



Customized software for you

## Custom Software Development - Laws Observed, Lessons Learned

There are several datums that appear to be important enough to the success of software development projects that they need to be communicated. I have endeavoured to arrange the list in a logical sequence.

1. TANSTAAFL There Ain't No Such Thing As A Free Lunch. Despite any and all appearances to the contrary, every benefit has a cost. It can be a direct or an indirect cost, an obvious or hidden cost, an up front or deferred cost, but every benefit has a cost. Somebody has to pick up the tab. If the diner doesn't pay the bill, the restaurant owner wears the cost either as a promotional expense or as a bad debt.
2. One man's meat is another man's poison. What is acceptable or desirable to one person is unacceptable or undesirable to another. In software development there are no limits as to what a person can prefer or dislike. This applies to colours, presentation format, functionality, menu layout and naming, sequence of actions, sequence of fields in a screen form, font size, pricing structure etc.
3. Without a single exception to date, every prospective client has come to us with a tall wish list and a short budget. Every single one want more functionality than their budget will allow.
4. Most clients want the software delivered as soon as possible. This item and the preceding one can combine into the occasional "Perfect Storm" of a client, a person who wants everything, yesterday for nothing.
5. I have yet to meet any client who can fully visualise their perfect software in minute detail then describe it completely and flawlessly to me so I can accurately create their vision. Most people are far better at correcting or improving a draft or existing product rather than designing a product from a blank sheet of paper.
6. Unless each and every aspect of the complete product is specifically detailed in writing, the client will assume the item to be included and will expect it, the programmer will assume it to be excluded and not deliver it. Refer item 2.
7. Both by virtue of the three previous items and due to the complexity of software coding it is both very difficult and very time consuming to accurately estimate the time it will take to create software prior to its completion. The best technique we have been able to find in an attempt to provide the answer to the question "How long will it take and how much will it cost?" is a three point estimate: fastest, most likely and worst case scenario. These are provided free as a guess.

Tel 02 9552 3311  
Fax 02 9566 2164

Po Box 470  
Glebe NSW 2037

tomgrim@just4usoftware.com.au  
www.just4usoftware.com.au

ABN: 71 971 940 807 195



Customized software for you

Detailed research to provide a more reliable and narrower range is still provided on an all care, no responsibility basis due to the almost limitless number of factors that can impact on a project.

8. According to research, the waterfall method of software development (where you specify up front the complete functionality of the software, then build it, then deliver it) has a "complete and utter failure" rate of 50%. Complete and utter failure is not just where it is over budget or takes longer, it is where the software is either never completed, is completed but never used or it is abandoned as unusable shortly after installation.
9. Software development is most often orders of magnitude more complex than either programmers or clients believe it is or should be. This is partly because a computer is a complete and utter moron. It is like the worst possible employee, someone who does specifically, only and exactly what you tell them to do. Humans can employ intelligence, judgement and discretion, allow for exceptions and formulate policy on the run to deal with unforeseen instances. Most non-programmers do not understand the necessity for a programmer to code for every possible eventuality.
10. No programmer writes perfect software code the first time. (Nobody writes a publishable book first time either. I'm told books go through various stages of editors, spell checkers etc. about 6 quality review processes in all. I read recently that even then there is hardly an author who does not find three errors in the first printing.)
11. There is no such thing as perfect software. There is always another defect that can be found and corrected. Software code has to comprise perfect logic and perfect expression of the logic to work perfectly. A single character is all it takes to create a defect. (A book reader can mentally insert a missing comma to have a sentence make sense. Software cannot.) Billions were wasted when a space probe went in the wrong direction because a programmer typed a full stop instead of a comma and it was never picked up.
12. There is no such thing as finished software. There is always more functionality that can be added.
13. The earlier in the software development cycle you detect a design, logic or coding flaw the cheaper it is to remedy.
14. A dumb user always beats a smart programmer. Any user, randomly performing functions in the software program will be able to create an error not checked for by a programmer, no matter how smart the programmer.

Tel 02 9552 3311  
Fax 02 9566 2164

Po Box 470  
Glebe NSW 2037

tomgrim@just4usoftware.com.au  
www.just4usoftware.com.au

ABN: 71 971 940 807 195



Customized software for you

15. Writing the code to check for and trap all possible errors a user can make can take five to fifty times as long as writing the code to create the original feature. (This specification of this level of robustness can make custom software prohibitively expensive for people who are not running nuclear reactors or launching space flights and do not need that degree of perfection.)

16. Over the years we have unfortunately had a couple of "I win, you lose" clients. These will endeavour to negotiate a fixed price contract and then try to get as much functionality out of the programmer, (with zero regard for the programmer's profitability or survival) regardless of whether it was specified up front. The "win-win" client will be prepared to take responsibility for paying for those features that are important and leaving undeveloped the trivial ones.

17. The nature of the client who commissions custom software tends to be more towards the perfectionist end of the scale rather than the "she'll be right mate" end of the scale.

18. Most clients resent software defects. Clients resent having to pay the programmer to rectify defects. (They will drive over or around potholes for years without ever thinking to ring the council about them. They will tolerate dropped calls from the mobile phone provider, typographical errors in written articles and street lights being out when they should be on much more easily than they will tolerate software defects. I haven't yet figured out why. Maybe it has something to do with the relatively high cost of custom development or the fact that it is functionality for which they are paying directly. But these same people, when asked, do not direct their salaried staff to work for nothing to correct an error where they will ask a programmer to do so.)

19. Custom software development is a service, not a product. The client is paying for the time it takes to create software to their specification; they are not purchasing a completed product that comes warranted to this or that standard stated by the manufacturer. The client determines the standard by what level of robustness to which they ask the programmer to program in the specification.

20. Given that a programmer uses an accurate basis for costing the service they provide, the two things most certain of driving up a quote and development costs are a fixed price contract and a warranty.

A fixed price contract has as its other side a fixed list of features to be delivered for the agreed price. This forces a complete design up front wherein every feature to be delivered has to be specified at the outset. This categorises the development methodology as waterfall and, as per item 8 above, therefore instantly reduces the project success likelihood to 50%. A 50% chance of the project failing is a very high price to pay for a fixed price quote.

Tel 02 9552 3311  
Fax 02 9566 2164

Po Box 470  
Glebe NSW 2037

tomgrim@just4usoftware.com.au  
www.just4usoftware.com.au

ABN: 71 971 940 807 195



Customized software for you

It also means and any and all modifications have to be documented, checked against the original specs, quoted on and a contract variation approved. This adds an enormous administrative overhead to a project.

A specific example comes to mind. One organisation came to us having been quoted \$350,000 for a project. The development organisation was prepared to give a fixed price quote and had a much larger process overhead. Maybe that's why their quote was more than double the price for which we are budgeted to complete the job.

A warranty necessitates the inclusion within the quote of an allowance for more testing than might otherwise be the case and the inclusion of an amount to cover the cost of providing post delivery defect remediation. For a product sold to a number of clients this warranty cost can be amortised or spread over all the items sold. For custom software provided to one client, maximum allowance has to be made for time that may or may not be used by that one client and the client billed accordingly.

21. Normally the client wants the product as soon as possible. To ensure that the completed product will most closely match the client's expectations, the programmer needs continual feedback from the client. In order to satisfy both of these requirements the programmer gives the client unfinished software to play with. This software has missing features and untested functionality but is the most effective way to give the client the opportunity to "test drive" the software under development.

22. Software development takes as long as it takes, no more no less. It does not take as long as we predict, as long as the client wishes or the client is prepared to pay for. It takes as long as it takes until the client says, "That will do, I'm happy enough with that!" or "Stop developing, we've reached the end of the budget." That point is, as it should be, 100% at the determination and discretion of the client. So should the cost of obtaining it.

Because I have always created software for smaller businesses, including my own, from day one I have had the following objectives:

Minimise process overhead and administration so as to

Maximise the amount of functionality for the dollars spent.

Minimise the possibility for disagreement and conflict between client and programmer.

Ensure the both the client and I share responsibility for the success of the project and our mutual profitability by discouraging the client from commissioning functionality that will be of insufficient benefit for the cost of developing it, working rapidly and diligently and having the client pay for the functionality that they commission so I do not work for peanuts or nothing.

Tel 02 9552 3311  
Fax 02 9566 2164

Po Box 470  
Glebe NSW 2037

tomgrim@just4usoftware.com.au  
www.just4usoftware.com.au

ABN: 71 971 940 807 195