4. tétel **Writing a Technical Specification document for our software project**

I have seen this post on the Internet. I would like to summarize it for you. I am sure it can help us to make the documentation of our software project. The writer asks about the Technical Specification and looks for any practice suggestion or real experience.

At first, the Technical Specification should contain the **project goals, the system architecture/infrastructure, the user dialogs and the control flow.** It should be a kind of **manual** for the users, end users or clients. Next, it is advisable to draft it **clearly** so that the readers can understand the text. The documentation should be **short** and written in **everyday English**. Also, it should be **detailed** because it is worth explaining everything that the users could see on the screen. We must **avoid technical terms** but we need to compose the text accurately so that the average user could understand it and follow what to do.

In my opinion, we should write about only the **tasks of the subprograms** and the **lists of the parameter**. It is important to determine the **minimum and maximum** hardware and software conditions. If we need an **external program**, we must know its version and the location of the downloading (e.g.PDF reader). The program must be **compatible** with the software browser and the hardware. We should write about the **database model,** the **background tasks** and the **interfaces** to other systems in the Functional Specification (not in the Technical Specification).

In addition, it is also important to **give examples**. They are an excellent way for end users to quickly grasp concepts that they may not fully understand. It is essential to have the section of the Frequently Asked Questions (**FAQs)** but it is sufficient if the users can access this part through the web.

Furthermore, it is useful **to test** the documentation. We can ask an average user to check on the text (e.g. its vocabulary).We should finalize the document **together with the tester.**

It is also a good idea to make our document with **pictures** if it is possible. In this way, we can minimize the length and complexity of our documentation. System users like having pictures, diagrams and bulleted lists for quick reference.

Finally, we can **anticipate problems**. We will not only save a lot of frustration for the end users but also a lot of extra calls to the help desk.