An Simple 5 Step Development Process for Engineering Consulting Projects

A Lextel, Inc. White Paper

Lextel, Inc.
131 Main St B475
N. Andover, MA 01845
781-245-5017
info@lextel.com
www.lextel.com

Introduction

Engineering projects involving digital electronics and software are notoriously unpredictable with respect to time and cost to complete. The end results are often not the same as those envisioned by the customer paying for the project. In one notable example, hundreds of millions of dollars of taxpayer funded software development effort was scrapped after many years of work when the 'end' result proved to be unusable.

As a practical matter, organizations and individuals involved in engineering projects need a process for insuring that projects are completed on time, within budget, and with results that accurately reflect the project requirements. The need for a simple process is especially important for the vast majority of projects that do not involve hundreds of engineers and millions of dollars in anticipated cost, but are instead on the scale of one to ten engineers working for less than 12 months.

It is just these types of projects that can easily end up taking two to five times the original time and cost estimates because the project is viewed as too small to warrant a 'formal' planning and scheduling process. However, when a large number of such projects are behind schedule and over budget, the end result can be costly.

Here we present a simple process that can be used by engineering firms, individual consultants and contractors, and the organizations that hire them. Implementing the process does not need to become a 'project in and of itself', and therefore can be used to guide a project without overburdening it.

The Five Step Development Process

The Lextel development process consists of the following five steps:

1. Information Gathering
2. Specification and Project Planning
3. Implementation
4. Final Test and Delivery
5. Maintenance and Support

The overall process is shown in the flow diagram in Figure 1. Each step will be described in detail in the following sections.

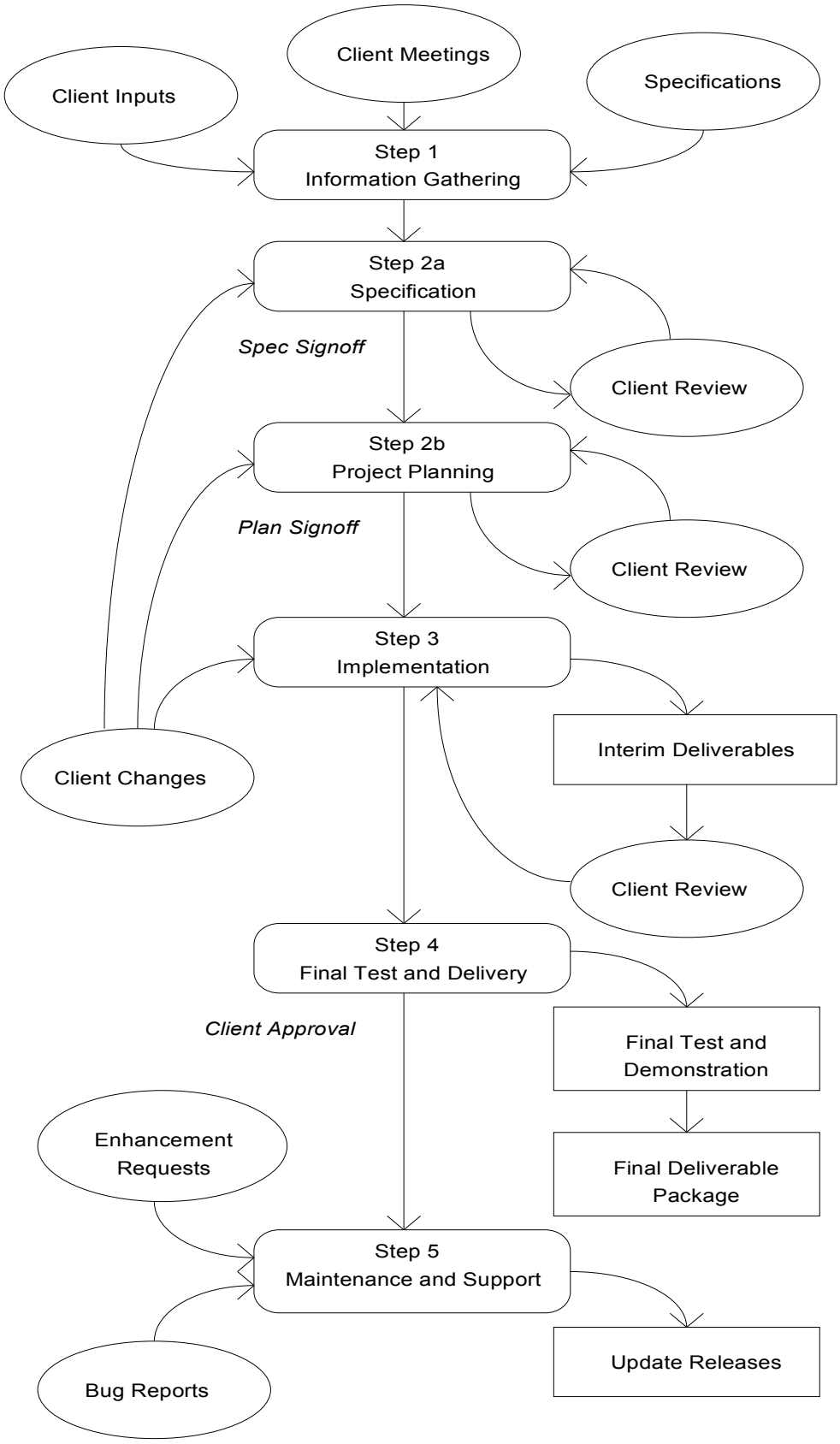

Figure 1. The Lextel 5 Step Development Process

Information Gathering

What to do

All projects must begin with information gathering. During this step, the consulting firm assembles all available information regarding the project. In a well-organized customer firm, a Product Functional Specification, RFP Document, or the like will have been prepared in advance. This document and all available relevant specifications need to be made available to the consulting firm at the start of a project. Other forms of input may include power point slides and presentations, which would be typical when a user interface is needed. Specifications of integrated circuits, industry standards, and the like must at least be referenced by the customer firm so that the consulting company can review them.

During an initial meeting, the customer firm should present all information to the consulting firm, as well as give a verbal summary overview of the project, noting priorities and any specific facts that may not be self evident from the written documentation.

All knowledgeable persons from the customer firm should be present at the information gathering session with the consulting firm. In a typical organization, important knowledge is dispersed among the people that have had some degree of participation in the project. It is not enough for the project manager to be present. Usually the project manager will not be aware of all important details.

The consulting firm may have questions during the initial meeting, but the main purpose is for the consulting firm to absorb as much as possible about the project goals. After the initial meeting, the consulting company should study the materials and reflect on the project. A set of detailed questions can be developed for further clarification on various aspects of the project that are unclear. Various trade offs regarding product features, time to develop, development priorities, and other issues will become clearer to the consultant.

After this initial review, a subsequent meeting or set of questions from the consultant to the customer should be prepared and responded to by the customer.

What not to do

Customers may have a desire to withhold information from the consulting company due to confidentiality or to keep things on a ‘need to know’ basis. However, this is usually a mistake. By withholding information, the customer may reduce the chances that the consulting firm will produce the desired result. For example, the customer may want to disclose only what it considers necessary to producing the desired output.

As a concrete example, a customer contracting for the design of a user controlled instrument may try to describe or document the specific control buttons and user interface elements and their workings, without fully disclosing who the end customers are, what their goals are, what is important to them, how and why they are using the end product, how existing products work, etc. This extra information may seem to be either superfluous to the task at hand – producing the product – since the customer has already thought that aspect of the situation out.

However, during the course of specifying and developing and testing the product, if the consultant tries to blindly follow the customer's directions without an understanding of the ‘why’ behind them, the result will be unsatisfactory. This is because things come up during development and testing that were unforeseen at the initial phase of the project. The developer needs to know ‘why’ a product feature was requested in order to implement it in a useful way.

As mentioned above, customers must make available all people with knowledge of the project. The customer may be reluctant to do this for various reasons, including time management, issues related to the use of an outside firm in favor of inside engineers, and the like. However, the success of the project may hinge on a piece of information known only to one person. That person must be present during initial project meetings.

Another 'withholding' of information can occur when the customer company is actually developing a product for yet another end customer. A concrete example of this is when a subcontractor to a prime government contractor brings in an outside consulting firm to help out with part of a large contract. The natural reaction of the subcontractor is to limit any contact between the outside consulting firm and the prime contractor, who is the real 'end-customer' of the customer firm. However, if possible, the consulting firm should have access to the representatives of the prime contractor, in order to prevent any important details from being 'lost in translation' when communicated via the subcontractor.

Specification

What to do

During the Specification Phase, the consulting firm generates a project specification.

There are a number of important goals for the specification. First is to provide a clear definition of what the consulting firm will be delivering to the customer. The customer eventually needs to 'sign-off' on the specification. This will prevent misunderstandings down the road. Secondly, by providing a clear definition of the deliverables, all parties who are in some way dependent on the result will be able to perform their own work with confidence and in parallel. For example, if a piece of hardware is being developed that interfaces with other parts of a system being developed, then the interfaces will be clearly defined so that development of the rest of the system can proceed. Thirdly, the specification is used as a guide during the implementation phase. The consulting firm engineers will need to refer to the specification on a continuing basis to insure they are implementing what was required. It is really too much to expect that this information can be accurately 'remembered' in the heads of all participants.

Of key importance is the format and level of detail in this specification. Not enough detail will leave open too many unanswered questions so that the end result may differ from that required. Too much detail may 'turn-off' the customer from providing a thorough review, require a large amount of consulting company and customer company time to produce the specification, and lastly, many of the details will prove to be incorrect when the project implementation and testing gets underway. So, the level of detail and format of the specification needs to be appropriate to the project.

At a minimum, the specification must clearly define all customer visible aspects of the project. If there is a user interface, it needs to be defined. If there are hardware interfaces to external components, these need to be spelled out. If there is a software interface (a.k.a. API or Application Programming Interface) it needs to be clearly defined.

A Functional Description of the result must also be provided. This plus the external interfaces really provides the core of the 'What' the customer should be expecting the result to be.

Details regarding the 'How' of the project; i.e. how the implementation will provide the required functionality and interfaces should only be described to the extent that this has an effect on the customer, or provides some valuable insight into the result. Some examples of information that should be provided are description of software tools that may be used during development, because these may need to be compatible with the customer's own development system so they can maintain the project. Or, for a hardware project, any 'special' components in use should be listed, including single source components, as these will have an effect on the customer's ability to produce the end product.

What not to do

Although it may seem counterintuitive, providing too many details in a project specification may hinder it's effectiveness, for the reasons described above. So, an important 'do-not' do item is - do not provide too much detail or otherwise irrelevant information. Keep in mind the task at hand, which is to come to agreement with all interested parties on what will be delivered as opposed to how it will be implemented. When writing the specification, always keep the user of the deliverable item in mind.

Another important issue is, do not proceed to the next phase of the project, project planning and implementation, without getting customer approval and agreement on the

specification. During the ensuing work, the specification is very likely to change. Changes may involve additional engineering work, beyond that initially envisioned. It is important for all parties involved that the effects of changes on schedule and budget are accounted for.

Therefore, a baseline is required as a point of comparison. The well known 'feature-creep' phenomenon in many software projects can result in the ultimate failure of the project. A project can fail due to the attempt to implement features that are of negligible benefit but were suggested as something 'nice to have'. So, in summary, do not proceed without an approved specification. Doing so will prove to be a problem down the road.

Project Planning

What to do

With a firm specification in hand, it is possible to determine with some range of accuracy the work entailed to create the deliverable.

The first step is to break up the deliverable into sub-deliverables or sub-projects. This step may entail some degree of 'high level' design, which usually involves the identification of a number of design options, identification of trade offs between the options, and finally selection of an overall design.

Risk factors in the sub projects must also be identified. There could be the risk that the anticipated sub project cannot be made to work due to technical factors. There could be vendor risk, for example, if a purchased component may not be available when needed. There is the risk that engineers are not available to work on the project for various reasons. All risks need to be identified and estimated. Once they are identified, the project can proceed or perhaps change direction if the risks are perceived as too great.

In order to create a project plan, the degree of inter-dependence between the sub projects must also be determined. To the extent that sub projects can be made independent, opportunities for parallelization in development will materialize. This will enable the option to add additional engineers in order to complete the project at an earlier date than performing the sub projects serially.

Another important type of interdependence is that for equipment or receipt of outside deliverables. A common example is an embedded software development project that is dependent on a piece of custom hardware being developed by the customer company engineers. This dependency will greatly impact the software delivery date. It is an extremely important, and often overlooked issue. In some cases, this dependency will warrant the development of a simulation system in order to enable software development to proceed in parallel with custom hardware development.

Finally, a project plan can be developed and approved. Various software tools are available for this effort, however, depending on the scope of the project, use of such a tool may not be necessary. However, it is important to put together some kind of plan on paper, in order to come up with a scheduled delivery date. It may be as simple as listing the project steps or sub-projects, time estimates, and end dates of each.

What not to do

Do not overlook man power loading. Many projects plans make the mistake of allocating engineers to multiple tasks or sub projects in parallel. Sometimes this can happen when an engineer has to be pulled off of one project to handle an emergency on another. If this possibility has a reasonable chance of occurring, the engineer should not be assigned to the new project at 100% of his or her time.

Do not delay the execution of sub projects that have significant risk. It is best to work out the 'question marks' in the project earlier rather than later. Plan to do these sub projects early since they may have significant impact in the overall project plan.

Do not underestimate man hours or time to complete sub projects. A conservative rule of thumb would be to make your best estimate and perhaps double it. It is better to come in ahead of schedule than behind. Of course, there is a downside to overestimating. The customer may decide that if the project will

take too long, it is not worth doing at all. Or, if the schedule is not aggressive enough, the engineers on the project may be less aggressive in striving to meet deadlines and the project can take even longer than the 'loose' schedule indicates.

Do not 'dictate' sub task time estimates to engineers performing the work. Rather, request the buy in of the effected engineers by getting their estimates. Often, the engineer performing the work is the only one who can come up with an accurate estimate due to his or her specialized knowledge. It is also vital to have the engineer's commitment and it is more likely to occur when the engineer has made the estimate in the first place.

Implementation

What to do

With a firm specification and project plan in hand, the implementation phase can proceed.

We have found that one of the best ways to shorten project duration and improve the likelihood of success is to follow a methodology that stresses fast turnaround of design, test, and demonstration. That is, if a sub-project is planned to require two man-months with a single engineer on the sub-project, the implementation should proceed in a way that allows for demonstrable results on a weekly or bi-weekly basis.

For example, if the sub-project is a software project, the code should be broken down into modules such that the individual modules can be coded, tested, and demonstrated in days rather than weeks. Similar issues exist for hardware projects although full testing and demonstration may not be possible. However, even with a hardware project, demonstration of simulation results of sub sections of the total design can be accomplished.

The order of implementation is also critical. For example, if the sub-project involves a user interface and the underlying functionality, the user interface code should be developed first, and demonstrated to the end user. Even with the underlying functional code incomplete, or simply 'stubbed out', the availability of the user interface for the end user to 'play with' will provide important feedback that will impact the ensuing work. Very often, the customer will request a change that will have a large ripple effect on the design. The sooner this change is determined, the less implementation time and effort will be wasted.

Short weekly or bi-weekly status reports are a useful tool. These should take no more than 15 minutes of an engineer's time to prepare. In order to prepare them, engineers on a contract or consulting project should keep activity logs. These can take the form of notes in an engineering notebook or in a word processor document. There are a number of reasons for doing this. One is to have something to refer back to when questions arise regarding technical issues. Another is to have a way to keep track of how long a task took in order to compare it with the original estimates, which may result in other time estimates being adjusted. Finally, for billing purposes, the customer may require time logs.

Finally, as the project proceeds it is inevitable that the original estimates and plans will need to be adjusted. These adjustments to the estimates and schedule should be made on an on-going basis, perhaps weekly, and reported as part of the weekly status report to the customer.

What not to do

One of the most important things not to do is to carry out the full implementation of a sub-project without intermediate testing and demonstration. As noted above, specification level changes, if made near the end of the scheduled work period for a task, can, in the worst case, render all of the work done to date useless. We have found this to be one of the most common causes of schedule slip.

A second common scenario resulting in failure is leaving the 'hard part' to the end. In other words, if the risky or technically difficult part of a sub-project is left until the end, and then it is determined that it can't be made to work, or perhaps will require a lot more effort than originally anticipated, the project success will be in jeopardy and completed work may need to be thrown out, wasting valuable time and effort.

Finally, be sure that schedule updates and project status is communicated on an on-going basis. Surprises are perhaps one of the worst things that can happen in the consultant – customer relationship. Even if the news is bad, it is better to know it as soon as possible. On the other hand, it is often a mistake for the consultant to provide overly optimistic news. For example, it is good for the consultant to be able to report a task is complete, but not so good to report it is 'just about' done, and then have to take that statement back when the project is not completed as anticipated.

Final Test and Delivery

What to do

If the prior process steps have been followed, Test and Delivery should be little more than a formality. That is, the customer should have already seen just about everything that has been accomplished.

If possible, a demonstration of the developed item should be given to the customer. In preparation for this step, the consultant should prepare an outline of the demonstration and do a run through prior to the customer meeting. This should be done well in advance in case last minute changes are discovered and need to be implemented. A review of the design specification at this time is in order. The demonstration should proceed in a way that proves that the elements of the specification have been implemented.

During final test and delivery, all design documentation, test documentation, software source code, etc. should be formally handed off to the customer. Typically this would involve the delivery of cd's containing all documentation, along with an inventory, or list of delivered items.

If required, a formal acceptance document for sign off can be prepared for the customer to sign and authorize. In some cases this may be a prerequisite for consultant payment. If a formal acceptance document is not going to be created, then at the very least some email communication should occur that indicates the customer's acceptance of the delivery.

What not to do

Do not skip the final test and delivery step. Even if it is a mere formality, it serves as a critical juncture marking the end of the project. This is particularly important in fixed bid contracts.

Do not wait until the 'official' demonstration to find out that there is a problem. We usually try to give a run through of the demonstration to one or two of the customer engineers or managers in advance of the final demonstration. This will lower the chances that a problem is found at the final hand-off point.

In general, there are several constituents in the customer firm. Typically this would include the customer engineers and project leader, whom the consultant has had weekly or even daily contact, and the customer managers. The customer engineers and consulting firm should be working together as a team to deliver the project that will be approved by the customer managers. Therefore, the customer engineers and consultant need to be 'on the same page' when presenting the demonstration to the customer managers, and so should work together in advance of the final demonstration.

Maintenance and Support

What to do

It is usually a win-win situation for the customer to engage the consultant again when the developed product needs modification or bug fixing. It is good for the customer because the consultant has the best knowledge of the product details and will thus be the fastest, most efficient option for performing additional work. It is good for the consultant because it provides additional work. It is good for both because it solidifies a hopefully productive, on-going work relationship.

However, in the interest of being able to support the product in-house, the customer may prefer to develop in-house expertise. This objective may be best served by having an in-house engineer work in

tandem with the consultant. Often this can be accomplished by assigning and in-house engineer to do the work, but have the consultant available for phone or in-person assistance when needed.

The consultant can foster the on-going relationship with periodic follow up.

What not to do

The customer should not be reluctant to contact and possibly engage the consultant for on-going support and maintenance. As stated above, the consultant may be the most efficient option. The customer should also realize that most consultants and contractors are very ethical and need happy and satisfied customers to make a living. Therefore, often when a problem arises, the consultant may be willing to fix it at no extra charge in order to solidify the relationship, and ensure the customer would provide a good reference to other customers.

The Consultant should not view the end of the Test and Delivery phase as the end of the customer relationship. Existing customers are the best source of new work for consultants and contractors. Therefore the consultant should do whatever it can to ensure the delivered product continues to meet expectations.

Summary

The 5-Step Development process described above provides a framework for customer companies to execute engineering projects with the use of an outside contractor or consultant. The process does not require a great deal of overhead or extra work and can greatly contribute to successful projects. We welcome any questions or comments.

# # #