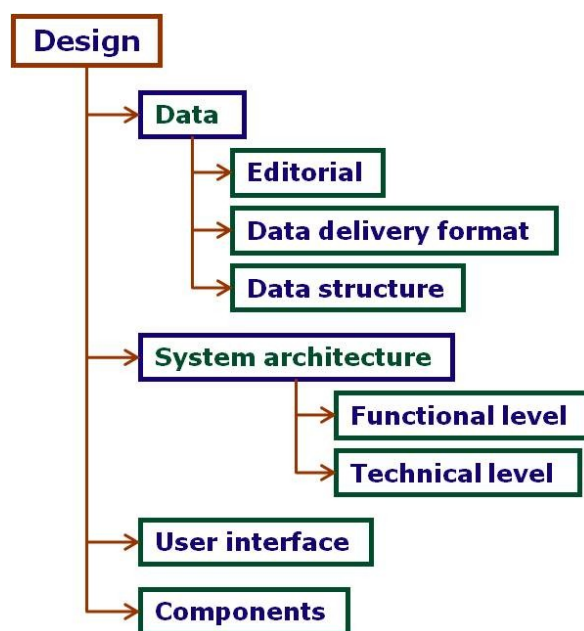# 13) Design specifications

General remarks

This chapter addresses issues related to design specifications that are of interest to a PM, more precisely in the case of a project involving content creation and software development (in the general sense of the word, ie including website development). It does not describe how to create content or design software, which are subjects generally covered in more specialized courses and manuals.

Design specifications are the "**blueprint**" for the work to be undertaken in order to **meet the needs expressed in a requirements specification**. As such, the design specifications describe in detail **how meeting the requirements will be achieved**.

Design specifications generally cover the following areas:

```
Design
   └──> Data
            └──> Editorial
            └──> Data delivery format
            └──> Data structure
   └──> System architecture
            └──> Functional level
            └──> Technical level
   └──> User interface
   └──> Components
```

The purely functional description of the system architecture and the details of the user interface may be contained in a single document called "**functional specifications**".

The functional specifications and data delivery format specifications are of interest to all parties, ie not only the developers and the development subproject manager, but also the PM leading the overall project, as well as other subproject managers (and possibly team members) whose job involves a close relationship with developers.

The editorial specifications are generally of interest to content creators exclusively.

The data structure specifications, the system architecture specifications at detailed technical level, and the component-level technical design specifications are generally of interest to developers exclusively.

Specifications of interest to the project owner and the project executors should be **reviewed and approved by the PM** and the project management team.

It is essential to ensure that design specifications are **exhaustive and precise**, are documented at an **adequate level of detail**, and **comply with the requirements** document.

Ambiguities and "grey areas" that might lead to misinterpretation should be clarified before sign-off.

It is strongly recommended to spend **as much time as necessary** on the design specifications, in order not only to **facilitate the subsequent product creation process**, but also to **avoid unpleasant surprises** with deliverables, and to **avoid potential conflicts** between the project owner and project executors such as content creators and software developers.

As a rule, a project executor should never accept requirements that appear impossible to meet (given technical, schedule, budget or other constraints).
Likewise, a project owner should never accept design specifications that do not comply with the requirements document.

Note that an **iterative process** is generally necessary in order to obtain a "perfect match" between design and requirements specifications. Indeed, specialists (editors, developers, etc.) involved in the design process may establish that some of the requirements cannot be met (due to technical or other reasons). If such a situation arises, solutions must be devised and agreed upon by all parties concerned, and the requirements document may need to be amended accordingly.

Once they have been agreed upon, the final design specifications of interest to both parties may be appended to the project execution contract.

Vocabulary/abbreviations

It is useful to provide a list of specific vocabulary (with precise definitions) and any abbreviations used in the specifications, in order to avoid any ambiguity or misinterpretation. That list may appear at the beginning of each of the design specifications documents or as an appendix, and/or may be stored for easy access in a central location (eg on an intranet or extranet site).

Editorial specifications

For a product with editorial content, for example an encyclopedia or an online newspaper or magazine, or an e-commerce website with a description of goods or services, as opposed to a software application without much editorial content such as a spreadsheet or a text processor or a web browser, it is necessary to write editorial specifications intended for authors, editors, etc.

Editorial specifications may cover for example (non-exhaustive list):

- the nomenclature (list of elements of content),
- content classification/hierarchy (sections, categories, topics…),
- guidelines relating to size, editorial style and other characteristics of texts,
- the style sheet for the presentation of texts,
- more generally, a style guide (or style manual),
- language(s) and localization constraints,
- links,
- metadata,
- DTD(s),
- style and format of multimedia assets,
- scenarios for animations.

Note that products generally assumed to be "without content", such as those mentioned above, may in fact include some content, for instance Help text, possibly illustrated, for which editorial specifications should be written.

Editorial specifications are a reference and a framework for people involved in **creating or editing content**. In particular, complying with the specifications will ensure a maximum degree of consistency among all elements of content, eg in terms of style.

Editorial specifications should match, but with more detail, the editorial section of the requirements document.

Editorial specifications are generally intended exclusively for people involved in content creation, not for software developers. They are usually written by the person in charge of the editorial subproject.

Data delivery format specifications

This document describes, at the lowest possible level of detail, how the **data** that will be delivered by the content provider to the software developer will be structured, tagged and formatted. It also describes the **metadata** associated with each data item or data set.
It may include a "**Document Type Definition (DTD)**" with a clear explanation of the meaning of each of the tags that will be used to mark up the data.
It is essential that both parties involved agree upon the data delivery format specifications before content creation and software development work is started, in order to avoid the cost of changes in the structure and tagging of content.

The data delivery format specifications may be written by the editorial subproject manager (or data engineering subproject manager, if any) or by the development subproject manager, or written as a collaborative effort between the two parties (content provider and software developer).

Note that the data delivery format specifications may actually be provided as part of or as an appendix to the requirements specification, as mentioned in chapter 8.

Data structure specifications

The data structure specifications are generally intended exclusively for software developers. They define the detailed structure of data for implementation purposes. These specifications should describe how data will be organized, stored, indexed, compressed, decompressed, encrypted, decrypted, retrieved, etc.

The data structure specifications are usually written by developers and/or the development subproject manager. In the case of complex data structures, an expert "database designer" may be involved.

Functional specifications

Functional specifications describe at a functional level the **architecture** of the software application to be developed, the detailed **behaviour** of its functional components, as well as the details of its **user interface**.
They describe how the **functions and features** expressed in the requirements specification will be implemented from a purely functional viewpoint.

The overall **architecture** of a software application (or "system") is comparable to the work breakdown structure of a project in that it represents the manner in which the system is divided into components (or units or modules) and structured. It should describe the manner in which components are related to one another and how they interact.

The system architecture can generally be described at the highest level with a **summary diagram**, and at lower levels with a series of more **detailed diagrams**.

Diagrams however are generally not sufficient to describe the precise behaviour of functional components. **Detailed narratives** must therefore be provided.

As regards the **user interface** of the software application (or system), a precise and exhaustive description of its elements must be documented, both **in graphical form and narrative form**.

The purpose of each of the **active graphical user interface** ("GUI") elements (menu items, buttons, checkboxes or tickboxes, data entry/input areas, etc.) must be explained in detail, as well as the corresponding function to be performed by the software.

Mock-ups of the user interface are obviously very helpful. In particular, they may be used as **guidelines for the graphic design work**, which also requires a precise description of the parts into which the GUI should be divided ("cut") and delivered for integration with the software by the developers.

The functional specifications are usually written by the development subproject manager (who may be working for a software development contractor). They should be reviewed and approved by the overall PM and the project management team.

In the case of complex systems, an expert "system architect" may be involved in writing system architecture specifications.

Note that the development team may require a more detailed and more technical description of the system architecture than that included in the functional specifications. The detailed system architecture may be a separate document or it may be included in the system component design specification.

Component specifications

This document (or set of documents) describes the functions that each of the software application's (or system's) components should perform and how such functions will be implemented.
It should also describe the relationships and interfaces between components.

The component design specifications are usually written by developers, for themselves or for other developers. If the development subproject manager is himself a developer, which is often the case, he may be involved in writing and reviewing the component specifications.