

# COMBINING INTERACTION DESIGN AND AGILE METHODS FOR IMPROVED PROJECT MANAGEMENT IN INTERNET DEVELOPMENT

Namgyal Damdul, Dundalk Institute of Technology, Software Technology Research Centre  
(STORC), Dublin Road, Dundalk, Ireland. namgyal.damdu@dkit.ie

Frank Keenan, Dundalk Institute of Technology, Software Technology Research Centre  
(STORC), Dublin Road, Dundalk, Ireland. frank.keenan@dkit.ie

## Abstract

*Internet applications are increasingly used in everyday business and life. However, despite their obvious potential, the successful management of large-scale internet projects using traditional software project management practices has proved problematic, as demonstrated by the Irish government's eGovernment initiative. To address these weaknesses Agile Methods (AMs) for software development consider customer commitment and short development cycles as guiding principles. The intention is that close collaboration allows input from people who will actually use the final system and that the short cycles allows this feedback to influence development, thus facilitating project management. However, various difficulties have been reported with these approaches, particularly the customer role. Essentially, there are few guidelines on the implementation of the customer role to suggest how interaction with developers should occur. The research problem is to propose a systematic way of analyzing user groups and their expectations to enhance existing project management approaches within AMs, particularly for the development of internet projects.*

## 1 INTRODUCTION

The Internet has a major influence on almost every part of our lives. Generally, Internet applications are network intensive, content-driven and continuously evolve (Pressman 2000). Internet application development is a “multidisciplinary approach requiring knowledge and expertise from different areas” (Kappel et al 2006). Here, it is difficult to have a “rigid, predefined project plan” as this would not permit a development team to “react flexibly to changing conditions” (Kappel et al 2006). Generally, internet applications are designed and developed to be used by a particular audience. Before beginning development it is best to clearly identify the main user groups and their anticipated or expected business outcomes so that subsequent effort can be more focused and better managed. However, despite widespread internet usage there are many examples of spectacular failure in the development of internet projects.

In Ireland, for example, the recent report of the Comptroller and Auditor General has highlighted the general dismal performance of the Irish government's ambitious eGovernment project (Collins 2008). Of the 161 projects approved by mid-2006, less than half (74) were fully delivered and operational. Also, almost one third (44) were only partly implemented while 23 were abandoned. Costs were 20% more than budgeted and average actual project duration was 25% more than planned. The main recommendations reported here for future internet projects were in the area of project management. Specifically, it called for appropriate use of *agile* development methodologies and the definition of

*target business* benefits in advance of implementation. This however, is currently a weakness with Agile Methods (AMs).

Interaction Design (ID) is the discipline of defining the behavior of products and systems that a user can interact with (Cooper 2004). ID is concerned with making the software easy, effective and enjoyable to use (Sharp 2007). “Usability and user experience” are becoming critical factors in the success of internet applications (Constantine 2004). However, traditionally, and in contrast to agile development, ID recommends comprehensive up-front design by interaction designers before actual construction of the software by the programmers. Also, the focus with ID is on the *user* whereas with agile practitioners priority is given to the *customer*.

The research under investigation here proposes to address the following research question: *can the activities of Interaction Design (ID) be combined with Agile Methods to link the software development process to higher level goals, thereby improving the management of internet projects?*

To achieve this following objectives have been identified:

- To identify and understand the tools and activities of Interaction Design and how they are used in the industry.
- To identify and understand agile software development methods, particularly the Scrum process.
- To investigate and develop a combined ID-Scrum process to improve the requirements management.
- To test and evaluate the ID-Scrum process in an appropriate project.

## **2 AGILE DEVELOPMENT**

Traditional *plan-driven* approaches to software development organise the main activities of *analysis, design, implementation* and *testing* in a linear manner. That is, when one activity completes for the whole project effort is then concentrated on the next with little opportunity for iteration. This clear distinction between activities, work products and roles facilitates project management. However, many problems have been reported, which has led to the recent emergence of the agile approaches – the so-called Agile Methods (AMs). Here, software is developed in short iterations throughout the project, providing an opportunity for the customer and users to evaluate a working product providing feedback, which, generally, leads to further changes to the software. Each iteration involves analysis, design, implementation and testing. AMs do not expect static requirements; rather it considers that requirements evolve during the process. Enough detail is elicited at the start of the project to allow overall project planning. Generally, AMs advocate close cooperation between interested stakeholders including business people, software developers, customer and users at the different stages of the process, and face to face communication and interaction among the development teams are preferred (Jeffries 2001). In particular, they discourage comprehensive up-front analysis and design, as the design changes are considered inevitable during the process.

### **2.1 Agile Project Management**

Traditionally in plan-driven software development requirements are documented in “dry, formal language” that describes the software system and acts as both means of communication and data storage. However, comprehensive documentation of requirements is not always desirable because the

documents are selective, unidirectional and may be ambiguous and vague (Davies 2006). In Agile software development teams “expect change, accepting that requirements will evolve throughout the project” (Ambler 2007). Extending the challenge with Web development, in particular, is more complex due to need for considering multimedia and marketing aspects in the development of requirements (Escalona and Koch 2004). Moreover, in these projects, “clients did not well understand the capabilities of the technologies” and their “own needs as they related to technology” (Lowe and Eklund 2002). It is recommended that for web development design activities and requirements elicitation should be handled and managed simultaneously.

Generally, with AMs requirements are recorded in brief format as, for example, *user stories*. A user story is a non-technical and “high-level definition of requirements, containing just enough information so that the developers can produce a reasonable estimate of the effort to implement it.” (Ambler 2008). The main activities of project management are applied to these stories in consultation with the customer. Detailed project management takes place within each iteration when stories are examined in detail. Project management and requirements development evolve together.

Scrum is an overarching process for *planning* and *managing* software development projects (Martin 2004). It is a team based agile process to develop software systems iteratively and incrementally, and manage and control development work (Schwaber 2008). Work is prioritised on the basis of business value, effectiveness, increasing profit, customer needs. Integrating changing requirements is an important feature of Scrum (Sutherland and Schwaber 2007). Frequently, in practice, it is necessary to combine Scrum with other approaches that focus on programming and individual team practices, such as eXtreme Programming (XP) and FDD (Martin 2004, Kerbs 2005 and Kniberg 2007). However, Martin (Martin 2004) warns that as development in Scrum is very fast; *tight control* and *feedback* are required to keep that speed from creating quality problems. Additional practices need to be added to achieve this. In (Kerbs 2005), for example, various artifacts and activities are added to supplement Scrum.

In Scrum requirements are organized in a Product Backlog (PG), which is, essentially, a list of features requested including both functional and non-functional requirements. Anyone can add items to the list but it is prioritized only by the product owner. The items which have highest priority in PG are broken down into small chunks so that they can be estimated and testable. Frequently, user stories are used to record requirements (Schwaber 2008). For each iteration or sprint a subset of PG features are implemented. A Sprint Backlog is a list of tasks that must be completed to realize the sprint’s goal. Within *Sprint Goals* are specific and measurable goals derived from the negotiation between the PO and ST.

Various roles are recommended in Scrum. The Product Owner is responsible for defining the features of the product, deciding on release date and content, prioritizing the features in accordance with market value. A Scrum Master plays the role of supporting, coaching and guiding the team through the process and is also responsible for removing obstruction faced during the process (Waters 2007). The Scrum Team, of seven or so developers, has the right to function in any manner it likes to achieve the *sprint goal* and work result specified by them (Sutherland and Schwaber 2007).

The Scrum Process is carried out in three main phases *Sprint Planning*, *Sprint Review*, and the *Daily Scrum Meeting* (Scrum Alliance 2008). A Sprint Planning Meeting (SPM) SPM begins after the Product Backlog is defined and prioritized to launch for the thirty day sprint with a detailed plan for the iteration developed. A new SPM is held after each Sprint until the whole project is completed. It is the responsibility of the Product Owner to *review the vision*, roadmap, release plan in addition to reviewing the Product Backlog with the Scrum team for estimates for features and decide how much work can be done in a sprint based on team size, time available and team’s productivity. Scrum

master leads the team to break down Product Backlog features into tasks which forms the individual Sprint backlog.

During development the Scrum Master leads the 15 minute long Daily Scrum Meeting (DSM) where team members assess progress by addressing three questions, *what did I do yesterday, what will I do today and what impediments got in my way?* At the Sprint conclusion a Sprint Review Meeting is held where the Product Owner demonstrates the potentially shippable code and he determines which Product Backlog items have been completed in the Sprint and discusses with the Scrum team and stakeholders to reprioritize the Product Backlog for the next Sprint.

### **3 INTERACTION DESIGN (ID) PROCESS**

Cooper (Cooper 2004) describes the three main stages in ID as:

- Conceptual design concerns with what is valuable for the user in the first place.
- Behavioral design which concerns with how the elements of the software act and communicate.
- Interface design which concerns with the interface of the program.

Typically, interaction design tools are applied in a project to optimize the design of software. Important interaction design tools include *personas, goals* and *scenarios*. Personas are hypothetical characters which are defined with precisions after detail investigation on the potential users and are the most important design tools if applied with “some sophistication” (Cooper 2004). It is recommended that software is developed for the defined persona and not for actual user. A persona exists to achieve his *goals*, and the goals exist to give meaning to a persona and the most important goal is the personal goal which is to “not feel stupid” when interacting with the system (Cooper 2004). *Scenarios* provide descriptions of how a persona interacts with a system and scenarios are envisaged by Interact Designer who thinks in the way personas would think. Also, scenarios help in eliminating unnecessary tasks by identifying overlap and interaction.

Sharp (Sharp 2007) suggests a more evolutionary approach to ID with four main activities:

1. Identifying needs and establishing requirements for the user experience.
2. Developing alternative designs that meet those requirements.
3. Building interactive versions of the designs so that they can be communicated and assessed.
4. Evaluating what is being built throughout the process and the user experience it offers.

### **4 INTERACTION DESIGN AND AGILE DEVELOPMENT**

Despite certain similar characteristics the two approaches differ. Significantly, interactions designers recommend that substantial design takes place up-front, at the beginning of the project, and that this changes little during development. On the other hand agile practitioners recommend minimal design at the beginning, expecting that this will evolve during development. Also, ID is *user-focused* while agile development is *customer-focused*.

Ambler (Ambler 2007) does however suggest that with minor operational changes, it is possible for Interaction Designers and Agile practitioners to work together. Broad, high-level modeling at the

beginning can help to identify key issues, with the details examined in *Just-in-Time* manner during the development process.

An important issue for collaboration is that since agile development follows flexible and highly collaborative organizational strategy, the designs specified in ID are likely to be changed by software developers (Ambler 2007). ID is holistic in nature involving specialized skills while Agile is incremental involving a team of generalists.

There has been few works on combining aspects of ID with agile development. Patton (Patton 2002) described an ID-XP combined process by defining the user roles and writing their detailed tasks based on the interaction design contexts and creating an abstract prototype with the help of the user tasks. However, he does warn that this process “takes long for those who don’t do it often”. Blomkvist (Blomkvist 2002) explained the key aspects of the integration of HCI and agile development highlighting the need for specifying the *usability goals* that can direct the course of the iterative development and usability validation but it is reported that usability methods can be “overly elaborated” and “highly lengthy” in duration. Patton (Patton 2004) stressed the need for clarity of the goals of the users roles and a “common mistake is for tasks to “wander” in goal level – some very high level, some very low”. Constantine (Constantine and Lockwood 2004) discussed about reincarnated agile usage-centered design using card based modeling and decision making and prioritizing roles and their tasks and creating prototypes and highlighted the need for a minimum up-front design. Martin (Martin 2004) used mainly Scrum for managing project and XP for development in a Scrum-XP combined process. XP provided the feedback and tight control over quality in fast pace Scrum development. However, difficulties have also been reported with XP in relation to the customer role and in identifying overall goals.

## 5 WORK IN PROGRESS

Work to progress this research has been made on a number of fronts. A poster titled “Building a XP Metaphor” will be presented this June at XP2008 Conference in Limerick, Ireland. This examined how a group of students managed user stories to develop an internet site. Particular problems that were reported by the members were that *scope* was difficult to manage, the overall *goal* and *direction* of the project were unclear and that it was difficult to provide iteration end checks to show progress.

To overcome these difficulties a combined ID-Scrum process to enhance project management for internet development is being investigated. However, it is important that this is conducted in an agile manner particularly in terms of the time and effort required. The proposed solution uses ID to specify a high level goal and specific goals based on individual user scenarios to provide a good foundation for forming requirement lists, or stories, for the Product Backlog, as this is the crucial initial step of the Scrum process. The proposed solution shall address agile requirements development by having a shared guiding *high-level user goal* throughout the process and *specific user goals* during the individual iterations. To clarify the issues reported in the literature interviews have been arranged with experts in ID and the AMs, particularly Scrum.

The proposed ID-Scrum process for software development will be used in a case study with a group of final year computing students. During the case study, the participants will be observed and then questionnaires will be used to collect their experiences. This will be followed by interviews of the participants to learn what they think about the process.

## 6 CONCLUSION

The Scrum process promotes iterative software development and extensive customer interaction to facilitate project management. ID has been proposed as a candidate approach to be used in combination with Scrum to overcome the reported difficulties in agile development. In doing this it is important to remain consistent with the agile principles; any process used should not cause excess effort or delay. ID is used to support the development of the product backlog of Scrum. This helps Scrum teams to decide on the tasks for implementation based on the user goals. Sprint goals, targets for individual sprints, are derived from this and used to assist project management. Trials are being arranged to assess this approach.

### References

- Ambler, S. (2008), "User Stories"[Online], Available:  
<http://www.agilemodeling.com/artifacts/userStory.htm> [accessed 12th May, 2008].
- Ambler, S. W. (2007), "User Experience Activities on Agile Development" [Online],  
<http://www.agilemodeling.com/essays/agileUsability.htm> [accessed 10th October, 2007].
- Blomkvist, S. (2006), "User-Centred Design and Agile Development of IT Systems" [Online],  
Available: <http://www.it.uu.se/research/publications/lic/2006-012/> [accessed 13th November, 2007].
- Collins, J. (2008) "Failure of e-government" The Irish Times [Online], 11 January, available from:  
<http://www.ireland.com/newspaper/finance/2008/0111/1199917844950.html> [accessed 20 February, 2008].
- Constantine, L. and Lockwood L. (2004), "Usage-Centered Engineering for Web Applications",  
Available: <http://www.foruse.com/articles/webapplications.pdf> [accessed 7th May, 2008].
- Constantine, L. (2006), "Activity Modeling: Toward a Pragmatic Integration of Activity Theory with Usage-Centered Design", Available: <http://www.foruse.com/articles/activitymodeling.pdf> [accessed 30th May, 2008].
- Cooper, A. (2004), "The Inmates Are Running the Asylum", Sams Publishing, Indiana.
- Davies, Rachel. (2005), "Agile Requirements" [Online], Available:  
<http://www.methodsandtools.com/archive/archive.php?id=27> [accessed 13th November, 2007].
- Escalona, M.J, Koch N, (2004), "Requirements Engineering for Web Applications – A Comparative Study", <http://www.pst.informatik.uni-muenchen.de/~kochen/KochEscalonaJWE-rev.pdf> [accessed 20th Feb, 2008].
- Jeffries, R. (2001), "What is Extreme Programming?" [Online], available from:  
<http://www.xprogramming.com/xpmag/whatisxp.htm> [accessed 30th September, 2007].
- Kerbs, J. (2005), "RUP in the dialogue with Scrum" [Online], Available: <http://www128.ibm.com/developerworks/rational/library/feb05/krebs/> [accessed 27th March, 2008].
- Kniberg, H. (2007), "Scrum and XP from Trenches" [Online], Available:  
<http://www.crisp.se/henrik.kniberg/ScrumAndXpFromTheTrenches.pdf> [accessed 27th March, 2008].
- Lowe, D. B, Eklund, J (2002), "Client needs and the Design Process in Web Projects"  
<http://services.eng.uts.edu.au/~dbl/archive/2003-Low03i.pdf> [accessed 20th Feb, 2008].
- Pressman R. S, (2000), "Software Engineering, A Practitioners Approach". Berkshire, England: McGraw-Hill Publishing Company.
- Kappel. G, Proll B et al (2006), "Web Engineering", John Wiley & Sons Ltd, Germany.
- Martin, R. (2004), Object Mentor Inc. "Best Practices in Scrum Project Management and XP Agile Software Development" [Online], Available:

- <http://www.controlchaos.com/download/Primavera%20White%20Paper.pdf> [accessed 14th December, 2007].
- Patton, J. (2002), "Hitting the Target: Adding Interaction Design to Agile Software Development" [Online], Available: <http://oopsla.acm.org/extra/pracreports/HittingTheTargeReport.pdf> [accessed 13th November, 2007].
- Patton, J. (2004), "Interaction Design Meets Agility: Practicing Usage-Centered Design in an Agile Software Development Environment" [Online], Available: <http://www.agilealliance.com/system/article/file/1368/file.pdf> [accessed 13th November, 2007].
- Sharp, H. (2007), "Interaction Design: Beyond Human-Computer Interaction", John Wiley & Sons, Ltd, England.
- Schwaber, K. (2008), "What is Scrum?" Available: <http://www.controlchaos.com/about/> [accessed 27th March, 2008].
- Scrum Alliance, (2008), "Scrum Ceremonies" Available: [http://www.scrumalliance.org/view/scrum\\_ceremonies](http://www.scrumalliance.org/view/scrum_ceremonies) [accessed 2nd April, 2008]
- Sutherland, J., Schwaber, K. (2007), "The Scrum Papers: Nuts, Bolts, and Origins of an Agile Process", Available: <http://jeffsutherland.com/scrum/ScrumPapers.pdf> [accessed 27th March, 2008].
- Waters, K. (2007), "How to implement scrum in 10 easy steps" Available: [http://kw-agiledevelopment.blogspot.com/2007/09/how-to-implement-scrum-in-10-easy-steps\\_20.html](http://kw-agiledevelopment.blogspot.com/2007/09/how-to-implement-scrum-in-10-easy-steps_20.html) [accessed 27th March, 2008].