# Why Flexible Development for Custom Software Projects

The Flexible Development approach to custom software is a proven method which enables us to deliver our expertise in a manner tried and tested in over 30 years of delivering projects.

Flexible Development is our own unique delivery method and describes a practical blend of waterfall and agile software development.

This paper by Ballard Chalmers' Managing Director, Andrew Chalmers, provides a detailed explanation for why Flexible Development is the right approach for custom software development projects.

Evolution into Flexible Development
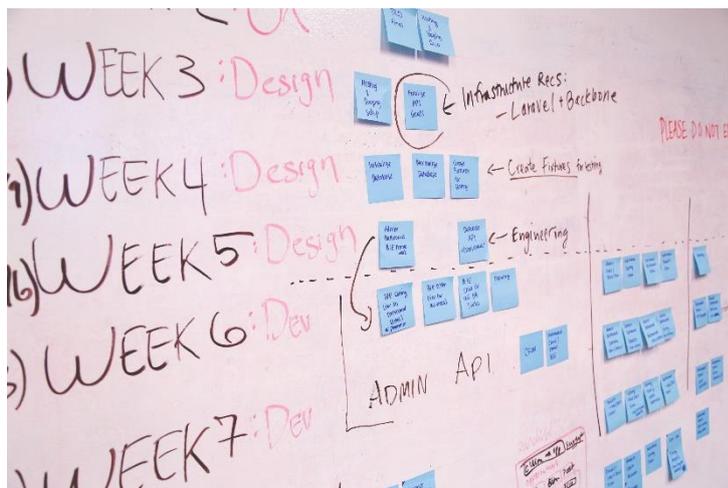
In the early days of web-based software development, along with most of our peers, we usually worked to a traditional 'waterfall' software development process and fixed-price project delivery. In waterfall methodology, you moved from one stage of the process onwards, completing each one fully before starting the next. You undertook the system design fully before starting development, completed development in full before testing, and delivered the near-complete system to the client at the end of the testing phase. The idea was that you could control the scope and thus the costs by not letting it change at all along the way, or at least following a very strict process if you did.

This was theoretically sound. However, in practice it never really worked. This is probably the reason behind much of the bad reputation large software projects have today. In reality, you could not design a system in enough detail before starting development to be sure you had it completely right first time. What's more, you could not be sure it was really what the client wanted before they could see it working in detailed actuality on a computer screen and not just on paper.

Then, if you did try to get the system design and specification totally complete before undertaking the development, you regularly found yourself '[painting the Forth Bridge](#)' because once you had completed the specification with the client, so much time had passed that the business itself had changed and you had to go back to the beginning and rewrite all the business processes again. We found ourselves on several occasions having spent a year writing a system specification with a client, only to find at the end of it that it was no longer relevant.

This clearly was not practical for either the client or the supplier, and over the last two decades or so 'Agile' development methodologies have gained prominence and are now pretty well universally accepted as the way to go for custom-developed software projects.

Given that even industry-specific 'off-the-shelf' business software written precisely for a particular business sector or type typically still requires a lot of customisation and integration to work for the particular client, then it is clear that custom software development will remain central to successful business software systems for the foreseeable future.

Pure agile development was perhaps a swing too far the other way for commercial considerations; though done correctly, I am sure, experts would be confident to deliver on time and in budget using proper agile methodologies. My personal view is that agile works very well for internal development teams where contracts and budgets are not so strictly delineated at the outset of the project, and, because an external consultancy is not being used, the cost-per-day can be lower; so, you have a lot more time potentially for the same budget. Though the down side can be that the urgency factor required to achieve time and cost targets can potentially be missing, so you do not save money in the end anyway.

With a fixed-price project you often found yourself in a potential confrontation scenario, with the client demanding as much functionality as possible for the agreed budget, and the supplier trying to stick to what he understood the scope that was agreed in the first place, to stay in budget!

As someone wise in years of software development projects once said to me, "In a fixed-price project it is the clients' intention to get as much work done as possible for the budget, and the suppliers intention to do as little as possible for the same budget, and they will fight over that dichotomy to the end."

And as a final note, with agile development, in reality it was very hard to make work as a remote supplier on a contract and get the job completed; not usually because of any failing on our side, but because it was so hard to get enough of the client staff's time and input needed to finalise the system design and specification as the development moved forward, and to get the consensus within the business, within the sprint times needed to ensure the development process was not delayed or slowed up.

Bridging the Gap with the Flexible Development Process

Out of these experiences we started to evolve our own methodology as a practical blend of waterfall and agile development, that somewhere along the line I labelled as 'flexible development', which seems to have stuck. The process is typically three-phased, after each one the project can either progress or not:

1. An initial outline scope and high-level budget for a project, which will usually only take a few hours of consultant time to complete, depending on the level of client documentation and information available at that stage. This is normally free of charge.

2. If that initial scope is acceptable and indicates the project can be delivered in the budget and timeframes needed, we then undertake a more detailed initial scoping, working together with the client, which, depending on the size of the project, is usually just a few days of consultancy. This delivers a reasonably detailed scope of work, and a budget estimate with a % accuracy estimate.

    For example, this project will cost in the region of £50,000 with an accuracy estimate at this stage (depending on the level of detail of the scope and design), of for example 20%. So, in this example we are saying it will cost between £40-60k and will take x weeks or months to complete.

    The delivery schedule is dependent on the size of the development team feasible to assign to the project. That is, it is usually not the case that the more people you put on the development the faster it will be done; there is typically an ideal number to ensure the most productive progress is achieved (for those of us of an older generation, "The Mythical Man-Month" book is our mentor!).

3. If that budget and time estimate is now still within the original expectations, or is otherwise acceptable, we can move into the next phase of System Design. The design phase is intended to validate the initial scoping and detail that fully enough to move into system coding proper. This phase typically will take several weeks, depending on the size of the project. And on smaller jobs, it will not be required.

This still does not provide for a fixed-price, but it does give the budget parameters, and a clear change-control process for the client and developers to work within to stay in budget. That is, if it turns out the scope is incomplete in practical terms during the process of development, then the budget can be increased, or the scope can be decreased to maintain budget.

Also, it is clear the budget must remain estimated: None of us (neither client nor supplier) can guarantee to have covered every aspect of the requirement in this type of scoping, and it is understood that we are going to uncover things we did not think of as we progress. It does sound simple really, and it is in practice!

With Flexible Development we are guaranteeing our expertise, honesty, integrity, and partnership to work together to deliver what is needed and wanted to the best of our abilities, and with the client fulfilling their obligations to do the same. Working together on this basis we will all do our best to get through the process of developing the custom business software needed, or integrating the systems, whatever the adventure we have undertaken together may be.

The Software Development Phase

The development phase is itself semi-agile...flexible! If we have not done wire frames (outline mock ups of user screens) already in the design phase, we will develop these, as well as user interface designs and outline the business processes to provide the next level of design detail and a proof of concept. These are done to confirm that we are on the right track.

With that signed off we can start development in earnest. Depending on the size of the project we will deliver several releases of the system to the client in agile 'sprint' fashion, demonstrate the system as it moves forward, and review and take feedback as we progress. Out of that often comes Change Requests, which are estimated for time and cost and then either approved and added to the project or rejected and omitted. These can include functionality requests to be added by the client, or functionality the developers have seen is required but not included, or where they have found the original assumptions were incorrect and additional time (and cost) will be required to meet the objectives.

In this process our guiding concept is 'Prediction' — knowing and alerting what is expected in the future in both time and cost terms. What we don't want is any last-minute surprises for

the client, or ourselves. A part of that is our unique weekly reporting system, which includes an 'uplift factor' that factors in the current progress of the project and extrapolates that out through the rest of the project time and budget. So, if we are currently running at 5% under the time and budget, that 5% will be applied through the remainder of the project estimates and the 5% reduction applied to the final cost. With that you can always track your final expected end cost, even without a fixed-price.

We have an online project management system that we share with clients for project reporting and use Microsoft Excel and Project heavily to track progress and ensure all parties remain informed and comfortable with the project as it moves through development.

Once the main development is completed we have the standard Testing and User Acceptance Testing phases, and closely involve the client in this process. You will often find clients in our offices working together with our project team on testing and acceptance to ensure a top-quality end-result that meets the desired outcome.

Of course, once we have completed a system development we provide ongoing support for that system.

In Closing

Our aim is to provide what is best for our clients for the ongoing secure and robust live deployment of the system that we have developed. We also want to ensure that all the effort put into the development of the system is now rewarded with the business-use that it was intended for, and – most importantly – the resultant return on investment!

We see custom software development as a Partnership between our clients and ourselves. We expect our clients to buy in to the Flexible Development process at the outset of the project, but where it is demanded we can and do provide fixed-price project work. However, the costs, to cover the contingencies and warranties and so on, are in reality always higher than with Flexible Development.

Once a client has experienced this with Ballard Chalmers, I am confident they will be convinced of the benefits.

And we become their long-term partner and trusted expert, ready and available whenever we are needed.

That is certainly our aim anyway; and one thing is certain — we will always try our best to do so.

By Andrew Chalmers, Managing Director at Ballard Chalmers

Andrew Chalmers is a co-founder and the Managing Director of Ballard Chalmers. With over 40 years' experience in business management, Andrew brings the business-side experience to Ballard Chalmers to complement our Chief Technical Officer's technical leadership.

In 1998 Andrew created and established one of the very first cloud software development consultancies in the UK, which developed a very early-adoption corporate intranet for Hewlett Packard Europe amongst many other ground-breaking projects.

---

ballardchalmers
software, engineered

Ballard Chalmers is one of the UK's leading custom software development companies, specialising in engineering enterprise applications on the Microsoft Platform.

We are principally dedicated to custom software development and integration for the Microsoft Cloud, as well as on-premises and hybrid systems. Our expertise is in Microsoft .NET and Azure, SQL Server, BizTalk Server, SharePoint Server and Xamarin. With over 30 years' industry experience and as a Microsoft Gold Partner, we have a well-earned reputation for delivering our expert service in these technologies.

w. ballardchalmers.com
e. enquiries@ballardchalmers.com
t. 01342 410 223

Gold
Microsoft Partner
Microsoft