tecurat

TECURAT'S GUIDE TO DIGITALIZATION

PART 2: USABILITY ENGINEERING

COVER USABILITY IN THE REQUIREMENTS
SIMPLIFY YOUR USABILITY ANALYSIS
AUTOMATE YOUR FIRST DOCUMENTATION STEPS

Efficient Usability Engineering

In our last post, we discussed the structuring of requirements and the levels used. Did you miss it?

Here's the first part of the series: Part #1.

In this post, we explain why this division is beneficial and how it helps you meet the usability requirements of IEC 62366-1.

Core functions

As described in Part #1, the requirement structure includes a step-by-step implementation and detailing of requirements according to the design. This step-by-step detailing applies not only to design requirements but also to usability requirements.

This simplifies the development of individual requirements and the overall requirement documentation, as you work directly with the requirements themselves.

Structure of requirements

The requirement structure follows not only the design but also the functionality of the product. We start at the highest level with the core tasks (Core-Tasks) of the product, which are derived from the purpose of the product. These describe:

- **Who**: Which user...
- **How**: ...through which interaction with the product...
- **What**: ...can fulfill a specific purpose or achieve a benefit

Parallel to the development of the design architecture on three levels, these actions are also defined at the corresponding levels.

Unlike Use Cases in UML (Unified Modeling Language), we consider the user interaction and the respective design component together.

The core tasks focus exclusively on user interactions and form the basis for usability considerations. Since interactions are directly analyzed and defined with the requirements, the device behavior with further dependencies is also defined.

This allows you to directly fulfill the requirements of FDA HFE (Human Factors Engineering) and consider cybersecurity aspects (more on this later).

Requirements and multiple levels

Requirements form the binding element of a product specification. A key requirement here is that higher-level design levels must be captured and further specified. This must be implemented for each level until a concrete technical implementation is defined.

Requirements and multiple levels (contd.)

The dependencies between the requirements of these individual levels are captured, so it can be monitored whether all product requirements are technically implemented.

- **Highest level**: Requirements are posed by stakeholders (e.g., doctors, nurses, patients). Example: The product should provide a suction function to support an operation. (the specific medical application should be defined here)
- **Lowest level**: A specific technical solution is described. Example: A colored button.

Each requirement must be associated with the architecture and a core function, such as a system component and a user action. This approach is similar to the double-entry method in accounting. By switching the perspective from requirement to product architecture, you can better analyze whether a system component or interface is sufficiently specified.

This method can be easily implemented using a database.

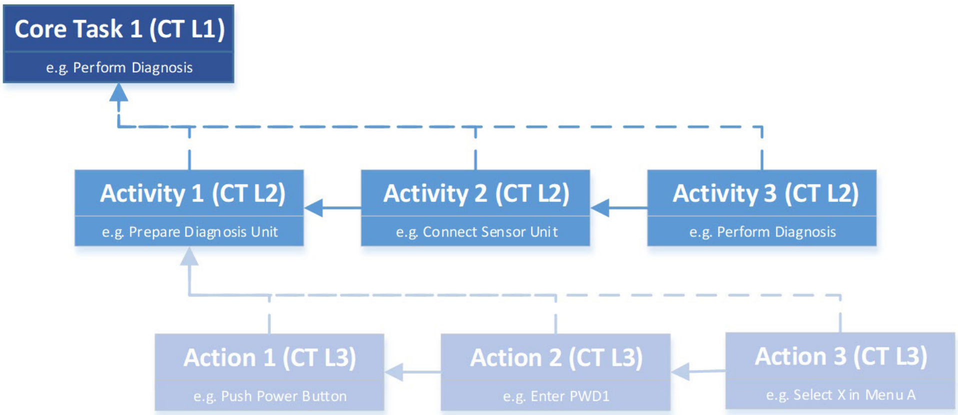

Fig. 1: Core tasks along the design levels