

Virtual Walkthrus Assist with the ‘Yes...But’ Issue

If you have worked on a software development project then you have been in this position before: your users agree with documented requirements but later change their mind or introduce new requirements when they see the software demonstrated for the first time (**Yes**, that’s what we agreed **But** now that we see it ...).

Why do we have this ‘Yes...But’ issue? Requirement documents are an abstract representation of a software solution and the users’ ‘real’ requirements become much clearer for them when they see a ‘real’ tangible solution. The more tangible the better, it’s just human nature.

The problem is that changing and introducing new requirements once the software is developed creates much pain for everyone involved, particularly on contractual fixed-price projects.

Recently, Equinox’s Bob Mortimer gave an IIBA® presentation on developments in user interface (UI) tools, specifically [Balsamiq Mockup](#) and [Microsoft's Expression Blend 3 with SketchFlow](#). We believe these tools, with the approaches outlined in this article, can be used to create ‘Virtual Walkthrus’ for users to help overcome the ‘Yes...But’ issue. This article outlines the tools and approach and the benefits that can be achieved.

Tools

Balsamiq Mockup allows UI designers to mock-up software UI in minutes, even real-time during requirements elicitation workshops. The mock-ups have a ‘sketched’ feel so that users understand they are not final and are easily changed. Balsamiq Mockup is so fast and easy to use that it can change the way we elicit requirements.

Microsoft Expression Blend 3 is a rich UI design tool for Silverlight and .NET development technologies. The tool is generally used for the development of the UI layer. But with the SketchFlow facility it allows the UI designer to ‘build the system’ without actually ‘building the system’ and so has a massive potential for requirements validation

In IIBA® terms validation is part of the Requirements Analysis knowledge area, and so these tools affect the two core areas of Software Requirements Analysis: Elicitation and Analysis (Validation).

So here’s how the process can work:

1. The UI designer works with analyst during Inception to create small cosmetic prototypes as necessary, to aid high-level requirements elicitation and to create UI guidelines.
2. The UI designer uses Balsamiq Mockup to quickly draw screens during Elaboration use case workshops. In this way the UI designer effectively replaces whiteboard story boarding and creates something more tangible, which is included in the validation package and which users find easier to relate to. As a bonus the analyst is freed up to facilitate the workshop and focus on capture interaction and other requirements.

3. Subsequently, while the analyst undertakes analysis and use case specification, the UI designer turns the mock-ups into animated 'Virtual Walkthrus' of the use cases using Expression Blend.
Note that in order to gain approval and sign off of the full set of software requirements we believe it may be necessary to simulate up to 80% of the specified system. So this is, in more ways than one, not just a cosmetic prototype.
4. The analyst and UI designer present a package of requirements made up of requirements documentation and executable screen flow simulation (in sketch format).
5. After updating the Expression Blend screens with user feedback, the UI Designer can add any finishing touches and use this as the production-ready presentation layer for the developers to wire up business logic and so forth. In other words the additional work to achieve requirements sign-off is not wasted, but rather contributes to the working solution.

Benefits

Resolution of the 'Yes... But' issue – the ability to mock-up and then simulate a system that looks and feels real means there is a greater chance of achieving accuracy with requirements, decreasing the number of change requests and scope creep during construction.

Minimisation of rework due to 'Just in Time' requirements gathering – whether built into the process in an agile manner or due to a change request process, late requirement changes almost always cause re-work and this is expensive.

In fact, early sign-off of requirements is necessary in order to fix price, which is the case in almost every commercial software project. Using the approach outlined here gives greater confidence to both the development team and the business customer that the requirements being signed off for a contractual fixed-price project are accurate. So it's OK to elicit requirements in a waterfall or formal iterative manner.

User buy-in to the requirements activities and requirements documents – buy-in is much higher, with many of those requirements being represented, communicated, and validated in a manner that the user group can more easily interpret.

In essence, we now have the tools that will allow a pragmatic and integrated approach to software requirements and UI design. This can radically improve a project team's ability to produce quality software solutions within the constraints of real world financial models.



Please [contact Craig McLean](#) or [Bob Mortimer](#) with any questions or comments relating to this article.