## Overview

Some companies have specific, written requirements regarding documentation. That is wonderful, we encourage that, and we will conform to those specifications, whatever they are. Some companies have guidelines for documentation, and that is OK too. Companies that have never done an electronic project, or have never done one with software, probably don't have any standards at all. That really isn't OK, for reasons that will be detailed below. In that case, we can help you develop a reasonable set of standards that are not too onerous, but still offer a reasonable assurance of getting the job done in a timely fashion.

## The Case for Adequate Documentation

There definitely is such a thing as too little documentation. The most important thing that the documentation tells you is when the project is done. Blame for continually adding clever features that hardly anybody uses is frequently placed on engineers, and that is sometimes justified. However, marketing and top management also have to take their share of the blame for this creeping featurism. Everyone needs to understand that a product must ship to make any money. You can have the best product in the world sitting on the lab bench, but if the competition is making a killing selling a product that is only half as good, guess who gets the Christmas bonus? Besides, shipping a product that has only 90% of what everyone wants leaves more room for an expanded product line with more features and higher profit margin.

Proper documentation up front also gives everyone involved a more complete picture of the amount of work needed to do the project before the project starts. If the client thinks that the job consists of pulling a canned design out of the filing cabinet, and the engineer thinks that he is going to be advancing the state of the art, the situation is bound to end with recriminations, finger pointing, and a product that cost a fortune and still doesn't work.

In addition, the U.S Air Force has invested considerable effort in figuring out where software bugs originate. This research shows that 70% of all software faults can be traced directly to inadequate requirements. That means every fault ever found over the entire life of the product begins before the first line of code is written.

Note that all of the above is true even if (maybe especially if) the project is purely for research purposes, with no production intended. In this case, it is especially important to keep the effort on track, and not to stray off into interesting side paths.

## IEEE weighs in

The Institute for Electrical and Electronic Engineers has issued a series of dozens of specifications for the development of software. If your project will have thousands of programmers working on several continents, or if the failure of your product means that your company will be the lead item on the evening news for a week and the subject of lawsuits for a decade, then all of these specifications are probably

justified. But most projects aren't like that, so a subset of these specifications is more reasonable and achievable.

The IEEE specifications for software can be reduced to a Software Requirements Specification (SRS), a Software Description Document (SDD), a validation test plan, and the results of the test plan. The essential thrust of this list is: plan your work, do the work according to the plan, and prove that the work was done properly.

Additional documentation that isn't directly about the software includes the schematic, and a system requirements specification (SYRS). A variety of production related documents may also be required, but these tend to be company specific and so are not covered here. In this document, the words "engineer" and "programmer" may refer to in-house employees, an outside consultant, or both at once.

## Specifying System Requirements

The System Requirements Specification (SyRS) is actually the first document to be done. It explains what the product will do in terms that everyone involved can understand. It details price, features, estimated sales levels, and possibly a brief survey of the competition. If there are government or industry standards that must be met, they are listed here. If compatibility must be maintained with other products, that is mentioned here also.

The marketing department has the main responsibility for writing this document, but must have considerable input from engineering. How long should the SyRS be? We have seen two page documents that were very complete. Of course, these were for products with no user interface of any sort, and no government standards to conform to. A more typical length might be 10 to 15 pages.

## Specifying Software Requirements

The Software Requirements Specification (SRS) is the earliest document that is specifically related to the program. The SRS is derived from the SyRS, and explains what the software will do, but not how it will do it. The SRS may suggest multiple solutions to a problem, but does not dictate any one solution.

This document bridges the gap between the client, who understands what the product must do, and the programmer, who understands how the software must make the product meet the specification. The target audience is the programmer, but the client must understand it completely.

IEEE has an excellent standard (IEEE Std. 830), with a very useable template.

The length of the SRS seems to be very dependent on the complexity of the user interface. If all communication with the outside world is through some sort of serial interface, then 10 to 20 or so pages are

usually plenty. A serious user interface will require a much longer document. We had one design that featured a graphic LCD, with text in half a dozen languages. The user interface portion of the SRS was over 50 pages, and took over two months to write. All of the rest of the SRS amounted to three pages, and was written in one day.

## Documenting the finished software

The Software Description Document (SDD) explains how the software does its job. If there are going to be flowcharts, data flow diagrams, etc anyplace, they go here. The programmer writes the SDD. The target audience is the programmer who has to maintain the product in the future, who may or may not be the same person as the original programmer.

IEEE has a standard (IEEE Std. 1016), but unfortunately, it isn't as useful as the SRS standard. The essential element is to have a good text explanation of what each routine actually does. This document is also the best place for the programmer to explain why a solution has the particular form that it does, and to issue cautions to future programmers about things to look out for.

The SDD is the most difficult document to get done, because most programmers lose interest when the program is "done". But, a proper SDD can save a fortune if another programmer has to take over, or if the same programmer has to revisit the code after an extended time away from it.

## Testing

The Verification and Validation (V&V) test plan tells everyone when the work is finished. The IEEE standard is Std 1012 and an implementation guide is in Std 1059. However, neither of these is particularly practical for normal projects. According to 1012, V&V begins when the program development begins, and ends when the last unit is hauled to the crusher. If you need that level of commitment, we will provide it. A more practical document is written with the SyRS and SRS close by, and answers the question, is the project done? Naturally, government and other legal requirements may have a lot of influence on the shape of this document.

The target audience for the test plan is the technician who will run the tests.

The acceptance test results prove that the work was finished. A spreadsheet listing each test, the allowable range of results and the actual results for each test is usually sufficient.

The target audience for the test results is everyone who has to sign off on the project, engineering, marketing, and top management.

## More information

The referenced IEEE documents, as well as all of the others, can be found at www.ieee.org. They are free to IEEE members, but not to non-members.