Integrating a SCRUM-based process with Human Centred Design: an Experience from an Action Research Study

Carmelo Ardito[1], Maria Teresa Baldassarre[1], Danilo Caivano[1,2], Rosa Lanzilotti[1]
[1]University of Bari, Italy
[2]Ser&Practicess S.r.l. Spin Off, Bari, Italy
carmelo.ardito, mariateresa.baldassarre, rosa.lanzilotti @uniba.it
d.caivano@serandp.com

***Abstract— Context*: Agile software processes are more frequently being integrated with aspects of human centered design given the commonalities that the two approaches share, such as user involvement, importance attributed to testing and prototyping practices, iterative design and so on.**
***Goal*: tailor a SCRUM-based process by integrating it and adapting it with Human Centered Design (HCD) practices.**
***Method*: action research experience carried out in an industrial case study involving researchers, an Italian SME and stakeholders of the final product (a web portal).**
***Results*: The research has led to the definition of the HCD-SCRUM process. The classic process has been tailored by: including staff dedicated to the project, considering flexible duration of sprints based on the workload and priority of features to be developed, using an iterative-incremental progression of the SCRUM development practices and including customer interviews.**
***Conclusion*: The tailored process outlined in this paper has been substituted to the classic SCRUM process that the company was adopting before this research study.**

Keywords-component; Agile process, SCRUM, Human centered design, action research

I. INTRODUCTION

Software development trends are moving more and more towards agile methods pushing, at the same time, for a constant integration of customer feedback and HCD within the agile approach practices themselves. Interest in the integration of HCD and agile development approaches is growing, as demonstrated by the number of papers published in the last decade [5]. Although agile and HCD are two different software development approaches, they share common principles. Just to name a few: iterative design, user involvement, continuous testing and prototyping [6], which are analyzed from different viewpoints on behalf of both communities. As researchers we are confident that their integration will hopefully lead to positive results: HCD can improve agile development by providing a systematic way to analyze end-user needs, whereas agile can improve HCD by providing more frequent iterations, and therefore more frequent usability evaluations [7].

There are some evidences of successful integration of Human-Centered Design (HCD) techniques with software engineering practices [2], provided that a deep analysis of company practices is performed by HCD researchers together with the practitioners, working synergically during the entire software development process from inside the company. Studies have investigated various aspects of the integration of HCD and agile approaches and suggest the most appropriate manners for enhancing such integration. In many cases, researchers report about their experience and provide recommendations suggesting how HCD can be effectively integrated in agile approaches. An interesting systematic review in this sense carried out by Silva da Silva et al. [8] identified six main aspects concerning the integration of these two approaches. Nonetheless, they seldom find application in industrial settings [1].

Specifically, in this paper we report an action research case study conducted in a Small-Medium sized Enterprise (SME) that migrated from a SCRUM-based methodology towards what the authors have defined HCD-SCRUM process. The revised tailored process integrated points between SCRUM and HCD activities. It was applied to a real project concerning the development of a web portal for retrieving data pertaining public institutions, comparing and analyzing them according to a set of specific economic indexes (e.g., number of employees, incomes, outcomes, taxes issued, etc.).

The concept underlying the process is its adherence to SCRUM based practices, which have been adapted to conform to the needs of the SME. Key points are: constant involvement of the customer; rapid development of code through iterative stepwise refinements; high frequency of releases due to the continuous iterations.

The main novelty of the tailored process is related to the need of an initial step, longer than the following development sprints, for creating a software system skeleton (i.e., the basic architecture for enabling the core functionalities and the wireframe of the main user interfaces) according to the requirements identified during the initial meetings with the customer. Starting from the next sprints, an evolutional prototype is evaluated with the customer, as usually occurs in SCRUM methodologies [3, 4].

978-1-5386-1546-1/17 $31.00 © 2017 IEEE
DOI 10.1109/CESI.2017.7

Another significant result showed that both face-to-face and remote informal Verification&Validation (V&V) sessions performed during a sprint were very valuable for improving usability and user experience. The customer was involved and actively participated in the co-design of the prototype. The tailored process has been further experimented in other projects and is currently being used by the software company.

The rest of the paper is organized as follows. Section 2 contextualizes the background on integrating HCD and Agile development processes. In section 3, after outlining a typical SCRUM-Like process, we illustrate how it has been tailored to the needs of an Italian SME called SER&Practices (from here on SER&P) and how customer feedback and HCD have been integrated into the agile development approach. Section 4 illustrates the experience pertaining an action research case study that shows how the process has been applied to a real case. Lessons learned and conclusions are finally drawn.

II. Background

There are various evidences that show how the community has demonstrated interest towards the integration of HCD and agile development processes [5, 6]. Despite their specificities, they can be seen as two faces of the same coin as they share common principles such as iterative design, user involvement, continuous prototyping just to mention a few.

Furthermore, some studies indicate how the integration of "HCD with Agile" presents, among others, issues such as: (i) communication between developers and designers [12, 13], which sees HCD experts prone towards user performance and user satisfaction while agile practitioners are more concentrated on transforming functional requirements into a running software system; (ii) differentiation between customer and user roles [14].

In a recent systematic literature review carried out by Silva da Silva et al. [8] authors identified six main aspects concerning the integration of HCD and Agile approaches: (1) Little Design Up Front (LDUF) i.e. user research activities should be performed before the project kickoff meeting [15] or in a Sprint 0 through user interviews [12]; (2) Prototyping i.e. generate prototypes from the user stories starting from the initial stages of the development process as they represent a preferred communication means between developers and HCD specialists [2]. Prototypes can be generated from persons and user stories; (3) User stories which in turn become functional for creating system prototypes [7]; (4) User testing. There are different opinions on this point. Some suggest carrying out evaluation on paper prototypes keeping in mind what the user interface will be like in the next iterations [7], others suggest executing user testing only on interactive prototypes, others yet suggest integrating user testing into the acceptance tests to validate the user interface [16]; (5) Inspection evaluation is recommended to be performed on paper prototypes until it can be used for implementing the user interface; 6) one sprint ahead, i.e. HCD specialists work one sprint ahead of the development team and recommend that this practice begins in Sprint 0 or two or three iterations ahead of the rest of the team [12].

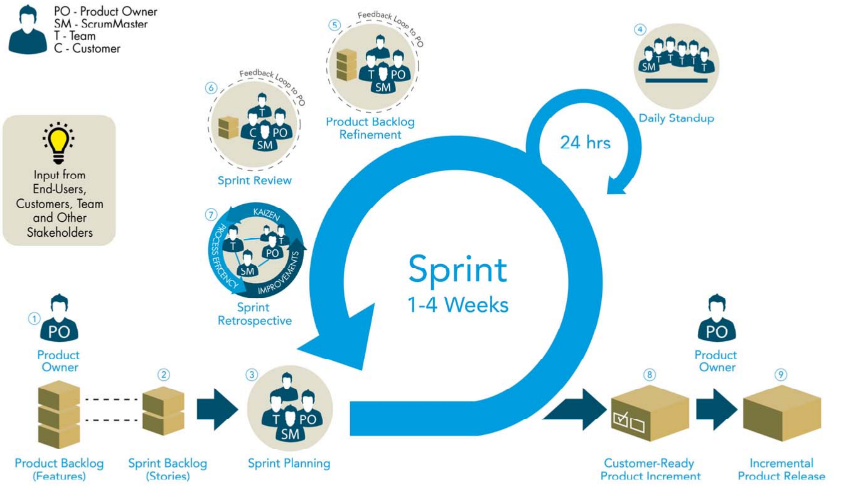
In this work we have reported our experience in integrating HCD activities in an agile software development process. In doing so we have considered the lessons learned and recommendations reported in literature summarized in this section.

III. From SCRUM to HCD-SCRUM

A typical SCRUM-Like process (Fig. 1) is made of three major steps - Inception, Development, Deliver and involves the following key roles: Product Owner, SCRUM Master, Team, Customer. There are also a set of time-box events: Sprint Planning meeting, Daily Standup meeting, Sprint Review, Sprint Retrospective – and a couple of fundamental artifacts – Product Backlog and Sprint Backlog. A key aspect of SCRUM is the use of self-organized, cross-functional and empowered teams who organize their work into short development cycles called Sprints.

In the current experience with SER&P, since the main goal of the research was "to identify integration points between SCRUM and HCD activities", we (researchers) performed an action research study in collaboration with experienced project managers of the company. Action research [9] is a social science methodology designed to help communities and organizations improve the way they address issues and solve problems and at the same time develop scientific knowledge about the problem and its solution. Researchers and practitioners work together as a team to assess current practices, propose a new course of action, implement changes, and evaluate the results. As final results, we defined a tailored process, called HCD-SCRUM, which led to significant improvements from both HCI and SE point of view.

The process is respectful of SCRUM based practices that have been reconsidered and tailored to the needs of the company. Moreover, main concerns and good practices see customers being constantly involved; code being developed through iterative stepwise refinements; continuous iterations leading to frequent releases.



1. The Product Owner creates a working vision of the end product defined as a set of prioritized features in the Product Backlog.
2. The highest priority features from the Product Backlog are pulled into the Sprint Backlog.
3. At Sprint Planning, the team makes final refinment, confirms the Sprint Backlog, commits to complete the work, and kicks off the Sprint.
4. Each day, the team meets and answers:
   What did I do yesterday?
   What do I plan to do today?
   What impediments are in my way?
5. During the Sprint, the team refines the Product Backlog and refines the definition of features so that they are ready to be pulled into a future Sprint.
6. At Sprint Review, the team demos features completed during that Sprint and collects feedback to update remaining features in the Product Backlog.
7. The Scrum Master facilitates a Retrospective that identifies process improvements to accelerate the team in the next sprint.
8. Each Sprint results in a customer-ready product increment.
9. The team repeats the Sprint cycle until the Product Owner decides to release the product.

Figure 1. A SCRUM process representation adapted from http://www.scruminc.com

The tailored process is depicted in Fig.2 and described as follows:

1) *Customer Committee*. The Product Owner covers one of the key roles of the process as he represents all the stakeholders and is the Customer's voice. He or she is accountable for ensuring that the team delivers value to the business. The product owner writes customer-centric items (typically user stories), ranks and prioritizes them, and adds them to the product backlog. SER&P has introduced the concept of Customer Committee that includes a Product Owner selected within the Scrum Team staff and at least two people from the customer side, desirably an end user and a business domain expert. The Customer Committee is actively involved in the Product and Sprint Backlogs definition and Sprint Review.

2) *Inception*. In SCRUM based processes software product or service development usually starts with stakeholder inputs. In SER&P the inception phase consists of interviews to stakeholders, as well a context and field study carried out jointly with the customer. The study provides the essential context background and knowledge necessary to design the application being developed and enable the Customer Committee and Team to evaluate and prioritize the issues/features to develop. It consists of market study, cost, portfolio and competitor analyses. In this way SER&P merges the needs and modus operandi usually adopted by HCI communities more prone to referring to end users and customers during requirement definition, and software engineers who usually have limited contact with final users. This way the Customer Committee can pin down priorities, backlogs, user stories and features that customers consider most relevant. Given this iterative characteristic of gradually adding features to the final product, customers are continuously involved in designing and approving developed features and versions before they are ultimately released.

3) *Sprint n.0*. In a SCRUM like process, during this step a preliminary architecture of the application is conceived and main modules and sub-systems are identified. The HCD-SCRUM process explicitly identifies a Sprint n.0 in which a high level prototype is built. The basic architecture and infrastructure of the system are developed at this stage. This assures a highly modularized software system skeleton with basic software services such as data, communication, reporting, computation services etc. that lead to developing effective software modules and functions during the next Sprints.

4) *SCRUM Islands*. SER&P typically implies Teams of 4 persons or less. One person is the Product Owner included in the Customer Committee who also covers the role of team member when not engaged in any committee activity. Another team member is a graphic designer with basic knowledge of HCI techniques. Others are software engineers. When a project begins, the selected team members are transferred in a "SCRUM Island" i.e. a round 4-seat working desk. This arrangement maximizes the information flow, improves communication and collaboration between team members. Consequently, in SER&P Daily Standup Meetings are not formally adopted. If a member is included in different islands or involved in more than one project, he/she physically moves between islands. As policy, the company tries to minimize sharing staff across ongoing projects.

5) *1 Week Time Boxed Sprint*. SER&P adopts sprints with a fixed duration of 1 week. Every week the sprint outputs are reviewed and the following sprint backlog is defined. It includes the new tasks along with the ones that have not been closed yet. Sprint duration is predefined, while

the Customer Committee at the end of every sprint estimates the entire project, during the Product Backlog Refinement. Thus the estimations about project termination and release deliver are carried out at the beginning of the project and updated weekly. A first draft of issues and their priority is made during the so called Inception phase in order to have a general idea of the roadmap to follow and a rough estimate of the effort required.

6) *(IN)Sprint Review*. The SER&P process includes a continuous verification and validation (V&V) activity. This activity temporally spans along the entire sprint. When a first/draft working software is available (after Sprint n.0 or few sprints later), the end users of the Customer Committee are asked to use the system and carry out a user test or functional system test. This stresses the functionalities developed during a sprint and verifies their proper integration with what already exists. The results of the V&V activities are then discussed and analyzed within the Sprint Review at the end of the Sprint. Here the Customer Committee is involved in the analysis of the results, from both functional and technical point of view.

7) *Project Retrospective*. A SCRUM process typically includes the Sprint Retrospective that is an official event in the SCRUM methodology where all the parties (Product Owner, SCRUM Master, Team) involved in the development try to improve the process by sharing strengths and weaknesses. SER&P's process doesn't include it. SCRUM means self-organized, cross-functional and empowered teams who organize their work autonomously. The Customer Committee, (IN)Sprint Review and SCRUM Island assure both information flow and adequate communication as well as coordination and prompt feedbacks from both team members and customer. Indeed, the Team can share, analyze and improve the process continuously without a formal Sprint Retrospective event. On the other hand, SER&P foresees a Project Retrospective, i.e. all the available project data are analyzed, strengths and weaknesses are highlighted and improvement opportunities and initiatives are defined and executed. This creates an external feedback loop that involves the entire organization [10, 11]. In this step customers are actively involved in the retrospective analysis through a thematic focus group i.e. customers are informally interviewed in order to explore improvement opportunities and better address customer satisfaction and product quality improvements. The feedback collected are then used by SER&P to globally improve its strategies, management organization and general process.

SER&P uses an Application Lifecycle Management (ALM) tool to manage its SCRUM-based process called Redmine (www.redmine.org). In accordance to the tool characteristics, requirements are classified into categories and issues are then prioritized based on a criticality scale (low, normal, high, urgent, and immediate). The ALM becomes the Product Owner's central channel towards the development team. The Product Owner uses the tool to operatively formalize backlogs, classify issues and assign them to development team members who, in turn, use their accounts to track the issues to develop during each sprint. In this way the project status and progress can be monitored through a SCRUM board interface which tracks information such as tasks by category, status, workload of developers, issue tracker and so on. On the other hand, SER&P uses a software called Kiuwan (www.kiuwan.com) to address software product quality during development. The Product Owner and Team use it during each Sprint Review. Moreover, a quality report is produced for each sprint. Data automatically collected by using Redmine and Kiuwan are also used in the Project Retrospective for deciding on possible improvements.

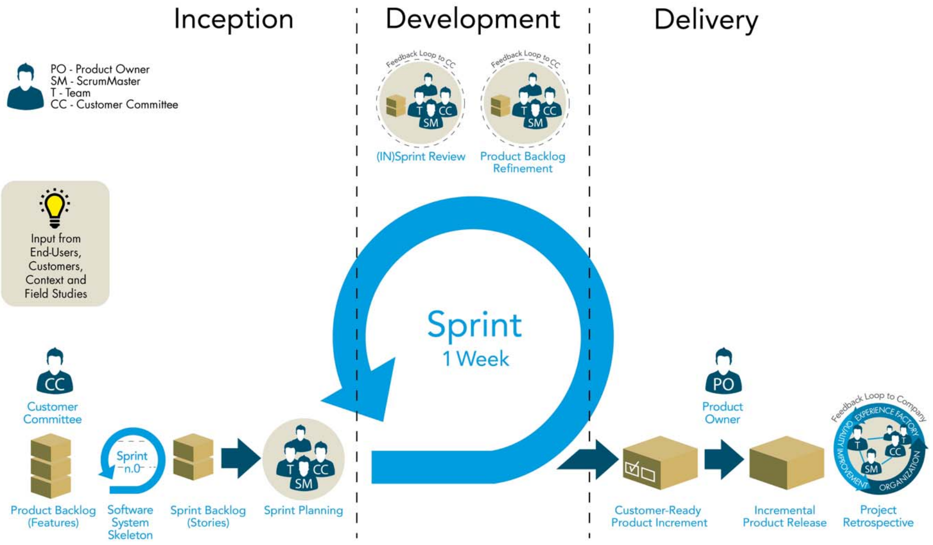


Figure 2. HCD-SCRUM process tailored in SER&P

IV. Industrial Case Study

In this section we illustrate a case study showing how HCD-SCRUM has been applied by SER&P within an industrial project called "PublicAccounts" in order to understand how stakeholder input and feedback was integrated into the company's software practices.

The *PublicAccounts* project consisted in designing a web portal that integrated various data sources containing information concerning public administrations and organizations. The portal draws data in the form of open data from official sources on the web such as www.soldipubblici.it, databases provided by the Central Bank of Italy as well as the Ministry of Economy and Finance. It elaborates and classifies the data according to specific criteria and produces reports showing economical values of public institutions such as expenses, revenue, how public money is spent and invested, and so on. The application currently integrates data of 8000 Italian cities. It provides a general overview of each single administration in that it compares data from public administrations, produces reports, elaborates statistics and ranks virtuous cities compared to non-virtuous ones. The long run users of the web portal are citizens, journalists, and public administration employees.

As of now, the project has been running for 10 months and is expected to last a total of 18 months implying a total of 5 staff with at least five years of experience. Overall up to now a total of 1000 person days (including management effort) have been implied.

SER&P adopted the SCRUM-based process, illustrated in the previous section, to develop the portal. In accordance to the process, the work was carried out in incremental iterations. Two researchers participated in the weekly meetings with the company staff during which customer feedback and experience was exploited in order to define features to implement. The work for the current Sprint and the next activities were also planned.

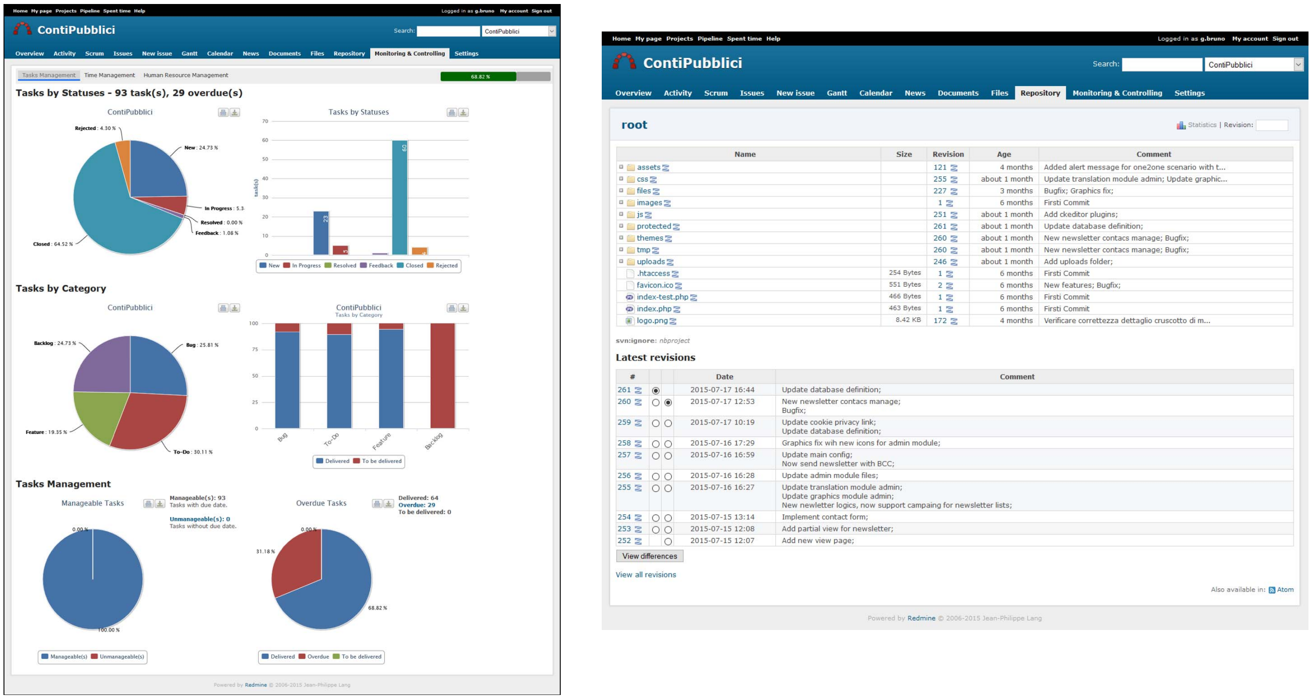
The Customer Committee was appointed with one SER&P staff, an end user, i.e. a technician of the customer company who used the portal daily, and a top manager ingrained in the business domain that the portal referred to. Three development team members were dedicated to the project. The Daily Standup meeting was not conducted because of the SCRUM Island structure. A weekly Sprint Review and Product Backlogs were scheduled.

Weekly meetings with Customer Committee were scheduled. In particular, in Sprint n.0 they were systematic and feedback was continuous and focused on defining a general picture of the project, customer needs as well as the software system skeleton. Sprint n.0 was the longest (40 days). The longer duration allowed to define a more detailed picture of requirements, architecture and front-end design, and to produce a first working prototype. In this sprint, more than in others, a greater portion of time and effort was arranged for planning and gathering customer data to produce upfront design and a starting-point prototype. Meetings were arranged in both SER&P sites and in the customer offices and were mainly face to face. Video conferencing, email and teleconference were also implied mostly during the later project sprints and especially during the (IN)Sprint Review in order to jointly analyze the defects, bugs or non conformities discovered by the end user during the continuous V&V activities. End users included in the Customer Committee and Product Owner from SER&P were involved in this task.

A total of 17 sprints were carried out, the first lasting 40 working days (in about 3 months) and one week each for the other ones. The ALM tool, Redmine, was used to conduct and manage the project in term of sprints, tasks assigned to the development teams, prioritization of user stories, features and tasks, customer feedback and suggestions. To this end, figure3 provides an example of how project activities were monitored and controlled with Redmine. Moreover, thanks to the features of the process, the Product Owner was able to have an overview of the entire project. In particular, tasks were grouped by status, category and management. It helped summarize the information of the project sprint board related to all of the Sprints, and identify for example which amount of tasks were new, in progress, resolved, closed or rejected; which categories they belonged to, and what their delivery status was.

As the project went on, at the end of each (IN)Sprint Review features were verified by Customer Committee and Product Owner. In particular, product quality was verified with respect to usability and internal product quality through the Kiuwan platform. It analyzed source code with respect to five quality characteristics (maintainability, reliability, portability, efficiency and security), validated the measures and compared them to baseline target threshold values, which could be set on a scale from 0-100. In this case study, the target values were all set to 70.

The quality assessment report generated by the Kiuwan software summarized general characteristics of the software's source code such as: number of lines of code, number of files, level of complexity and amount of duplicated source code. The report produced was used as discussion point for the (IN)Sprint Review. It was also used in the Project Retrospective phase of the Delivery step, before releasing any version of the product to the customer. This activity enforced interaction between Customer Committee, Product Owner, and Team in the light of a continuous improvement cycle towards the development and release of the final product. In other words, final users, customers and professionals collaborated for the entire duration of the project.


Home My page Projects Pipeline Spent time Help
Logged in as g.bruno My account Sign out
ContiPubblici
Search:
ContiPubblici
Overview Activity Scrum Issues New issue Gantt Calendar News Documents Files Repository Monitoring & Controlling Settings
Tasks Management Time Management Human Resource Management
68.82 %
ContiPubblici
Rejected : 4.30 %
New : 24.73 %
In Progress : 5.3
Resolved : 0.00 %
Feedback : 1.08 %
Closed : 64.52 %
Tasks by Statuses
task(s)
0 10 20 30 40 50 60 70
23
5
60
4
New In Progress Resolved Feedback Closed Rejected
ContiPubblici
Backlog : 24.73 %
Bug : 25.81 %
Feature : 19.35 %
To-Do : 30.11 %
ContiPubblici
Tasks by Category
0 25 50 75 100
Bug To-Do Feature Backlog
Delivered To be delivered
Manageable Tasks
0.00 %
100.00 %
Manageable(s) Unmanageable(s)
Overdue Tasks
0.00 %
31.18 %
68.82 %
Delivered Overdue To be delivered
Powered by Redmine © 2006-2015 Jean-Philippe Lang

Tasks by Statuses - 93 task(s), 29 overdue(s)

Tasks by Category

Tasks Management

Manageable(s): 93
Tasks with due date.

Unmanageable(s): 0
Tasks without due date.

Delivered: 64
Overdue: 29
To be delivered: 0

Home My page Projects Pipeline Spent time Help
Logged in as g.bruno My account Sign out
ContiPubblici
Search:
ContiPubblici
Overview Activity Scrum Issues New issue Gantt Calendar News Documents Files Repository Monitoring & Controlling Settings
Statistics | Revision:

root

Name	Size	Revision	Age	Comment
assets		121	4 months	Added alert message for one2one scenario with t...
css		255	about 1 month	Update translation module admin; Update graphic...
files		227	3 months	Bugfix; Graphics fix;
images		1	6 months	Firsti Commit
js		251	about 1 month	Add ckeditor plugins;
protected		261	about 1 month	Update database definition;
themes		260	about 1 month	New newsletter contacs manage; Bugfix;
tmp		260	about 1 month	New newsletter contacs manage; Bugfix;
uploads		246	about 1 month	Add uploads folder;
.htaccess	254 Bytes	1	6 months	Firsti Commit
favicon.ico	551 Bytes	2	6 months	New features; Bugfix;
index-test.php	466 Bytes	1	6 months	Firsti Commit
index.php	463 Bytes	1	6 months	Firsti Commit
logo.png	8.42 KB	172	4 months	Verificare correttezza dettaglio cruscotto di m...

svn:ignore: *nbproject*

Latest revisions

#			Date	Comment
261	●		2015-07-17 16:44	Update database definition;
260	○	●	2015-07-17 12:53	New newsletter contacs manage; Bugfix;
259	○	○	2015-07-17 10:19	Update cookie privacy link; Update database definition;
258	○	○	2015-07-16 17:29	Graphics fix wih new icons for admin module;
257	○	○	2015-07-16 16:59	Update main config; Now send newsletter with BCC;
256	○	○	2015-07-16 16:28	Update admin module files;
255	○	○	2015-07-16 16:27	Update translation module admin; Update graphics module admin; New newletter logics, now support campaing for newsletter lists;
254	○	○	2015-07-15 13:14	Implement contact form;
253	○	○	2015-07-15 12:08	Add partial view for newsletter;
252		○	2015-07-15 12:07	Add new view page;

Figure 3. Redmine ALM Tool for managing HCD-SCRUM

V. LESSONS LEARNED

Strategically speaking, we have learned the following lessons from the action research experience conducted conjunctly by a group of academic researchers, SER&P staff and the stakeholders (users and customers) of the final product where HCD practices have been integrated into SCRUM-based process:

i. customer feedback is valuable for HCD and in turn agile practices seem well suited to be mapped to customer involvement;

ii. feedback and customer involvement should be established at multiple points throughout the development process. To this end, HCD-SCRUM adopted by SER&P includes several points in all steps (Inception, Development, Delivery) that take into account such aspects;

iii. interviews/meetings/field studies should be carried out in the first part of the project, during the first sprint in order to have a clear picture of what is going to be developed. For this reason, the first sprint is most likely longer than the subsequent ones which focus more on development, evaluation, implementation and integration aspects;

iv. customer feedback should be filtered, i.e., not everything a customer wants can be carried out or is actually a good idea. Decision points should be set where suggestions can either lead to a new issue a change request or even be rejected;

v. the project status and evolutions should be tracked as the project proceeds. Decision making and project status monitoring on behalf of the Project Owner with respect to the development team members can be simplified if ALM tools are adopted to support organization of sprints, daily meetings and issue assignments as they provide an important infrastructure for the entire team;

vi. make use of short iterations, meetings and focus groups between "stakeholders" during system development in order to reduce the risks of omitted requirements and deliver unusable systems. Stakeholders obviously include among others, users, lead users and customers as well as developers and experts;

VI. CONCLUSIONS

In this paper we have presented and discussed an action research study conducted in an Italian Small-Medium sized Enterprise (SME) focused on integrating HCD practices into the company's ingrained agile SCRUM process. The research has led to the definition of the HCD-SCRUM process. Changes to the basic process have involved including staff that is totally dedicated to the project, possibility of considering flexible duration of sprints based on the workload and priority of features to be developed, use of an iterative-incremental progression of the SCRUM development practices and most of all, customer interviews/field studies concentrated especially early on in the project.

The tailored process applied to a web application development project is a first attempt of how changes have been integrated in the SME's ingrained SCRUM practices. The HCD-SCRUM process described in this paper has been substituted to the classic SCRUM process that the company was adopting before this research study. It is currently used by SER&P as standard development process.

REFERENCES

[1] C.Ardito, P.Buono, P., D.Caivano, M.F. Costabile, R. Lanzilotti, A. Bruun, J.Stage. “Usability evaluation: a survey of software development organizations”. In: International Conference on Software Engineering and Knowledge Engineering (SEKE '11). (Skokie, IL, USA), 2011, p.282-287.

[2] C.Ardito, P. Buono, D. Caivano, MF. Costabile, R. Lanzilotti, “Investigating and promoting UX practice in industry: An experimental study”. International Journal of Human-Computer Studies 72(6), 2014, p.542-551.

[3] K.Schwaber, M. Beedle. “Agile Software Development with Scrum”. Prentice Hall PTR, Upper Saddle River, NJ, 2011, USA.

[4] R.Dybå, T. Dingsøyr. “Empirical studies of agile software development: A systematic review”. Information and Software Technology 50(9–10), 2008, pp.833-859.

[5] G.Jurca, T.D.Hellmann, F.Maurer, “Integrating Agile and User-Centered Design: A Systematic Mapping and Review of Evaluation and Validation Studies of Agile-UX”. In: Proceedings of the Agile Conference (AGILE '14). 2014, p.24-32.

[6] S. Blomkvist, “Towards a Model for Bridging Agile Development and User-Centered Design” In: Seffah, A., Gulliksen, J., Desmarais, M. (eds.), Human-Centered Software Engineering — Integrating Usability in the Software Development Lifecycle. Springer Netherlands, Vol. 8, 2005, pp.219-244

[7] Z.Hussain, H. Milchrahm, S. Shahzad, W. Slany, M. Tscheligi, P. Wolkerstorfer, “Integration of Extreme Programming and User-Centered Design: Lessons Learned”. In: Abrahamsson, P., Marchesi, M., Maurer, F. (eds.), Agile Processes in Software Engineering and Extreme Programming. Springer Berlin. Vol. 31, 2009, pp.74-179.

[8] T. Silva da Silva, A. Martin, F. Maurer, M. Silveira, “User-Centered Design and Agile Methods: A Systematic Review”. In: Agile Conference (AGILE '11). 2011, pp.77-86.

[9] P.Checkland, S. Holwell, “Action Research: Its Nature and Validity”, Systemic Practice and Action Research 11(1), 9-21, 1998

[10] VR.Basili, G.Caldiera, HD Rombach, “Experience Factory. Encyclopedia of Software Engineering”. John Wiley & Sons, 2002

[11] K.Schneider, JP.Hunnius, V. Basili, V. “ Experience in Implementing a Learning Software Organization”. IEEE Software. 19(3), 2002 pp. 46-49

[12] S.Chamberlain, H.Sharp, N.Maiden. “Towards a Framework for Integrating Agile Development and User-Centred Design”. In: Abrahamsson, P., Marchesi, M., Succi, G. (eds.), Extreme Programming and Agile Processes in Software Engineering - XP '06. Vol. LNCS 4044, Springer, 2006. pp. 143-153.

[13] T.Memmel, F.Gundelsweiler, H.Reiterer. “Agile human-centered software engineering”. In: British HCI Group Annual Conference on People and Computers: HCI...but not as we know it - Volume 1 (BCS-HCI '07). British Computer Society, Swinton, UK 2007, pp. 167-175.

[14] J.Highsmith. “Agile software development ecosystems”, Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA. 2002

[15] M. Detweiler. “Managing UCD within agile projects”. interactions 14(3), 2007. pp. 40-42

[16] M. Federoff, C.Villamor, L.Miller, J.Patton, A.Rosenstein, K.Baxter, K.Kelkar, “Extreme usability: adapting research approaches for agile development”. In Extended Abstracts of SIGCHI Conference on Human Factors Human Factors in Computing Systems (CHI EA '08). pp. 2269-2272. ACM, New York, NY, USA.