

PERSONALITY TRAITS OF SCRUM ROLES IN AGILE SOFTWARE DEVELOPMENT TEAMS – A QUALITATIVE ANALYSIS

Complete Research

Baumgart, Ruth, Goethe-University, Frankfurt am Main, Germany, baumgart@wiwi.uni-frankfurt.de

Holten, Roland, Goethe-University, Frankfurt am Main, Germany, holten@wiwi.uni-frankfurt.de

Hummel, Markus, Goethe-University, Frankfurt am Main, Germany, hummel@wiwi.uni-frankfurt.de

Abstract

Software development teams must be able to react rapidly to changing customer requirements. Therefore, agile software development methodologies have evolved in the last decades. Interpersonal and social skills, which are influenced by personality traits, are of fundamental importance for a successful agile software development. While the significance of the human factor is widely acknowledged, scant studies investigate the impact of personality factors on software development success and those few studies report contradictory results. Hence, we conducted interviews with eleven Scrum team members from seven different companies to investigate which personal characteristics are important for agile software development success. We use the five-factor model as a theoretical basis for our investigation, more specifically, we have applied the facets developed by Costa and McCrae. This more detailed approach enables us to give an explanation of the conflicting outcomes of prior investigations. Our study contributes to existing research by suggesting that the most important facets for single Developers are altruism, compliance, tender-mindedness, dutifulness and openness to values. The Scrum Master needs tender-mindedness, assertiveness, dutifulness, achievement striving and stability. In contrast, straightforwardness, compliance, modesty, order and assertiveness are important for the Product Owner.

Keywords: Agile Software Development, Scrum, Personality traits, five-factor model.

1 Introduction

Competition and the unpredictability of economic change force information system development (ISD) to react rapidly to shifting customer requirements during the development process (Balijepally et al., 2006). Therefore, agile software development methodologies such as Scrum, Extreme Programming and Lean Software Development have evolved in the last decades (Lee and Xia, 2010). In contrast to traditional software development, which tries to determine the complete specification at the beginning of the project, the agile software development process facilitates later requirement changes through incremental development in short iterations (Balijepally et al., 2006). Agility in this context means the ability to react efficiently and effectively to requirement changes (Lee and Xia, 2010).

However, the implementation of an agile approach is not a guarantee for success. Between 2003 and 2012, approximately 6% of agile software projects failed and 48% were categorised as challenging (Standish Group International, 2013). It turned out that the human factor is of fundamental importance for the success or failure of agile software development (Acuña et al., 2008). This is especially caused by the team oriented development and reliance on individuals and interaction rather than predefined processes, which demands additional social and interpersonal skills from software developers (Bali-

jepally et al., 2006). Van der Linden et al. (2014) found a significant relationship between personality and social skills. Furthermore, Chandra Misra et al. (2009) detected a significant relationship between personal characteristics and success in agile software development and Ones et al. (1993) found out that integrity tests, which include the FFM, predict supervisor's assessment of job performance in a variety of settings. While the significance of the personal characteristic is widely acknowledged, scant studies investigate the impact of personality factors on agile software development success and those few studies report contradictory results (Acuña et al., 2015; Acuña et al., 2009; Balijepally et al., 2006; Salleh et al., 2014). The goal of this research therefore is to explore the personality traits of Scrum team members, differentiated by Scrum roles, which are important for successful (in regards to working atmosphere, software quality and goal achievement) software development in more detail. Earlier work in this area was based on the five-factor model (FFM) of personality, which distinguishes between the dimensions openness to experience, conscientiousness, extraversion, agreeableness and neuroticism.

Openness to experience measures the tendencies for inquisitiveness and creativity (Balijepally et al., 2006). Conscientiousness, in contrast, describes the extent of organization, commitment and persistence shown by individuals. People scoring highly in the factor of extraversion tend to be disclosed and enjoy the interaction with others. Agreeableness is the extent of friendliness as well as trust displayed by individuals and neuroticism represents the tendency towards experiencing negative emotions. While these factors represent broad but also abstract qualities, our research examines personality characteristics in more detail by applying 30 facets suggested by Costa and McCrae (1995), which are a more precise specification of the FFM. The analysis on a more detailed level is to prefer if there is a wide scatter among facet scores because in these cases the interpretation of factors, which are merely the aggregation of the facets, could be misleading (Costa and McCrae, 1995). Conflicting results of prior studies could be due to high relevance of some facets combined with low relevance of others. Since FFM averages facet scores, persons with high scores in relevant facets and low scores in irrelevant facets could have the same factor score like persons with high scores in irrelevant facets and low scores in relevant facets. As shown by Lynch (1999) external validity is strictly limited, when the analysis is on a too general level. A more detailed analysis will reveal which facets are important for agile software development and which are not. Through a more precise differentiation, our study suggests an explanation of the conflicting results of previous studies. Furthermore, we investigate the diverse attributes, which are important for different Scrum roles. To the best of our knowledge, this differentiation has also not been examined in previous investigations.

This leads to our central research questions: *Which facets of the FFM, differentiated by Scrum roles, are important for a successful Scrum team and why this is the case?*

In order to address this question, we conducted semi-structured interviews with 11 Scrum team members from seven different companies and analysed the results using the pragmatic analysis method developed by Mühlfeld et al. (1981). Qualitative methods are useful where social factors are involved, as this is the case in the present study (Seaman, 1999).

The remainder of the paper is structured as follows. In the next section, we provide the theoretical background and related work in the field of agile software development and FFM. After that, we describe our research methodology. Next, we present our findings and discuss the results as well as the limitations of our investigation. Finally, we give a short conclusion.

2 Theoretical Background and Related Work

2.1 Agile Software Development

Agility is “the continual readiness of an ISD method to rapidly or inherently create change, proactively or reactively embrace change, and learn from change while contributing to perceived customer value (economy, quality, and simplicity), through its collective components and relationships with its environment.” (Conboy, 2009)

Agility in the ISD domain is an umbrella term for different methods like Scrum, Extreme Programming, Crystal, FDD, DSDS and Adaptive Software development (Hoda et al., 2011). We will focus on Scrum because it is the most widely used agile method (Barlow et al., 2010). It distinguishes between three different Scrum roles, namely the Product Owner, the Developer and the Scrum Master (Schwaber, 2004). The Product Owner represents the interests of the stakeholders and is responsible for the return of investment. The Developer creates the functionalities and the Scrum Master ensures that everyone follows the Scrum process. Thereby, the Scrum process consists of different meeting practices, which are presented in Table 1.

Meeting	Description
Sprint Planning meeting	The Developers and the Product Owner decide which requirements will be completed in the next sprint.
Daily Scrum meeting	A daily meeting, which lasts 15 minutes. In this time interval, every Developer has to report the respective status quo.
Sprint Review meeting	In the Sprint Review meeting, the Developers present the functionalities developed during the sprint to the Product Owner and stakeholders.
Sprint Retrospective meeting	The Developers review the process to make the next sprint more effective.

Table 1. Scrum meetings (Schwaber 2004)

2.2 Facets of the five-factor model of personality

Despite the differences between the diverse methods, all of them rely on the Manifesto for Agile Software development, which emphasises the significance of motivated individuals and the superiority of “individuals and interaction over processes and tools” (Beck et al., 2001).

Although the meaningfulness of the human factor in agile software development has been widely accepted, we only found few studies exploring personal characteristics of agile software development team members. And these investigations used the FFM, which has broad but also abstract qualities. To arrive at a more precise result, we use in the following the facets developed by Costa and McCrae (1995), which subdivide the five factors into more detailed facets. Table 2 gives a description of the five factors and their facets. This differentiation intends to resolve inconsistent results of prior research.

Balijepally et al. (2006), for example, propose a theory about useful personality profiles for successful agile software development teams based on the FFM. They derived their theory through a literature review of prior management and group research which investigated teams in a non software development context. The factors conscientiousness, agreeableness and extraversion were seen as important for team success. In the field of openness to experience and neuroticism, they found contradictory implications. For example, while Barrick et al. (1998) found a negative relationship between a variability in emotional stability and team performance, Neuman and Wright (1999) discovered a positive effect of variability in emotional stability and team performance. Similarly, Van Vianen and De Dreu (2001) found no significant relationship between openness to experience and team performance, whereas Neuman and Wright (1999) discovered a significant relationship between openness to experience and team performance.

In contrast, Acuña et al. (2009) detected, in an agile software development context, no significant result for openness to experience and neuroticism but a positive correlation between agreeableness and conscientiousness with job satisfaction and a significant relationship between extraversion and software quality. In a recent meta-analysis Acuña et al. (2015) found no significant relationship between conscientiousness and job satisfaction.

For pair programming, which is an agile software development technique, Salleh et al. (2014) found no significant effect of neuroticism and conscientiousness on team performance but a significant effect

of openness to experience on performance. In addition, other studies found out, that a diversity of personality attributes lead to better performance (Choi et al., 2008; Sfetos et al., 2009).

To reveal possible reasons behind these contradictory findings we analyze personality traits on the more detailed level of facets.

FFM	Facets	Description
Openness to experience	Openness in Fantasy Openness in Aesthetics Openness to Feeling Openness to Action Openness to Ideas Openness to Values	Vivid imagination, tendency to daydreaming Sensitivity to art and beauty Experiencing and valuing own feelings strongly Willing to try new dish etc. Willing to think of different possibilities Liberal in values, not everything is right for everyone
Conscientiousness	Competence Order Dutifulness Achievement Striving Self-Discipline Deliberation	See themselves as competent Organised, efficient in work Scrupulously adhering to their moral precepts Pursuing excellence in their duties Able to accomplish their goals Making plans in advance
Extraversion	Warmth Gregariousness Assertiveness Activity Excitement Seeking Positive Emotion	Friendly style of personal interaction Desire to be with other people, sociability Natural leaders, making up their own minds Like to keep busy Search for excitement like fast cars etc. Feeling of joy and delight
Agreeableness	Trust Straightforwardness Altruism Compliance Modesty Tender-mindedness	Rarely suspecting hidden intents Trustworthiness and candour Selflessness, desire to help others Willing to compromise and get along with others Humble, meek, deferring to others Sentimentality and charity
Neuroticism	Anxiety Angry Hostility Depression Self-Consciousness Impulsiveness Vulnerability	Fearful, nervous, tense Angry, irritable, ill tempered Sadness, hopelessness Emotion of shame Little self-control Inability to deal with stress

Table 2. Differentiation of the Domains in Facets (Costa and McCrae, 1995)

3 Research Approach

We conducted semi structured interviews in Germany with eleven Scrum team members to arrive at a deeper understanding of the facets leading to a successful software development. Qualitative research methods are useful in studies that include social factors' (Seaman, 1999) and well-accepted in information system research studies (Sarker et al., 2013). They allow investigators to obtain a deep understanding of the phenomena in their socially embedded context and to answer "how" and "why" questions (Myers, 2013).

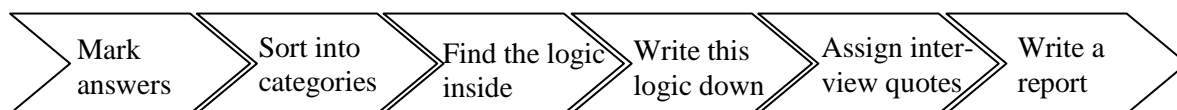
In order to not influence the participants we developed a interview guideline, which does not provide the thirty facets developed by (Costa and McCrae, 1995); instead, we used open questions about the working atmosphere, the fluctuation and the performance of the team and asked which influences are responsible for this. Following that, we enquired which personality traits the single Developers, the Scrum Master and the Product Owner should have and how they influence team success. The assignment of the mentioned characteristics to the thirty facets followed during the analysis phase. The interviews lasted 30 to 70 minutes. Some of the participants were questioned repeatedly to corroborate findings of consecutive interviews. All interviews focused on practical experience with the agile software development approach. In the first company, we conducted five interviews with five different persons. After theoretical saturation, we continued with six further interviews in six different companies to test whether the initial findings are also applicable to other companies. The number of employees, which is a general criterion for the size of a company, varies from 5 to 280.000. All interviews were recorded and transcribed as well as translated from German into English. Because the research question covers the entire Scrum team, the sampling contains all Scrum roles. Table 3 provides background information and demographics on the interviewees.

	Industry	Firm size	Team size	Agile Process	Team roles of interviewees	Highest degree	Experience in years
I1	Consulting and development	Large	6	Scrum	Developer	Bachelor	2
I2	Consulting and development	Large	6	Scrum	Developer	Master	4
I3	Consulting and development	Large	5	Scrum	Project leader	Master	1
I4	Consulting and development	Large	5	Scrum and XP	Developer	Master	2
I5	Consulting and development	Large	5	Scrum and XP	Scrum Master	Master	4
I6	Finance	Large	7	Scrum	Developer	Bachelor	4
I7	Consulting and development	Large	8	Scrum	Lead developer	Bachelor	1
I8	Consulting and development	Large	5	Scrum	Developer	Master	3
I9	Consulting and development	Small	2-3	Scrum	Diverse roles in different projects	Master	4
I10	Finance	Large	5	Scrum	Project leader	Professor	2
I11	Manufacturing	Large	7	Scrum	Product Owner	Doctor	2

Table 3. Characteristics of the teams and companies

The interviews were analysed by one author using Mühlfeld's pragmatic analysis method, a six step analysing procedure, which enables the implementation of a predefined code-scheme (Mühlfeld et al., 1981). We employed this method rather than open coding because we wanted to use the facets developed by Costa and McCrae (1995) to obtain a subdivided FFM so that we receive more details but also comparableness with previous studies, which used the original FFM. Furthermore, this method enables an economic and structured proceeding. Figure 1 presents the steps of the pragmatic analysis method. In the first step, every text passage constituting an answer to the interview guideline was marked. After that, the selected text was assigned to the thirty facets developed by Costa and McCrae (1995). In order to allocate the interview passages to these categories, the software MAXQDA was used. In a next step, the logic between the single information was analyzed by comparing different text passages and answers from different interviewees. In this step we investigated why and how different facets are

relevant for team success and whether the different interviewees are align with each other. Subsequently a text was written to explain the received logic and interview passages were assigned to it. In the last step we wrote a report of this analysis which presents the results of the investigation.



4 Figure 1. Mühlfeld’s pragmatic analysis method (Mühlfeld et al., 1981)Results

In the following, we present the results for each factor of the FFM differentiated by the facets defined by Costa and McCrae (1995) (see Table 2). For our analysis, we distinguish between the three Scrum roles Developer, Scrum Master and Product Owner.

4.1 Openness to experience

Openness to experience is crucial for every Developer. This factor is mentioned seven times. The most relevant facet is *openness to values* followed by *openness to ideas*.

Openness to values is fundamental for ensuring that every Developer accepts unequal norms and values of different people. This tolerance facilitates an unobstructed communication, which is important because communication in the agile software development is seen as the “most efficient and most effective method of conveying information to and within a development team” (Beck et al. 2001)

“For an agile development, regardless of whether you use FDD or XP or any other methods, you have to communicate a lot, you have to be open to the counterpart and you have to be relatively open-minded, if you want it to work.” (I2, Developer)

Openness to ideas is also fundamental because customer requirements can change after every sprint in the agile development process. This leads to unpredictability and insecurity regarding the project. In addition, it is not certain that the programmed code will be integrated into the end-product. Consequently, the Developer should have openness to ideas.

“I think that openness to changes, especially openness to plan changes, is something which you need especially for agile software development. [...]. Every time you go beyond two sprints [...], it is difficult and impossible to say, what we do, in fact. This is just an attribute of agile software development.” (I11, Product Owner)

Furthermore, *openness to ideas* is important because the single Developers have to refine themselves constantly and deal with new technologies in order to keep up to date and achieve good results.

“It is a positive issue if the members of the [development] team are also flexible to the extent that they are willing to learn, willing to deal with new technologies and try to solve problems technically well and not just apply the obvious strategy. They need a little bit of exploratory urge.” (I9, diverse roles)

We could only find the two facets *openness to values* and *openness to ideas* as relevant for the single Developers. No other facet of the FFM dimension openness to experience (*openness in fantasy, openness in aesthetics, openness to action* and *openness to feeling*) was mentioned as important for the Developers. No single facet of this factor was seen as fundamental for the Scrum Master and the Product Owner.

4.2 Conscientiousness

For the factor conscientiousness, *dutifulness* and *achievement striving* were mentioned as being important for the Developers. Every Developer should be responsible for the end product and be evaluated for overall results.

“There are a lot of good Developers who know exactly whose fault it is if a problem occurs [...]. These [kind of Developers] are not as useful as those who [...] commit themselves to being overall successful.” (I3, project leader)

Furthermore, *achievement striving* is fundamental for agile software development as well as traditional development. Developers have to be motivated to achieve their goals.

“One of the most important attributes is motivation.” (I6, Developer)

For the Scrum Master, conscientiousness is seen as the most relevant factor, five of six facets were seen as relevant for him/her. Thereby, *dutifulness* and *achievement striving* are of special importance, closely followed by *order* and *self-discipline*. The Scrum Master has to be *dutiful* because he/she supervises the development progress as well as the work of the Developers. In addition, he/she has to motivate the development team and needs to ensure that the Developers achieve the objectives in every sprint.

“He has the overall responsibility for the things that occurred in the sprints to some extent and he has to motivate.” (I11, Product Owner)

Good *achievement striving* is also mentioned as a basic facet for the Scrum Master. He needs to have the intention to achieve process improvements and to facilitate the work of the Developers.

“He needs the will to push something through. This is important. Someone who sits down and says I can’t change anything because we are in a big enterprise will not shift great things.” (I4, developer)

The Scrum Master has to be *orderly* because he/she has to undertake meetings systematically and note important problems.

“If you have a well structured man, who notices everything [...] then it is the best you can have.” (I2, Developer)

Self-discipline is also needed to facilitate a strict procedure. For example, the Scrum Master is accountable for the time restriction set for the Daily Scrum meetings. Because these meetings should only last fifteen minutes, he/she has to take care that every Developer delivers his status report during this time interval.

“There are rules e.g. for the Daily Scrum, which define for how long everyone can report his status and so on. He has to ensure that everyone follows this restriction. He is responsible for the execution and he should do this with discipline.” (I1, Developer)

For the Product Owner, conscientiousness is not seen as important as it is for the Scrum Master. The only facet mentioned is *order*. He has to be orderly and structured to be able to manage the Backlog. Thereby, it is important that priorities are clear and changes do not occur too frequently. Changes should also be reproducible because otherwise this can lead to a worse working atmosphere.

“[The Product Owner] should be structured in his handling of the administration of the Backlogs.” (I2, Developer)

4.3 Extraversion

The factor extraversion is essential for every Developer. Especially the facets *gregariousness* and *assertiveness* were mentioned by our interviewees, followed by *activity* and *positive emotion*.

The Developers have to be able to show *gregariousness* because every Developer needs to have the ability to communicate since they cannot write a software program in isolation, as used to be the case

in the traditional development method; rather, they have to talk with each other according to agile development.

“Yes, they need soft skills, the ability to accept criticism and the ability to communicate, these things are more important than in a classic [development] team.” (I10, project leader)

However, too much of an extrovert character trait in single Developers can lead to over-communication in the Daily Scrum meetings. This can have the consequence that time constraints cannot be met or that other Developers do not have the time to talk about important topics and problems. Therefore, a moderate degree of extraversion is recommendable.

“As is often the case, it’s the dose that makes the poison. And when you have a jumping jack [...] who talks too much, he will be harmful to the others who are not as extrovert [...], and are not able to get a word in edgeways and for this reason topics will not be discussed and therefore problems will not be solved or other things will not be implemented.” (I2, Developer)

Assertiveness is an important attribute for the Developers provided the assertiveness is equally distributed. If single members of the development team subordinate themselves too much or others are too dominant, the collective work is impeded. It is important that the single Developers follow the same level of assertiveness because this ensures good teamwork and a good working atmosphere.

“It is very important that everyone is perceived as equal, feels valued and sees himself equally valued, too.” (I9, diverse roles)

Single Developers that are too dominant could obstruct the objective of achieving a fair task allocation because they could enforce their goals too strongly against those of more introvert development team members. This danger exists particularly for self-organised teams because the Developers allocate the tasks themselves.

“In an agile approach [...] the [development] team makes arrangements among themselves as to who completes which task at what time. I see there a certain danger, that for example especially extrovert [development] team members [...] assert themselves and perhaps the calmer ones are dominated.” (I1, Developer)

In addition, having an excessively dominant Developer on board can lead to decisions that are not made by the whole development team but by one person only. This is contradictory to the agile principle which demands that the development team should be self-organised.

“If single persons are too controlling, too dominant, then this stands in conflict with the idea of Scrum because Scrum builds on self-organised teams and if it works in a way in which one person decides on behalf of the team, you have missed the topic.” (I3, project leader)

The activity of the single Developers is also a relevant facet for agile software development. It works twofold: on the one side, active Developers are important for the interactive Scrum process; on the other side, Scrum and the accompanied team could help motivate introvert Developers to become more active.

“I think that an agile approach or the Scrum approach should also motivate [introvert] people to play a part in the team, just because the whole team takes on a task for one sprint. This should also encourage a silent, introverted type to contribute something.” (I2, Developer)

Positive emotions are profitable for a good working atmosphere and a good collaboration in the development team.

“Yes, well if the people are sociable and understand humour and can handle criticism then this is surely a benefit.” (I11, Product Owner)

For the Scrum Master, extraversion is one of the most important factors. The crucial facets are, in descending importance, *assertiveness* and *gregariousness*.

Assertiveness is a determining facet for the Scrum Master because he/she has to ensure the wellbeing and success of the team. In addition, it is important that he exudes authority inside the team and outside.

“He should be competent, accredited by his colleagues inside the team but also outside the team. He should be ready to stand up for his opinion as much as he sees a benefit for the team or for the success of the team.” (I11, Product Owner)

Furthermore, the Scrum Master needs interpersonal skills and *gregariousness* because he has to interact with the Developers as well as the Product Owner.

“The Scrum Master [...] has to be communicative. He also needs leadership skills and [...] he has to listen [...] to improve communication.” (I8, developer)

Also, the Product Owner needs self-confidence and *assertiveness*. These attributes are relevant for the interaction with the Developers and the customer because he/she has to be demanding in regards to the Developers to enforce customer requirements. In contrast, he/she has to be able to talk the customer out of some ideas if they are uneconomic.

“He should be self-confident when facing the customer and self-confident when facing the Developer, also demanding. And what is terrible is someone who can't say no and absorbs everything the customer wants. He has to stand up for reasonable issues and talk the customer out of not so meaningful things.” (I2, Developer)

4.4 Agreeableness

Agreeableness is the most important factor for every Developer. The relevant facets in diminishing importance are *altruism, compliance, straightforwardness, tender-mindedness*.

Altruism is the most crucial attribute of the 30 facets. The qualities which are important for this facet are mutual assistance, fair allocation of tasks and knowledge transfer. A mutual endorsement is central because the collaboration stands in the foreground of agile software development. Egoistic Developers, who sabotage tasks to perform better than their team members are, in contrast, impedimental.

“Or that you have no interpersonal skills, i.e. that you don't want to work with the others, rather do everything by yourselves or try the whole time to sabotage projects or task, for a better status. Well, egoism is a really bad attribute in an agile team.” (I2, Developer)

Altruism is fundamental because of the self-organisation of the development team and free task allocation. The single Developers have to be willing to also take on displeasing tasks because an inequitable allocation of tasks leads to a worse working atmosphere and has a negative effect on the collaboration.

“On the other side, what is also important, is not everyone being too egoistical and saying I just do the things that are interesting, because even in a self-organised team you have to do displeasing tasks, which are part of every project.[...] You have to take care, that this does not always hit the same person.” (I1, Developer)

Furthermore, it is important that the single Developers share their know-how and are cooperative because in agile software development, human interaction is an important condition for agile software development (Beck et al., 2001).

“[...] To encapsulate your know-how from others; sort of, not helping others for tactical reasons or not wanting to help too immediately, rather after a lot of asking, this is what you can't do in the Scrum process, such things will be eliminated through it.” (I6, Developer)

Compliance is also important for every Developer. Especially new team members have to adapt to the working habits of the team and should be willing to learn as well as not challenge every decision. Furthermore, it is important that there is some give-and-take between the members of the team.

“If someone new joins the team, he is at first a bit of a stranger and not really well versed. If this one is willing to learn and adapt and brings in own ideas, but nevertheless accepts that

there is a team that works and they do it this way and this person is not challenging everything but tries to go along with the team, then I think it will be work.” (I11, Product Owner)

Tender-mindedness is also important for the Developers because good collaboration needs team members with interpersonal skills.

“[It is important that] you mix well with the team and that you have a good interpersonal relationship.” (I2, developer)

The important facets for the Scrum Master in order of decreasing importance are *tender-mindedness*, *altruism* and *modesty*.

Altruism is crucial for the Scrum Master because it is good for the motivation of the team if the Scrum Master cares not only for functional things but also looks after the team’s overall wellbeing.

This also means that he maintains a good atmosphere and creates a good working environment as well as eliminates interpersonal disturbances.

“Well, understanding for everyone, a little bit of an initiator who cares for functional things as well as social things and provides coffee and looks that there are no draughty windows and the people feel well.”(I10, project leader)

Furthermore, he/she has to be *modest*. He/she should hold back from obstructing the Developers from working.

“Well, not detain the development team from work, this means he should not be obtrusive, rather reserved and moderate quietly.” (I2, Developer)

The participants see *tender-mindedness* as important for the Scrum Master because he/she has to interact more with people than technology. He/she also should have empathy with the development team and an understanding of the customer.

“You have to treat people right [...], because as a Scrum Master you have to interact more with people than with technology.” (I5, Scrum Master)

“I would say [the Scrum Master needs] an overview of the project and its content, an understanding of the customer and of course empathy for the staff and the members of the team.” (I10, project leader)

For the Product Owner, the facets *trust*, *compliance*, *modesty* and *tender-mindedness* are relevant.

Trust was mentioned as the most important attribute for the Product Owner because the Developers knows best where redundant complexity may be cut and where the complexity of the requirements is too high and not economic.

“He has to trust the development team, if they say this is not going to work then he has to accept it.” (I2, Developer)

In order to prevent conflicts the Product Owner has to have *compliance*. It is important that the Product Owner has a similar understanding of the product’s quality level as the development team, although he has to keep the customer’s perspective in mind as well. Customers generally want as much functionalities as possible and do not honour refactoring and documentation. But these are basic requirements for quality. Because the capacity of the development team is limited the Product Owner has to decide to what extent refactoring and documentation is more important than more features for the final product.

“[The Product Owner should have] a similar understanding as the [development] team. The Product Owner sees the product and the outside perspective as well as the customer’s use and his requirements. [...] Finally, the functional features stand in competition with quality, with the internal quality, feasibility, sustainability of the product. It is like weighing up the short against the long-term benefit.” (I11, Product Owner)

The Product Owner has to be *modest*. It is important that he does not adopt a leader role so that the development team is able to remain self-organised and take over responsibility because a self-organised development team leads to a better architecture and improved drafts (Beck et al. 2001).

“A traditional project leader [who adopts the role of the Product Owner] has to step back from his direct leader role so that the [development] team can organise itself and takes on responsibility for it.” (I3, project leader)

Furthermore, he/she should have *tender-mindedness* and the ability to work in a team. It is important that he/she does not overcharge the Developers.

“He should cooperate with the [development] team. There is no use if he demands too much from the team. He also needs social ability to work with the team.” (I5, Scrum Master)

4.5 Neuroticism

Every Developer should have low *vulnerability*. This is an important facet in order to insure a constructive teamwork. If there is considerable pressure, conflicts are hard to avoid, but it is important that the Developers are not resentful to ensure that cooperation is not affected.

“Well, there is no gain in being permanently in trouble with a person but having to work with this person. You have to be professional.” (I7, Lead Developer)

Also, the Scrum Master has to have low *vulnerability*. He has to be stable and stress resistant. These attributes are important because he has to act as the mediator between the Developers and the Product Owner. Therefore, he has to find solutions if the Developers cannot implement the desired requirements of the Product Owner in time.

“[He/she has to be] stress resistant because he/she acts as the buffer between two glaring sides. On the one side, there is the Product Owner, who wants everything done at once, and on the other side, there is the Developer, who says that something doesn’t work the way he hoped or intended and he has to bring both parties together to find a solution.” (I7, Lead Developer)

In addition, the Product Owner is the negotiator between customer, Developers and Scrum Master. He has to have a low *vulnerability* when trying to find a compromise between the opposed interests.

“Always the mediator – you can be caught between two stools, you need strong nerves and you have to stand above such things, otherwise you go crazy.” (I10, project leader)

Table 4 summarizes the relevant facets of the big five factors and shows how often they are mentioned for every Scrum role.

Attribute/frequency	Developers	Scrum Master	Product Owner	Total
Openness to Experience	7	0	0	7
1. Openness to Ideas	3	0	0	3
2. Openness to Values	4	0	0	4
Conscientiousness	7	10	3	20
1. Order	0	1	3	4
2. Dutifulness	5	4	0	9
3. Achievement Striving	2	4	0	6
4. Self-Discipline	0	1	0	1
5. Deliberation	0	0	0	0
Extraversion	11	8	2	21
1. Gregariousness	4	2	0	6
2. Assertiveness	4	6	2	12

3. Activity	2	0	0	2
4. Positive Emotion	1	0	0	1
Agreeableness	26	8	15	49
1. Trust	0	0	5	5
2. Straightforwardness	5	0	0	5
3. Altruism	9	1	0	10
4. Compliance	7	0	4	11
5. Modesty	0	1	4	5
6. Tender-mindedness	5	6	2	13
Neuroticism	1	3	1	6
Anxiety	1	0	0	1
Vulnerability	1	3	1	5

Table 4. *Relevance of the single facets*

5 Discussion

Our study investigates the important personality factors of the different roles in the Scrum team which lead to a successful team performance. We found that all five factors of the FFM are to some extent relevant for Developers. Agreeableness is the primary factor for the Developers because they need to possess altruism, compliance, straightforwardness and tender-mindedness to ensure good teamwork, which is of fundamental importance for agile software development. In terms of the factor extraversion, the facets gregariousness, assertiveness, activity and positive emotion are relevant. Only openness to values and openness to ideas were seen as important for the factor openness to experience. The fundamental facets for conscientiousness are dutifulness and achievement striving. For the factor neuroticism merely the facets anxiety and vulnerability have a negative influence on the development team performance.

While agreeableness is the most important factor for the Developers, conscientiousness is the most eminent factor for the Scrum Master because he/she has to possess order, dutifulness, achievement striving and self-discipline. The basic attribute for the Product Owner is agreeableness and, more specifically the facets trust, compliance, modesty and tender-mindedness. In contrast, the factor openness to experience was not mentioned as important for the Scrum Master or the Product Owner. This could be the case because the Scrum Master has to be in line with the Scrum rules and the Product Owner should act on behalf of the stake holders. Therefore, scant openness to experience but rather a strict alignment is beneficial. Our results show a wide scatter among the importance of the different facets. In the case of the Developers, only two of six facets for the factors openness to experience, neuroticism and conscientiousness are seen as relevant. This variation of high and low relevance of single facets of one factor impede the interpretation if the analysis only use factors because they reveal only an aggregation of the facets and this can lead to wrong conclusions. For example, a relatively high score in the factor openness to experience may also appear if a person is not willing to think of different possibilities (openness to ideas) or not liberal in values (openness to values), which are important facets for good agile team work, but has a high tendency to daydreaming (openness in fantasy), values own feelings strongly (openness to feelings) and is very sensitive to art (openness in aesthetics), which are dispensable attributes. In this case, a high score in the factor openness to experience would not lead to better team performance. In contrast, a low value for the factor openness to experience may result from high values of openness to ideas and openness to values but low values in the four residual facets of openness. In this case a low value in the total factor openness to experience would lead to better team performance. The same explanation can be given for the factor neuroticism and conscientiousness.

Previous studies (Acuña et al., 2015; Acuña et al., 2009; Balijepally et al., 2006; Salleh et al., 2014) show conflicting results exactly in these factors (openness to experience, neuroticism and conscientiousness) where we found that only two facets are relevant. The wide scatter of the importance of single facets could therefore explain why some investigator found a significant relationship between these factors and team performance while others did not. We, therefore, agree with Costa and McCrae (1995), who suggest that the use of facets is more appropriate if there is a wide scatter between the facets because the use of the more abstract factors in this case can be misleading. Future research should build on these facets to obtain more precise results, for example, by replicating our study in a case study design. Thereby, a larger sample from different roles could be interviewed to get a greater external validity. Furthermore, a measurement of the FFM profile of respondents could be done first to control for the preference of own personality traits. Quantitative examinations, e.g. surveys or experiments, could also strengthen our initial results by deriving and testing hypothesis. Further investigation could also explore if the same facets are relevant for teams in other countries and for distributed teams.

For practice, this research helps companies to build effective Scrum teams from existing IT employees. Therefore, personality tests can be used to find out which of the staff members are most suitable for working in an agile ISD environment. In addition, training could be introduced to improve particular attributes, e.g. yoga can reduce neuroticism in a short period of time (Telles et al., 2012). Furthermore, our results for the Developers are also applicable for other agile software development methods like XP and FDD because the basic values of the agile approach, which are defined in the Agile Manifesto, also exist in other agile methods. But personality traits are not the only characteristics which have to be taken into account. Other attributes like expertise, programming skills and learning abilities (Hannay et al., 2009) must also be considered for a successful agile software development.

This study has several limitations. We interviewed only eleven participants so external validity and generalizability are limited. Nevertheless, we used data triangulation to mitigate this problem. This procedure also diminishes the problem of biased answers because different interviews from different companies were compared. Furthermore, only one author coded the interviews. In addition, the interviews were conducted in Germany and therefore the validity for other countries is also constricted because of cultural differences. To minimize this problem, we chose companies for our study which have an international standing and experiences with offshore team members from other countries.

In sum, our findings contribute to the agile software team research by providing a qualitative analysis of the personality traits, which are important for a successful Scrum team. By applying the 30 facets suggested by Costa and McCrae (1995) this study gives an explanation of contradictory results of previous investigations, which use the more general FFM. Future research in the agile software development context could build on our findings to achieve more specific results.

Acknowledgements

The German Research Foundation (DFG) funded parts of this study under record no. HO 2196/4-1.

References

- Acuña, S., M. Gómez, J. E. Hannay, N. Juristo and D. Pfahl (2015). "Are team personality and climate related to satisfaction and software quality? Aggregating results from a twice replicated experiment." *Information and Software Technology* 57, 141-156.
- Acuña, S., M. Gómez and N. Juristo (2008). "Towards understanding the relationship between team climate and software quality—a quasi-experimental study." *Empirical Software Engineering* 13 (4), 401-434.

- Acuña, S. T., M. Gómez and N. Juristo (2009). "How do personality, team processes and task characteristics relate to job satisfaction and software quality?" *Information and Software Technology* 51 (3), 627-639.
- Balijepally, V., R. Mahapatra and S. Nerur (2006). "Assessing Personality Profiles of Software Developers in Agile Development Teams." *Communications of AIS* 18 (4), 2-40.
- Barlow, J. B., J. S. Giboney, M. J. Keith, D. W. Wilson, R. M. Schuetzler, P. B. Lowry and A. Vance (2010). "Overview and Guidance on Agile Development in Large Organizations." *Communications of AIS* (2010) 29, 25-44.
- Barrick, M. R., G. L. Stewart, J. M. Neubert and M. K. Mount (1998). "Relating Member Ability and Personality to Work-Team Processes and Team Effectiveness." *Journal of Applied Psychology* 83 (3), 377-391.
- Beck, K., M. Beedle, A. van 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 and D. Thomas (2001). *Agile Manifesto*. URL: <http://www.agilemanifesto.org/> (visited on 11/28/2014).
- Chandra Misra, S., V. Kumar and U. Kumar (2009). "Identifying some important success factors in adopting agile software development practices." *The Journal of Systems and Software* 82, 1869-1890.
- Choi, K. S., F. P. Deek and I. Im (2008). "Exploring the underlying aspects of pair programming: The impact of personality." *Information and Software Technology* 50 (11), 1114-1126.
- Conboy, K. (2009). "Agility from First Principles: Reconstructing the Concept of Agility in Information Systems Development." *Information Systems Research* 20 (3), 329-354.
- Costa, P. T. and R. R. McCrae (1995). "Domains and Facets: Hierarchical Personality Assessment Using the Revised NEO Personality Inventory." *Journal of Personality Assessment* 64 (1), 21-50.
- Hannay, J. E., E. Arisholm, H. Engvik and D. I. Sjöberg (2009). "Effects of Personality on Pair Programming." *Software Engineering* 36 (1), 61 - 80.
- Hoda, R., J. Noble and S. Marshall (2011). "The impact of inadequate customer collaboration on self-organizing Agile teams." *Information & Software Technology* 53 (5), 521-534.
- Lee, G. and W. Xia (2010). "Toward Agile: An Integrated Analysis of Quantitative and Qualitative Field Data." *MIS Quarterly* 34 (1), 87-114.
- Lynch, J. J. G. (1999). "Theory and External Validity." *Journal of the Academy of Marketing Science* 27 (3), 367-376.
- Mühlfeld, C., P. Windolf, N. Lampert and H. Krüger (1981). "Auswertungsprobleme offener Interviews." *Soziale Welt* 32 (3), 325-352.
- Myers, M. (2013). *Qualitative Research in Business and Management*. 2nd Edition. London: SAGE Publications Ltd.
- Neuman, G. A. and J. Wright (1999). "Team Effectiveness: Beyond Skills and Cognitive Ability." *Journal of Applied Psychology* 84 (3), 376-389.
- Ones, D. S., C. Viswesvaran, and F. L. Schmidt (1993). "Comprehensive meta-analysis of integrity test validation: Findings and implications for personnel selection and theories of job performance." *Journal of Applied Psychology* 78, 679-703.
- Salleh, N., E. Mendes and J. Grundy (2014). "Investigating the effects of personality traits on pair programming in a higher education setting through a family of experiments." *Empir Software Eng* 19, 714 - 752.
- Sarker, S., X. Xiao and T. Beaulieu (2013). "Qualitative Studies in Information Systems: A Critical Review and Some Guiding Principles." *MIS Quarterly* 37 (4), iii-xviii.
- Schwaber, K. (2004). *Agile Project Management with Scrum*. Washington: Microsoft Press.
- Seaman, C. (1999). "Qualitative Methods in Empirical Studies of Software Engineering." *IEEE Transaction on Software Engineering* 25 (4), 557-572.
- Sfetsos, P., I. Stamelos, L. Angelis and I. Deligiannis (2009). "An experimental investigation of personality types impact on pair effectiveness in pair programming." *Empir Software Eng* 14, 187-226.

Standish Group International (2013). *The Chaos Manifesto*. URL:

<http://www.versionone.com/assets/img/files/CHAOSManifesto2013.pdf/> (visited on 11/28/2014).

Telles, S., K. V. Naveen, N. Kumar, A. Balkrishna, and Acharya. (2012). "The effect of yoga on neuroticism in an Indian population varies with socio-demographic factors." *Journal of Cultural Diversity* (19) 4, 118-123.

Van der Linden, D., J. K. Oostrom, M. Born, H. T. Van der Molen and A. Serlie. (2014). "Knowing What to Do in Social Situations. The General Factor of Personality and Performance on Situational Judgment Tests." *Journal of Personnel Psychology* 13 (2), 107 - 115.

Van Vianen, A. E. M., and C. K. W. De Dreu. (2001). "Personality in Teams: Its Relationship to Social Cohesion, Task Cohesion, and Team Performance." *European Journal of Work & Organizational Psychology* 10 (2), 97-120.