The Journal of Systems & Software 198 (2023) 111592

Contents lists available at ScienceDirect

The Journal of Systems & Software

journal homepage: www.elsevier.com/locate/jss

Investigating acceptance behavior in software engineering—Theoretical perspectives[☆]

Jürgen Börstler[a,*], Nauman bin Ali[a], Martin Svensson[b], Kai Petersen[a,c]

[a] Department of Software Engineering, Blekinge Institute of Technology, Sweden
[b] Department of Industrial Economics, Blekinge Institute of Technology, Sweden
[c] University of Applied Sciences Flensburg, Germany

ARTICLE INFO

Article history:
Received 1 April 2022
Received in revised form 17 November 2022
Accepted 21 December 2022
Available online 2 January 2023

Keywords:
Acceptance behavior
Dual process theory
Technology acceptance
Theory
TAM
UTAUT
TPB

ABSTRACT

Background: Software engineering research aims to establish software development practice on a scientific basis. However, the evidence of the efficacy of technology is insufficient to ensure its uptake in industry. In the absence of a theoretical frame of reference, we mainly rely on best practices and expert judgment from industry-academia collaboration and software process improvement research to improve the acceptance of the proposed technology.
Objective: To identify acceptance models and theories and discuss their applicability in the research of acceptance behavior related to software development.
Method: We analyzed literature reviews within an interdisciplinary team to identify models and theories relevant to software engineering research. We further discuss acceptance behavior from the human information processing perspective of automatic and affect-driven processes ("fast" system 1 thinking) and rational and rule-governed processes ("slow" system 2 thinking).
Results: We identified 30 potentially relevant models and theories. Several of them have been used in researching acceptance behavior in contexts related to software development, but few have been validated in such contexts.

They use constructs that capture aspects of (automatic) system 1 and (rational) system 2 oriented processes. However, their operationalizations focus on system 2 oriented processes indicating a rational view of behavior, thus overlooking important psychological processes underpinning behavior.
Conclusions: Software engineering research may use acceptance behavior models and theories more extensively to understand and predict practice adoption in the industry. Such theoretical foundations will help improve the impact of software engineering research. However, more consideration should be given to their validation, overlap, construct operationalization, and employed data collection mechanisms when using these models and theories.

1. Introduction

Software engineering is defined as "the application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is, the application of engineering to software" (ISO/IEC, 2017, 3.3810, 2.). This means that the adoption of new technologies, methods, languages, tools or ways of working (or practice for short[1]) should be informed by evidence (Dybå et al., 2005). In the past, software engineering research has primarily approached practice introduction from a rational perspective. The emphasis was mainly on providing evidence of the efficacy of a practice, and sufficient evidence was assumed to lead to adoption (Basili, 1996; Ali, 2016; Dybå et al., 2012).

However, it has been observed that, in practice, adoption decisions are rarely informed by evidence alone (Devanbu et al., 2016; Zelkowitz et al., 2003). This is a well-recognized problem that has been discussed and investigated in the context of technology transfer, industry-academia collaboration, and change management in the context of software process improvement. There is a large body of literature on "success factors" for quality and process improvement (Dybå, 2005; Khan et al., 2017; Aquilani et al., 2017). This literature highlights the importance of people factors like management support and communication in general terms but rarely addresses in detail aspects like attitudes

☆ Editor: Raffaela Mirandola.

* Corresponding author.
E-mail addresses: jurgen.borstler@bth.se (J. Börstler), nauman.bin.ali@bth.se (N.b. Ali), martin.svensson@bth.se (M. Svensson), kai.petersen@bth.se, kai.petersen@hs-flensburg.de (K. Petersen).

[1] In the remainder of this paper, we will use the term "practices" to refer to specific practices as well as to technologies, methods, languages, tools or ways of working, in general.

https://doi.org/10.1016/j.jss.2022.111592
0164-1212/

(a psychological construct (Breckler, 1984)) or beliefs (a form of cognitive tacit knowledge (Fagerholm et al., 2022)) that may affect individuals in making a change. A change in behavior, like introducing new practices that change established ways of working, cannot simply be introduced by instruction, even when there is evidence of the efficacy of the change (Stelzer and Mellis, 1998).

Software engineering is not unique in facing the challenge that practice adoption is not a purely rational process (Dessart et al., 2019; Whitehead and Russell, 2004). Strength of evidence, cost, and risk (Ali, 2016) may be necessary factors for practice adoption but are certainly insufficient to explain adoption or lack thereof. Resistance to change, for example, is a key factor in implementing software process improvements (Lavallée and Robillard, 2012) as are developers' attitudes towards (organizational) change for increasing the success of a change (Lenberg et al., 2017). It is therefore important to understand, not only, the evidence of the efficacy of a practice as a driver for developers' intentions, but also the individual processes that might interfere with implementing these intentions (Howe and Krosnick, 2017). The study of human behavior, therefore, plays a significant role in the research on software engineering practices (Lenberg et al., 2015).

Theories of behavior, e.g., the Theory of Planned Behavior (TPB) (Ajzen, 1991a), are used widely in the social, learning and health sciences to explain people's behavior (Conner and Norman, 2015). In the context of information systems and information technology, researchers have adapted these theories to explain or predict the usage or adoption of (information) technology (Taherdoost, 2018), e.g., the Technology Acceptance Model (TAM) (Davis, 1989) or the Unified Theory of Acceptance and Use of Technology (UTAUT) (Venkatesh et al., 2003a).

In this study, relying on several recent and extensive literature reviews, we identify and describe the various models and theories of acceptance behavior. Specifically, we:

- Catalog models and theories on acceptance behavior; from a range of available models, we identify those of particular interest for software engineering research.
- Reason on the potential relevance of the models and theories for software engineering.
- Analyze models and theories on acceptance behavior used in software engineering from a human information processing perspective.

The remainder of this paper is organized as follows: In Section 2, we present the background and related work. Among others, we discuss behavioral software engineering 2.2, the basic concepts of acceptance behavior models and theories 2.4, and the role of individual information processing 2.5. In Section 3, we describe our research approach, and in Section 4, we describe and briefly reflect on the relevance of the identified models. In Section 5, we discuss acceptance behavior models and theories that have been used in software engineering research from the system 1 and system 2 perspectives. Next in Section 6 we discuss the findings, and we conclude the paper in Section 7 with a summary, conclusion and directions for further research.

2. Background and related work

2.1. On the need and role of theories in software engineering

There is an increasing realization that software engineering research needs strong theories (Schmid, 2021; Stol and Fitzgerald, 2015; Johnson et al., 2012). Theories provide a framework for building hypotheses and reasoning about what might have caused certain results and eventually explain certain phenomena within the constraints of the theory. In the last decade, several theories in software engineering have been presented (Bjarnason et al., 2016; Johnson and Ekstedt, 2016; Hall and Rapanotti, 2017; Wohlin et al., 2015; Ralph, 2016). However, by and large, a theoretical perspective to understand and predict acceptance behavior is fragmented in software engineering research. For example, research on agile practice introduction shows that practices that are claimed to be used might actually be used differently or not at all (Nurdiani et al., 2019). Therefore, an important question is not only under which conditions a practice is likely to be accepted but also why a practice intended to be used or claimed to be accepted under prevailing conditions is not used.

The models and theories on understanding and predicting behavioral intention take such factors into account. Some of these theories are used in the computing literature, but focus mainly on consumer technology or the end-users of a software system, not practices that are used by software developers in software development. The Technology Acceptance Model (TAM) is one frequently used model that predicts behavioral intention (Davis, 1989). A systematic literature review on TAM (Turner et al., 2010) suggests that TAM's behavioral intention "is likely to be correlated with actual usage" but also cautions that "[c]are should be taken using the TAM outside the context in which it has been validated".

2.2. Behavioral software engineering

Software development is typically considered a complex and predominantly intellectual activity accomplished through cognitive processing abilities (Khan et al., 2011). It is a human activity that clearly moves beyond the view of generating and ranking alternatives to find optimal solutions that are effortlessly implemented in practice (Bazerman and Moore, 1998; Simon, 1968). The study of human aspects, therefore, plays a significant role in the research on software development practices (Lenberg et al., 2015).

The importance of "people factors" in software development has long been recognized and the psychology of programming was an established research field[2] in the 1970s (Weinberg, 1971; Blackwell et al., 2019). There is also a large body of literature on "success factors" for quality and process improvement (Dybå, 2005; Khan et al., 2017; Aquilani et al., 2017; Unterkalmsteiner et al., 2011) that highlights the importance of people factors like management support and communication in general terms, but rarely addresses in detail the social, cognitive or behavioral aspects that affect individuals in making a change.

Resistance to change, for example, is a key impact factor in implementing software process improvements (Lavallée and Robillard, 2012). Furthermore, developers' attitudes towards (organizational) change (Lenberg et al., 2017) and humans' attitudes in general have also been described as important predictors for behavior (Fishbein and Ajzen, 2010). Devanbu et al. (2016) found that a priori opinions and beliefs of developers that are not based on empirical evidence affect their practice. It is therefore important to understand, not only, the processes that enable developers to act according to their intentions but also the processes that might interfere with implementing those intentions (Howe and Krosnick, 2017) as well as how they process information.

In recent years, the study of behavioral aspects of software engineering has received renewed attention. Lenberg et al. (2015) define behavioral software engineering as the "study of cognitive, behavioral and social aspects of software engineering performed by individuals, groups or organizations" and emphasize that this definition includes "other aspects of mind, such as emotions

[2] The Psychology of Programming Interest Group (PPIG, https://www.ppig.org) was established in the late 1980s and runs regular workshops until today.

and values etc." Their literature review showed that behavioral aspects in software engineering have dominantly been studied in 4 of SWEBOK's 15 knowledge areas (SE management, SE models and methods, SE professional practice and SE economics). Nine of the remaining 11 areas were covered by 1% or less of the 250 publications they studied. To support research in behavioral software engineering Graziotin et al. (2021) provide guidelines for studies that involve instruments for measuring behavioral aspects (psychometrics).

Overall, the recent work on behavioral software engineering indicates a growing awareness of and interest in explanatory models that go beyond purely rational models for explaining behaviors.

2.3. Related literature reviews

To the best of our knowledge, there are no literature reviews on models and theories of acceptance behavior within the field of software engineering. However, several literature reviews on the topic have been conducted in the IS/IT and manufacturing literature (Rad et al., 2018; Seuwou et al., 2017; Taherdoost, 2018). We capitalize on these reviews and further analyze the identified models and theories that might be relevant to software engineering research.

There are several literature reviews on the usage of specific models and theories of acceptance behavior in CS/IS/IT/SE research, though. To the best of our knowledge, Turner et al.'s systematic review on TAM (Turner et al., 2010) is the only such review in the field of software engineering. However, only a fraction of the primary studies included in those secondary studies target aspects of software development. For UTAUT, citation analyses also show that only few of the research papers citing UTAUT actually use it in some way, at least partially; Tamilmani et al. (2017): 22.6%, Venkatesh et al. (2016): 4.1%, Williams et al. (2011): 9.6%. To the authors' best knowledge, similar analyses regarding the usage/citation of other models or theories are not available for research within CS/IS/IT/SE.

In a recent paper on social science theories in software engineering, Lorey et al. (2022) argue that "software engineering research becomes more concerned with the psychological, sociological and managerial aspects of software" and therefore they investigate "which, how, where, and to what extent social science theories are used in software engineering research". They provide a list of the theories encountered and elaborate on the most common. They further explain how inattention to social science leads software engineering researchers to oversimplify and over-rationalize core phenomena.

Compared to the related work, our study should not be considered an additional literature review on acceptance models in software engineering. We found that the existing reviews in software engineering only looked at a limited set of models. Turner et al. (2010), for example, only investigated TAM. In this paper, we look at a broader set of models and theories of acceptance behavior and reflect on their application in software engineering. Specifically, we highlight and discuss the role and implications of individual information processing in decisions (system 1 and system 2 thinking, see Section 2.5). It is also essential to reflect on the intention to use and actual usage, as this has not been sufficiently addressed in the software engineering literature.

2.4. Predicting behavior

The key goal of models and theories for acceptance behavior is the prediction of *behavior* and/or explanation of a behavioral change. Most models and theories have in common that they use *(behavioral) intention* as the key predictor for actual behavior, see Fig. 1. The relationship between behavioral intention and actual behavior is well established in psychology (Sheeran, 2002). However, researchers also caution that "intentional control of behavior is a great deal more limited than previous meta-analyses of correlational studies have indicated" (Webb and Sheeran, 2006, p. 262). Behavioral intention is influenced by different core constructs in different models/theories, e.g., attitudes, motivation, social factors and expectations/beliefs regarding the effect of the behavior, and the effort to carry out the behavior. Some of the core constructs also affect the behavior more directly (e.g., habit in UTAUT2 (Venkatesh et al., 2012)). Mediators or moderators like age, gender and experience affect the influence of the core constructs on behavioral intention and behavior. Furthermore, it has been shown that conceiving of implementation intentions supports goal attainment, i.e. implementing the intended behavior.

Since the relationship between behavioral intention and actual behavior is well established in the IS research literature, behavioral intention is often used as a proxy for actual behavior (Tamilmani et al., 2021a). However, in a meta-analysis on behavioral intention and actual behavior, Wu and Du found that behavioral intention is not a good surrogate for actual behavior (system usage more specifically) (Wu and Du, 2012). This is also corroborated by Tiefenbeck et al. (2016) who found "only a mild correlation of self-reported usage and log data" and conclude that "while intention is indeed well explained by UTAUT factors, it does not necessarily predict usage".

Research also shows that practices that are claimed to be used not always are actually used (Nurdiani et al., 2019). Therefore, an important question is not only under which conditions a practice is likely to be accepted. We also need to understand why a practice that is intended or even claimed to be accepted under prevailing conditions is not used.

The intention-behavior gap, captured by the critical question *why and when do people fail to act on their intentions?*, is discussed thoroughly in a review by Sheeran (2002). Research shows that intentions only account for about 20%–30% of the variance in behavior (Gollwitzer, 1999). Fishbein and Ajzen discuss many potential explanations for this gap, like, for example, how intentions and behaviors are measured or the degree of volitional control over performing the behavior (Ajzen, 1991a, Chp. 2). In TPB, the concept of *actual behavioral control* "moderates the effect of intentions on behavior". Since "it is usually much more difficult to measure actual behavioral control than perceived behavioral control, most studies rely on perceived behavioral control as a proxy for actual control" (Ajzen, 2020). However, there might be many factors influencing the transition from perceived to actual behavioral control (Carrington et al., 2010). In health-related research, TPB is, for example, used to investigate behaviors related to physical activity/exercising or dietary behaviors (Conner and Sparks, 2005), in particular to find out why people fail to act on intentions. The results can then be used to improve behavior change interventions (Hagger et al., 2020). The acceptance behaviors studied in health-related research or consumer behavior have many striking similarities to acceptance behavior in software engineering, e.g., practice adoption in software development.

2.5. The role of individual information processing in decisions

Research in psychology shows that individuals process information by switching between affective and cognition-laden systems. Theories in psychology distinguish, for example, between controlled and automatic processes (Shiffrin and Schneider, 1977), cold and hot processes (Metcalfe and Mischel, 1999),

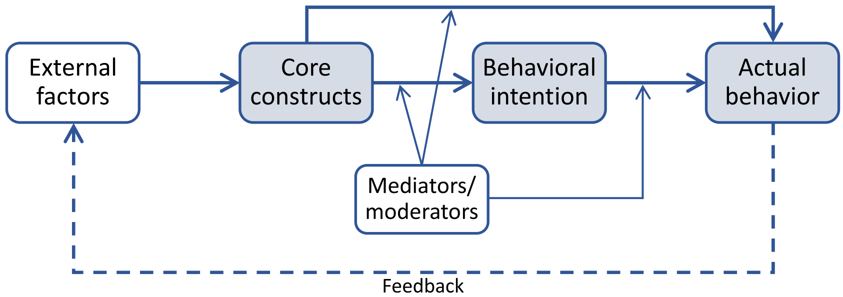

Fig. 1. Basic concepts of acceptance behavior models and theories, e.g., UTAUT and TPB.

and planners and doers (Thaler and Shefrin, 1981). Other theories conceptualize the differences of individual information processing into "systems", like the experiential versus the rational system (Epstein, 1994) and the (fast) system 1 versus the (slow) system 2 (Kahneman, 2003; Stanovich and West, 2000, 2002). At their cores, these conceptualizations point out distinctions between affective and cognitive processes (the psychological source), how and how fast information is processed (type and speed of processing), the level of detail in processing (granularity), the level of consciousness involved and the degree of automaticity (flexibility) of thought. Inspired by Evans (2008, 2011), we draw on system 1 and system 2 processing as two broad, but conceptually different, systems for processing information. In broad terms, system 1 functions in terms of generating alternatives, and encompasses various affective processes such as emotions, moods and intuitions, while system 2 serves to regulate through cognitive processes (memory, reasoning and decision-making). The exact influence of one system over the other is complex, though, and a matter of ongoing research (Lerner et al., 2015; Angie et al., 2011; Darlow and Sloman, 2010).

Metcalfe and Mischel (1999), for example, pointed to a 'cold' system as the seat of mainly cognitive and affect-neutral processing. On the contrary, the 'hot' system is the seat of emotional, impulsive and reflexive processing, which is fundamental for emotional (classical) conditioning and undermines efforts of self-control. The distinction between system 1 and system 2 shares some characteristics with that. System 1 (Kahneman, 2003, 2011), or the experiential system (Epstein, 1994), has been argued to stand for pre-conscious, rapid, automatic, holistic, typically nonverbal and mostly affect-driven, processes. System 2, or the rational system, emphasizes cognitive characteristics; slow, serial, effortful, rule-governed, flexible and neutral processes.

Dual process theories, such as system 1 and system 2, refer to information being processed on different levels and with different efforts. System 1 operates automatically, quickly, and intuitively with little or no effort and no sense of voluntary control. It also houses affective processes (Darlow and Sloman, 2010). System 2 regulates and allocates attention to the effortful mental activities that demand it, including complex computations. Kahneman (2011) uses the following common example to exemplify the functioning of the two systems: "A bat and a ball together cost $1.10. The bat costs $1 more than the ball. How much does the ball cost?". Intuitively, system 1 leads one to believe the ball costs 10 cents, but when letting the processes underpinning system 2 further scrutinize the example, it becomes clear that the ball costs 5 cents and the bat $1.05.

The similarities and differences of system 1 and 2 processing are summarized and outlined in Table 1.

Despite complex definitions, the viability and importance of individual information processing theories, are reflected in the IS-(Ferratt et al., 2018) and behavior change literature (Hagger et al., 2020). For example, regarding the role of converting intentions into actions regarding IT use, Ferratt et al. (2018) proposed that system 1 processing is likely to occur when there is no prior exposure to using an IT system and that conscious and effortful system 2 processing needs to override the fast, automatic and unconscious system 1 responses. Thus, system 2 is the most dominant during initial IT use. However, its influence gradually decreases as learning occurs and behavior becomes more habitual. Consequently, studying system 1 and 2 processes becomes relevant for predicting acceptance behavior in software engineering. Acceptance behavior in software engineering often concerns changing an established practice. For the individuals who need to change the practice, such a change usually requires conscious effort and also management of emotions due to reactions to the change. Ferratt et al. (2018) claim that the influence of system 1 decreases and becomes substituted by system 2 processing as the behavior becomes an accepted standard of operations. These arguments emphasize the need for studying and predicting acceptance behavior within the software engineering domain. Implementing new practices in the software engineering domain typically deals with the substitution of a "competing" established and potentially automated behavior.

In the recently proposed Technology Integration Model (TIM), Shaw et al. (2018) argue along similar lines. Their model focuses on the continued use of technology. The authors argue that common technology acceptance models, like TAM, are not well suited for the prediction of sustained technology use, since other constructs become important. A literature review on user resistance in IT concludes that user resistance is a complex phenomenon and not just a reaction to change and uncertainty and that further research is necessary to better understand user experience and behavior (Ali et al., 2016).

In the software engineering domain, Sánchez-Gordón and Colomo-Palacios (2019) reviewed the literature on software developers' emotions and mapped the kinds of emotions investigated and the measures/instruments to assess them. In the 66 selected primary studies, spanning 2005–2018, they identified 40 discrete emotions which they grouped into 15 categories, of which anger, fear, disgust, sadness, joy, love and happiness were the most commonly reported ones. Of the 66 selected studies, 34 were published 2017–2018, suggesting that the study of emotions in software engineering is a relatively new area within software engineering. Although the authors did not investigate the relationship between emotions and specific behaviors, they stated a need to map software developers' emotions to their performance, productivity, quality and well-being as well as how cognitions affect emotions.

To the best of our knowledge, human information processing theories, such as system 1 and system 2 processing have not been explicitly applied in the software engineering context. Implicitly, though, system 1 and system 2 are discussed in the literature on cognitive biases in software engineering since (fast and automatic) system 1 processing is particularly prone to cognitive bias. Mohanani et al. (2018) mapped cognitive biases in software

Table 1
Characteristics of system 1 and system 2 processing.

Characteristic	System 1 processing	System 2 processing
Psychological source	Behavior influenced by "vibes", "hunches" or affect	Neutral, cognitive and reason-based
Type of processing	Dominated by association and context specific	Logic, conceptual and context independent
Speed of processing	Fast and direct	Slow and step-wise due to sequential processing
Granularity	Holistic and pattern-based	Detail focused
Level of consciousness	Unconscious/preconscious/reflexive processing	Conscious and reflective
Flexibility in processing	Automatic, hard to overrule and neglect	Flexible, changes with speed of thought

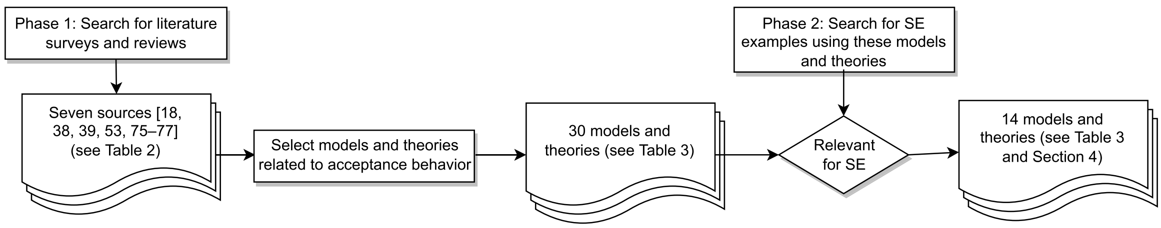

Fig. 2. A high-level overview of the research approach.

engineering (e.g., ignoring change requests, inaccurate effort estimation). While Mohanani et al. mention various biases that may affect acceptance behavior, e.g., interest biases (related to individual preferences) or action-oriented biases (taking premature decisions not considering all relevant information), they did not discuss the potential effects of biases on technology acceptance. In a recent field study, Chattopadhyay et al. (2020) investigated how cognitive biases influence developers. They conclude that developers recognize the frequent occurrence of cognitive biases and "deal with such issues with ad hoc processes and sub-optimal tool support".

3. Research approach

The objective of this paper is to provide an overview of models and theories that might be useful for investigating phenomena related to acceptance behavior in software engineering. Furthermore, we aim to describe the most relevant models and theories with respect to the characteristics of individual information processing outlined in Section 2.5. We used literature review as our research method. It should be noted that we did not conduct a systematic literature review according to the guidelines by Kitchenham and Charters (2007). However, we followed a systematic approach when studying the literature (e.g., using systematic searches and criteria to identify the relevant theories) to raise the trust in the results, to facilitate revisiting the literature using a similar approach later, and to allow other authors to "replicate" our work. Fig. 2 provides an overview of our approach.

3.1. Search for models and theories and their use in software engineering

There are several secondary studies about the usage of specific technology acceptance models/theories (like TAM and UTAUT) within the CS/IS/IT/SE literature. However, only a fraction of the research included in those secondary studies target topics related to software development. Due to the relevance of these domains to software development, we consider that the models and theories reported in those secondary studies can be candidates for software engineering.

For this study, we searched in two phases. In the first phase, to identify candidate models and theories, we searched broadly for reviews on technology adoption and/or acceptance behavior as described in Section 3.1.1. We used these reviews as a source for extracting models and theories relevant for investigating acceptance behavior. In the second phase, we searched if the candidate models and theories identified in the first phase have been used in the software engineering literature.

3.1.1. Phase 1 (search for existing literature reviews as a source for candidate models and theories)

We searched Scopus (due to its comprehensive coverage of peer-reviewed scientific literature (Adriaanse and Rensleigh, 2013; Ali and Tanveer, 2022; Moed et al., 2016)) for surveys and reviews on technology acceptance or adoption and on acceptance behavior. From the results, we selected publications based on three criteria: (1) the publication is potentially related to the software engineering domain (e.g., we excluded literature on farming, consumer electronics, etc.), (b) the publication provides a comprehensive list of models, frameworks or theories and (c) the publication provides some details of the process used for compiling this list. This left us with Davis et al. (2015), Seuwou et al. (2017), Rad et al. (2018) and Taherdoost (2018). We also added two key publications on the "integrative" theory UTAUT since they provide overviews of the models and theories that were considered for integration (Venkatesh et al., 2003b; Dwivedi et al., 2019). Furthermore, we added the "Handbook of behavior change" (Hagger et al., 2020) to ensure a comprehensive overview of theories on behavior-related theories being well aware that the handbook targets a different audience and also covers many theories beyond acceptance behavior.

Based on these sources, we compiled an overview of models and theories related to acceptance behavior that were cited/described in these sources, see Table 3.

3.1.2. Phase 2 (search for primary studies reporting use of candidate models and theories in software engineering)

To identify studies in SE that used any of the models and theories in Table 3, we searched Scopus using the search string *TITLE-ABS-KEY(⟨model/theory⟩ AND ⟨software engineering⟩)*. The first part of the search string (⟨model/theory⟩) was operationalized using the names of the models and theories in our list of models and theories (see Table 3) with some name variants to increase recall and to avoid excessive noise. The second part of the search string (⟨software engineering⟩) was operationalized by a list of software engineering venues. For the venue list, we used the union of the lists of software engineering conferences and -journals by Karanatsiou et al. (2019) and the list of software engineering journals by Archambault et al. (2011). Both lists have been used by other researchers to map venues to fields of research (e.g., Ampatzoglou et al. (2019), Mohammadi et al. (2020) and Petersen and Ali (2021)). The search was done on December 09, 2021.

Table 2
Literature describing models and theories with potential relevance for research on acceptance behavior in software engineering (in alphabetic order of first author).

Ref. (year)	Study type	Research field	Description
Davis et al. (2015)	Systematic literature review	Health, psychology, sociology, anthropology, economics	Reviews models and theories of behavior and behavior change of potential relevance to health interventions.
Dwivedi et al. (2019)	Literature review	Information systems	Reviews models and theories of IS/IT acceptance and the role of attitude in those. Proposes a revised model for UTAUT incorporating attitude which is seen as a central concept in behavior.
Hagger et al. (2020)	Expert reviews	Health, psychology, sociology, behavioral science, economics, philosophy, implementation science	"[S]ummarizes current evidence-based approaches to behavior change in Venkatesh et al. (2016) chapters by leading theorists, researchers, and practitioners from multiple disciplines". Focuses on interventions to support behavior change.
Rad et al. (2018)	Systematic literature review	Information systems	Categorizes papers on technology adoption along (among others) their theoretical foundation, i.e. the models/theories they use.
Seuwou et al. (2017)	Literature review	Information security	Overview of technology acceptance models and theories.
Taherdoost (2018)	Literature review	Engineering, information technology	Overview of general adoption models and theories focusing on user acceptance of technology.
Venkatesh et al. (2003b)	Literature review	Information systems	Compilation of models and theories considered for integration/unification in UTAUT.

Table 3
Models and theories of acceptance behavior discussed in the reviews/overviews listed in Table 2, (Davis et al., 2015; Dwivedi et al., 2019; Hagger et al., 2020; Rad et al., 2018; Seuwou et al., 2017; Taherdoost, 2018; Venkatesh et al., 2003b in alphabetical order).

Model/theory name (and acronym)	Covered in	Comment	Ref
Big Five theory (BIG5)	Rad et al. (2018)	Focuses on factors defining human personality.	–
Compatibility-UTAUT (C-UTAUT)	Taherdoost (2018)	Extension of UTAUT, see UTAUT.	4.9
Decomposed Theory of Planned Behavior (DTPB)	Seuwou et al. (2017)	Refinement of TPB, see TPB.	4.8
Expectation Confirmation Theory (ECT)	Rad et al. (2018)	Focuses on post-purchase (or post-adoption) satisfaction.	–
Flow theory	Rad et al. (2018)	–	–
Health Belief Model (HBM)	Davis et al. (2015) and Hagger et al. (2020)	Custom model for analyzing and promoting health-related behavior change with key constructs do not fit SE research. No SE-studies found.	–
Igbaria's Model (IM)	Taherdoost (2018)	Highly cited extension of TAM, see TAM.	4.7
Information-Motivation-Behavioral skills model (IMB)	Davis et al. (2015)	–	–
Innovation Diffusion Theory (IDT)	Dwivedi et al. (2019), Seuwou et al. (2017), Taherdoost (2018), Venkatesh et al. (2003b) and Rad et al. (2018)	Covers organizational aspects.	4.1
Inter-Organizational Relationship theory (IOR)	Rad et al. (2018)	Focuses on relationships between organizations, rather than Human's acceptance behavior.	–
IS Success model (ISS)	Dwivedi et al. (2019) and Rad et al. (2018)	Highly cited model (in Rad et al. (2018)) that also covers organizational aspects.	4.2
Model of Acceptance with Peer Support (MAPS)	Dwivedi et al. (2019)	–	–
Model of PC Utilization (MPCU)	Dwivedi et al. (2019), Seuwou et al. (2017), Taherdoost (2018) and Venkatesh et al. (2003b)	Adaptation of TIB to PC usage. Constructs of MPCU are covered in UTAUT.	4.3

(*continued on next page*)

3.2. Selection of a subset of models and theories

We defined a model/theory as most relevant when it fulfills one of the following two criteria: (1) the model/theory is covered in at least 3 of the reviews/overviews in Table 2; *OR* (2) the model/theory is covered in 2 of the reviews/overviews in Table 2 **and** has been in at least one study in SE.

We applied these criteria to 30 models and theories listed in Table 3. The variants of various models, e.g., UTAUT and C-UTAUT, were treated as one group. Nine groups (discussed in Section 4)

Table 3 (*continued*).

Model/theory name (and acronym)	Covered in	Comment	Ref
Motivational Model (MM)	Seuwou et al. (2017), Taherdoost (2018) and Venkatesh et al. (2003b)	–	4.4
Perceived Characteristics of Innovating (PCI)	Dwivedi et al. (2019) and Taherdoost (2018)	Extension of IDT, see IDT.	4.1
Perceived value model	Rad et al. (2018)	–	–
Social capital theory	Rad et al. (2018)	–	–
Social Cognitive Theory (SCT)	Davis et al. (2015), Dwivedi et al. (2019), Seuwou et al. (2017), Taherdoost (2018), Venkatesh et al. (2003b), Rad et al. (2018) and Hagger et al. (2020)	–	4.5
Social identity theory	Rad et al. (2018) and Hagger et al. (2020)	No SE-studies found.	–
Stages of Change Model (SoC)	Davis et al. (2015) and Hagger et al. (2020)	Commonly used in the health sciences, also known as the transtheoretical model (TTM). No SE-studies found.	–
TAM Extension (TAME)	Dwivedi et al. (2019) and Taherdoost (2018)	Extension of TAM, see TAM.	4.7
Task-Technology Fit model (TTF)	Dwivedi et al. (2019) and Rad et al. (2018)	Focuses on individual performance.	4.6
Technology Acceptance Model (TAM)	Dwivedi et al. (2019), Seuwou et al. (2017), Taherdoost (2018), Venkatesh et al. (2003b) and Rad et al. (2018)	–	4.7
Theory of Interpersonal Behavior (TIB)	Taherdoost (2018)	–	–
Technology-organization-environment framework (TOE)	Rad et al. (2018)	Concerned with organizational level of IT adoption.	–
Theory of Planned Behavior (TPB)/Theory of Reasoned Action (TRA)	Davis et al. (2015), Dwivedi et al. (2019), Seuwou et al. (2017), Taherdoost (2018), Venkatesh et al. (2003b), Rad et al. (2018) and Hagger et al. (2020)	–	4.8
Trust model	Rad et al. (2018)	–	–
Unified Theory of Acceptance and Use of Technology (UTAUT)	Dwivedi et al. (2019), Seuwou et al. (2017), Taherdoost (2018), Venkatesh et al. (2003b) and Rad et al. (2018)	–	4.9
Uses and Gratification theory (U&G)	Rad et al. (2018) and Taherdoost (2018)	Focuses on mass communication media. No SE-studies found.	–

met the criteria. Column "Ref" in Table 3 indicates the section in which the model is discussed further.

3.3. Validity threats

The following are the main validity threats of the study and how we addressed them:

Interpretive validity: Studying acceptance theories is outside the field of software engineering. Therefore, there is a risk that we misinterpret or misunderstand theories. Thus, we conducted our study jointly within an interdisciplinary team, where one of the co-authors has a background in behavioral sciences. All interpretations and categorizations of models and theories were reviewed by multiple co-authors. Any disagreements were resolved through discussion and consulting the original sources.

Generalizability: The goal of this study was not to provide a systematic and exhaustive overview of acceptance theories. Rather, we focused on identifying a broader set of different theories and also highlighting specific aspects that are overlooked in software engineering (e.g., system 1 and 2 thinking for understanding intended versus actual use).

Instead of directly searching for models and theories, we leveraged existing literature reviews from related fields (as there are not any such literature reviews in SE). As it takes time to conduct, report, and publish a literature review, a limitation of our search approach is that we may miss the most recent contributions to the field. Furthermore, each literature review has its own research questions and study design, which means that these reviews excluded certain models and theories that may be relevant to our study. While this threat exists, we contend it is minimized as we identified and used several comprehensive surveys and reviews on the topic from different fields.

We leveraged the "subject area" classification by Scopus to include papers published in venues that are considered related to computer science. This classification is inclusive and entirely transparent; each venue can have multiple subject tags.[3] However, there is a chance that a venue is wrongly classified or that a relevant paper was published in a venue not typically considered related to computer science.

In the next phase (see Section 3.1.2), we searched for applications of models and theories (identified in the first phase of search see Section 3.1.1) in the software engineering literature. We, however, limited the search to the metadata (title, abstract, and keywords). Publications that do not mention the used model or theory in their metadata will be missed. This is a limitation of our search strategy and our decision to only search in Scopus (as it only indexes the publications' metadata). Similarly, while Scopus has good coverage (Adriaanse and Rensleigh, 2013; Ali and Tanveer, 2022; Moed et al., 2016), it may have some delays in indexing publications compared to primary databases like IEEE or Springer.

[3] Scopus Content https://www.elsevier.com/solutions/scopus/how-scopus-works/content.

Table 4
Some examples from the SE literature of models and theories used in acceptance behavior research.

Model/theory group	Examples from SE literature	See subsection
Theories of innovation diffusion and adoption	Wan et al. (2020)	4.1
Information systems success models	Polančič et al. (2010)	4.2
Model of PC Utilization	Riemenschneider et al. (2002)	4.3
Motivation models	Beecham et al. (2008), Hall et al. (2009) and França et al. (2011)	4.4
Social cognitive theory	Hadar et al. (2018)	4.5
Task-Technology fit	Dishaw and Strong (1998)	4.6
Technology acceptance model (and extensions)	Wallace and Sheetz (2014)	4.7
Theory of planned behavior/Theory of reasoned action	Passos et al. (2013) and Sojer et al. (2014)	4.8
Unified theory of acceptance and use of technology (and extensions)	Pano et al. (2018)	4.9

4. Models and theories used in acceptance behavior research

Theories of behavior and behavior change are researched and used heavily in other areas, e.g., in the health sciences. One application area, for example, is to investigate how people adopt and sustain healthier lifestyles or abandon them despite good intentions and despite being well aware of the risks involved. We think that this has striking similarities to software process improvement initiatives that fail, despite research showing the advantages of the improvement and developers' good intentions to carry out the improvements.

Two models used frequently in the health sciences are the Health Belief Model (HBM) and the Trans-Theoretical Model (TTM). However, "[t]here is evidence that the Theory of Planned Behaviour has greater predictive power than the Health Belief Model or the Theory of Reasoned Action (TRA). Nevertheless, neither the TPB nor the TRA or the HBM is specified to offer insight into how health behavioural change can most effectively be facilitated. In this respect, the Trans-Theoretical Model (which embodies both 'stage-of-change' and 'process of change' constructs) is fundamentally different in terms of its structure, and how it can be used to define and manage the delivery of health behaviour change interventions" (Taylor et al., 2006, p. 16).

In this section, we provide an overview of the models and theories of acceptance behavior that we identified in the literature described in Table 2.

Table 3 provides a compilation of the models and theories discussed in these overviews and reviews. The most relevant models and theories (meeting our criteria for judging relevance as described in Section 3.2) are listed in Table 4 and described briefly in the following subsections. For each of the models and theories, we also indicate their prevalence in IS/IT adoption research by using a recent and comprehensive literature review by Rad et al. (2018).

4.1. Theories of innovation diffusion and adoption

Description. Theories of innovation diffusion or innovation adoption try to explain the implementation, adoption and dissemination of innovations by individuals and organizations. The theories have in common a staged adoption model, proposed initially by Rogers (2003). The five stages lead from becoming initially aware of an innovation to continued use of an innovation with the adoption decision as the stage in the middle. Besides the perceived characteristics of the innovation, the factors affecting adoption can be external/environmental, or related to the organization or the individuals affected by an innovation (Moore and Benbasat, 1991). Wisdom et al. conducted a review of innovation adoption frameworks "to identify elements across adoption frameworks that ... might be employed to improve the adoption of evidence-based practices" (Wisdom et al., 2014). They found that "[c]onstructs of leadership, operational size and structure, innovation fit with norms and values, and attitudes/motivation toward innovations each are mentioned in at least half of the theories, though there were no consistent definitions of measures for these constructs".

Prevalence in IS/IT. In Rad et al.'s review (Rad et al., 2018), theories of innovation diffusion and adoption (IDT/ DIT/ DOI, PCI) are used in 44 of 330 papers (13.3%).

Examples from SE. Within software engineering, DOI has, for example, been used to investigate practitioners' adoption of defect prediction techniques (Wan et al., 2020). An interesting result of this research was that the authors found an inconsistency between practitioners' behavior and their perception regarding defect prediction.

Reflection. Most of the research in innovation diffusion deals with "market penetration" of new products or product categories and competition between markets/providers and is somewhat out of the scope of software engineering. Innovation adoption, on the other hand, is more related to individuals' perceptions and behaviors regarding the adoption of innovations (or practices). Regarding innovation adoption, the characteristics of the innovation and the characteristics of the (potential) adopters are more important than the communication channels and diffusion networks.

4.2. IS success models

Description. Information system success modes (IS success models) try to model and predict the success or net benefits of information system usage (DeLone and McLean, 1992, 2003). Two important constructs in IS success models are system usage and user satisfaction and their net benefits for users and organizations. IS success models go beyond adoption or implementation decisions and consider "usage factors", i.e. how a system is used. This might make it difficult to use IS success models for researching adoption behavior since the targeted practices might not be in use (yet). Since usage is difficult to capture, DeLone and McLean suggest using intention to use as a proxy but also point out that "'[i]ntention to use' is an attitude, whereas 'use' is a behavior" but that "attitudes, and their links with behavior, are notoriously difficult to measure" (DeLone and McLean, 2003). While one meta-analysis of the IS success model at the individual level found support for most of the relationships between the constructs in the IS success model (Petter and McLean, 2009), a more recent one is more critical (Jeyaraj, 2020). They particularly point out that the constructs are used inconsistently and question whether intention to use is a suitable construct for IS success. Sabherwal et al. (2006) provide another meta-analysis and extension of the model with further constructs on the individual- and organizational levels.

Prevalence in IS/IT. In Rad et al.'s review (Rad et al., 2018), the IS success model (ISS) is used in 36 of 330 papers (10.9%).

Examples from SE. Within software engineering, ISS has, for example, been used (together with TAM) to investigate the continuous use of application frameworks (Polančič et al., 2010).

Reflection. DeLone and McLean already remarked (DeLone and McLean, 2003) that research "demonstrate[s] that early use and continued use can differ". Shaw et al. (2018) also argue that common technology acceptance models are not well suited for the prediction of continued use since other constructs become important sustained use is investigated. When using IS success models for investigating acceptance behavior, care must therefore be taken to distinguish between early use (the acceptance phase) and sustained usage.

4.3. Model of PC utilization

Description. The Model of PC Utilization (MPCU) was originally developed as an adaptation of Triandis' theory of interpersonal behavior to investigate the factors that affect the volitional use of PCs introduced in organizations (Thompson et al., 1991). While many models and theories presented here focus on behavioral intention as a proxy for behavior, the MPCU focuses on actual PC utilization and, therefore, lacks behavioral intention as a mediating construct.

Prevalence in IS/IT. Rad et al.'s review (Rad et al., 2018) lacks a category for MPCU. However, key parts of MPCU have been integrated into other models or theories presented here, e.g., UTAUT.

Examples from SE. Within software engineering, MPCU has, for example, been used to investigate software developers' acceptance of a comprehensive, structured software development methodology (Riemenschneider et al., 2002).

Reflection. Riemenschneider et al. (2002) compared the MPCU with four other acceptance models/theories (TAM, TAM2, PCI and TPB). The comparison showed that all models had a construct capturing "usefulness" that was a significant predictor of behavioral intention. The construct for "ease of use" (present in four of the models) was an insignificant predictor of behavioral intention, which is interesting since ease of use is a key construct in most models and theories of acceptance behavior. Although Riemenschneider et al. point out MPCU as the only model using affect as a construct, affect is also used in other models, e.g., as a background factor influencing the core constructs in TPB.

4.4. Motivation models

Description. Motivation plays an important role in the behavior of humans. Theories of motivation distinguish intrinsic (induced from "within" the human) and extrinsic (induced externally) motivation (Davis et al., 1992). Motivation and job satisfaction have been studied in many areas (Locke and Latham, 1990; Steers et al., 2004) and it is argued that existing motivation models do not fit knowledge workers like, e.g., software engineers well (Wallgren and Hanse, 2007).

Prevalence in IS/IT. Rad et al.'s review (Rad et al., 2018) lacks a category for motivation models.

Examples from SE. Motivation has been studied extensively in the context of software engineering and several secondary studies on motivation in software engineering have been published (Beecham et al., 2008; Hall et al., 2009; França et al., 2011).

Reflection. Based on existing work on motivation in software engineering, Sharp et al. (2009) and França et al. (2020) proposed motivation models that include motivational factors intrinsic to software engineers' job characteristics. Since motivation is a complex construct (like most others) and likely also interacts with other constructs, it has been included as a factor in several models and theories discussed here, e.g., in UTAUT and later versions of TAM and its derivates. UTAUT, for example, treats extrinsic motivation as an element of its construct "performance expectancy;" "hedonic motivation" was added as a separate construct in UTAUT2 (Venkatesh et al., 2012).

4.5. Social Cognitive Theory

Description. Social Cognitive Theory (SCT) is based on the concept of observing behaviors and their consequences in a social context (Bandura, 2001). Observers may decide to engage in the same behaviors depending on the expected outcomes of those behaviors (in terms of task performance and personal consequences) and personal factors, like affect, anxiety and self-efficacy. SCT has been applied in investigating communication, learning and health-related behaviors.

Prevalence in IS/IT. In Rad et al.'s review (Rad et al., 2018), SCT is used in 4 of 330 papers (1.2%).

Examples from SE. Within software engineering, SCT has, for example, been used to investigate factors that influence developers' in their design/development decisions regarding privacy (Hadar et al., 2018).

Reflection. The concepts underlying SCT have been included in many contemporary models and theories of acceptance behavior, e.g., UTAUT (see Section 4.9).

4.6. Task-Technology Fit

Description. The Task-Technology Fit (TTF) model links user tasks to the functionality of technology that supports those tasks and investigates how well the technology assists a user in performing these tasks (Goodhue and Thompson, 1995). Thus, TTF is concerned with how well a technology supports the needs of a user to perform a particular task, rather than acceptance of a technology per se. However, since TTF's task-oriented view complements other views of technology acceptance, it has been proposed to be integrated with TAM, see Dishaw and Strong (1999). There are general and specific task and technology characteristics, i.e. a TTF model needs to be adapted to the context in which it is used. The factors that influence the key constructs in TTF need to be adapted for each type or class of technology and task. Venkatesh and Bala (2008), for example, write that TAM "theorizes that the effect of external variables (e.g., design characteristics) on behavioral intention will be mediated by perceived usefulness and perceived ease of use".

Prevalence in IS/IT. In Rad et al.'s review (Rad et al., 2018), TTF is used in 12 of 330 papers (3.6%).

Examples from SE. Within software engineering, TTF has, for example, been used to investigate the usage of software maintenance tools (Dishaw and Strong, 1998).

Reflection. TTF is closely related to cognitive fit (Vessey and Galletta, 1991) postulating that a close correspondence between a problem or task and the mental representation of it held by the problem solver, leads to better performance. This correspondence has, for example, been demonstrated in the context of software maintenance (Shaft and Vessey, 2006).

4.7. Technology Acceptance Model (and extensions)

Description. The Technology Acceptance Model (TAM) (Davis, 1989) is an extension and adaptation of the Theory of Reasoned Action (TRA) (Fishbein and Ajzen, 2010) to the context of technology adoption. Instead of TRA's general construct "attitude toward behavior", TAM uses the more technology-related constructs "perceived usefulness" and "perceived ease of use". In its initial version, TAM did not include TRA's construct "subjective norm" which considers socio-cultural influences. It was, however, added in later versions of TAM (Venkatesh and Davis, 2000) to better explain factors that contribute to "perceived usefulness".

According to Ajzen (2020, Sect. 2.17) "perceived usefulness deals with possible consequences of accepting the technology and hence has some relation to attitude toward the behavior whereas perceived ease of use is allied with the concept of perceived behavioral control. In contrast to the technology acceptance model, which is content-specific, applying mainly to the acceptance of technology, the TPB[4] is formulated at a very general level. Its constructs are content-free, assumed to be applicable to any behavior of interest to social and behavioral scientists".

Prevalence in IS/IT. In Rad et al.'s review (Rad et al., 2018), TAM and its variants and extensions are used in 168 of 330 papers (50.9%) and, according to the authors the most commonly used model or theory in the technology acceptance or adaptation literature.

Examples from SE. Within software engineering, TAM has, for example, been used to study the adoption of software measures (Wallace and Sheetz, 2014).

Reflection. TAM has successively been extended. A literature review on TAM (covering 1986–2013) (Marangunić and Granić, 2015), for example, lists 32 publications addressing the development and/or extension of TAM. Although its core constructs have remained the same, factors from other theories have been incorporated as moderators for behavioral intention. A secondary study published 2010 (Turner et al., 2010), identified 79 empirical studies using TAM and concluded that although behavioral intention "is likely to be correlated with actual use … [c]are should be taken using the TAM outside the context in which it has been validated".

4.8. Theory of Planned Behavior/Theory of Reasoned Action

Description. The Theory of Planned Behavior (TPB) (Ajzen, 1991a) as well as the Theory of Reasoned Action (TRA) (Fishbein and Ajzen, 2010) use behavioral intention as the key predictor for actual behavior. Behavioral intention is determined by an individual's attitude towards the behavior and subjective norms. TPB extends TRA with the construct of "perceived behavioral control" to capture that individuals may not have full volitional control in realizing behavioral intentions.

Prevalence in IS/IT. In Rad et al.'s review (Rad et al., 2018), TPB, TRA and TIB are used in 50 of 330 papers (15.2%).

Examples from SE. Within software engineering, the TPB has, for example, been used to study beliefs about software practices in three Brazilian software companies (Passos et al., 2013) and led to recommendations to support the adoption of new practices. In another study, the TPB has been used to better understand why developers copy code from the Internet without carefully checking licenses (Sojer et al., 2014).

Reflection. TPB as well as TRA have been widely and successfully used to study behavior and behavior change, in particular in the health sciences, social sciences and psychology. Although the theories have been criticized a lot (Sniehotta et al., 2014; Ajzen, 2015; Trafimow, 2015; Conner, 2015; St Quinton et al., 2021), they have made a significant contribution to studies of behavior and have been extended with constructs from other major theories behavior, e.g., into an Integrated Behavioral Model (IBM), which is mainly used in the health sciences (Montaño and Kasprzyk, 2015; Hagger and Hamilton, 2020). As discussed at the end of Section 2.4, there is a long tradition of using TPB/TRA in the study of health behavior that has striking similarities to acceptance behavior related to software process improvement.

[4] The Theory of Planned Behavior (TPB) is an extension of TRA, see Section 4.8.

4.9. Unified theory of acceptance and use of technology (and extensions)

Description. UTAUT is an attempt to integrate existing models and theories of (individual) technology adoption into a single unified theory (Venkatesh et al., 2003a). Originally, it considered and validated concepts taken from TPB/TRA, TAM, MM, MPCU, DIT and SCT and has since been successively extended.

Prevalence in IS/IT. In Rad et al.'s review (Rad et al., 2018), UTAUT, UTAUT2 and C-UTAUT are used in 40 of 330 papers (12.1%).

Examples from SE. Within software engineering, UTAUT has, for example, been used to guide the data analysis of an empirical study that investigated the adoption of a JavaScript framework (Pano et al., 2018).

Reflection. Since many concepts have been integrated for different purposes in UTAUT2 (Venkatesh et al., 2012), the concepts and constructs might not all be applicable in all contexts or situations. In a review of empirical UTAUT2-studies, Tamilmani et al. (2018), for example, concluded that "[r]esearchers studying early stages of technology adoption in mandatory user settings should refrain from using [the] habit construct. On the other hand, the usage of [the] habit construct is encouraged in research to examine established technologies driven by consumer intrinsic motivation". A recent review of UTAUT2 and its extensions can be found in Tamilmani et al. (2021b). A recent meta-analysis can be found in Blut et al. (2021). The latter provides recommendations regarding the usage of constructs in certain contexts.

4.10. Summary

The literature provides many models and theories that can be applied for investigating acceptance behavior in software engineering contexts but they have been used sparsely. The models and theories discussed above overlap substantially regarding the constructs they use and the factors affecting these constructs. The constructs are, furthermore, not defined consistently; the same construct may have different meanings and operationalizations in different models and theories and different constructs and factors may have similar meanings and operationalizations. This is also indicated in Riemenschneider et al.'s study on the acceptance of software development methodologies (Riemenschneider et al., 2002). They employed a "unified" concept of usefulness that they operationalized in the same way for the five theories they compared, although only two theories define usefulness as a construct (TAM, TAM2). The other three use the constructs of relative advantage (PCI), attitude (TPB) and job fit (MPCU), respectively.

Ambiguous definitions of constructs and operationalizations can make it difficult to determine which model or theory fits a particular software engineering problem. The proliferation of constructs and their operationalizations can also make it difficult to apply a chosen model or theory and draw valid conclusions based on the results. It also makes comparisons of research results more challenging.

For TAM alone, a meta-analysis (Yousafzai et al., 2007) found 78 external variables that were proposed to affect the central TAM constructs perceived usefulness and perceived ease of use. Looking at criteria for tool acceptance and acceptability beyond the models and theories discussed above, a literature review (Alexandre et al., 2018) spanning 182 articles published 1959–2016 identified as many as 142 criteria that could be categorized along five meta-criteria; utility, ease of use, aesthetics, contextual and social differences, and overall judgment. A review by Jeyaraj et al. (2006) on IT innovation adoption research identified 99

Table 5
System 1 and system 2 processes in frequently used models and theories of acceptance behavior used in software engineering.

Model/theory	Key constructs & definitions	Sys 1/2 influence
Technology Acceptance Model (TAM)	**Perceived usefulness**: "the degree to which a person believes that using a particular system would enhance his or her job performance"; **Perceived ease of use**: "the degree to which a person believes that using a particular system would be free of effort" (Davis, 1989). **Subjective norm**: "the person's perception that most people who are important to him think he should or should not perform the behavior in question" (Fishbein and Ajzen, 1975).	Primarily system 1 processing.
Innovation Diffusion Theory (IDT)	**Relative advantage**: "the degree to which an innovation is perceived as being better than its precursor" (Moore and Benbasat, 1991). **Ease of use**: "the degree to which an innovation is perceived as being difficult to use". **Image**: "the degree to which use of an innovation is perceived to enhance one's image or status in one's social system". **Visibility**: "the degree to which one can see others using the system in the organization". **Compatibility**: "the degree to which an innovation is perceived as being consistent with the existing values, needs, and past experiences of potential adopters". **Demonstrability**: "the tangibility of the results of using the innovation, including their observability and communicability". **Voluntariness of use**: "the degree to which use of the innovation is perceived as being voluntary, or of free will".	Mixed influence of system 1 and system 2 processing with a tilt towards system 1 processing.
Task-Technology Fit (TTF)	**Quality of data** refers to the "currency of data quality, maintenance of the necessary fields or elements of data, and maintenance of data at the right level or levels of data". **Locatability** is the ease of determining what data is available and where, as well as the ease of determining what a data element on a report or file means, or what is excluded or included in calculating it. **Authorization** is obtaining authorization to access data necessary to carry out a task. **Compatibility** is how data from different sources can be consolidated or compared without inconsistencies. **Ease of use/training** refers to the ease of using the system hardware and software for submitting, accessing, and analyzing data and whether the participant can get the kind of quality computer-related training when they need it. **Production timeliness** relates to whether information systems (ISs) meet pre-defined production turnaround schedules. **Systems reliability** refers to dependability and consistency of access and up-time of systems and relationship with others. **Relationship with users** refers to how well ISs understand the participant's unit's business mission and its relation to corporate objectives, interest and dedication to supporting customer business needs, turnaround time for a request submitted for IS service, availability and quality of technical and business planning assistance for systems, and how well ISs keep their agreements (Goodhue and Thompson, 1995).	Primarily system 2 processing.
Theory of Planned Behavior (TPB)	**Attitude toward behavior** refers to "an individual's positive or negative feelings (evaluative affect) about performing the target behavior"; **Subjective norm** refers to "the person's perception that most people who are important to him think he should or should not perform the behavior in question" (Fishbein and Ajzen, 1975). **Perceived behavioral control** refers to "the perceived ease or difficulty of performing the behavior" (Ajzen, 1991b).	Primarily system 1 processing.

empirical studies published 1992–2003 using 135 independent variables. Of the variables that were used in at least five studies the following were rated as the best predictors of individual IT adoption: Perceived usefulness, top management support, computer experience, behavioral intention, and user support. The best predictors of IT adoption by organizations were top management support, external pressure, professionalism of the IS unit, and external information sources. Overall, innovation characteristics and organizational characteristics were good predictors of both individual and organizational IT adoption.

Another issue in using the models and theories is that they target different stages of acceptance behavior; adoption decisions, initial use (related to an adoption decision or not) or continued use. Research "demonstrate[s] that early use and continued use can differ" (DeLone and McLean, 2003). Continued use affects learning which affects ease of use which affects behavioral intention which predicts behavior, i.e. use. The relationship between behavioral intention and behavior should therefore be treated differently in studies dealing with initial use and continued use. Furthermore, it is important to consider feedback loops between the constructs as indicated in Fig. 1. It is therefore important to be context-aware and ensure that the employed models and theories fit the problem at hand.

Despite the abundance of research on acceptance behavior and the factors that help predict behavioral intention, there is little research on the factors or circumstances that might explain why people fail to act on their intentions. We argue that we might need to pay more attention to how information is processed by humans and that actual behavioral might not follow behavioral intention because it seems rational.

5. Information processing types in models and theories of acceptance behavior used in software engineering

In this section, we discuss the characteristics of individual information processing, as outlined in Section 2.5, for the models and theories of acceptance behavior that have been most frequently used in the software engineering literature. According to our search (see Section 3.1.2), these four (groups of) models and theories that have been used more than 10 times (in descending order of uses): TAM, innovation diffusion models (e.g., IDT), TPB/TRA and fitness models (e.g., TTF). These four (groups of) models and theories are also among the most used theories in a recent review about the usage of social science theories in software engineering (Lorey et al., 2022).

Each of the models and theories is underpinned by constructs that draw on aspects of system 1 and system 2 processing (see Table 5). The key constructs for TAM (perceived usefulness, perceived ease of use and subjective norm) refer to snapshot evaluations and perceptual processes rather than analytical and reason-based processes. In other words, such assessments align with the concepts outlined in Table 1 as being fast, automatic, holistic, direct "hunches", rather than deliberate, slow, step-wise cognition, logic, conceptual and detail-focused reason-based processing. Thus, the constructs of TAM conceptualize system 1-oriented processes to a higher degree than system 2 concepts. However, it is important to note that conceptualizations does not necessarily generate accurate operationalizations. Table 1 shows that many system 1 processes are subjective and beyond respondents' awareness and therefore difficult to measure accurately. A similar observation was made by Belletier et al. (2018) who suggest using implicit measures, rather than explicit measures to elicit behavioral intention, to capture attitudes that respondents may have but are unaware of.

Similar reasoning applies to several of the other theories. For instance, early versions of IDT included five broad areas; relative advantage, compatibility, complexity, observability and triability (Rogers et al., 1983). Later developments, identifying a 34-item 7-scale instrument to measure adoption of information technology, include the constructs relative advantage, ease of use, image, visibility, compatibility, demonstrability and voluntariness of use (Moore and Benbasat, 1991).

The influence of system 1 and 2 processing in IDT is mixed, but the overall assessment leans towards domination of system 1 processing. For instance, the construct relative advantage refers explicitly to perceptual processes and snapshot evaluations, in terms of that it addresses decreases in discomfort, social prestige, saving time and effort and immediacy of rewards. Furthermore, the constructs ease of use, image, visibility, compatibility and voluntariness all refer to perceptual processes. The compatibility refers to perceptual processes per se, but the combination and complexity of such judgments make it likely for comparison, analysis and reasoning to be involved.

However, the construct of relative advantage is also often operationalized from a perspective of economic profitability and initial cost. The latter constitutes circumstances which enable systematic comparisons and conscious reasoning — and thus influence objective measures. Similarly, demonstrability refers to tangibility and observability (Moore and Benbasat, 1991) and thereby relates to objective measures and system 2 processing.

Interestingly, Moore and Benbasat (1991, p. 194) claim that primary attributes are intrinsic to an innovation independent of their perception by potential adopters. "The behavior of individuals, however, is predicated by how they perceive these primary attributes. Because different adopters might perceive characteristics in different ways, their eventual behaviors might differ. This is the root of the problem of using primary characteristics as research variables". Similar views are echoed in the review of Wisdom et al. (2014), stating that IDT mixes objective and subjective concepts, noting considerable heterogeneity in the definitions of underlying constructs.

TTF focuses on performance rather than adoption. It focuses on the performance of various technologies rather than the intentions of individuals per se. The influence of system 2 processing is clear since assessing/measuring most key constructs requires conscious cognitive effort and attention to detail (with ease of use as an exception).

TPB extends TRA and is the basis for TAM (or rather TAM2). TPB's three key constructs: attitude toward behavior, subjective norm and perceived behavioral control emphasize individuals' perceptual processes rather than reason-based processes (Ajzen, 1991b). For instance, the role of individuals' emotions and feelings, perceptions of subjective norms and judgments, as well as ease/difficulty of performing an action point to snapshot evaluations related to system 1 processing.

6. Discussion

Research from the social and the behavioral sciences shows that behavior can be predicted by intentions and that intentions are formed by attitudes, beliefs and norms (Ajzen, 1991a). The same research, however, also shows that intentions are insufficient to explain or predict behavior sufficiently well. Turner et al.'s systematic literature review on TAM (Turner et al., 2010) echoes that behavioral intention "is likely to be correlated with actual usage". However, there is much variance left that classical models and theories of behavior or behavior change do not account for Wu and Du (2012). Turner et al. (2010) also caution that "[c]are should be taken using the TAM outside the context in which it has been validated".

Most models and theories in this study have been extended to fit specific research contexts. This leads to a proliferation of versions of models and theories. Some of these versions may lead to more accurate predictions in certain contexts. However, using different versions of a model or theory also makes it more difficult to compare and evaluate research results.

The models and theories discussed here use a wide range of constructs that draw on aspects of (automatic) system 1 and (rational) system 2 processing. To predict actual behavior both aspects are needed which raises the question of suitable approaches for data collection. Using surveys may be problematic, since individuals may make substantial and systematically flawed self-assessments about the degree of influence on behavior. This includes underestimation, overestimation as well as failing to detect an influence at all (Dunning et al., 2004). System 1-laden processes are therefore difficult to capture with self-assessments alone. The reliability of an intuitive judgment (a system 1-laden process) also depends on the predictability of the environment in which the judgment is made and of the individual's opportunity to learn the regularities of that environment (Kahneman and Klein, 2009). Outside stable environments, subjective experience may not be a reliable indicator of intuitive judgment (Dane, 2011).

Thus, relying solely on surveys for perception-based phenomena may cause problems since perceptions and beliefs are not easily detectable, and when detected, their influence on behavior may be either underestimated, overestimated or in the worst case not detected at all. To tap into software developers' minds, complementary data collection methods are needed, like, e.g., observational or ethnographic studies. Similar issues have been noted in the evaluation of user experience in virtual-, augmented and mixed reality applications (Law et al., 2021).

Developing data collection methods that give a more complete picture of the factors that influence behavioral intention and behavior may help us to decrease the variance in models and theories for predicting acceptance behavior.

7. Conclusions and future work

Our review and discussion show that software engineering research might learn a lot from other areas about the usage of models and theories of acceptance behavior. Our field has much to gain from looking beyond rationality when introducing new practices. Providing empirical evidence for the superiority of a new practice does not necessarily imply that it "naturally" will be adopted since there are convincing rational arguments. Research about practice adoption in software development is typically treated as a system 2 question or problem. Yet, when assessing the models and theories of acceptance behavior that are commonly used in software engineering research, we can see that they provide a number of constructs that aim to tap into system 1. Capturing system 1 concepts is difficult since there are limited ways to measure them objectively, reliably and non-invasively.

Self-evaluations are highly dependent on individual differences and socio-cultural factors "but show little systematic variation with personally endorsed values" (Becker et al., 2014).

Keeping these issues in mind, we could take a different angle on practice adoption research. When an evidently or presumably superior practice is not adopted or abandoned after initial adoption, despite good intentions to adopt and use it, there might be system 1-related processes in play that cannot be easily explained using traditional empirical research.

Without looking into the social, cognitive or behavioral aspects of individuals' acceptance behavior, conclusions about practice adoption might be drawn prematurely. Looking into social, cognitive and behavioral aspects might help us to distinguish issues directly related to the practice from issues related to the realization of behavioral intentions to adopt/use it. The models and theories discussed here can, for example, be used to inform the study design and data collection at an early stage. The models and theories may help determine the kind of data that is needed and how it may be collected to calculate suitable values for the constructs involved. The models and theories can, furthermore, inform the data analysis phase and the type of conclusions that can be drawn. To reap these benefits, it is necessary to gain more experience from using the models and theories and eventually create valid and reliable data collection instruments.

We foresee the following potentially useful future research directions:

- Integration of theories: Since there are a number of models and theories that already have been successfully used in the research of adoption behavior in the context of software development, gaining more experience should not primarily drive a further proliferation of constructs and their operationalizations (see discussion in Section 4.10). Instead, we would argue either for using models and theories that have been used to some degree in our field already (e.g., the ones discussed in Section 5) or for further unification of models and theories, as, for example, in recent developments of UTAUT (Blut et al., 2021; Tamilmani et al., 2021b; Dwivedi et al., 2019; Venkatesh et al., 2016). Integrative theories are also suggested in the health sciences (Hagger et al., 2020, Chp. 15), e.g., the Integrated Behavior Model (Montaño and Kasprzyk, 2015).
- Terminology alignment: There is an evident need for more consistent terminology as the definitions of the terms depend on the specific discipline and sometimes even the particular model or theory that uses them.
- Construct operationalization and their validation: Often, surveys are used to operationalize the data collection for the various constructs in the theories and models. However, there is a need to support the development and validation of such survey instruments in the software engineering context.
- Measurement of system 1-laden processes: As discussed in Section 6 relying solely on surveys and respondents' self-assessment for perception-based phenomena has limitations. Therefore, future research may use other non-invasive objective psychometrics (see, e.g., Graziotin et al. (2021)) and conduct more observational or ethnographic studies.
- Develop guidelines: Software engineering researchers need guidance to use/test the models and theories of acceptance behavior.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgment

This work has been supported by ELLIIT, Sweden, the Strategic Research Area within IT and Mobile Communications, funded by the Swedish Government.

References

Adriaanse, L.S., Rensleigh, C., 2013. Web of science, scopus and google scholar: A content comprehensiveness comparison. Electron. Libr..

Ajzen, I., 1991a. The theory of planned behavior. Organ. Behav. Hum. Decis. Process. 50 (2), 179–211.

Ajzen, I., 1991b. The theory of planned behavior. Organ. Behav. Hum. Decis. Process. 50 (2), 179–211.

Ajzen, I., 2015. The theory of planned behaviour is alive and well, and not ready to retire: A commentary on Sniehotta, Presseau and Araújo-Soares. Health Psychol. Rev. 9 (2), 131–137.

Ajzen, I., 2020. The theory of planned behavior: Frequently asked questions. Hum. Behav. Emerg. Technol. 2 (4), 314–324.

Alexandre, B., Reynaud, E., Osiurak, F., Navarro, J., 2018. Acceptance and acceptability criteria: a literature review. Cogn. Technol. Work 20 (2), 165–177.

Ali, N.b., 2016. Is effectiveness sufficient to choose an intervention? Considering resource use in empirical software engineering. In: Proceedings of the 10th International Symposium on Empirical Software Engineering and Measurement. pp. 1–6.

Ali, N.b., Tanveer, B., 2022. A comparison of citation sources for reference and citation-based search in systematic literature reviews. E-Inform. Softw. Eng. J. 16 (1).

Ali, M., Zhou, L., Miller, L., Ieromonachou, P., 2016. User resistance in IT: A literature review. Int. J. Inf. Manage. 36 (1), 35–43.

Ampatzoglou, A., Bibi, S., Avgeriou, P., Verbeek, M., Chatzigeorgiou, A., 2019. Identifying, categorizing and mitigating threats to validity in software engineering secondary studies. Inf. Softw. Technol. 106, 201–230.

Angie, A.D., Connelly, S., Waples, E.P., Kligyte, V., 2011. The influence of discrete emotions on judgement and decision-making: A meta-analytic review. Cogn. Emot. 25 (8), 1393–1422.

Aquilani, B., Silvestri, C., Ruggieri, A., Gatti, C., 2017. A systematic literature review on total quality management critical success factors and the identification of new avenues of research. TQM J. 29 (81), 184–213.

Archambault, E., Beauchesne, O.H., Caruso, J., 2011. Towards a multilingual, comprehensive and open scientific journal ontology. In: Proceedings of the 13th International Conference of the International Society for Scientometrics and Informetrics. pp. 66–77.

Bandura, A., 2001. Social cognitive theory: An agentic perspective. Annu. Rev. Psychol. 52, 1–26.

Basili, V.R., 1996. The role of experimentation in software engineering: past, current, and future. In: Proceedings of the 18th International Conference on Software Engineering. pp. 442–449.

Bazerman, M.H., Moore, D.A., 1998. Judgment in Managerial Decision Making, fifth ed. John Wiley & Sons, New York, NY, USA.

Becker, M., Vignoles, V.L., Owe, E., Easterbrook, M.J., Brown, R., Smith, P.B., Bond, M.H., Regalia, C., Manzi, C., Brambilla, M., et al., (40 authors), 2014. Cultural bases for self-evaluation: Seeing oneself positively in different cultural contexts. Personal. Soc. Psychol. Bull. 40 (5), 657–675.

Beecham, S., Baddoo, N., Hall, T., Robinson, H., Sharp, H., 2008. Motivation in software engineering: A systematic literature review. Inf. Softw. Technol. 50 (9–10), 860–878.

Belletier, C., Robert, A., Moták, L., Izaute, M., 2018. Toward explicit measures of intention to predict information system use: An exploratory study of the role of implicit attitudes. Comput. Hum. Behav. 86, 61–68.

Bjarnason, E., Smolander, K., Engström, E., Runeson, P., 2016. A theory of distances in software engineering. Inf. Softw. Technol. 70, 204–219.

Blackwell, A.F., Petre, M., Church, L., 2019. Fifty years of the psychology of programming. Int. J. Hum.-Comput. Stud. 131, 52–63.

Blut, M., Chong, A., Tsiga, Z., Venkatesh, V., 2021. Meta-analysis of the unified theory of acceptance and use of technology (UTAUT): Challenging its validity and charting a Research Agenda In The Red Ocean. J. Assoc. Inf. Syst. forthcoming.

Breckler, S.J., 1984. Empirical validation of affect, behavior, and cognition as distinct components of attitude. J. Personal. Soc. Psychol. 47 (6), 1191–1205.

Carrington, M.J., Neville, B.A., Whitwell, G.J., 2010. Why ethical consumers don't walk their talk: Towards a framework for understanding the gap between the ethical purchase intentions and actual buying behaviour of ethically minded consumers. J. Bus. Ethics 97 (1), 139–158.
Chattopadhyay, S., Nelson, N., Au, A., Morales, N., Sanchez, C., Pandita, R., Sarma, A., 2020. A tale from the trenches: cognitive biases and software development. In: Rothermel, G., Bae, D. (Eds.), Proceedings of the 42nd International Conference on Software Engineering. ACM, pp. 654–665.
Conner, M., 2015. Extending not retiring the theory of planned behaviour: A commentary on Sniehotta, Presseau and Araújo-Soares. Health Psychol. Rev. 9 (2), 141–145.
Conner, M., Norman, P., 2015. Predicting and Changing Health Behaviour: Research and Practice with Social Cognition Models, third ed. Open University Press.
Conner, M., Sparks, P., 2005. Theory of planned behavior and health behavior. In: Conner, M., Norman, P. (Eds.), Predicting Health Behavior, second ed. Open University Press, pp. 170–222.
Dane, E., 2011. Capturing intuitions 'in flight': Observations from research on attention and mindfulness. In: Sinclair, M. (Ed.), Handbook of Intuition Research. Edward Elgar Publishing, Cheltenham, UK, pp. 217–226.
Darlow, A.L., Sloman, S.A., 2010. Two systems of reasoning: Architecture and relation to emotion. Wiley Interdiscip. Rev.: Cogn. Sci. 1 (3), 382–392.
Davis, F.D., 1989. Perceived usefulness, perceived ease of use, and user acceptance of information technology. MIS Q. 319–340.
Davis, F.D., Bagozzi, R.P., Warshaw, P.R., 1992. Extrinsic and intrinsic motivation to use computers in the workplace. 22 (14), 1111–1132.
Davis, R., Campbell, R., Hildon, Z., Hobbs, L., Michie, S., 2015. Theories of behaviour and behaviour change across the social and behavioural sciences: A scoping review. Health Psychol. Rev. 9 (3), 323–344.
DeLone, W.H., McLean, E.R., 1992. Information systems success: The quest for the dependent variable. Inf. Syst. Res. 3 (1), 60–95.
DeLone, W.H., McLean, E.R., 2003. The DeLone and McLean model of information systems success: A ten-year update. J. Manage. Inf. Syst. 19 (4), 9–30.
Dessart, F.J., Barreiro-Hurlé, J., van Bavel, R., 2019. Behavioural factors affecting the adoption of sustainable farming practices: a policy-oriented review. Eur. Rev. Agric. Econ. 46 (3), 417–471.
Devanbu, P., Zimmermann, T., Bird, C., 2016. Belief & evidence in empirical software engineering. In: Proceedings of the 38th International Conference on Software Engineering. pp. 108–119.
Dishaw, M.T., Strong, D.M., 1998. Assessing software maintenance tool utilization using task-technology fit and fitness-for-use models. J. Softw. Maint.: Res. Pract. 10 (3), 151–179.
Dishaw, M.T., Strong, D.M., 1999. Extending the technology acceptance model with task–technology fit constructs. Inf. Manage. 36 (1), 9–21.
Dunning, D., Heath, C., Suls, J.M., 2004. Flawed self-assessment: Implications for health, education, and the workplace. Psychol. Sci. Public Interest 5 (3), 69–106.
Dwivedi, Y.K., Rana, N.P., Jeyaraj, A., Clement, M., Williams, M.D., 2019. Re-examining the unified theory of acceptance and use of technology (UTAUT): Towards a revised theoretical model. Inf. Syst. Front. 21 (3), 719–734.
Dybå, T., 2005. An empirical investigation of the key factors for success in software process improvement. IEEE Trans. Softw. Eng. 31 (5), 410–424.
Dybå, T., Kitchenham, B., Jørgensen, M., 2005. Evidence-based software engineering for practitioners. IEEE Softw. 22 (1), 58–65.
Dybå, T., Sjøberg, D.I., Cruzes, D.S., 2012. What works for whom, where, when, and why? On the role of context in empirical software engineering. In: Proceedings of the 6th International Symposium on Empirical Software Engineering and Measurement. pp. 19–28.
Epstein, S., 1994. Integration of the cognitive and the psychodynamic unconscious. Am. Psychol. 49 (8), 709.
Evans, J.S.B., 2008. Dual-processing accounts of reasoning, judgment, and social cognition. Annu. Rev. Psychol. 59, 255–278.
Evans, J.S.B., 2011. Dual-process theories of reasoning: Contemporary issues and developmental applications. Dev. Rev. 31 (2–3), 86–102.
Fagerholm, F., Felderer, M., Fucci, D., Unterkalmsteiner, M., Marculescu, B., Martini, M., Tengberg, L.G.W., Feldt, R., Lehtelä, B., Nagyváradi, B., et al., 2022. Cognition in software engineering: A taxonomy and survey of a half-century of research. ACM Comput. Surv. http://dx.doi.org/10.1145/3508359, Accepted Dec 2021.
Ferratt, T.W., Prasad, J., Dunne, E.J., 2018. Fast and slow processes underlying theories of information technology use. J. Assoc. Inf. Syst. 19 (1), 3.
Fishbein, M., Ajzen, I., 1975. Belief, Attitude, Intention and Behavior: An Introduction to Theory and Research. Addison-Wesley, Reading, MA.
Fishbein, M., Ajzen, I., 2010. Predicting and Changing Behavior: The Reasoned Action Approach. Taylor & Francis.
França, C., Da Silva, F.Q., Sharp, H., 2020. Motivation and satisfaction of software engineers. IEEE Trans. Softw. Eng. 46 (2), 118–140.
França, A.C.C., Gouveia, T.B., Santos, P.C., Santana, C.A., da Silva, F.Q., 2011. Motivation in software engineering: A systematic review update. In: Proceedings of the 15th Annual Conference on Evaluation & Assessment in Software Engineering. pp. 154–163.
Gollwitzer, P.M., 1999. Implementation intentions: Strong effects of simple plans. Am. Psychol. 54 (7), 493–503.
Goodhue, D.L., Thompson, R.L., 1995. Task-technology fit and individual performance. MIS Q. 19 (2), 213–236.
Graziotin, D., Lenberg, P., Feldt, R., Wagner, S., 2021. Psychometrics in behavioral software engineering: A methodological introduction with guidelines. ACM Trans. Softw. Eng. Methodol. 31 (1), 1–36.
Hadar, I., Hasson, T., Ayalon, O., Toch, E., Birnhack, M., Sherman, S., Balissa, A., 2018. Privacy by designers: software developers' privacy mindset. Empir. Softw. Eng. 23 (1), 259–289.
Hagger, M.S., Cameron, L.D., Hamilton, K., Hankonen, N., Lintunen, T., 2020. The Handbook of Behavior Change. Cambridge University Press.
Hagger, M.S., Hamilton, K., 2020. Changing behavior using integrated theories. In: Hagger, M.S., Cameron, L.D., Hamilton, K., Hankonen, N., Lintunen, T. (Eds.), The Handbook of Behavior Change. Cambridge University Press, pp. 208–224.
Hall, T., Baddoo, N., Beecham, S., Robinson, H., Sharp, H., 2009. A systematic review of theory use in studies investigating the motivations of software engineers. ACM Trans. Softw. Eng. Methodol. 18 (3), 1–29.
Hall, J.G., Rapanotti, L., 2017. A design theory for software engineering. Inf. Softw. Technol. 87, 46–61.
Howe, L.C., Krosnick, J.A., 2017. Attitude strength. Annu. Rev. Psychol. 68 (1), 327–351.
ISO/IEC, 2017. ISO/IEC/IEEE 24765:2017 Systems and Software Engineering – Vocabulary. Standard, International Organization for Standardization, Geneva, Switzerland.
Jeyaraj, A., 2020. DeLone & McLean models of information system success: Critical meta-review and research directions. Int. J. Inf. Manage. 54, 102139.
Jeyaraj, A., Rottman, J.W., Lacity, M.C., 2006. A review of the predictors, linkages, and biases in IT innovation adoption research. J. Inf. Technol. 21 (1), 1–23.
Johnson, P., Ekstedt, M., 2016. The Tarpit — A general theory of software engineering. Inf. Softw. Technol. 70, 181–203.
Johnson, P., Ekstedt, M., Jacobson, I., 2012. Where's the Theory for Software Engineering? IEEE Softw. 29 (5), 96.
Kahneman, D., 2003. A perspective on judgment and choice: mapping bounded rationality. Am. Psychol. 58 (9), 697.
Kahneman, D., 2011. Thinking, Fast and Slow. Farrar Straus Giroux.
Kahneman, D., Klein, G., 2009. Conditions for intuitive expertise: a failure to disagree. Am. Psychol. 64 (6), 515.
Karanatsiou, D., Li, Y., Arvanitou, E.-M., Misirlis, N., Wong, W.E., 2019. A bibliometric assessment of software engineering scholars and institutions (2010–2017). J. Syst. Softw. 147, 246–261.
Khan, I.A., Brinkman, W.-P., Hierons, R.M., 2011. Do moods affect programmers' debug performance? Cogn. Technol. Work 13 (4), 245–258.
Khan, A.A., Keung, J., Niazi, M., Hussain, S., Ahmad, A., 2017. Systematic literature review and empirical investigation of barriers to process improvement in global software development. Inf. Softw. Technol. 87 (C), 180–205.
Kitchenham, B., Charters, S., 2007. Guidelines for Performing Systematic Literature Reviews in Software Engineering. Report EBSE-2007-1, Keele University.
Lavallée, M., Robillard, P.N., 2012. The impacts of software process improvement on developers: A systematic review. In: Proceedings of the 34th International Conference on Software Engineering. pp. 113–122.
Law, E.L., Thanydait, S., Heintz, M., Campbell, A., Wild, F., 2021. Beyond questionnaires: Innovative approaches to evaluating mixed reality. In: 34th British HCI Workshop and Doctoral Consortium 34. pp. 1–5.
Lenberg, P., Feldt, R., Wallgren, L.G., 2015. Behavioral software engineering: A definition and systematic literature review. J. Syst. Softw. 107, 15–37.
Lenberg, P., Tengberg, L.G.W., Feldt, R., 2017. An initial analysis of software engineers' attitudes towards organizational change. Empir. Softw. Eng. 22 (4), 2179–2205.
Lerner, J.S., Li, Y., Valdesolo, P., Kassam, K.S., 2015. Emotion and decision making. Annu. Rev. Psychol. 66 (1).
Locke, E.A., Latham, G.P., 1990. Work motivation and satisfaction: Light at the end of the tunnel. Psychol. Sci. 1 (4), 240–246.
Lorey, T., Ralph, P., Felderer, M., 2022. Social science theories in software engineering research. In: Proceedings of the 44th International Conference on Software Engineering. pp. 1994–2005.
Marangunić, N., Granić, A., 2015. Technology acceptance model: a literature review from 1986 to 2013. Univ. Access Inf. Soc. 14 (1), 81–95.
Metcalfe, J., Mischel, W., 1999. A hot/cool-system analysis of delay of gratification: dynamics of willpower. Psychol. Rev. 106 (1), 3.
Moed, H.F., Bar-Ilan, J., Halevi, G., 2016. A new methodology for comparing Google Scholar and Scopus. J. Informetr. 10 (2), 533–551.

Mohammadi, E., Gregory, K.B., Thelwall, M., Barahmand, N., 2020. Which health and biomedical topics generate the most facebook interest and the strongest citation relationships? Inf. Process. Manage. 57 (3), 102230.

Mohanani, R., Salman, I., Turhan, B., Rodríguez, P., Ralph, P., 2018. Cognitive biases in software engineering: A systematic mapping study. Trans. Softw. Eng. 46 (12), 1318–1339.

Montaño, D.E., Kasprzyk, D., 2015. Theory of reasoned action, theory of planned behavior, and the integrated behavioral model. In: Glanz, K., Rimer, B.K., Viswanath, K. (Eds.), Health Behavior: Theory, Research, and Practice, fifth ed. John Wiley & Sons, pp. 95–124.

Moore, G.C., Benbasat, I., 1991. Development of an instrument to measure the perceptions of adopting an information technology innovation. Inf. Syst. Res. 2 (3), 192–222.

Nurdiani, I., Börstler, J., Fricker, S., Petersen, K., 2019. Usage, retention, and abandonment of agile practices: A survey and interviews results. E-Inform. Softw. Eng. J. 13 (1), 7–35.

Pano, A., Graziotin, D., Abrahamsson, P., 2018. Factors and actors leading to the adoption of a JavaScript framework. Empir. Softw. Eng. 23 (6), 3503–3534.

Passos, C., Cruzes, D.S., Hayne, A., Mendonça, M., 2013. Recommendations to the adoption of new software practices: A case study of team intention and behavior in three software companies. In: Proceedings of the 7th International Symposium on Empirical Software Engineering and Measurement. pp. 313–322.

Petersen, K., Ali, N.b., 2021. An analysis of top author citations in software engineering and a comparison with other fields. Scientometrics 126 (11), 9147–9183.

Petter, S., McLean, E.R., 2009. A meta-analytic assessment of the DeLone and McLean IS success model: An examination of IS success at the individual level. Inf. Manage. 46 (3), 159–166.

Polančič, G., Heričko, M., Rozman, I., 2010. An empirical examination of application frameworks success based on technology acceptance model. J. Syst. Softw. 83 (4), 574–584.

Rad, M.S., Nilashi, M., Dahlan, H.M., 2018. Information technology adoption: A review of the literature and classification. Univ. Access Inf. Soc. 17 (2), 361–390.

Ralph, P., 2016. Software engineering process theory: A multi-method comparison of sensemaking–coevolution–implementation theory and function–behavior–structure theory. Inf. Softw. Technol. 70, 232–250.

Riemenschneider, C.K., Hardgrave, B.C., Davis, F.D., 2002. Explaining software developer acceptance of methodologies: a comparison of five theoretical models. IEEE Trans. Softw. Eng. 28 (12), 1135–1145.

Rogers, E.M., 2003. Diffusion of Innovations, fifth ed. The Free Press, New York, NY, USA.

Rogers, E.M., Burdge, R.J., Korsching, P.F., 1983. Diffusion of Innovations, third ed. The Free Press A Division of Mc Millan Publishing Co. Inc.

Sabherwal, R., Jeyaraj, A., Chowa, C., 2006. Information system success: Individual and organizational determinants. Manage. Sci. 52 (12), 1849–1864.

Sánchez-Gordón, M., Colomo-Palacios, R., 2019. Taking the emotional pulse of software engineering – A systematic literature review of empirical studies. Inf. Softw. Technol. 115, 23–43.

Schmid, K., 2021. If you want better empirical research, value your theory: On the importance of strong theories for progress in empirical software engineering research. In: Proceedings of the 25th Conference on Evaluation and Assessment in Software Engineering. pp. 359–364.

Seuwou, P., Banissi, E., Ubakanma, G., 2017. User acceptance of information technology: A critical review of technology acceptance models and the decision to invest in information security. In: Proceedings of the International Conference on Global Security, Safety, and Sustainability. pp. 230–251.

Shaft, T.M., Vessey, I., 2006. The role of cognitive fit in the relationship between software comprehension and modification. MIS Q. 29–55.

Sharp, H., Baddoo, N., Beecham, S., Hall, T., Robinson, H., 2009. Models of motivation in software engineering. Inf. Softw. Technol. 51 (1), 219–233.

Shaw, H., Ellis, D.A., Ziegler, F.V., 2018. The technology integration model (TIM). Predicting the continued use of technology. Comput. Hum. Behav. 83, 204–214.

Sheeran, P., 2002. Intention–behavior relations: A conceptual and empirical review. Eur. Rev. Soc. Psychol. 12 (1), 1–36.

Shiffrin, R.M., Schneider, W., 1977. Controlled and automatic human information processing: II. Perceptual learning, automatic attending and a general theory. Psychol. Rev. 84 (2), 127.

Simon, H.A., 1968. Judgment in Managerial Decision Making. Macmillan, New York, NY, USA.

Sniehotta, F.F., Presseau, J., Araújo-Soares, V., 2014. Time to retire the theory of planned behaviour. Health Psychol. Rev. 8 (1), 1–7.

Sojer, M., Alexy, O., Kleinknecht, S., Henkel, J., 2014. Understanding the drivers of unethical programming behavior: The inappropriate reuse of internet-accessible code. J. Manage. Inf. Syst. 31 (3), 287–325.

St Quinton, T., Morris, B., Trafimow, D., 2021. Untangling the Theory of Planned Behavior's auxiliary assumptions and theoretical assumptions: Implications for predictive and intervention studies. New Ideas Psychol. 60, 100818.

Stanovich, K.E., West, R.F., 2000. Individual differences in reasoning: Implications for the rationality debate? Behav. Brain Sci. 23 (5), 645–665.

Stanovich, K.E., West, R.F., 2002. Individual differences in reasoning: Implications for the rationality debate? In: Gilovich, T., Griffin, D., Kahneman, D. (Eds.), Heuristics and Biases: The Psychology of Intuitive Judgment. Cambridge University Press, New York, NY, USA, pp. 421–440.

Steers, R.M., Mowday, R.T., Shapiro, D.L., 2004. The future of work motivation theory. Acad. Manag. Rev. 29 (3), 379–387.

Stelzer, D., Mellis, W., 1998. Success factors of organizational change in software process improvement. Softw. Process: Improvement Pract. 4 (4), 227–250.

Stol, K.-J., Fitzgerald, B., 2015. Theory-oriented software engineering. Sci. Comput. Program. 101, 79–98.

Taherdoost, H., 2018. A review of technology acceptance and adoption models and theories. Procedia Manuf. 22, 960–967.

Tamilmani, K., Rana, N.P., Dwivedi, Y.K., 2017. A systematic review of citations of UTAUT2 article and its usage trends. In: Proceedings of the Conference on E-Business, E-Services and E-Society (I3E). pp. 38–49.

Tamilmani, K., Rana, N.P., Dwivedi, Y.K., 2018. Use of 'habit' is not a habit in understanding individual technology adoption: A review of UTAUT2 based empirical studies. In: Proceedings of the International Working Conference on Transfer and Diffusion of IT. pp. 277–294.

Tamilmani, K., Rana, N.P., Dwivedi, Y.K., 2021a. Consumer acceptance and use of information technology: A meta-analytic evaluation of UTAUT2. Inf. Syst. Front. 23 (4), 987–1005.

Tamilmani, K., Rana, N.P., Wamba, S.F., Dwivedi, R., 2021b. The extended unified theory of acceptance and use of technology (UTAUT2): A systematic literature review and theory evaluation. Int. J. Inf. Manage. 57, 102269.

Taylor, D., Bury, M., Campling, N., Carter, S., Garfied, S., Newbould, J., Rennie, T., 2006. A Review of the use of the Health Belief Model (HBM), the Theory of Reasoned Action (TRA), the Theory of Planned Behaviour (TPB) and the Trans-Theoretical Model (TTM) to Study and Predict Health Related Behaviour Change. Technical Report, National Institute for Health and Clinical Excellence, London, UK.

Thaler, R.H., Shefrin, H.M., 1981. An economic theory of self-control. J. Polit. Econ. 89 (2), 392–406.

Thompson, R.L., Higgins, C.A., Howell, J.M., 1991. Personal computing: Toward a conceptual model of utilization. MIS Q. 125–143.

Tiefenbeck, V., Kupfer, A., Ableitner, L., Schöb, S., Staake, T., 2016. The uncertain path from good intentions to actual behavior: A field study on mobile app usage. In: Proceedings of the Twenty-Sixth DIGIT Workshop. p. 17.

Trafimow, D., 2015. On retiring the TRA/TPB without retiring the lessons learned: A commentary on Sniehotta, Presseau and Araújo-Soares. Health Psychol. Rev. 9 (2), 168–171.

Turner, M., Kitchenham, B., Brereton, P., Charters, S., Budgen, D., 2010. Does the technology acceptance model predict actual use? A systematic literature review. Inf. Softw. Technol. 52 (5), 463–479.

Unterkalmsteiner, M., Gorschek, T., Islam, A.M., Cheng, C.K., Permadi, R.B., Feldt, R., 2011. Evaluation and measurement of software process improvement – a systematic literature review. IEEE Trans. Softw. Eng. 38 (2), 398–424.

Venkatesh, V., Bala, H., 2008. Technology acceptance model 3 and a research agenda on interventions. Decis. Sci. 39 (2), 273–315.

Venkatesh, V., Davis, F.D., 2000. A theoretical extension of the technology acceptance model: Four longitudinal field studies. Manage. Sci. 46 (2), 186–204.

Venkatesh, V., Morris, M.G., Davis, G.B., Davis, F.D., 2003a. User acceptance of information technology: Toward a unified view. MIS Q. 27 (3), 425–478.

Venkatesh, V., Morris, M.G., Davis, G.B., Davis, F.D., 2003b. User acceptance of information technology: Toward a unified view. MIS Q. 425–478.

Venkatesh, V., Thong, J.Y., Xu, X., 2012. Consumer acceptance and use of information technology: Extending the unified theory of acceptance and use of technology. MIS Q. 36 (1), 157–178.

Venkatesh, V., Thong, J.Y., Xu, X., 2016. Unified theory of acceptance and use of technology: A synthesis and the road ahead. J. Assoc. Inf. Syst. 17 (5), 328–376.

Vessey, I., Galletta, D., 1991. Cognitive fit: An empirical study of information acquisition. Inf. Syst. Res. 2 (1), 63–84.

Wallace, L.G., Sheetz, S.D., 2014. The adoption of software measures: A technology acceptance model (TAM) perspective. Inf. Manage. 51 (2), 249–259.

Wallgren, L.G., Hanse, J.J., 2007. Job characteristics, motivators and stress among information technology consultants: A structural equation modeling approach. Int. J. Ind. Ergon. 37 (1), 51–59.

Wan, Z., Xia, X., Hassan, A.E., Lo, D., Yin, J., Yang, X., 2020. Perceptions, expectations, and challenges in defect prediction. IEEE Trans. Softw. Eng. 46 (11), 1241–1266.

Webb, T.L., Sheeran, P., 2006. Does changing behavioral intentions engender behavior change? A meta-analysis of the experimental evidence. Psychol. Bull. 132 (2), 249–268.

Weinberg, G.M., 1971. The Psychology of Computer Programming, Vol. 29. Van Nostrand Reinhold, New York, NY, USA, A silver anniversary edition was published by Dorset House 1998.

Whitehead, D., Russell, G., 2004. How effective are health education programmes—resistance, reactance, rationality and risk? Recommendations for effective practice. Int. J. Nurs. Stud. 41 (2), 163–172.

Williams, M., Rana, N., Dwivedi, Y., Lal, B., 2011. Is UTAUT really used or just cited for the sake of it? A systematic review of citations of UTAUT's originating article. In: Proceedings of the European Conference on Information Systems. p. 231.

Wisdom, J.P., Chor, K.H.B., Hoagwood, K.E., Horwitz, S.M., 2014. Innovation adoption: A review of theories and constructs. Adm. Policy Ment. Health Ment. Health Serv. Res. 41 (4), 480–502.

Wohlin, C., mite, D., Moe, N.B., 2015. A general theory of software engineering: Balancing human, social and organizational capitals. J. Syst. Softw. 109, 229–242.

Wu, J., Du, H., 2012. Toward a better understanding of behavioral intention and system usage constructs. Eur. J. Inf. Syst. 21 (6), 680–698.

Yousafzai, S.Y., Foxall, G.R., Pallister, J.G., 2007. Technology acceptance: a meta-analysis of the TAM: Part 1. J. Model. Manage. 2 (3), 251–280.

Zelkowitz, M.V., Wallace, D.R., Binkley, D.W., 2003. Experimental validation of new software technology. In: Lecture Notes on Empirical Software Engineering. World Scientific, pp. 229–263.