Towards Better Understanding of Agile Teams through Behavior Change Models

Kashumi Madampe
kashumi.madampe@monash.edu
Monash University
Melbourne, Australia

Rashina Hoda
rashina.hoda@monash.edu
Monash University
Melbourne, Australia

John Grundy
john.grundy@monash.edu
Monash University
Melbourne, Australia

ABSTRACT

Agile software development welcomes changes throughout software development – but this implies that agile teams face several dilemmas. When to respond to a change; how to respond; how to manage the change. Our current understanding and support for agile teams during such change management is very limited. Psychological behavioral change models can be used to better understand the behavior of agile teams. Combining our understanding of agile teams and practices with a review of behavior change models, we propose several avenues for studying behavior and behavioral changes in agile teams. Our proposed interdisciplinary approach provides a much needed avenue to acknowledge and address the psychological and behavioral aspects of the humans central to the software engineering process, ultimately assisting with their well-being and productivity.

CCS CONCEPTS

• **Software and its engineering → Agile software development**; **Programming teams**.

KEYWORDS

agile, software development teams, human behavior, behavior change models

1 INTRODUCTION

Agile software development is popular for its perceived ability to embrace changes throughout the software development process. Agile teams face many changes such as organizational changes, process changes, changes in practices, and requirements changes in part due to stakeholder and market demands, and in part due to the highly iterative nature of the agile approach. But how do individuals and teams cope with the demands these high-paced changes place on them as humans in the software engineering process? We want to better understand the nature of these changes and team responses to them, with a view to ultimately help agile teams better handle change management.

Of the four core values from the agile software development manifesto [4], three (*individuals and interactions over processes and tools, customer collaboration over contract negotiation*, and *responding to change over following a plan*) focus on human behavior. Agile team responses to change can thus be understood through the following, the last being our central focus: roles of the individuals; how they interact within the team; how they interact with customers; and *how they respond to the requirements changes.*

The need to examine and better understand human behavior in software engineering has long been supported [7, 9, 10]. Psychology helps our understanding of how a human being responds in different situations and stimuli. We propose that psychology-derived *Behavior change models* [5] can help better understand human behavior and changes within agile software teams. The recent universal experience in remote working in response to the global COVID-19 pandemic has further highlighted the need to focus on human well-being and handling change in the work context. An international survey on how COVID-19 has impacted software professionals [16], established a clear link between improvement in well-being and improvement in productivity. Studying behavior and behavior changes can help uncover the related to promoting and sustaining both well-being and productivity.

We summarise some key behavior change models that seem well suited to study the behavior of agile teams from a Psychological perspective (section 2). Based on our analysis of fundamental agile guides [4, 6, 17], we propose a number of behavioral aspects that can be studied using behavior change models (section 3). We present preliminary findings from a pilot study (section 4), and proposed directions for agile team behavior research (section 5).

2 BEHAVIOR CHANGE MODELS

A review study [5] on behavior change models found 82 models. We selected the highly cited papers to identify the most prevalent psychological behavior change models. Below we summarise the *stages of change model* – also known as the *transtheoretical model* [15], *social cognitive theory* [3], and the *theory of planned behavior* [1], the leading behavior change models identified. In the next section we provide examples of how each of these are particularly suited to studying aspects of agile team behaviors.

Stages of Change Model/Transtheoretical Model [15] This six stage model is the dominant behavior change model where a person is highly unlikely to change in the early stages but committed to the

change in the later stages. By applying this model, understanding the behavior of the individual and/or improving the behavior of the individual during the process of changing, is possible. As the dominant model, we elaborate its main stages here.

Stage 1 – *precontemplation.* Here acceptance of the behavior is denied. Stage 2 – *contemplation* is where the advantages of the behavior change are understood, but conflicting emotions exist. Stage 3 – *preparation* is where a person starts to make minute changes in behavior and collects information about the change. Stage 4 – *action* is taken by the person to achieve the goal. Stage 5 – *maintenance* is where the new behavior is maintained and reverting to previous behavior is avoided. Stage 6 – *relapses* may occur. These can occur through feeling frustrated, disappointed, and failed.

Social Cognitive Theory [3] This is not a "staged" behavior change model, but instead is a highly applied behavior change theory where three factors are triangulated. The key constructs of *Social Cognitive Theory* are observational learning, reinforcement, self-control, and self-efficacy. In simple terms, the theory states that an individual's learning depends on the person being confident of enacting the behavior and persisting it, the response received after he/she enacts the behavior, and the environment around him/her. An interesting fact is that the individual can be either an agent for change and/or the respondent to change. Importantly, learning is central to this.

Theory of Planned Behavior [1] The *Theory of Planned Behavior* is applied in cases where it is required to predict the intention of an individual to participate in certain behavior. *Theory of planned behavior* is based on six constructs namely, *attitudes, behavioral intention, subjective norms, social norms, perceived power*, and *perceived behavioral control. Attitudes* define the evaluation of the interest towards the behavior. The motivational factors to enact the behavior is defined by the behavior intention. The individual's approval/disapproval of the behavior depending on his/her own beliefs are called as the *subjective norms. Social norms* come to play when the individual's behavior is performed within a group. The team culture is a great factor when it comes to social norms. The required factors to perform the behavior are called as the *perceived power. Perceived control* is the individual's perception of performing the behavior – the individual's perception of easiness/difficulty in performing the behavior at a given time and a given circumstance.

3 AGILE SOFTWARE DEVELOPMENT THROUGH BEHAVIOR CHANGE MODELS

The *internal* stakeholders of an agile software development team are agile coach, product owner, and the development team. The *external* stakeholders of an agile team are the customer, customer representative, and others interacting with the team. The behavior change of an individual can be due to a change in the internal agile environment or external factors impacting the agile environment. For instance, requirements change being a change in the agile environment and the behavior of the individuals affected may change as the change is introduced, being developed, and when delivered. On the other hand, due to agile teams being cross-functional, the behavior of the individual may change. In both cases, behavior change models can be employed to better understand and affect

behavior change. We now propose specific behavioral change aspects in agile teams that can be studied using the behavior change models summarised in the previous section.

3.1 Agile Coach Behavior Changes

How does the behavior of the agile coach change due to the practice of having a rotating agile coach in the team? The agile coach role can be a dedicated or a rotating one. When having a rotating agile coach, the social cognitive theory can be applied to see how the new agile coach's behavior is impacted by the previous agile coach's behavior. The captured learning can be useful for the team. By understanding the impact, a decision can be made on the suitability of having a rotating agile coach within the team.

How to improve the skills of the agile coach? The agile coach's skills can be improved by applying the *stages of change* model. For instance, an agile coach may fail in performing his/her duties such as facilitating agile events. At some point, a stakeholder may accuse the agile coach for not playing the role correctly. As per the stages of change model, the agile coach may deny the acceptance of his/her failure, but later may accept and have their own strategies for improving their skills to perform better. As the stages of change model are well structured, the agile coach can use an improvement plan according to the stages in the model.

The overall behavior of the agile coach can be modelled by constructs in the *theory of planned behavior*, and an early prediction made on how the agile coach's behavior changed by executing the self-improvement plan using the *stages of change model.* Finally, the *theory of planned behavior* can be applied again to see whether the predictions on the coach's behavior have been made as expected.

3.2 Product Owner Behavior Changes

Does the product owner get influenced by customer pressure? The product owner mediates between internal and external stakeholders and thus the product owner's behavior can be influenced by both parties. Sometimes, more pressure can be expected from the customer, and the same pressure can be passed on to the development team by the product owner. This can be understood through *social cognitive theory*. If such behavior is found, then appropriate remedies can be used to minimize passing of customer pressure down to the development team through the product owner.

How to manage the external factors affecting the Product Owner's role? The product owner can also follow a similar skill improvement plan as the agile coach. To mitigate undue influence from external stakeholders, a plan can be defined to manage external factors which harm the agile team performance and applied throughout the project. This will benefit current and future projects.

Theory of Planned Behavior can be particularly useful in understanding the behavior changes of the product owner. Since the product owner deals with both internal and external agile environments, they come across the social norms from both the environments. Since social norms are a key construct in the *theory of planned behavior*, the degree of its involvement in the behavior change of the product owner can be effectively studied.

3.3 Development Team Behavior Changes

How to keep the development team motivated? "If he/she can do, so can I" [14]. For better or worse, this is what an individual does according to *social cognitive theory*. As different changes are introduced to the team, especially requirements changes, some members can become demotivated. By seeing individuals who are neutral or positively responding to requirements changes, demotivated members may benefit. The agile coach should be tactful enough to apply this by observing each individual in the team and subtly directing demotivated members towards motivated members. *Social cognitive theory* can be used to encourage inexperienced team members to work with experienced members to improve skills.

How to establish required behavior within the development team? The *stages of the change model* can be useful for studying and understanding team behavior understanding, both to the individual team members and the agile coach. A plan can be devised to achieve improvements in behaviors. Developers who are not willing to change their behavior can be carefully observed, and the agile coach may set different plans for their improvements. This can be applied for inexperienced members as well.

Typically, agile development teams are cross-functional and self-organizing. Within the team, a culture of its own exists; which includes social norms. Along with the other factors: attitude, behavior intention, subjective norm, perceived power, and perceived control, the team's self-organizing and cross-functional abilities can be determined using the *theory of planned behavior*. For example, a schema can be created by giving weights to each of these constructs and allocating points to individuals against each of these constructs. If expected improvements are not achieved, collaboratively agreed corrective measures can be applied to achieve the desired behaviors.

4 VISION: UNDERSTANDING AGILE TEAM BEHAVIORS IN RESPONSE TO REQUIREMENTS CHANGES

Requirements changes are an integral part of agile projects. We conducted a pilot study on agile team responses to requirements changes [12, 13] with the participation of ten agile practitioners and through applying grounded theory [8] and sentiment analysis [11]. Our preliminary findings highlighted that agile developers react

to requirements changes, both *technically* and *emotionally*. That is, they explicitly make several, diverse technical responses to address requirements changes and incorporate them into the software under development. They also react, implicitly or explicitly, emotionally to different changes, in different ways and at different times. Feedback from agile teams suggests these responses may have a significant impact on team performance and project outcome.

Our larger research vision is shown in Figure 1. We conceptualise a *requirements change* as a change in the agile environment, and *technical* and *emotional* responses which have been explicitly displayed are "human responses" to this change. Since human responses involve human behavior, we explored models from behavior psychology to better understand them. As depicted left in 1, we started by reviewing and developing an understanding of wider psychological theories (rectangle 1), then behavioral psychology (2), followed by behavioral psychological models (3), and then finally, behavior change models (4). We examined technical and emotional responses in *receiving, developing*, and *delivering* requirements changes [12], using three common change models.

Examination of these models made us realize that technical responses and emotions linked to requirements changes over time can be understood with the use of *social cognitive theory* where team members learn from each other over time. For example, some developers in the team may show a technical response by *conducting a feasibility study* [12] when requirements changes are *received*. Given other limitations, such as length of the iteration, others in the team may also learn this technical response and can emulate similar behavior. Some team members may be enthusiastic about *developing* the requirements change [13], while others become frustrated. If this happens, there is an opportunity for the frustrated developers to potentially learn from the enthusiastic ones. In order to encourage this, the agile coach may apply certain strategies from social cognitive theory. This might include techniques for observational learning from other teams/team members; reinforcement of positive benefits of incorporating changes; and feedback from team members to support self-observation and self-efficacy.

The *theory of planned behavior* comes into play when understanding factors affecting each team member's responses to requirements changes. The agile environment can be understood through the key constructs of this model; attitudes, behavioral intention, subjective norms, social norms, perceived power, and perceived behavioral control at the stages of the requirements changes (receiving, developing, delivering). For example, the reasons why the emotional response *no feeling* [13] is shown when the requirements changes are *received* can be identified in terms of the key constructs in the *theory of planned behavior*. Similarly, the reasons why some developers display a technical response of *questioning the product owner* [12] when requirements changes are received, and why the others do not, can be investigated through *theory of planned behavior*.

The *stages of change model* can be highly useful in classifying both technical and emotional responses of the agile team through the stages of receiving the requirements change to developing and delivering it. This classification will help determine the pattern of technical and emotional changes the team displays at different stages. For instance, considering the requirements change stage of receiving, the technical response of *developing the requirements change directly* can be classified into the action stage of the *stages*
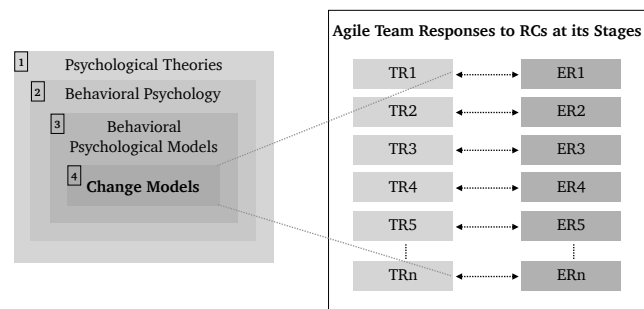
Figure 1: Our Vision (1-4: Steps in Narrowing Down Approach; RC: Requirements Change; TR: Technical Response; ER: Emotional Response)

of change model. Where the developer decides that he/she will not work on the requirements change by showing anger, it falls under the stage of pre-contemplation. Likewise, the technical and emotional responses can be classified to see the pattern of the responses at each stage of requirements change.

5 NEXT STEPS FOR AGILE TEAM BEHAVIOR RESEARCH

Further areas that we have identified and plan to explore via our empirical studies with agile software development teams include: **During requirements engineering activities, such as iteration planning**: as the product owner and team discuss work items for the upcoming iteration, disagreements can often occur around priorities, estimates, assignment, and so on. How do the product owner and the team navigate these possible disagreements and conflicts emotionally and technically? How do they reach a suitable consensus, especially when these may bring significant change for some?; **As teams develop their planned work:** individuals may identify dependencies, blockages, inaccurate estimations, and technical challenges. How does the team respond technically and emotionally to these issues and challenges during the development phase? How does the agile coach support them in overcoming ones which require – potentially very difficult - individual and/or team behavior changes? **As teams demonstrate planned work:** during an iteration. How does the product owner best provide reinforcement of desired product outcomes and the agile coach help the team more widely adopt changes in behavior, technical approach, communications strategy, and so on?; **As requirements changes are introduced:** by the product owner or customer. Does the team respond to product owner and customer differently? Does the rate of requirements change acceptance change depending on the person who introduced them?; **As teams work to implement:** the requirements changes. Is the behavior of the team same as the behavior they showed when they received/accepted the requirements changes? Has the behavior got evolved with the time?; **As teams deliver:** the implemented changes in a release. What differences in behavior can be seen when the release includes changes and when the release does not include any change?; **Toward the end of an iteration:** where the team is under iteration pressure [2]. How can the behavior change models be used to reduce the pressure the team has?; **As the project draws to a close:** what changes in behavior and affective states of the team can be expected as the team is getting ready for the final release of the product? Will there be some degree of project or team "withdrawal syndrome" that needs to be managed? Correlations between behavior change and other human factors, such as culture, personality, gender, and age, will also be important to study.

6 CONCLUSION

We have summarised three leading behavior change models from behavioral psychology that can be used to study various behavior and behavioral changes in agile teams. These are: *stages of change model* (also known as *transtheoretical model*), *social cognitive theory*, and *theory of planned behavior.* In our pilot study on handling requirements changes, we wanted to better understand technical and emotional responses to these changes by agile teams using

behavior change models. Practitioners can monitor and record their responses in different circumstances. They can analyse responses as a team and to different types of changes and combine those together and classify using stages of a change model. This allows agile teams to better see how they actually respond and to then better predict and manage their responses in different situations. Appropriate use of change models, individually and in combination, can help with self- and team improvement, risk mitigation, and contingency planning. Use of behavior change models can enable software developers, teams, and organisations to enhance their practices toward increasing team performance. They also provide avenues to acknowledge and address the psychological state of the humans central to the software engineering process, assisting with well-being and productivity.

REFERENCES

[1] Icek Ajzen. 1985. From Intentions to Actions: A Theory of Planned Behavior. In *Action Control.*
[2] Jeffry Babb, Rashina Hoda, and Jacob Nørbjerg. 2014. Embedding reflection and learning into agile software development. *IEEE Software* 31, 4 (2014), 51–57.
[3] Albert Bandura. 2012. Social Foundations of Thought and Action. In *The Health Psychology Reader.*
[4] K Beck, M Beedle, A Van Bennekum, A Cockburn, Ward Cunningham, Martin Fowler, James Grenning, Jim Highsmith, Andrew Hunt, Ron Jeffries, Jon Kern, Brian Marick, Robert C Martin, Steve Mellor, Ken Schwaber, Jeff Sutherland, and Dave Thomas. 2001. Manifesto for Agile Software Development.
[5] Rachel Davis, Rona Campbell, Zoe Hildon, Lorna Hobbs, and Susan Michie. 2015. Theories of behaviour and behaviour change across the social and behavioural sciences: a scoping review. *Health Psychology Review* (2015).
[6] Pete Deemer, Gabrielle Benefield, Craig Larman, and Bas Vodde. 2012. The Scrum Primer. *InfoQ* (2012).
[7] Luiz Fernando Capretz. 2014. Bringing the human factor to software engineering.
[8] Barney Glaser and Anslem Strauss. 1967. *Grounded Theory: The Discovery of Grounded Theory.*
[9] Per Lenberg, Robert Feldt, and Lars Göran Wallgren. 2015. Behavioral software engineering: A definition and systematic literature review. *Journal of Systems and Software* (2015).
[10] Per Lenberg, Robert Feldt, and Lars Goran Wallgren. 2015. Human factors related challenges in software engineering - An industrial perspective. In *Proceedings - 8th International Workshop on Cooperative and Human Aspects of Software Engineering, CHASE 2015.* Institute of Electrical and Electronics Engineers Inc., 43–49.
[11] Bing Liu. 2012. Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies* (2012).
[12] Kashumi Madampe, Rashina Hoda, John Grundy, and Paramvir Singh. 2020. Towards Understanding Technical Responses to Requirements Changes in Agile Teams. In *IEEE/ACM 42nd International Conference on Software Engineering Workshops (ICSEW'20).* ACM, New York, NY, USA, Seoul, Republic of Korea, 4.
[13] Kashumi Madampe, Rashina Hoda, and Paramvir Singh. 2020. Towards Understanding Emotional Response to Requirements Changes in Agile Teams. In *IEEE/ACM 42nd International Conference on Software Engineering New Ideas and Emerging Results (ICSE-NIER'20).* ACM, New York, NY, USA, Seoul, Republic of Korea, 4.
[14] Matthew J. Mimiaga, Sari L. Reisner, Laura Reilly, Nafisseh Soroudi, and Steven A. Safren. 2009. Individual interventions. In *HIV Prevention.*
[15] James O. Prochaska and Carlo C. DiClemente. 1983. Stages and processes of self-change of smoking: Toward an integrative model of change. *Journal of Consulting and Clinical Psychology* (1983).
[16] Paul Ralph, Sebastian Baltes, Gianisa Adisaputri, Richard Torkar, Vladimir Kovalenko, Marcos Kalinowski, Nicole Novielli, Shin Yoo, Xavier Devroey, Xin Tan, Minghui Zhou, Burak Turhan, Rashina Hoda, Hideaki Hata, Gregorio Robles, Amin Milani Fard, and Rana Alkadhi. 2020. Pandemic Programming: How COVID-19 affects software developers and how their organizations can help. (5 2020).
[17] Ken Schwaber and Jeff Sutherland. 2017. The Scrum Guide: The Definitive The Rules of the Game. *Scrum.Org and ScrumInc* (2017).