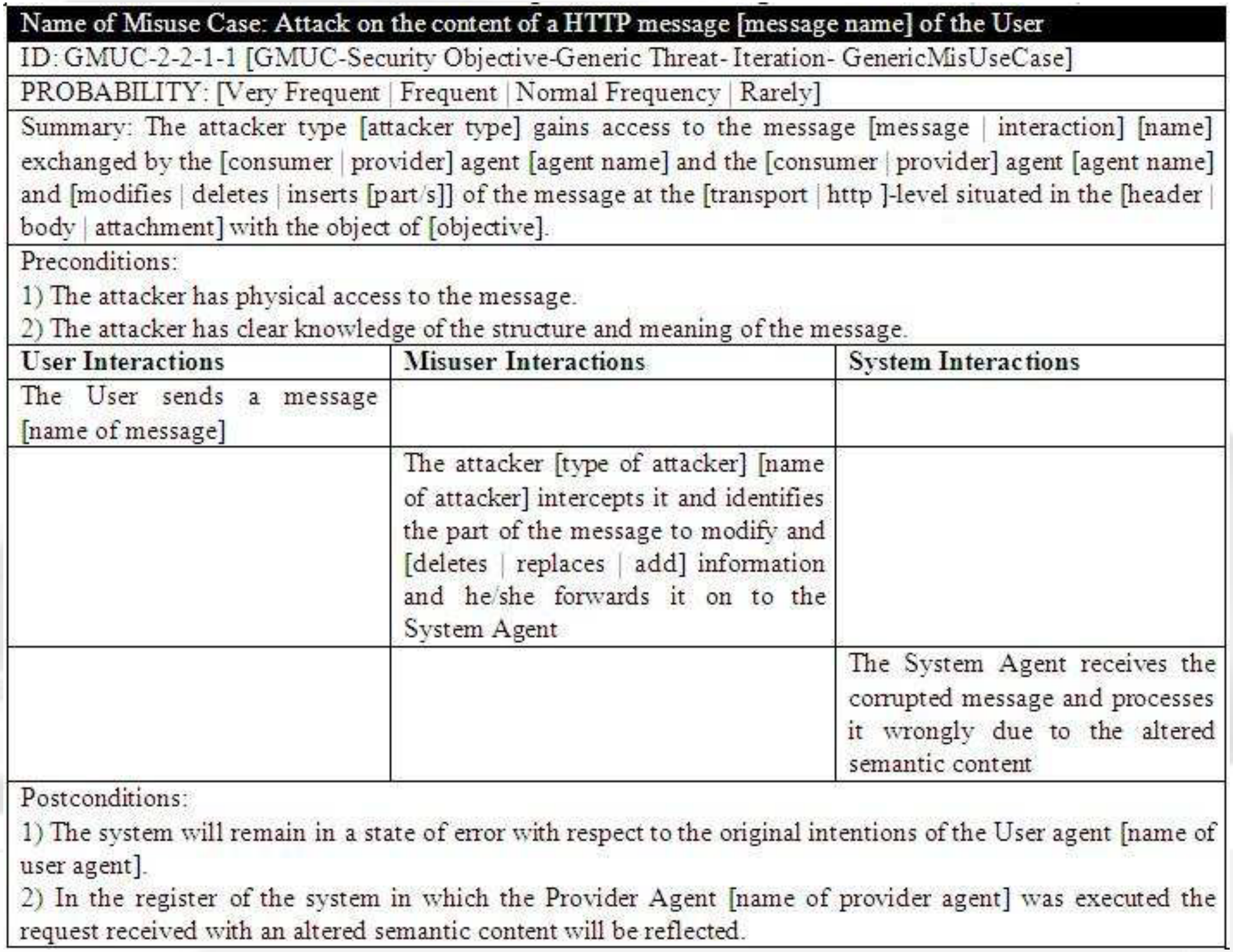
- **Activity 3: Identify security objectives and dependencies**
- In this activity the SRR can be used, so that if the type of assets identified in the previous activity are in the SRR we will be able to retrieve their associated security objectives (SO).
- We can identify the following security objectives:
  - SO1: Prevent unauthorised disclosure of information. (Confidentiality). Valuation – High.
  - SO2: Prevent unauthorised alteration of information. (Integrity). Valuation – High.
  - SO3: Ensure availability of information to the authorised users. Valuation – Medium.
  - SO4: Ensure authenticity of users. Valuation – High.
  - SO5: Ensure accountability. Valuation – Medium

- This list should be refined in subsequent iterations
- These security objectives will be written down in the *Security Objectives Document with the help of the CC* assurance classes (CC class ASE).

- **Activity 4: Identify threats and develop artifacts**
  - Generic Threat 1: Unauthorised disclosure of information.
  - Generic Threat 2: Unauthorised alteration of information.
  - Generic Threat 3: Unauthorised unavailability to information.
  - Generic Threat 4: Spoof user.

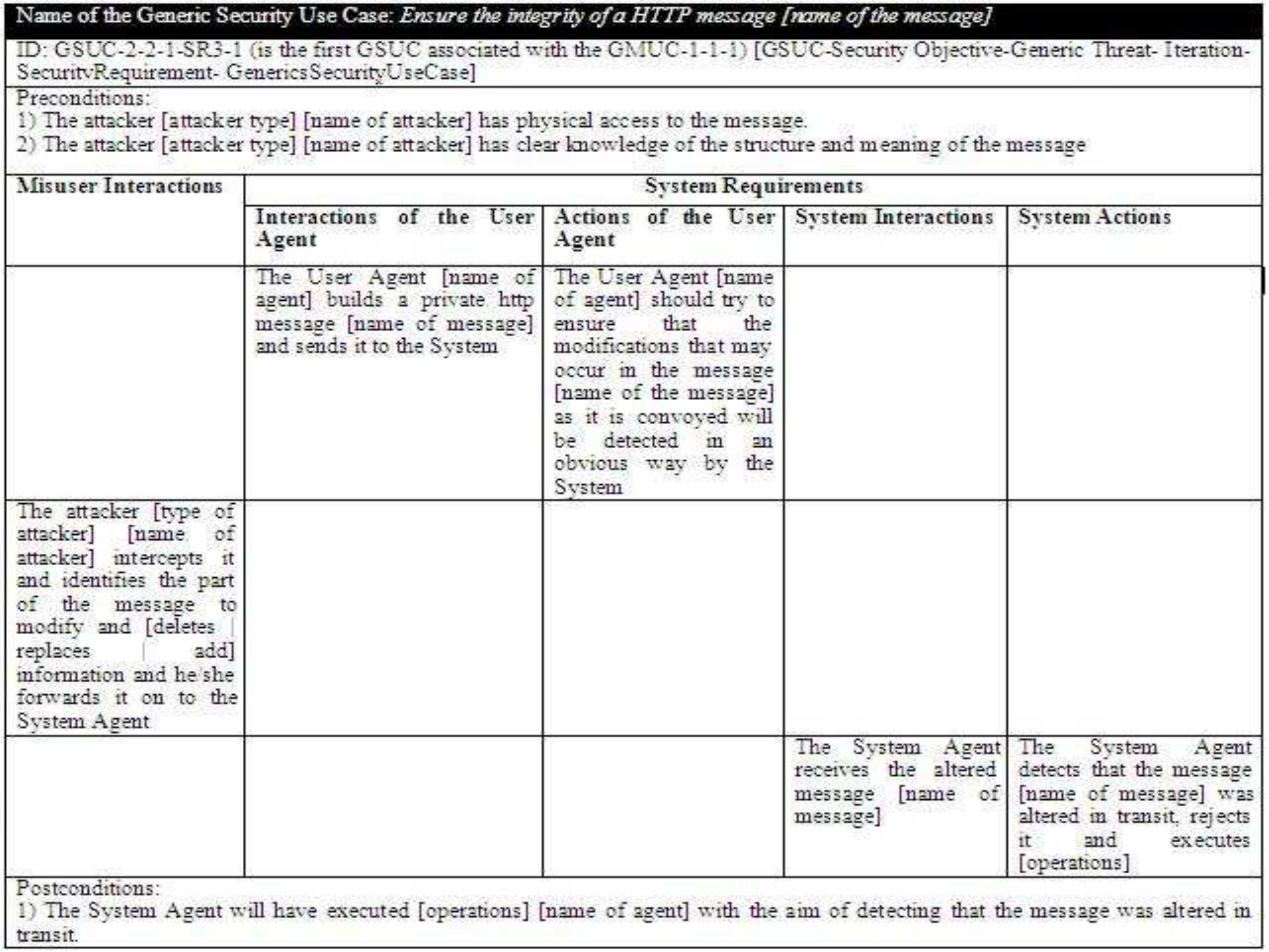


Case Study



<table>
<tr><th colspan="3">Name of Misuse Case: Attack on the content of a HTTP message [message name] of the User</th></tr>
<tr><td colspan="3">ID: GMUC-2-2-1-1 [GMUC-Security Objective-Generic Threat- Iteration- GenericMisUseCase]</td></tr>
<tr><td colspan="3">PROBABILITY: [Very Frequent | Frequent | Normal Frequency | Rarely]</td></tr>
<tr><td colspan="3">Summary: The attacker type [attacker type] gains access to the message [message | interaction] [name] exchanged by the [consumer | provider] agent [agent name] and the [consumer | provider] agent [agent name] and [modifies | deletes | inserts [part/s]] of the message at the [transport | http]-level situated in the [header | body | attachment] with the object of [objective].</td></tr>
<tr><td colspan="3">Preconditions:
1) The attacker has physical access to the message.
2) The attacker has clear knowledge of the structure and meaning of the message.</td></tr>
<tr><td><b>User Interactions</b></td><td><b>Misuser Interactions</b></td><td><b>System Interactions</b></td></tr>
<tr><td>The User sends a message [name of message]</td><td></td><td></td></tr>
<tr><td></td><td>The attacker [type of attacker] [name of attacker] intercepts it and identifies the part of the message to modify and [deletes | replaces | add] information and he/she forwards it on to the System Agent</td><td></td></tr>
<tr><td></td><td></td><td>The System Agent receives the corrupted message and processes it wrongly due to the altered semantic content</td></tr>
<tr><td colspan="3">Postconditions:
1) The system will remain in a state of error with respect to the original intentions of the User agent [name of user agent].
2) In the register of the system in which the Provider Agent [name of provider agent] was executed the request received with an altered semantic content will be reflected.</td></tr>
</table>

Case Study



Name of the Generic Security Use Case: *Ensure the integrity of a HTTP message [name of the message]*

ID: GSUC-2-2-1-SR3-1 (is the first GSUC associated with the GMUC-1-1-1) [GSUC-Security Objective-Generic Threat- Iteration-SecuritvRequirement- GenericsSecurityUseCase]

Preconditions:
1) The attacker [attacker type] [name of attacker] has physical access to the message.
2) The attacker [attacker type] [name of attacker] has clear knowledge of the structure and meaning of the message

Misuser Interactions	**System Requirements**			
	Interactions of the User Agent	**Actions of the User Agent**	**System Interactions**	**System Actions**
	The User Agent [name of agent] builds a private http message [name of message] and sends it to the System	The User Agent [name of agent] should try to ensure that the modifications that may occur in the message [name of the message] as it is conveyed will be detected in an obvious way by the System		
The attacker [type of attacker] [name of attacker] intercepts it and identifies the part of the message to modify and [deletes | replaces | add] information and he/she forwards it on to the System Agent				
			The System Agent receives the altered message [name of message]	The System Agent detects that the message [name of message] was altered in transit, rejects it and executes [operations]

Postconditions:
1) The System Agent will have executed [operations] [name of agent] with the aim of detecting that the message was altered in transit.

- **Activity 5: Risk assessment**
  - Having identified the threats, we shall now go on to determine the probability of each threat and to assess its impact and risk.
  - In order to carry out this task, we will use a technique proposed by the guide of techniques of MAGERIT v2.0 and which is based on tables to analyse impact and risk of threats.
- Risk and Impact: Very Low, Low, Medium, High, Very High
- Likelihood of a threat: Very Frequent (daily), Frequent (monthly), Normal Frequent (once a year), Rarely (once in several years)
- All of this is captured in the ***Risk Assessment Document***

Table of Threats, attacks and risks - Iteration 1				
Threat	Impact	Attack	Probability	Risk
1.2.1.1.1.1 Alteration of the information	LOW, if there is not pension information modified	S*MUC-2-2-1-1-1*	HIGH	LOW
	HIGH if the opposite is the case.	S*MUC-2-2-1-1-1*	HIGH	HIGH

- **Activity 6: Elicit Security Requirements**
- In order to derive security requirements, each security objective is analysed for possible relevance together with its threats which imply more risk.
- we will use domain knowledge to transform the entities described in the security objectives into entities in the functional requirement.
- We will transform the security objectives (Confidentiality, Integrity, Availability, Authenticity, Accountability) into constraints on the operations that are used in functional requirements.
- Additionally, we will search in the CC security functional requirements catalogue (which has been previously introduced together with the CC assurance requirements into the SRR) security requirements which mitigate the threats that can prevent the security objective from being achieved.

Case Study

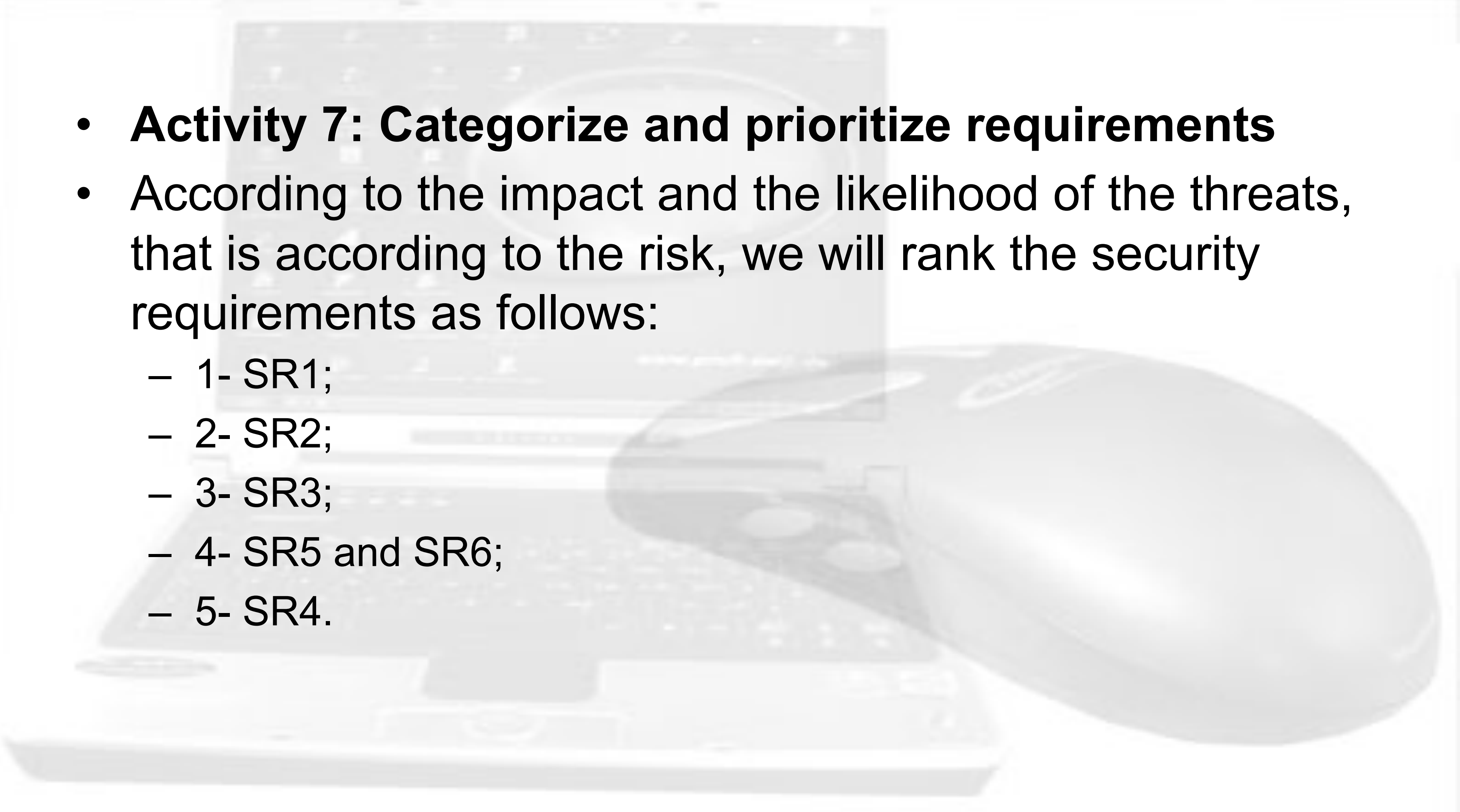
- The security requirements (SR) that we identify are the following ones:

- **SR1**: The security functions of PensionApp shall **use *cryptography*** [assignment: *cryptographic algorithm* and *key sizes*] to protect confidentiality of pension information provided by PensionApp to an EndUser. (CC requirement FCO_CED.1.1)
- **SR2**: The security functions of PensionApp shall **identify and authenticate an EndUser by using *credentials*** [assignment: *challenge-response technique based on exchange of encrypted random nonces, public key certificate*] before an EndUser can bind to the shell of PensionApp. (CC requirements FIA_UID.2.1 & FIA_UAU.1.1)
- **SR3**: When PensionApp transmits pension or pensioner's information to EndUser, the security functions of PensionApp shall provide that user with the ***means*** [assignment: *digital signature*] to **detect** [selection: *modification, deletion, insertion, replay, other integrity*] **anomalies**. (CC requirement FCO_IED.1.1)

- **SR4**: The security functions of PensionApp shall **ensure the availability** of the information provided by PensionApp to an EndUser within [assignment: *a defined availability metric*] given the following conditions [assignment: *conditions to ensure availability*]. (CC requirement FCO_AED.1.1)

- **SR5**: The security functions of PensionApp shall require **evidence** that PensionApp has pension **information** to an EndUser and he/she has **received** the information. (CC requirement FCO_NRE.1.1)

- **SR6**: The security functions of PensionApp shall store an **audit record** of the following events [selection: *the request for pension information, the response of PensionApp*] and each audit record shall record the following information: date and time of the event, [selection: *success, failure*] of the event, and EndUser identity. (CC requirements FAU_GEN)



- **Activity 7: Categorize and prioritize requirements**
- According to the impact and the likelihood of the threats, that is according to the risk, we will rank the security requirements as follows:
  - 1- SR1;
  - 2- SR2;
  - 3- SR3;
  - 4- SR5 and SR6;
  - 5- SR4.

- **Activity 8: Requirements inspection**
- We will generate the *Validation Report*
- thereby we will review the quality of the previous work with the help of the CC assurance requirements
- these assurance requirements will result from the determined EAL, which was agreed with the stakeholders in the first activity, although it could be modified in subsequent iterations.

- **Activity 9: Repository improvement**
- We will store in the SRR the new elements, in this case in this iteration the *Generic and Specific Threats* and *Requirements* which were developed in the activities 4 and 6 will be stored.
- After all, we will write the *Security Target document* of the CC.
- This activity will be performed coinciding with the milestone at the end of each phase of the UP.

Case Study
Lessons Learned

- To improve and refine the some activities of SREP
- Iterative e incremental → facilitates reuse and correct errors, risks are discovered and mitigated earlier, and the process itself can be improved and refined along the way.
- CC does not provide us with any method/guide to include them into the software development process, so that a modification in one document often leads to modify several other documents
- Tool support is critical for the practical application of this process in large-scale software systems due to the number of handled artifacts and the several iterations that have to be carried out

- Characteristics

- It lets us apply the SREP process by providing automated support to its activities.

- It facilities iterative and incremental development

- It facilitates the reusability → Security Resources Repository

- It facilitates the traceability of the security requirements and with the other functional and non-functional requirements → it is integrated with RequisitePro.

- The aim is to build security concepts in the first stages of the software development.

SREPTool - prototype

- Characteristics

- It supports and includes concepts and techniques of the field of Security Requirements Engineering and of the Risks Analysis and Management → security use cases, misuse cases…

- It facilities the implementation of Unified Process (RUP)

- It is conformed and integrated with some of the most important standards for the treatment of security requirements and it facilities that the IS developed are conformed with ISO/IEC 15408, ISO/IEC 17799, ISO/IEC 27001, IEEE 830: 1998.

- Technology: Visual Basic; RequisitePro Extensibility Server (RPX); RqGUIApp library; MS-Access;

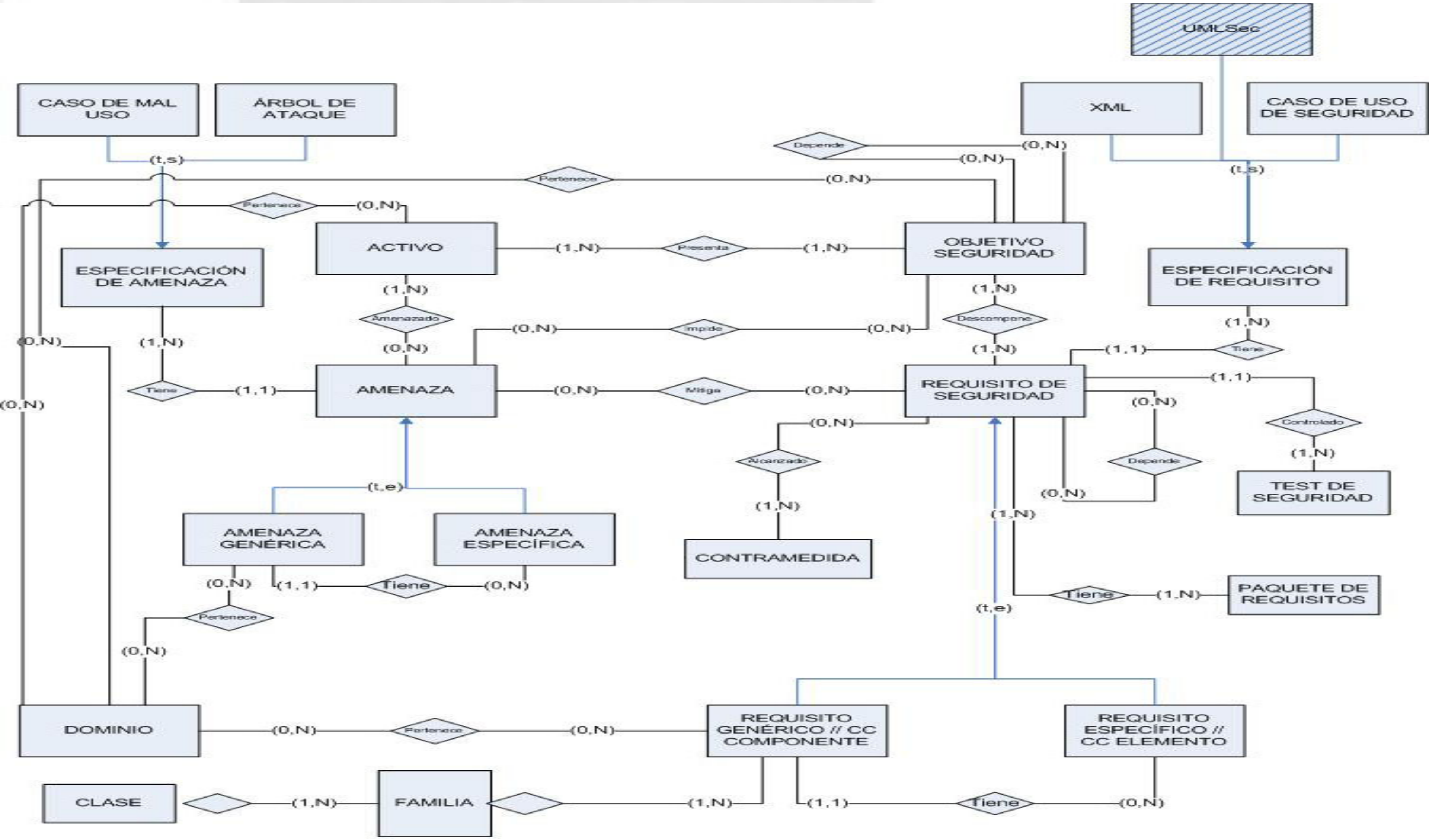
❖ Requirements Tools Comparision for extension

	RequisitePro	IRqA	DOORS	Caliber-RM
Extensibilidad de la funcionalidad	**Si, API basado en COM**	**Si, API basado en COM y JAVA**	**Si, API lenguaje DXL**	**Si, API basado en COM y JAVA**
Trazabilidad	**Si, entre los tipos de requisitos.**	**Si, entre tipos de requisitos, conceptos, UML, código, test.**	**Si, entre cualquier elemento del repositorio**	**Si, entre los tipos de requisitos y otros elementos.**
Integración con otras herramientas del ciclo de vida	**Si, con IBM Rational tools (Rational Rose, Rational TestManager, and Rational ClearQuest, MSProject, etc)**	**Si, con Office, IRqA-Rational Rose, Mercury TestDirector, CVS.**	**Si, mediante el lenguaje DXL.**	**Si, con Office, Project, Modelling, Testing e IDE tools.(Ej. Mercury TestDirector)**
Soporte reutilización	**No**	**No**	**No**	**No**
Repositorio del proyecto	**MS-Access, MS-SQL Server, Oracle**	**MS-SQL Server, Oracle, Informix, MySQL**	**Propietario**	**MS-Access y MS-SQL Server**
Validación de la especificación	**Si, con matriz de trazabilidad**	**Si, con matriz de trazabilidad**	**Si, con matriz de trazabilidad**	**Si, con matriz de trazabilidad**
Estándares de especificación	**Si, la salida puede adaptarse a las plantillas que se definan**	**Si, la salida puede adaptarse a diferentes formatos/ plantillas**	**Si, la salida puede adaptarse a diferentes formatos**	**Si, la salida puede adaptarse a diferentes formatos**
Experiencia previa. Facilidad uso e Interfaz de usuario	**Si. Requisitos, vistas y documentos en una sola vista. Acceso web colaborativo**	**No. Organizada en vistas.**	**No. Difícil de seguir. Módulos y objetos**	**No. Práctica y amigable, orientada al entorno web**
Importación de requisitos	**Si, Word y CSV**	**Si, Word, CSV, Excel, XML**	**Si, Word y ficheros delimitados**	**Si, Word y ficheros delimitados**
Control de versiones y líneas base	**Si pero no comparables**	**Si y comparables**	**Si y comparables**	**Si y comparables**
Control acceso por roles y usuarios	**Si**	**Si**	**Si**	**Si**
Requisitos parametrizados	**No**	**No**	**No**	**No**

SREPTool - prototype

- It decided to extend RequisitePro due to:
  - Extensibility
  - Automated integration with the rest of activities of the lifecycle
  - Previous experience
  - Usability and Multiuser
  - Trazability
  - Others: it allows the use of templates, comercial databases, etc…

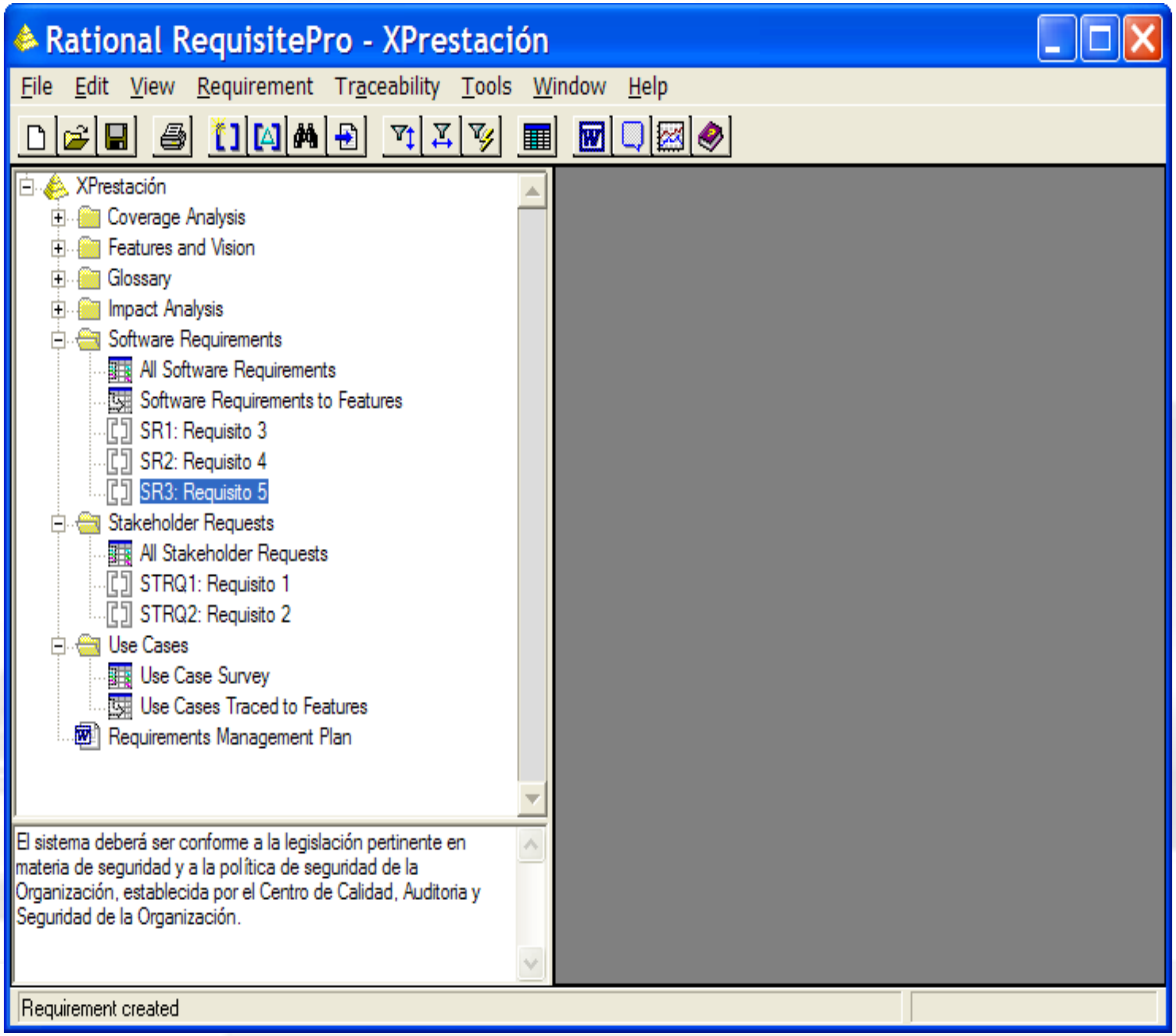
- SREP Tool Repository Implementation



1 2 3

SREPTool - prototype

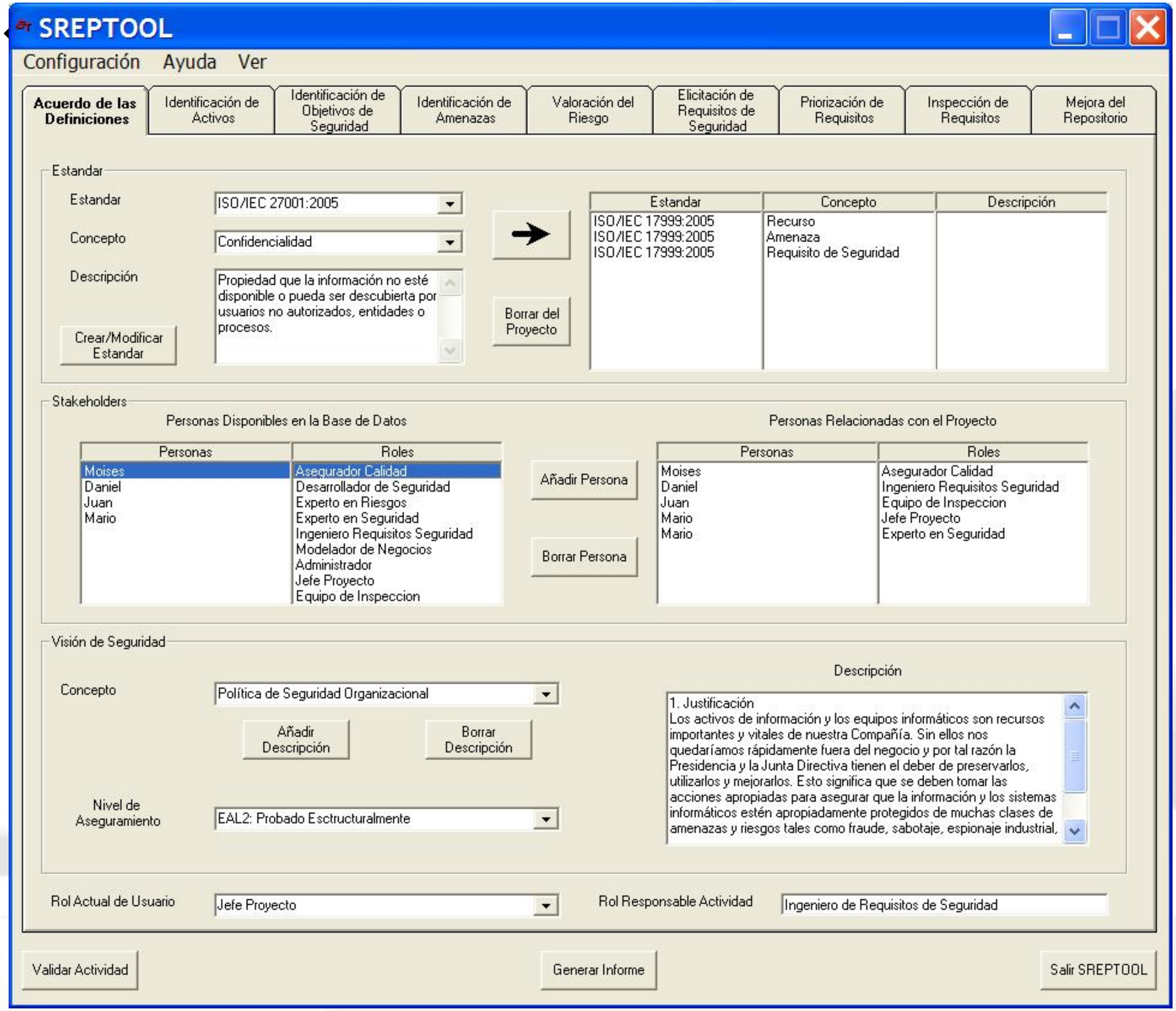
5 6



El sistema deberá ser conforme a la legislación pertinente en materia de seguridad y a la política de seguridad de la Organización, establecida por el Centro de Calidad, Auditoria y Seguridad de la Organización.

Requirement created

SREPTool - prototype



Estandar

Estandar: ISO/IEC 27001:2005

Concepto: Confidencialidad

Descripción: Propiedad que la información no esté disponible o pueda ser descubierta por usuarios no autorizados, entidades o procesos.

Estandar	Concepto	Descripción
ISO/IEC 17999:2005	Recurso	
ISO/IEC 17999:2005	Amenaza	
ISO/IEC 17999:2005	Requisito de Seguridad	

Stakeholders

Personas Disponibles en la Base de Datos

Personas	Roles
Moises	Asegurador Calidad
Daniel	Desarrollador de Seguridad
Juan	Experto en Riesgos
Mario	Experto en Seguridad
	Ingeniero Requisitos Seguridad
	Modelador de Negocios
	Administrador
	Jefe Proyecto
	Equipo de Inspeccion

Personas Relacionadas con el Proyecto

Personas	Roles
Moises	Asegurador Calidad
Daniel	Ingeniero Requisitos Seguridad
Juan	Equipo de Inspeccion
Mario	Jefe Proyecto
Mario	Experto en Seguridad

Visión de Seguridad

Concepto: Política de Seguridad Organizacional

Nivel de Aseguramiento: EAL2: Probado Esctructuralmente

Descripción

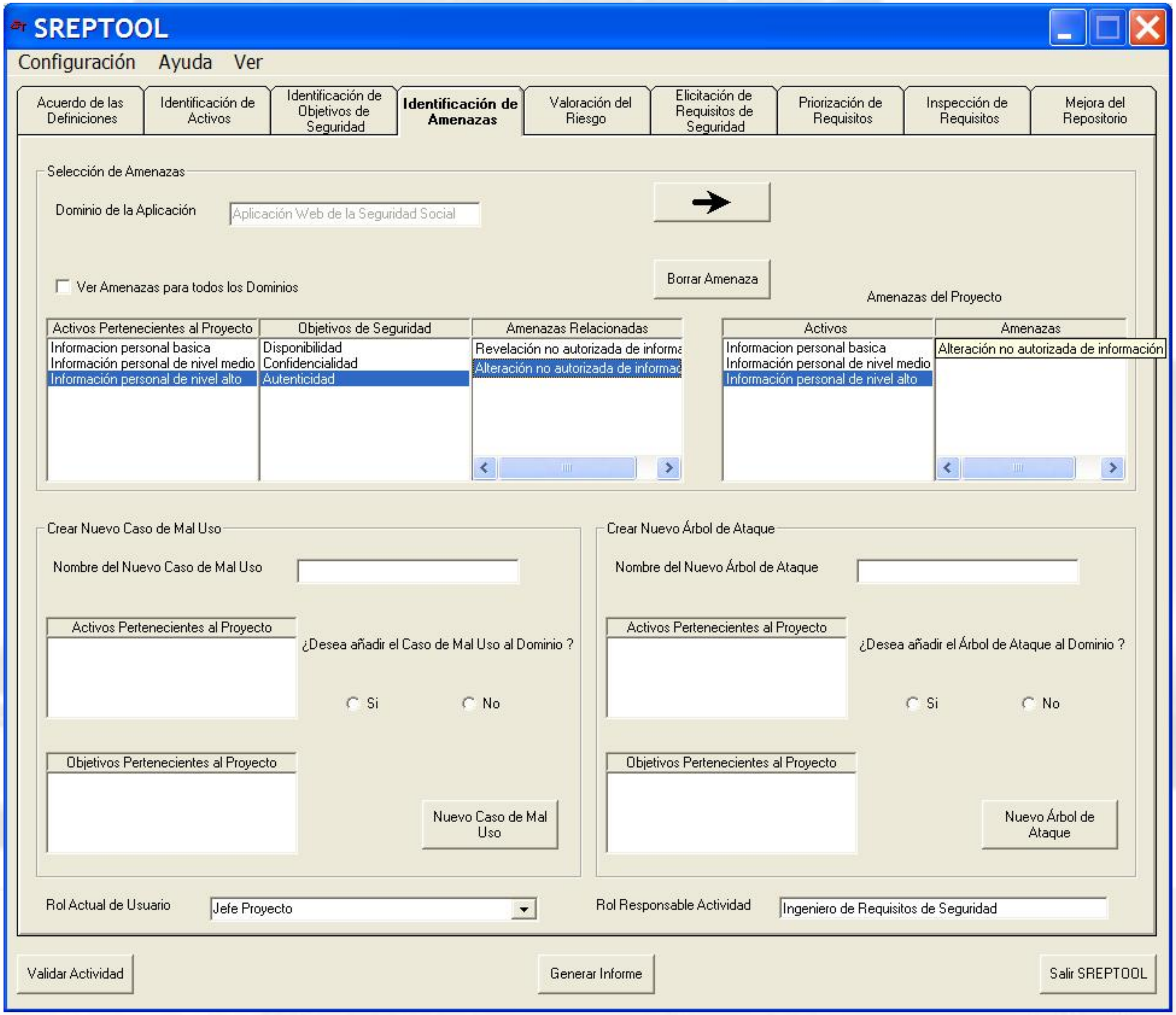
1. Justificación
Los activos de información y los equipos informáticos son recursos importantes y vitales de nuestra Compañía. Sin ellos nos quedaríamos rápidamente fuera del negocio y por tal razón la Presidencia y la Junta Directiva tienen el deber de preservarlos, utilizarlos y mejorarlos. Esto significa que se deben tomar las acciones apropiadas para asegurar que la información y los sistemas informáticos estén apropiadamente protegidos de muchas clases de amenazas y riesgos tales como fraude, sabotaje, espionaje industrial,

Rol Actual de Usuario: Jefe Proyecto

Rol Responsable Actividad: Ingeniero de Requisitos de Seguridad

SREPTool - prototype

❖



Selección de Amenazas

Dominio de la Aplicación: Aplicación Web de la Seguridad Social

□ Ver Amenazas para todos los Dominios

Activos Pertenecientes al Proyecto	Objetivos de Seguridad	Amenazas Relacionadas
Informacion personal basica	Disponibilidad	Revelación no autorizada de informa
Información personal de nivel medio	Confidencialidad	Alteración no autorizada de informac
Información personal de nivel alto	Autenticidad	

Amenazas del Proyecto

Activos	Amenazas
Informacion personal basica	Alteración no autorizada de información
Información personal de nivel medio	
Información personal de nivel alto	

Crear Nuevo Caso de Mal Uso

Nombre del Nuevo Caso de Mal Uso:

Activos Pertenecientes al Proyecto

¿Desea añadir el Caso de Mal Uso al Dominio ? ○ Si ○ No

Objetivos Pertenecientes al Proyecto

Crear Nuevo Árbol de Ataque

Nombre del Nuevo Árbol de Ataque:

Activos Pertenecientes al Proyecto

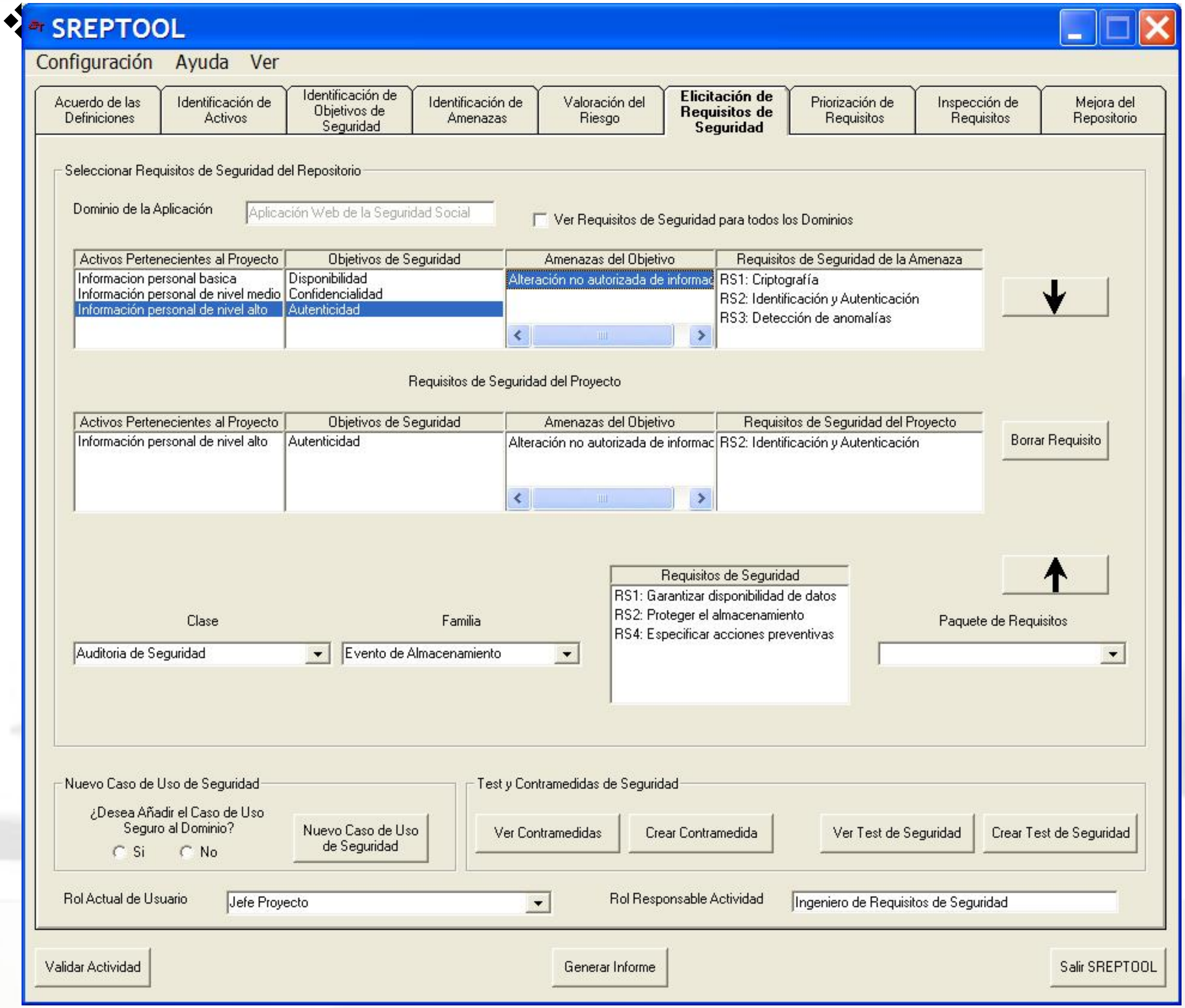
¿Desea añadir el Árbol de Ataque al Dominio ? ○ Si ○ No

Objetivos Pertenecientes al Proyecto

Rol Actual de Usuario: Jefe Proyecto

Rol Responsable Actividad: Ingeniero de Requisitos de Seguridad

SREPTool - prototype



Seleccionar Requisitos de Seguridad del Repositorio

Dominio de la Aplicación: Aplicación Web de la Seguridad Social

□ Ver Requisitos de Seguridad para todos los Dominios

Activos Pertenecientes al Proyecto	Objetivos de Seguridad	Amenazas del Objetivo	Requisitos de Seguridad de la Amenaza
Informacion personal basica	Disponibilidad	Alteración no autorizada de informac	RS1: Criptografía
Información personal de nivel medio	Confidencialidad		RS2: Identificación y Autenticación
Información personal de nivel alto	Autenticidad		RS3: Detección de anomalías

Requisitos de Seguridad del Proyecto

Activos Pertenecientes al Proyecto	Objetivos de Seguridad	Amenazas del Objetivo	Requisitos de Seguridad del Proyecto
Información personal de nivel alto	Autenticidad	Alteración no autorizada de informac	RS2: Identificación y Autenticación

Clase: Auditoria de Seguridad

Familia: Evento de Almacenamiento

Requisitos de Seguridad
RS1: Garantizar disponibilidad de datos
RS2: Proteger el almacenamiento
RS4: Especificar acciones preventivas

Paquete de Requisitos:

Nuevo Caso de Uso de Seguridad

¿Desea Añadir el Caso de Uso Seguro al Dominio?

○ Si ○ No

Test y Contramedidas de Seguridad

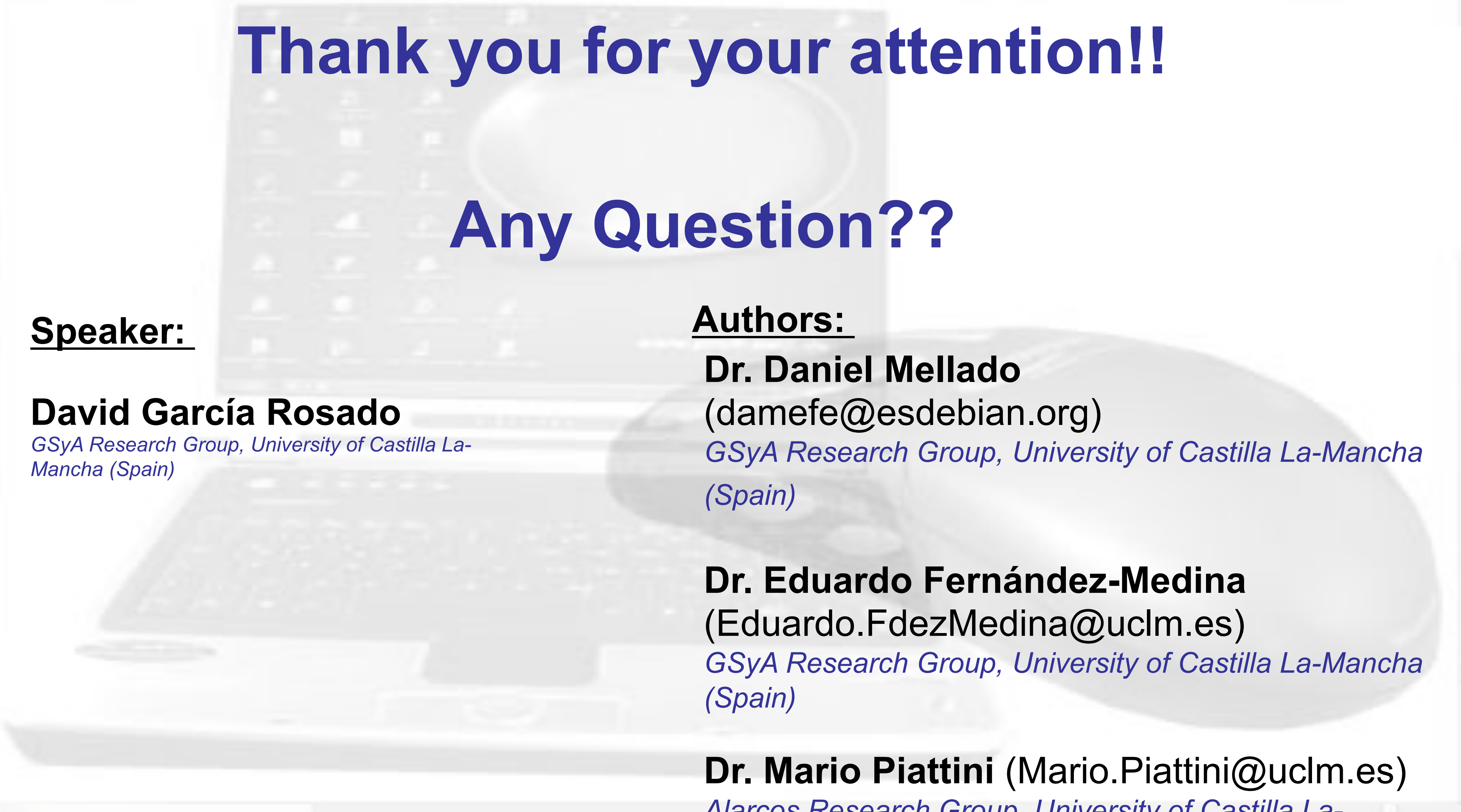
Rol Actual de Usuario: Jefe Proyecto

Rol Responsable Actividad: Ingeniero de Requisitos de Seguridad

Conclusions and Further Work

- The increasingly crucial nature of IS with corresponding levels of new legal and governmental requirements → development of more and more sophisticated approaches to ensuring the security of information → it is fundamental to deal with security at the early stages of software development
- We demonstrate how the security requirements for a security critical IS can be obtained in a guided and systematic way by applying SREP
- Our proposal deals with the security requirements at the early stages of software development in a systematic and intuitive way
    - reusability (Security Resources Repository).
    - integration of the CC
    - standards ISO/IEC 15408 (CC), ISO/IEC 17799:2005, and ISO/IEC 13335 (GMITS).
    - techniques (miss-use cases, UMLSec,…)
    - micro-process of 9 activities, iterative and incremental

- Further Work…
  - We are developing a CARE (Computer-Aided Requirements Engineering) tool which supports the process (current prototype: SREPTool)
  - A refinement of the theoretical approach by proving it in real case studies.
- More details with regards SREP (our suggested methodology) can be found in :
  - Daniel Mellado, Eduardo Fernández-Medina, Mario Piattini: A common criteria based security requirements engineering process for the development of secure information systems. Computer Standards & Interfaces 29(2): 244-253 (2007)



Thank you for your attention!!

Any Question??

Speaker:

David García Rosado
GSyA Research Group, University of Castilla La-Mancha (Spain)

Authors:

Dr. Daniel Mellado
(damefe@esdebian.org)
GSyA Research Group, University of Castilla La-Mancha (Spain)

Dr. Eduardo Fernández-Medina
(Eduardo.FdezMedina@uclm.es)
GSyA Research Group, University of Castilla La-Mancha (Spain)

Dr. Mario Piattini (Mario.Piattini@uclm.es)
Alarcos Research Group, University of Castilla La-Mancha (Spain)