

# Cross-cutting Concepts

## Cross-cutting Concepts

### *Content*

This section describes overall, principal regulations and solution ideas that are relevant in multiple parts (= cross-cutting) of your system. Such concepts are often related to multiple building blocks. They can include many different topics, such as

- domain models
- architecture patterns or design patterns
- rules for using specific technology
- principal, often technical decisions of overall decisions
- implementation rules

### *Motivation*

Concepts form the basis for *conceptual integrity* (consistency, homogeneity) of the architecture. Thus, they are an important contribution to achieve inner qualities of your system.

Some of these concepts cannot be assigned to individual building blocks (e.g. security or safety). This is the place in the template that we provided for a cohesive specification of such concepts.

### *Form*

The form can be varied:

- concept papers with any kind of structure
- cross-cutting model excerpts or scenarios using notations of the architecture views
- sample implementations, especially for technical concepts
- reference to typical usage of standard frameworks (e.g. using Hibernate for object/relational mapping)

### *Structure*

A potential (but not mandatory) structure for this section could be:

- Domain concepts
- User Experience concepts (UX)
- Safety and security concepts
- Architecture and design patterns
- "Under-the-hood"
- development concepts
- operational concepts

Note: it might be difficult to assign individual concepts to one specific topic on this list.

[Possible topics for crosscutting concepts]

<Concept 1>

<explanation>

<Concept 2>

<explanation>

...

<Concept n>

<explanation>

## Directory Structure and Required Files

```
| -dechat5a
|   | -chat1
|   |   | -conversacion.ttl
|   | -chat2
|   |   | -conversacion.ttl
|   | -chat3
|   |   | -conversacion.ttl
|
```