

Persona driven agile development

Build up a vision with personas, sketches and persona driven user stories

Dominique Winter

GreenPocket GmbH

Cologne, Germany

dominique.winter@greenpocket.de

Eva-Maria Holt

SEVEN PRINCIPLES

Ratingen, Germany

eva-maria.holt@7p-group.com

Jörg Thomaschewski

HS Emden/Leer

Emden, Germany

joerg.thomaschewski@hs-emden-leer.de

Abstract - In the beginning of a user centered agile development process, three important elements have to be defined. 1) A software product with a defined set of features, 2) the intended usage context and 3) the future software product user. The clear definition of these elements right from project start will lead to a more accurately developed software product. A combination of well-known tools like “personas” and sketches of the context of use as well as requirements as persona driven user stories make a comprehensive basis. During an iterative visioning phase, also referred to as sprint 0, directly before starting the first sprint, those tools can be integrated to assure a complete software product definition process. That will ensure that, even in this early stage, the agile project is focused on user needs represented by “personas”.

Keywords: Scrum, User Centered Design, Personas, User Stories, Sketching

I. INTRODUCTION

Agile project management is widely adopted. Scrum is a very popular agile project management framework. Furthermore agile project management argues that a project cannot be planned completely from the very beginning. According to this conviction Scrum follows an iterative and incremental planning approach. Scrum framework brings a small set of rules and roles together and these let teams be easily organized. Scrum also allows a quick change in direction if unknown obstacles suddenly appeared or the requirements changed [1]. Nevertheless this procedure is incomplete. Scrum describes how to organize an agile development process but neither exactly defines how to go on nor the needed tools to make the process user-centered.

Scrum is focused on initiating communication. It releases creativity and team spirit and “makes the world a better place to live and work” [2]. But successful product development needs more than organizing teams and producing software. In practice the presence of the user is often missing. Especially in new agile teams [3] the missing presence of the user mostly leads to a product with poor usability which does not match user needs and expectations. To improve the startup of an agile project it is possible to establish a preliminary visioning phase with focus on user needs represented by “personas” as sprint 0.

II. VISIONING PHASE

At the beginning of the development process there should be a clear vision about what the software product has to be and what it should not be. This is critical to the success of an agile development process [3]. From the user’s viewpoint the vision describes how the new system should work and also includes potential changes in technology as well as a high level presentation of functional and non-functional requirements. The Scrum project should start with a visioning phase to define the vision. There are different options how to structure this visioning phase [4]. The authors suggest a new approach to center the tasks of the visioning phase on “personas”. “Personas” can represent the user during the development process of a new product and should be integrated as early as possible. There are also significant advantages of using “personas” during the product conception stages of the development process [7].

Using a combination of best practices in agile software development and tools for user-centered design, the visioning phase is inevitable to get detailed information about user needs. Moreover it can be organized similar to a Scrum flow. Therefore this phase can be placed as sprint 0.

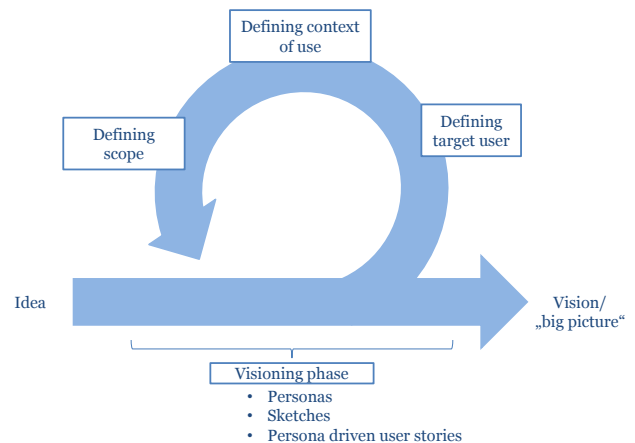


Figure 1. Visioning phase

The first step in a user centered design process is to get information about the future user, the tasks and the context of use. These are very important tasks which should be done at the beginning. The target user should be defined as someone who will use the developing product. This is important to keep the user needs in mind during the following steps. To understand in which situation the final product will be used it is recommended to record a description of the context of use. Finally the resulting requirements should be available to all project team members. This will enable proceeding in a user centered way.

These tasks are not isolated and the results of the three tasks affect each other. For example the definition of the context of use will affect and be affected by the user concept. The context will also have an impact of the definition of the project's scope. With regards to this fact, an iterative process is essential as seen as in Figure 1. By the end of this phase a big picture is created, which must be approved by the stakeholders.

The results of the visioning phase will be recorded as "personas", sketches and persona driven user stories. These tools are visual and emphatic and can be easily communicated to other stakeholders. The combination of these three tools with a "persona" centered approach has been successfully tested during many small projects.

Optimizing the process of iteratively adapting the big picture sketches can be a good alternative. Sketches are small pieces and can be put together to the big picture on a pin board. The advantage is to make it easier to change the vision iteratively. Following the iterative process the "personas", sketches and persona driven user stories are continuously reviewed and supplemented during the whole project and even the big picture can be changed in small dimensions.

III. ELEMENTS OF THE VISIONING PHASE

The following chapter gives an overview of the three tasks which should be done early during the visioning phase. The artifacts from these tasks are "personas", sketches and persona driven user stories.

A. Defining target user

In order to define the target user, the user-centered design offers a tool which enables visualizing the users and their needs in detail. This tool is called "personas".

A "persona" is a fictional model of a potential user, which is based on behaviors and motivations of real people [5]. "Personas" can represent the real users during the development process and give a face representing anonymous user [6]. Different people, who are involved in a project, can create the same picture of the user, so a "persona" can also be used as a communication tool. A "persona" helps to empathize with the user, understand user behavior and deeper needs.

First it is necessary to gather data from interviews and documents that provide information about the possible target group. Afterwards the information should be summed up to create different "personas". The information should be represented as if the "personas" are real people [7]. Therefore it is necessary to add or choose a picture, which does not appear unnatural and supports the textual information.

As Figure 2 shows, a "persona" can contain career information, skills, personal details, behaviors and a short story. Especially the personal details and the behaviors will strengthen the empathic link between future user and the team.

It is possible to differentiate "personas" because there are two different ways to create them. On one hand there are real "personas", which are based on data gathered from interviews

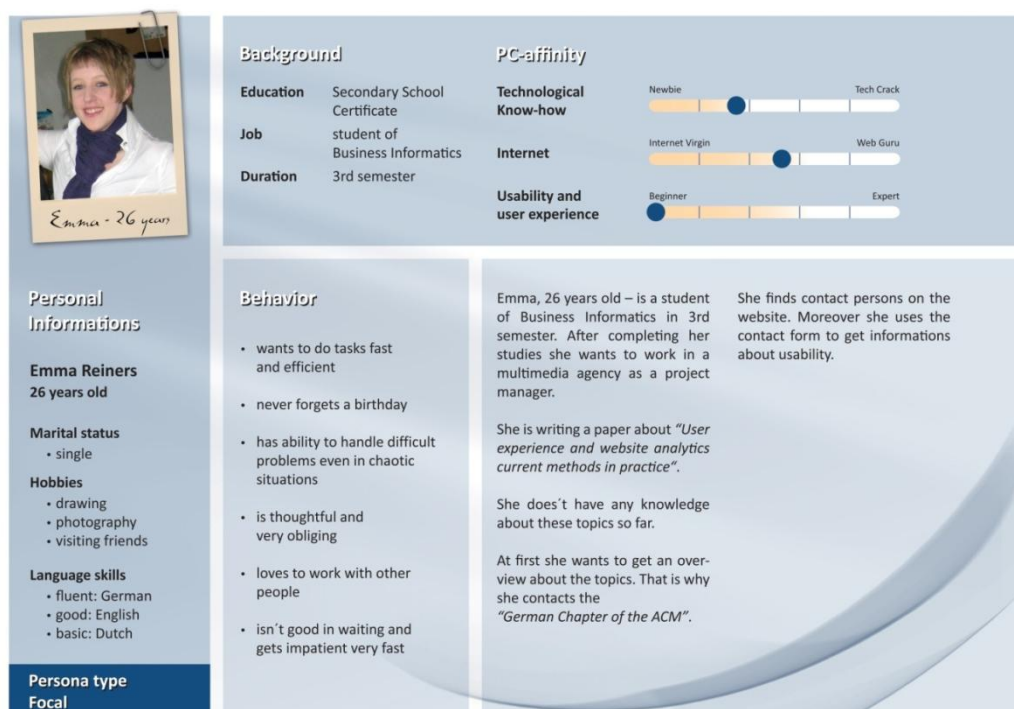


Figure 2. The "persona" Emma

with targeted users; on the other hand there are realistic “personas” [8] which are based on information from different people such as stakeholders, marketing or sales staff. Norman also refers to the realistic “personas” as “ad-hoc personas” [9]. Using a realistic “persona” entails the risk that the “persona” may appear as a stereotype and the fictional element could be dominant. In this case the method would lose its advantages.

According to the different levels of the user’s experience, the “personas” can be divided into different groups. These groups are beginners, intermediates and experts [5]. A user is a beginner when she interacts with a system for the first time. Normally this user does not stay in this status long time. After using the system regularly she will become intermediate.

In addition to the classification of “personas” with different experience levels, it is also possible to divide them into different persona types. Cooper and Reimann [5] propose six different types (Primary, Secondary, Supplemental, Customer, Served, and Negative). On the other side Olsen [10] suggests five groups (Focal, Secondary, Unimportant, Affected, and Exclusionary).

Briefly, tasks and goals of the “personas” support the decisions about system functionalities and make the communication between stakeholders, developers and designers easier. “Personas” is also a tool for measuring the effectiveness of the design solution. However it is important to keep in mind that a “persona” only represents a small portion of the target group and the relationship between target groups and “personas” is problematic [5,11].

B. Defining context of use

A product will be used within a certain range of different environments (technical, physical and social or organizational) that will affect its use [12]. It is therefore necessary to consider these conditions in the visioning session. This can be done by using sketches.

“Sketching” implies drawing an idea on paper and reducing its quality. Today sketching is a well-known tool to create a fast visualization of an idea and many people use it already in various parts of their life. Concerning this fact this simple tool is suitable to use in an early project phase particularly when many people from very different backgrounds are involved. Sketches can make the communication in a project much easier, especially when the team members speak different languages. Each of them can make their own sketches and share them. There is no need to be a good drawer.

Additionally, sketches are quick, timely, inexpensive, disposable, plentiful, and unambiguous. They have minimal details, clear vocabulary and an appropriate degree of refinement. Furthermore they suggest and explore, rather than confirming [13].

It is easy to assign a sketch to a “persona”. In a sketch about the expected context a “persona” can be the main protagonist. All user expectations and concerns using the product can be presented in a sketch. Also external conditions can be appropriately taken into consideration as seen in Figure 3. No matter whether there is time pressure or miserable weather.

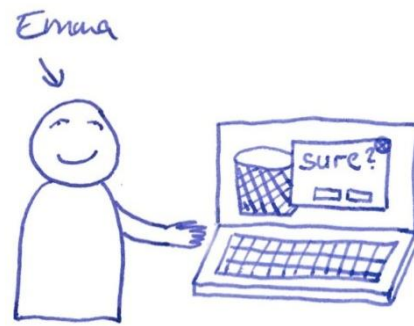


Figure 3. Sketch “Emma wants to delete a file permanently”

C. Defining scope

The scope of an idea will change depending on the people because everyone has their own mental model. To reach common understanding about the selection of possibilities, it is necessary to write down the requirements in a way all stakeholders can understand. A good way to reach this goal is writing user stories. They are more comprehensible than use cases or scenarios, because stories are short and written to demonstrate value from customer’s or user’s standpoint. This is the main reason why they are easily understandable by both business people and developers. [14].

A user story is composed of a written story, the conversation about it and tests that can be used to determine the completeness of a story [14]. They define requirements from a user’s point of view [15]. Normally user stories follow the form “As a [user role], I want [a feature] so that I can [achieve some goal]”. This short text form describes a user story as a piece of functionality that is of value to the user [16]. The short form of the written part results from the intention to be a reminder to the feature for everyone on the project. Also the size of the story must fit into a single sprint. If necessary the feature which the story describes must be broken down [17]. Therefore the user stories are explanatory and profound enough.

Because of changing requirements during the progress of the project they are intentionally inaccurate. That leaves room to discuss and to be flexible. For that reason the story needs the conversation. In the conversation all details, edge cases, and constraints are found out. This is necessary and will allow the team to estimate the effort needed to develop the story [16]. Nevertheless there should be a defined test, which will give the possibility to check if the story is completely implemented.

Concerning the three parts of the written user story there is a way to improve the consideration of future users. The user role describes in which context the user needs the feature. It is also possible to use “personas” instead of an anonymous user role (e.g. “Emma wants to...” instead of “As a visitor I want...” as seen in Figure 4). When using “personas” on the whole project this addition to user stories is the preferred way of requirement description. When using persona driven user stories the user is not an abstract construction anymore [3] and the story reflects the experience, backgrounds and goals of the users [14].

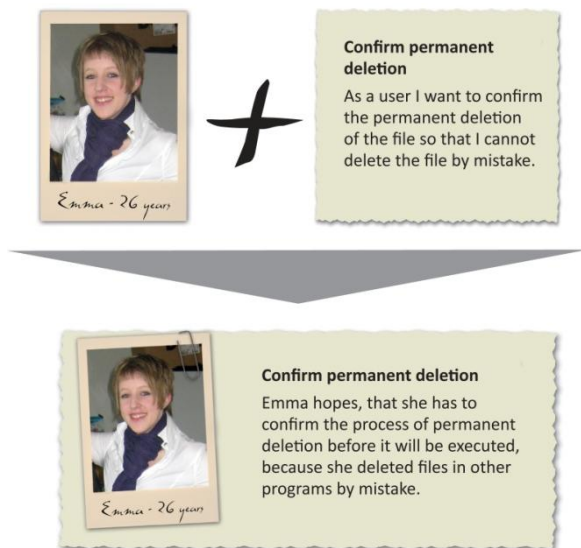


Figure 4. Persona driven user story

Cohn [14] suggests replacing one or two important user roles of user stories with “personas” depending on the role. Instead of this the authors approach is to replace the primary roles with different “personas”, because mostly the described feature of the user story will be used by different users. In this case the user story can be assigned to different “personas”.

It is also possible to describe non-functional requirements as user stories so all requirements look similar and can be understood by everyone involved in the project. Furthermore this method will also help developers to understand the user needs better [17] and especially when using persona driven user stories.

During the discussion about user stories between the product owner and the team they have to add acceptance criteria [18]. Again “personas” can help to represent the user in a more natural way by using persona driven user stories.

IV. CONCLUSION

This paper presents a way to integrate user centered design at the beginning of an agile development process with Scrum. At the end the authors think that adding a visioning phase as sprint 0 with a strong focus on “personas” to a Scrum project would be a good practice that lets the development process be more user-centered.

A significant point of this sprint 0 is to make the vision of the new system more concrete. This requires three steps:

- The first step is to define the target user. “Personas” are a practical way to document the results of the analysis.
- The second step is to define the context of use. Sketches are a good tool to document the results for further sprints. A sketch can be related to a “persona”.
- The third step is to define the scope using persona driven user stories. These are special types of user stories, which concretize the user role of a “persona”.

According to the principles of the agile manifesto [19] the results of the visioning phase are not static and are open for changes during the iterative development process. In most projects, requirements will change or will be specified in later project phases when knowledge about the project increases. “Personas”, sketches and persona driven user stories are fast created and can be easily developed in an iterative way. This is also needed because these artifacts influence each other.

REFERENCES

- [1] D. Friis, J. Ostergard, J. Sutherland, “Virtual Reality Meets Scrum: How a Senior Team Moved from Management to Leadership”, 44th hawaii international conference on system sciences, Ieee Press Books, New York, 2011.
- [2] J. Sutherland, K. Schwaber, “The Scrum Papers: Nut, Bolts, and Origins of an Agile Framework”, <http://jeffsutherland.com/ScrumPapers.pdf>, 1993-2011.
- [3] H. Beyer, “User-centered agile methods”, Morgan & Claypool Publishers, [San Rafael, Calif.], 2010.
- [4] K. Holtzblatt, J.B. Wendell, S. Wood, “Rapid contextual design: A how-to guide to key techniques for user-centered design”, Elsevier/Morgan Kaufmann, San Francisco, 2005.
- [5] A. Cooper, R. Reimann, D. Cronin, “About face 3: The Essentials of Interaction Design”, Wiley, Indianapolis (Ind.), op. 2007.
- [6] E.-M. Holt, D. Winter, J. Thomaschewski, “Personas als Werkzeug in modernen Softwareprojekten: Die Humanisierung des Anwenders”, In: Brau, H; Lehmann, A.; Petrovic, K.; Schroeder, M.C. (Eds.); Usability Professionals 2011, pp. 158 - 163, 2011.
- [7] F. Long, “Research Paper - Real or Imaginary: The effectiveness of using personas in product design” Proceedings of the Irish Ergonomics Society Annual Conference, pp. 1-10, 2009.
- [8] A. Beck, H. Eichstädt, B. Gaiser, P. von Savigny, P. Schubert, W. Schweibenz, “Personas in der Praxis, in: Usability Professionals 2005, pp. 98-102, 2005.
- [9] D. Norman, “Ad-Hoc Personas & Empathetic Focus”, from http://jnd.org/dn.mss/personas_empath.html, 2004
- [10] G. Olsen, “Persona Creation and Usage Toolkit”, from http://www.interactionbydesign.com/presentations/olsen_persona_toolkit.pdf, 2004.
- [11] C.N. Chapman, R.P. Milham, “The Personas’ new Clothes: Methodological and Practical Arguments against a Popular Method”, Proceedings of the Human Factors and Ergonomics Society, 50th Annual Meeting, pp. 634-636, 2006.
- [12] M. Maguire, “Context of Use within usability activities”, Int. J. Human-Computer Studies (2001) 55, 453-483, 2001.
- [13] B. Buxton, “Sketching user experiences: Getting the design right and the right design”, 3. printing., Morgan Kaufmann, Amsterdam, 2008.
- [14] M. Cohn, “User stories applied: For agile software development”, Addison-Wesley, Boston, Mass, 2004.
- [15] R. Wirdemann, “Scrum mit User Stories”, 2., erweiterte Auflage., Hanser, Carl, München, 2011.
- [16] K. Beck, M. Fowler, “Planning extreme programming: Includes index”, Addison-Wesley, Boston, 2001.
- [17] R. Davies, “Non-Functional Requirements: Do User Stories Really Help?”, from <http://www.methodsandtools.com/archive/archive.php?id=113>, Methods & Tools, 2010.
- [18] W.F. Nazzaro, C. Suscheck, “New to User Stories?”, from <http://www.scrumalliance.org/articles/169-new-to-user-stories>, 2010.
- [19] K. Beck, M. Beedle, A.v. Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, R. Jeffries, J. Kern, B. Marick, R. C. Martin, S. Mellor, K. Schwaber, J. Sutherland, D. Thomas, Manifesto for Agile Software Development, 2001

