Optimizing User Experience with Agile Design

By James Hobart, President, Classic System Solutions

April 2002

With all the press and attention recently given to Extreme Programming and Agile Modeling techniques, it seems very little attention has been given to User Interface design techniques which will blend rather than clash with these new development methods being used to create high quality, industrial strength and highly usable applications. As an offshoot to the Agile Modeling techniques, we are proposing teams explore the benefits of Agile Design to quickly establish a high quality user interface.

What is agile design?

We're faced with shorter timelines, more aggressive development goals and limited resources to complete our projects.   Each project requires a UI design strategy geared toward success implementation with the other aspects of the project. Agile design addresses these constraints by limiting the level of documentation to a point which works well for the team based on skill, complexity and proximity of the team participants. Agile Design acknowledges that great user interfaces are an iterative process and require time tested user-centered design techniques as a foundation for success.

Where does the UML fit it in?

We feel use of UML and specifically Use Cases on design projects can help create a common set of notations for documenting UI design decisions. We are particularly impressed with Use Case techniques covered in the book "Writing Effective Use Cases"
Unfortunately, the UML does not go far enough in providing useful documentation techniques for deriving a world-class user interface. Specifically, it lacks UI models we feel are critical when developing robust Web-based applications.   The purpose of these UI models is to allow for extensive recapturing of the UI prior to initial implementation. The models form the basis for the prototype and are driven from a starting point of User Centered Design.

Agile Design is built upon the principles established by a talented team of professionals who wrote a manifesto regarding Agile Development to address the needs of the current development environment we all seem to face. We often attempt to layer usability into a fast moving agile development environment where successive iterations of the code base are commonplace. This approach can backfire if we insist on heavy documentation and human factors studies before getting prototypes in front of users. On the other hand, just putting a few developers in a room with some end users can often result in a product which works for a subset of users on the team, but not a product which truly delivers the benefits of user-centered design. This is because specific design decisions need to validated with actual usability testing to verify how the users behave rather than what they think they may want.

Using models to refine your design

There are some basic models we use to support quick development of prototypes while adhering to the principles of Agile Design and usage centered design. First we develop a User Profile Matrix that allows the team to more clearly understand the unique tasks and characteristics of each user type for the affected system. We then develop a Presentation Model that documents major decisions about the system like platform, colors, fonts, and overall UI architecture. This model clearly demonstrates the key UI elements and often includes several design renderings so that everyone can visually validate the design direction. We then move on to develop specific Navigation Models which are supported by existing Visual Design Patterns and the documented Use Cases for the system under development. To flush out the models we develop Screen Layouts for specific pages of known complexity to validate both the navigation and presentation models. Of course all this modeling needs to be supported by a robust set of design standards and guidelines. This entire process can occur over a period of days or weeks depending on the scope and complexity of the specific design project. During this period we continually refine each of the models and strive for a strong congruency between them.

Why use Agile Design techniques?

Using these techniques with the appropriate skills, roles and levels of documentation can result in highly effective and usable software delivered under shorter timeframes. Your team members must commit to a strong discipline of design iterations highly focused interactive design sessions and fluid team communication for this approach to succeed.

We have been using these techniques on several projects over the past 3 years and have had success with small teams as well as very large enterprise application design efforts. In the end, it is really about achieving a successful design with the resources you are given. Please feel free to email me with questions or concerns you have experienced applying these techniques and I will follow up this article with some more detailed examples of how we implemented Agile Design techniques successfully on our client projects.

About the author:

James Hobart is an internationally recognized user interface design consultant based in California, USA. He specializes in the design and development of large-scale, high-volume client/server and web applications. He is an expert in GUI design for transaction processing systems and strategies for migration to thin-client graphical user interfaces. He can be reached at jimh@classicsys.com

Find more articles on Usability at www.classicsys.com