Law and the software development life cycle

November 25, 2017

Cesare Bartolini, Gabriele Lenzini

Interdisciplinary Centre for Security, Reliability and Trust (SnT),
University of Luxembourg

Outline

1. Legal requirements

2. The Software Development Life Cycle

3. Legal requirements in the Software Development Life Cycle (SDLC)

4. Putting it all together

Outline

1. Legal requirements

2. The Software Development Life Cycle

3. Legal requirements in the SDLC

4. Putting it all together

Requirements in software

- Functional
  - *What* the system must do
- Non functional
  - *How* the system must do it

Requirements in software

- Functional
  - *What* the system must do
- Non functional
  - *How* the system must do it

Typical non functional requirements

- Performance (good quality software)
- Security (confidentiality of information)
- Efficiency (limited use of resources)
- Cost-effectiveness (competitiveness on the market)
- Usability (easy to use for its target customers)
- ...

Requirements in software

securityandtrust.lu

- Functional
  - *What* the system must do
- Non functional
  - *How* the system must do it

Typical non functional requirements

- Performance (good quality software)
- Security (confidentiality of information)
- Efficiency (limited use of resources)
- Cost-effectiveness (competitiveness on the market)
- Usability (easy to use for its target customers)
- . . .
- Compliance with legal obligations

Ratio of legal requirements

- Laws set rules for enterprises
  - Obligations / prohibitions / permissions

Ratio of legal requirements

- Laws set rules for enterprises
  - Obligations / prohibitions / permissions
- Already happened in the past
  - Products (health, transparency, competition. . .)
  - Industrial processes (safety, environment. . .)
- Now happening in the digital world

Ratio of legal requirements

- Laws set rules for enterprises
  - Obligations / prohibitions / permissions
- Already happened in the past
  - Products (health, transparency, competition. . .)
  - Industrial processes (safety, environment. . .)
- Now happening in the digital world
- Growing number of digital policies
  - Especially in the European Union

Purposes

- Corporates
  - Security for trade secrets
  - E-commerce
  - Intellectual property
- Users
  - Data protection
  - Privacy
- Public safety
  - Cybersecurity
  - Data and news reliability
  - Social trust

Purposes (2)

- Crime control
  - Backdoors
  - Access to authorities
  - Notice and take down
- National security
  - Export control
  - Security in military / intelligence software

Legal sources

- Law
  - HIPAA
  - E-commerce Directive
  - General Data Protection Regulation (GDPR)
  - Export control (ITAR)
  - ...
- Policies / standards
  - Security standards
  - Sectorial standards
- Contracts
  - Service-Level Agreements (SLAs)

Standards and laws

Policies / standards may be mandated

- PCI DSS (payment cards) in Nevada & Washington
- A variant of ISO 13485 (medical devices) in Mexico
- . . .

Standards and laws

Policies / standards may be mandated

- PCI DSS (payment cards) in Nevada & Washington
- A variant of ISO 13485 (medical devices) in Mexico
- . . .

Problems

Mandatory standards can introduce limitations to competitivity due to stringent requirements that may limit the target market.

Two types of requirements

Organizational

- Concerns the structure of the enterprise or the business processes
- May introduce specific roles
- May introduce specific activities
- May introduce specific timings
- May depend on enterprise size and type

Two types of requirements

Organizational

- Concerns the structure of the enterprise or the business processes
- May introduce specific roles
- May introduce specific activities
- May introduce specific timings
- May depend on enterprise size and type

Technical

- Concerns specific activities to be put into place
- Depend on the technical state of the art
  - By means of a *relatio*
- May or may not evolve in time
  - Formal or substantive *relatio*
- May exclude from damage liability
- May be integrated into the SDLC

Outline

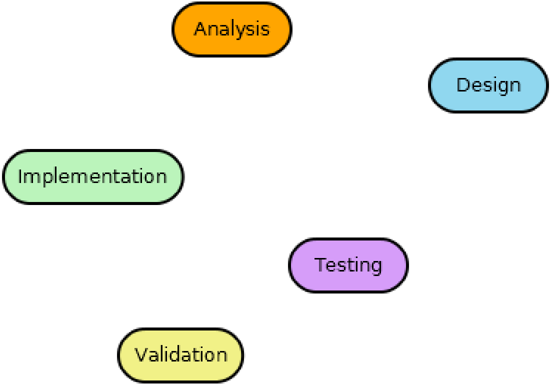
1. Legal requirements

2. The Software Development Life Cycle

3. Legal requirements in the SDLC

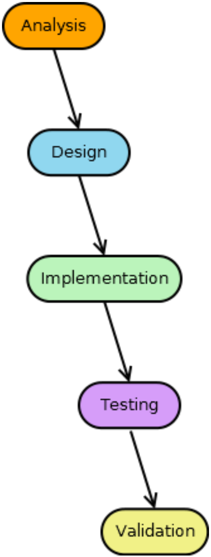
4. Putting it all together

SDLC concept



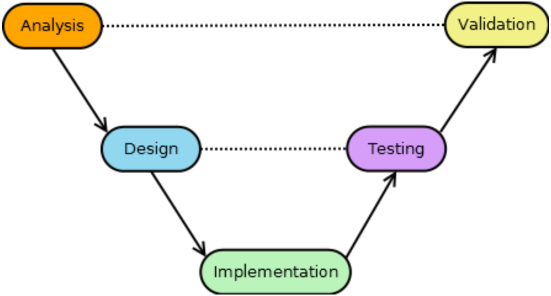
Stages of the SDLC.

SDLC structures



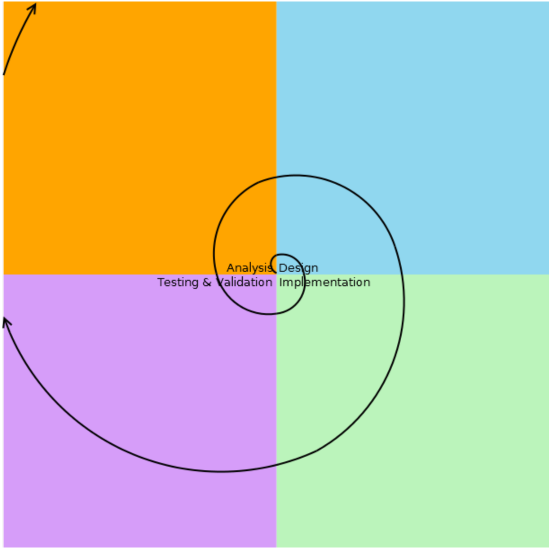
The waterfall model.

SDLC structures (2)



The V-model.

SDLC structures (3)



The spiral model.

Dealing with requirements

- Formal definition
- Representation (model)
- Implementation (measures)
- Assessment (metrics)
- Monitoring

Outline

1. Legal requirements

2. The Software Development Life Cycle

3. Legal requirements in the SDLC

4. Putting it all together

One objective, many solutions

- SDLC extension with legal requirements can happen in many ways
- Different methodologies for each SDLC stage
- Also depend on the software engineering approaches used
- Just a few guidelines

Definition

- Definition written in legal language
    - Especially when the source is the law
    - Standards and contracts may give an easier time
- Many possible technical definitions
    - Only partial overlap between legal and technical definitions
- Definition must be interpreted
    - May differ depending on interpretation

Definition

- Definition written in legal language
  - Especially when the source is the law
  - Standards and contracts may give an easier time
- Many possible technical definitions
  - Only partial overlap between legal and technical definitions
- Definition must be interpreted
  - May differ depending on interpretation

Examples

Service, cloud, database, file, request. . .

More than words

- Affects all of the following stages
  - Model
  - Implementation
  - Metrics
- Taken from literature or *ad hoc*
- May require feedback from later stages. . .
  - . . . if it proves too problematic to use
  - . . . if the scope is too broad or too narrow
  - . . . if it is not useful enough

Formal definition

Natural language

Ontologies

Ontologies

- Knowledge representation
- Descriptions of a knowledge domain
- Language used: Web Ontology Language (OWL)
  - (*Sic*)
- Representation of real-world objects
- They do not *define* anything
  - Objects are defined in the domain itself
- They *describe* relations

A parliament of OWLs

- Ontologies can be extended with *deontic rules*
  - *must*
  - *should not*
  - *may*
  - ...
- Legal ontologies
- These can describe duties etc.

A parliament of OWLs

- Ontologies can be extended with *deontic rules*
  - *must*
  - *should not*
  - *may*
  - ...
- Legal ontologies
- These can describe duties etc.

- They can be used to describe legal requirements

Representation

- Describes the requirement in formal terms
- Various degrees of detail
- *Can* include a destructuring
- *Can* include relationship with other requirements
- *Should* include metrics for evaluation
- *Should* connect to the design tools and models

Formal models

- Unified Modeling Language (UML)
  - Easy to connect with design tools
- i*
  - Highlights roles of stakeholders
- Goal model
  - Hierarchical representation
- 4-variable model
  - Strong connection between actual data and software
- . . .

Implementation

- The requirement must be implemented into the software
- Implementation differs depending on many factors
  - Development tools
  - Programming language
  - Content of requirement
  - Nature of requirement
    - Functionality
    - Performance
    - Restriction
    - . . .

Sample implementations

Right of access to personal data

- Requires a module that grants access
  - Front-end interface
  - Authentication method
  - Data base and query engine

Sample implementations

Right of access to personal data

- Requires a module that grants access
  - Front-end interface
  - Authentication method
  - Data base and query engine

Encryption protocol for secure payments

- Needs a component to process encrypted data
  - Encrypting module
  - Decryping module

Sample implementations

Right of access to personal data

- Requires a module that grants access
  - Front-end interface
  - Authentication method
  - Data base and query engine

Encryption protocol for secure payments

- Needs a component to process encrypted data
  - Encrypting module
  - Decryping module

Export control under ITAR regulations

- Access must be denied to non-citizens
  - Database of citizenships
  - Access limitations

Assessment

Compliance

- For every requirement in the specification
- Depending on its nature
  - Qualitative (e.g., the functionality is present / not present)
  - Quantitative (e.g., measure of the security strength)
- At different levels
  - Component
  - Integration

Assessment

Compliance

- For every requirement in the specification
- Depending on its nature
  - Qualitative (e.g., the functionality is present / not present)
  - Quantitative (e.g., measure of the security strength)
- At different levels
  - Component
  - Integration
- Metrics must be implemented

Assessment

Compliance

- For every requirement in the specification
- Depending on its nature
  - Qualitative (e.g., the functionality is present / not present)
  - Quantitative (e.g., measure of the security strength)
- At different levels
  - Component
  - Integration
- Metrics must be implemented
- At least for quantitative assessments

Work in progress

Examples

Reliability

- System must backup data in three different locations
  - Backup delay
  - Backup time
  - Security of transfer

Examples

Reliability

- System must backup data in three different locations
  - Backup delay
  - Backup time
  - Security of transfer

Transparency

- System must provide information in a clear and intelligible form
  - Usability of the interface
  - Detailedness of the information
  - Clarity of the language used

Monitoring

- Things change over time (e.g., functionality, hardware, laws)

Monitoring

- Things change over time (e.g., functionality, hardware, laws)
- And sometimes they just don't work as they appear on paper

Monitoring

- Things change over time (e.g., functionality, hardware, laws)
- And sometimes they just don't work as they appear on paper
- And sometimes a periodic check is mandated

Monitoring

- Things change over time (e.g., functionality, hardware, laws)
- And sometimes they just don't work as they appear on paper
- And sometimes a periodic check is mandated

- Evaluate compliance over time
- Implementation of monitoring tools
- Halfway between implementation and testing
- Reports

Outline

1. Legal requirements

2. The Software Development Life Cycle

3. Legal requirements in the SDLC

4. Putting it all together

The three-eyed researcher

Three different perspectives

- Analysis and formalization of legal requirements
- Modelling legal requirements and defining metrics
- Integrating legal requirements in all stages of the SDLC

How to achieve it

- Currently only some *ad hoc* solutions for specific requirements
- More standardized approach to legal requirements
- Techniques to model interpretation
- Classification of legal requirements
- Extending SDLC methodologies and tools