

AIS Project Management Process and Mythology



Development

Process activities/steps

❑ Software Elements Analysis:

The most important task in creating a software product is extracting the requirements. Customers typically know what they want, but not what software should do, while skilled and experienced software engineers recognize incomplete, ambiguous or contradictory requirements. Frequently demonstrating live code may help reduce the risk that the requirements are incorrect. All these techniques are well practiced by professional and experienced ASI Analysts.

❑ Specification:

Specification is the task of precisely describing the software to be written, possibly in a rigorous way. In practice, most successful specifications are written to understand and fine-tune applications that were already well developed, although safety- critical software systems are often carefully specified prior to application development. Specifications are most important for external interfaces that must remain stable.

❑ Software architecture:

The architecture of a software system refers to an abstract representation of that system. Architecture is concerned with making sure the software system will meet the requirements of the product, as well as ensuring that future requirements can be addressed. The architecture step also addresses interfaces between the software system and other software products, as well as the underlying hardware or the host operating system.

With a strong architectural and design background, ASI delivers successful projects.

❑ Implementation (or coding):

Reducing a design to code may be the most obvious part of the software engineering job, but it is not necessarily the largest portion.

We have clear activities and tools to decide the life cycle proportions.

❑ Testing:

Testing of parts of software, especially where code by two different engineers must work together falls to the software engineer. ASI has a well-dedicated team of test engineers.

❑ Documentation:

An important (and often overlooked) task is documenting the internal design of software for the purpose of future maintenance and enhancement. Documentation is most important for external interfaces. ASI follows a very well defined process related to documentation

❑ Software Training and Support:

A large percentage of software projects fail because the developers fail to realize that it doesn't matter how much time and planning a development team puts into creating software if nobody in an organization ends up using it. Users will have lots of questions and software problems, which leads to the next phase of software. ASI well acknowledges this fact and pays high attention to training.

❑ Maintenance:

Maintaining and enhancing software to cope with newly discovered problems or new requirements can take far more time than the initial development of the software. Not only may it be necessary to add code that does not fit the original design but just determining how software works at some point after it is completed may require significant effort by a software engineer. About $\frac{2}{3}$ of all software engineering work is maintenance.

ASI has experience in executing many maintenance projects alone.

