DEVELOPING AN INSTRUMENT TO MEASURE PERCEIVED INPUT CONTROL ON ONLINE PLATFORMS FROM THE APP DEVELOPER PERSPECTIVE

Research in Progress

Croitor, Evgheni, Technische Universität Darmstadt, Germany, croitor@ise.tu-darmstadt.de

Abstract

Over the last decade, applications of third-party app developers have increasingly become the cornerstone of software platform ecosystems' success and sustainability. Given their importance, the procedures and practices used on platforms to screen and sort out applications are crucial means to regulate their influx. Although traditional control mechanisms applied on software-based platforms have been widely studied in Information Systems (IS) research, there is still a lack of research on input control and how it is conceptualized and measured from an app developer perspective. In this research-in-progress paper, we present the initial stages of a scale development process and present an initial construct to measure perceived input control from an application developers' perspective. We accomplish this by first providing a structured literature review across the leading IS and management journals and conference papers. Based on this review and subsequent open-ended expert interviews, we developed a preliminary measurement instrument that not only captures app developers' overall perceptions of input control across different platform contexts, but also breaks these perceptions down into distinct input control factors. The results of our study are expected to contribute to a more comprehensive understanding of input control in general and platform-specific input control mechanisms in particular.

Keywords: Input Control, Scale Development, Literature Review, Platform Ecosystems

1 Introduction

During the past decade, software platforms and their corresponding ecosystems have fundamentally changed the way software-based applications and services are developed and distributed (Jansen et al., 2009). Platform operators deliberately open up their ecosystems and enable external, third-party developers to add functionality to the core product of the platform (Boudreau, 2012; Thies et al., 2016). These complementary subsystems are called apps and their developers 'app developers' (Tiwana, 2014). Furthermore, enabling apps had leverage on app developers' ingenuity, innovative capacity and skills (Ceccagnoli et al., 2012) to respond to rapidly changing markets and customer needs (Boudreau and Lakhan, 2009). A software platform is thereby defined as "*the extensible codebase of a software-based system that provides core functionality shared by the modules that interoperate with it and the interfaces through which they interoperate*" (Tiwana et al., 2010, p. 675). More broadly, and taking into account the various stakeholders participating in and contributing to a software platform, software ecosystems refer to the "*software and actor interaction in relation to a common technological infrastructure that results in a set of contributions and influences directly or indirectly the ecosystem*" (Manikas, 2016, p. 93).

A particularly critical interaction between app developers and a software platform is the ongoing supply of high-quality complements that are supposed to attract users to the platform to foment positive cross-side network effects (Boudreau, 2012). In order to sustain platform prosperity and health, a key challenge for platform operators is to align and harmonize the numerous and diverse goals and behaviors of app developers with the platform's strategies. Constantly attracting new developers or avoiding that existing ones churn to rival platforms are important goals for the platform's long-term viability and success (Benlian et al., 2015). Failing to do so may otherwise lead to the demise of a platform, as witnessed with platforms such as Nokia or Blackberry that missed among other problems to create and manage a persistent pipeline of high-quality apps (Tiwana, 2014). Against this backdrop, platform operators are well advised to nurture and shape an ecosystem that encourages app developers to continuously contribute innovative, useful and high-quality complements to the platform.

Control modes that help coordinate interactions between the various players on platforms are a central building block of platform governance, and platform providers exercise various forms of formal and informal control to influence behaviors and performance outcomes of app developers (Tiwana et al., 2010). Classical control modes (i.e., output, process, clan, and self-control) have been widely studied in various contexts such as IT projects, IT outsourcing, and on software platforms (Goldbach et al., 2017), and have yielded valuable findings. However, research on input control and how it is conceptualized and measured is conspicuously absent, even though IS scholars have repeatedly pointed to the importance of studying app screening and gatekeeping practices in the context of software platforms and how they affect app developers (Tiwana et al., 2010; Tiwana, 2015). Indeed, despite input control's apparent importance, the app developer perspective has been widely neglected in previous approaches that investigated input control (Tiwana, 2015; Wiener et al., 2017). In fact, there are as yet no validated measurement instruments for app developers' perceptions of concrete input control on online platforms.

Given these calls for research and the research gaps identified above, this paper aims to contribute to a richer understanding of app developers' perceptions of input control and how they relate to important individual downstream factors, such as satisfaction, commitment, and app quality. Hence, we ask the following research question:

RQ: What is the conceptual definition of perceived input control and how can it be measured?

The remainder of this research-in-progress paper is organized as follows. The next section presents the theoretical background and related IS literature on control mechanisms. Further, we describe our scale development process and present its preliminary results. Finally, we define future steps and discuss expected theoretical and practical contributions.

2 Theoretical Background and Related Literature

Control is the main building block of governance in platform ecosystems, therefore several control mechanisms are currently implemented by platform providers (Tiwana et al., 2010). We define Control as a controller's attempts to influence a controlees behaviour according to the controller's goals (Ouchi, 1979). Two main categories of control mechanisms are distinguished in extant literature, which are formal and informal modes of control (Kirsch, 1997). Formal control is commonly distinguished into output and process control. In terms of output control, output requirements and performance targets are pre-specified as objectives, which are then monitored, evaluated and rewarded accordingly. By contrast, under process control, no specific outcomes are pre-determined and are therefore free-to-be-chosen.

Informal control is categorized into self-control and clan control (Bergvall-Kåreborn and Howcroft, 2011). With self-control, controllers encourage individuals to set their own goals and self-regulate their activities and outcomes in achieving these goals (Henderson and Lee, 1992). In terms of clan control, members of a group commit themselves to mutual beliefs and goals and therefore often tend to engage in similar behaviors and produce comparable outcomes, based on shared values and norms (Kirsch et al., 2010). Informal control modes are particularly relevant when desired outcomes and behaviors are unknown or difficult to monitor (Chua et al., 2012).

Formal and informal control modes (and their combinations) have been widely studied in IS research and have been identified as effective governance mechanisms to coordinate the controller-controlee relationship in classical IS contexts (Roberts et al., 2006; Chua et al., 2012; Gregory et al., 2013). The emergence of software platform ecosystems has, however, brought about fundamental changes to the classical controller-controllee relationship and has made the applicability of traditional and well-studied control modes rather difficult and less worthwhile (Tiwana, 2015). Given the limited practicability of traditional control modes, operators of software platforms often resort to a control mechanism that is widespread in practice but has been largely overlooked in IS research so far: input control. Input control is abstractly defined as "*adjudicating which complements are granted access to an ecosystem*" (Tiwana, 2015, p. 266). It is a form of formal control or "gatekeeping" that regulates which complements are allowed into a software ecosystem and which ones are rejected. It usually involves formal application and selection processes (Cardinal et al., 2004).

Changing input control can have far-reaching and profound effects on the entire ecosystem, given that opening or closing an ecosystem has a direct bearing on the amount and quality of complements entering the ecosystem and thus on the attractiveness of a platform's profile and offering to platform users (Wessel et al., 2017; Thies et al., 2018). Strong changes in input control may even have the potential to unbalance an entire software ecosystem as well as transform its character, as too liberal input control may lead to coordination failures and quality issues, while too stringent input control may thwart diversity and innovation in an ecosystem (de Reuver and Bouwman, 2012).

Despite the pervasiveness of its application in practice and its fundamental implications for software ecosystems and its key actors, it comes as a surprise that research on input control is still in its infancy. In previous IS research, input control has sometimes been referred to implicitly without being acknowledged theoretically as such. For example, Choudhury and Sabherwal (2003) point to the importance of assessing the quality delivered by information technology vendors as well as vendors' project staffing choices. The large neglect of input control in IS research is however likely because it was not as visibly observed in traditional IT (outsourcing/offshoring) projects as it is in platform settings. Only scant literature in management and organizational research has investigated input control explicitly but used inconsistent terminology and various definitions to refer to screening (Sah and Stiglitz, 1986), selection of employees from an applicant pool (Snell, 1992), and bouncer rights (Boudreau, 2010). Recent studies in software ecosystems have started to examine input control amongst other control modes and provided largely anecdotal and qualitative insights. Ghazawneh and Henfridsson (2013) and Eaton et al. (2015), for example, have looked at the relationship between control and boundary resources on Apple's iPhone platform, whereas Wareham et al. (2013) investigated the tension between control and autonomy in a business software ecosystem. Based on results of our literature review, there is only minimal research that has brought input control and its effects to the forefront of its investigations (Boudreau, 2012; Tiwana, 2015). Despite these valuable initial efforts, however, the nature (i.e., conceptualization) of input control in software ecosystems remained rather vague and its measurement inconsistent.

Taken together, contemporary IS research has largely neglected input control and those few studies that have looked at it have examined input control based on an undifferentiated and high-level conceptualization without shedding light on the core practices that constitute input control. This lack of a clear and substantial conceptualization calls for the development of a deeper theoretical foundation of input control. Besides rather global and relatively lean approaches to conceptualize and measure input control (e.g., Tiwana, 2015), previous research has largely focused on the macro (i.e., platform) level of analysis to study the implications of input control, but neglected a more nuanced analysis that captures the micro (i.e., individual) perspective of platform app developers who are immediately affected by input control.

Despite recent calls for investigating the relationship between input control and app developers' attitudes and behaviors and for identifying effective input control practices used by platform operators (Hilkert et al., 2011; Benlian et al., 2015; Eaton et al., 2015), almost no studies have empirically and systematically explored the nature and implications of different input control mechanisms in software ecosystems. In order to fill this research gap, we aim to develop a construct and measurement instrument that should not only provide a consistent conceptualization and definition of input control, but should also precisely capture app developers' general and more specific perceptions of input control. A common understanding and consistent measurement of perceived input control will ultimately help future research and practitioners better accumulate and consolidate knowledge about input control practices and their implications for app developers on platforms.

3 Scale Development

The primary objective of this study is to systematically and rigorously develop and validate a scale for perceived input control (PIC). Several approaches in developing scales have been proposed in IS research (Moore and Benbasat, 1991; Segars, 1997; Boudreau et al., 2001; MacKenzie et al., 2011). Following these guidelines, we plan to conduct five steps to create an instrument to measure PIC. Figure 1 summarizes the main steps in the scale development and validation process and depicts that the first two steps are part of this research-in-progress paper.

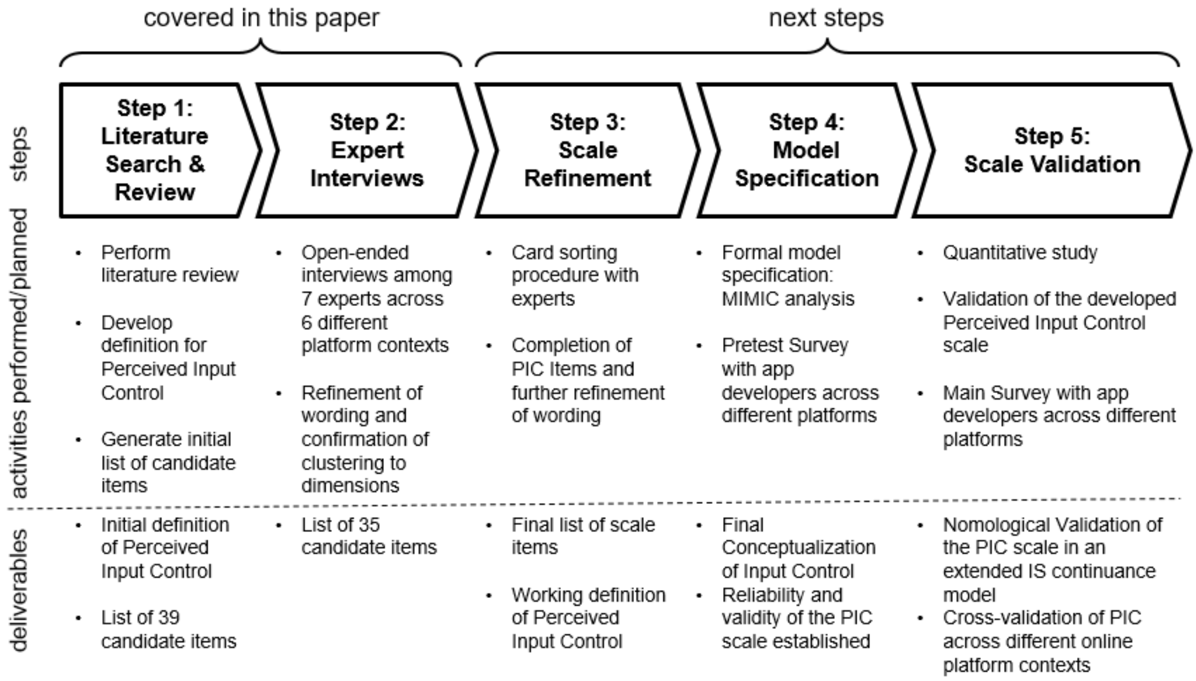

Figure 1. Overview of the Scale Development Process for Perceived Input Control.

In the first step, we performed a structured literature review and developed an initial working definition of PIC. Based on these results, we generated an initial list of candidate items. Second, we conducted expert interviews to purify and revise our list of items. A card-sorting method is going to be step three. Fourth, we are planning to formally specify the measurement model to evaluate and improve the scale based on a pretest survey across different platform contexts (e.g., app development, browser software, gaming platforms, crowdfunding) (step 4). The final step will include the validation of the developed input control scale in an extended nomological network. This network includes important antecedents and consequents of PIC and should be developed in later stages of the scale development process (see section Next Steps and Expected Contribution). In the following subsections, we present the results of the first two steps of the scale development process.

3.1 Development of Measures

In the first step, we performed a systematic literature review approach (Churchill, 1979; Webster and Watson, 2002) on input control and developed an initial list of scale items based on previous literature. In doing so, we followed the extant guidelines that have been presented in IS research (Okoli and Schabram, 2010; Boell and Cecez-Kecmanovic, 2015; Brocke et al., 2015).

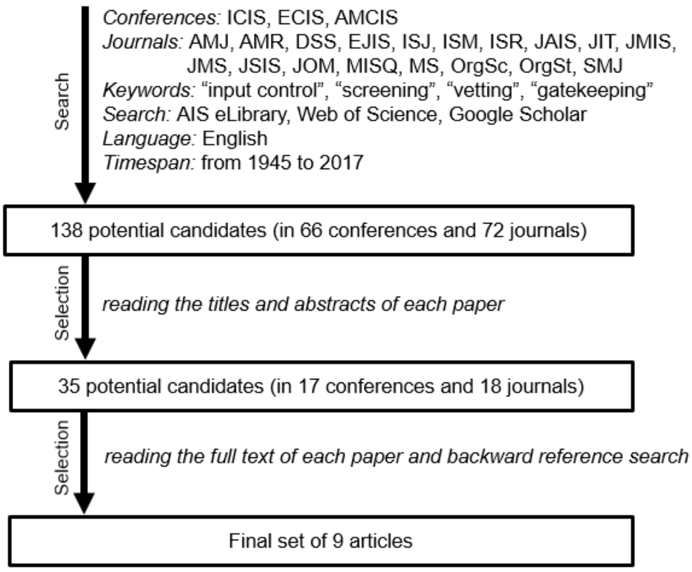

Figure 2. Search and Selection Process.

At the outset of our literature search, we included 17 highly-ranked IS and management journals (including the Senior Scholars’ Basket of IS Journals) and proceedings of three leading IS conferences (see Figure 2). In order to conduct the literature review, we used Web of Science, the AIS eLibrary and Google Scholar and applied keyword search using different combinations of related search terms such as “input control”, “screening”, “vetting” and “gatekeeping”. Our search was confined to articles written in English between 1945 and 2017. In this way, an initial pool of 138 search results was found. We divided the main selection process into two phases. After reading the titles and abstracts of each article in the first phase, we determined 103 irrelevant articles and excluded them from the pool. In the second phase of the selection process, we assessed the remaining 35 articles by reading the main content of the papers. We screened the resulting papers to keep only those that mentioned the term “Input Control” within the context of gatekeeping and screening process. We also performed backward reference search for relevant articles to identify other important documents. In doing so, we identified ten definitions of input control in nine articles (see Table 1).

Discipline	Definition	Author
Platform Ecosystems	*“We define input control as the degree to which a platform owner adjudicates allowing revisions of an extension into the ecosystem.”*	Tiwana (2015)
	“Formal control intended by the platform owner to regulate inputs into the ecosystem.”	Tiwana (2015)
	“Gatekeeping represents the degree to which the platform owner uses predefined objective acceptance criteria for judging what apps and app developers are allowed into a platform’s ecosystem.”	Tiwana (2014)
Project Organisation	*“Managing the resources that are used as inputs to project activities.”*	Wiener et al. (2017)
	“Mode of formal control that refers to the allocation and manipulation of human, financial, and material project resources.”	Jaworski (1988)
	“Input controls refer to the management of materials and human resources.”	Vlasic and Yetton (2004)
	“Input control is the use of control mechanisms to manage resources acquired by the firm; it focuses on human, material, and financial resources flowing into the firm.”	Cardinal et al. (2004)
	“Input control is seen as the control of a task through resources and other inputs, including selection and recruitment of personnel in relation to this task.”	Mähring (2002)

Human Resources	*"Input control system seek to control the selection and training process of an employee."*	Krausert (2009)
	"Input Control regulates the antecedent conditions of performance - the knowledge, skills, abilities, values and motives of employees."	Snell (1992)

Table 1. Definitions of Input Control identified by the literature review.

The next step of the literature review was to extract relevant demographic and research-related information. Demographic data includes title, author's name, year of publication, and publication type. Research-related data includes existing definitions of input control, as well as items measuring input control (if available). The definition of input control appeared mostly in the field of project organisation and human resources. Only few articles included platform-related definitions of input control (Tiwana, 2015). However, these articles transferred and adapted items from other organizational contexts (Cardinal, 2004), but did not engage in a thorough and rigorous scale development process in the context of online platforms.

As one of the main goals of this study is to systematically develop a conceptualization of perceived input control (PIC), we provide an initial definition based on our literature review and following Rossiter's (2002) recommendations on conceptualizing constructs by specifying the attribute of interest, the focal object and the rater entity. We consider the attribute of interest (i.e., input control), the focal object (i.e., online platforms), and the rater entity (i.e. third-party application developers) to define perceived input control as follows:

> *Perceived input control (PIC) is defined as third-party application developers' perceptions of the degree to which a platform provider uses gatekeeping and screening procedures to allow third-party developers' apps to enter a platform.*

Based on the initial definition of PIC, along with the principles of creating new construct items in the scale development literature (MacKenzie et al., 2011; DeVellis, 2012), we collected and modified 39 initial items measuring PIC as a candidate pool for future refinements. In particular, these items were used as a starting point for conducting expert interviews, which will be discussed in the next section.

3.2 Expert Interviews

The use of expert interviews in conjunction with a systematic literature review is a recommended method to create an initial set of candidate items. (Churchill, 1979) The use of these methods is believed to generate a list of candidate items with a relatively high degree of content validity (Moore and Benbasat, 1991). For this reason, and to help identify different dimensions and develop general perceptions of PIC, we performed open-ended expert interviews based on the literature review results of the first step.

The criteria to select our interview partners were led by the idea to cover the diversity of several different types of online platforms. Therefore, we drew on a convenience sample and interviewed a mobile app developer (Android/iOS/Windows), a game developer (Steam), a video producer (YouTube/MyVideo), two online retailers (Amazon), an extension developer (Chrome browser), and a fundraiser for a social project (Razoo/GoFundMe). Five of the interviews were conducted face-to-face and the remaining two by phone. All interview partners had less than two years of experience with online platforms, which was an important criterion for our interviews. We asked the interviewees to tell us their personal experience of submitting their first and last application to the platform. Applying open-ended interviews including think-aloud techniques enabled us to see perspectives we had not considered before and ensured us to dig deep into the target domain until we reached a satisfactory point of saturation (Hoffmann, 2007; Bogner et al., 2009).

As a key result, the interviews revealed that all interviewees had to undergo a more or less extensive online registration process on the platform before being able to submit their application. Although most platforms had only low requirements for the registration, some of them had a time-consuming screening process (e.g., Amazon, Steam). Moreover, the head of a social project failed to complete registration on several donation-based crowdfunding platforms due to burdensome developer requirements. These findings motivated us to divide PIC on platforms into two phases. The first phase is related to the developer-based screening process. It has to be passed through only once and includes app developers' registration and authentication on the platform, as well as all corresponding actions (i.e., paying registration fee, providing required license, certificates and additional accounts). In the second phase, gatekeeping refers to the application itself and has to be repeated for each application submitted to the platform. This phase involves fulfilling all application-specific requirements, which are imposed by the platform provider (i.e. paying submission fee, complying with copyright, performance, and security standards). Expert interviews also helped us to identify four major facets or dimensions of PIC, which are common across all platform contexts for both phases of the screening process. The resulting facets and examples for facets of PIC that emerged in the expert interviews are presented in Table 3.

Facets of PIC	**1. Phase: Developer**	**2. Phase: Application**
Financial Barrier	Developer Registration Fee	Application Submission Fee
Regulatory Requirements	License, Certificate	Copyright, Privacy, Safety
Technical Requirements	Bank Account, E-Mail	Performance, Design, Security
Temporal Expenditure	Developer Authentication Time	Application Review Time

Table 3. Facets of Perceived Input Control.

Considering the outcomes of the expert interviews, we revised our initial item pool and adapted them to the two phases of input control (i.e., developer-centric input control during one-off registration and app-centric input control during repeated app submissions). In doing so, we developed preliminary items that can comprehensively capture the most essential aspects of PIC from an app developers' perspective (see Table 4 with exemplary items for each facet of the PIC scale).

	1. Phase: Developer	**2. Phase: Application**
Financial Barrier	The fee for registering on this platform as a developer is high.	The cost of publishing an app is an entry barrier to the platform.
Regulatory Requirements	There are rules and procedures that complicate admission to the platform.	My apps have to undergo a series of regulatory evaluations before they can be published.
Technical Requirements	It was easy to fulfill all technical requirements to get a developer account.	The technical requirements for developing and publishing apps on this platform are challenging to meet.
Temporal Expenditure	The registration process for developers is fast and easy.	I do not have to wait long for the results of the app review process.
Overall Perception	In my opinion, it is hard to get access to the platform to publish my apps.	

Table 4. Exemplary items from the PIC scale (after step 2).

For each facet of PIC, we generated 3-5 items, as well as 5 additional items measuring overall PIC, ending up with a list of 35 items in total (a full list of items can be obtained from the authors upon request). We derived additional items for app developers' overall perception of PIC for two main reasons. First, following previous scale development studies (e.g., Barki et al., 2007), overall perceptions of a multi-dimensional construct help capture a phenomenon on a higher level of conceptualization and can therefore be more easily applied with a parsimonious set of items across different contexts. Second, given that we plan to estimate multiple indicators and multiple causes (MIMIC) models in order to evaluate our measurement instrument's external validity in step 4 of the scale development process, the development of a redundant set of more general, reflective indicators are required for identification purposes (Diamantopoulos and Winklhofer, 2001; Cenfetelli et al., 2009).

4 Next Steps and Expected Contribution

The remaining steps of the scale development process seek to empirically test the validity and reliability of the scale. As depicted in Figure 1, after we have refined the existing PIC items via a card-sorting (i.e., Q-Sort) procedure (Hinkin, 1998), we plan to thoroughly analyse the statistical properties of the instrument using data from a pre-test study across different online platforms. This requires the specification of a formal measurement model. Based on the multi-dimensionality of PIC, we plan to conduct a MIMIC model that allows us to assess the validity of a formative measurement instrument (i.e., the four sub-dimensions) using common fit indices. The general reflective indicators measuring the overall perception of PIC will then also help us ensure identification of the MIMIC model (Diamantopoulos and Winklhofer, 2001). In our main study (step 5), the refined instrument should be integrated into an extended nomological context, i.e., it should be tested whether the relationship between the instrument and other constructs is in line with the predictions of existing theories and whether it holds across different online platform contexts. A suitable theoretical context to test the scale we propose here would be the well-advanced model of IS continuance (Bhattacherjee, 2001). Transferring the logic of this model to our platform context, app developers' intentions to continuously contribute to a platform can be hypothesized to be influenced by their satisfaction with and perceived usefulness of the platform. By situating PIC into this nomological network, we aim to examine input control's effects on app developers' attitudes (i.e., satisfaction and perceived usefulness) and behavioral intentions (i.e., continuance intention) above and beyond alternative explanations.

With our proposal of a conceptualization and measurement instrument for PIC, we seek to provide both theoretical insights and a valuable tool for practitioners. Previous research has repeatedly called for the development of a measurement scale for input control, particularly for platform contexts, owing to a lack of consistency on what this concept means and how it should be measured (Tiwana et al., 2010; Tiwana, 2015). However, based on the evidence of our systematic literature review, no study has systematically investigated the conceptual foundations as well as the defining characteristics of input control as it relates to online platform contexts. Moreover, previous IS research related to input control have largely drawn on survey items from organizational contexts in an ad-hoc fashion without engaging in a more rigorous scale development process. In contrast, our PIC scale should provide a thoroughly validated instrument to capture the particularities and nuances of PIC in various platform contexts and thereby contribute to a comprehensive understanding of the properties that denote app developers' perceptions of online platforms' gatekeeping practices. We also hope that a rigorously developed input control scale will increase comparability of future research results across different platform contexts over time.

In online platforms in which persuading app developers to contribute to a platform on a regular basis is a cornerstone for sustainable success, a clear understanding of developers' perceptions of platform input control is also an invaluable tool for platform providers. In this regard, our PIC scale including a taxonomy of generic input control facets may serve as an analytical toolkit for platform providers. In the first place, to pinpoint missing or inadequately addressed input control facets on platforms. Secondly, to decide whether consideration or neglect of these facets can overcome shortcomings against the background of a more or less open platform strategy (Schlagwein et al., 2017). Finally, our investigations have so far indicated that two phases of input control (i.e., developer-centric input control during registration and app-centric input control during app submission) have to be taken into account when analyzing platform input control as perceived by app developers. This distinction may help platform providers allocate their attention and budgets more efficiently in their gatekeeping processes over time.

References

Barki, H., R. Titah and C. Boffo (2007). "Information System Use–Related Activity: An Expanded Behavioral Conceptualization of Individual-Level Information System Use." *Information Systems Research* 18(2), 173-192.

Benlian, A., D. Hilkert and T. Hess (2015). "How open is this platform? The meaning and measurement of platform openness from the complementors' perspective." *Journal of Information Technology* 30(3), 209-228.

Bergvall-Kåreborn, B. and D. Howcroft (2011). "Mobile Applications Development on Apple and Google Platforms." *Communications of the Association for Information Systems* 29.

Bhattacherjee, A. (2001). "Understanding Information Systems Continuance: An Expectation-Confirmation Model." *MIS Quarterly* 25(3), 351-370.

Boell, S. K. and D. Cecez-Kecmanovic (2015). "On being 'systematic' in literature reviews in IS." *Journal of Information Technology* 30(2), 161-173.

Bogner, A., B. Littig and W. Menz (2009). Introduction: Expert Interviews — An Introduction to a New Methodological Debate. In: A. Bogner, B. Littig and W. Menz (Eds.), *Interviewing Experts*, p. 1-13. London: Palgrave Macmillan UK.

Boudreau, K. (2010). "Open Platform Strategies and Innovation: Granting Access vs. Devolving Control." *Management Science* 56(10), 1849-1872.

Boudreau, K. J. (2012). "Let a Thousand Flowers Bloom? An Early Look at Large Numbers of Software App Developers and Patterns of Innovation." *Organization Science* 23(5), 1409-1427.

Boudreau, K. J. and K. R. Lakhan (2009). "How to Manage Outside Innovation." *MIT Sloan Management Review* 50.

Boudreau, M.-C., D. Gefen and D. W. Straub (2001). "Validation in Information Systems Research: A State-of-the-Art Assessment." *MIS Quarterly* 25(1), 1-16.

Brocke, J. v., A. Simons, K. Riemer, B. Niehaves and R. Platfaut (2015). "Standing on the Shoulders of Giants: Challenges and Recommendations of Literature Search in Information Systems Research." *Communications of the Association for Information Systems* 37, 205-224.

Cardinal, L. B., S. B. Sitkin and C. P. Long (2004). "Balancing and Rebalancing in the Creation and Evolution of Organizational Control." *Organization Science* 15(4), 411-431.

Ceccagnoli, M., C. Forman, P. Huang and D. J. Wu (2012). "Cocreation of value in a platform ecosystem: the case of enterprise software." *MIS Quarterly* 36(1), 263-290.

Cenfetelli, R. T., G. Bassellier, xe and ve (2009). "Interpretation of Formative Measurement in Information Systems Research." *MIS Quarterly* 33(4), 689-707.

Choudhury, V. and R. Sabherwal (2003). "Portfolios of Control in Outsourced Software Development Projects." *Information Systems Research* 14(3), 291-314.

Chua, C. E. H., W.-K. Lim, C. Soh and S. K. Sia (2012). "Enacting clan control in complex it projects: a social capital perspective." *MIS Quarterly* 36(2), 577-600.

Churchill, G. A. (1979). "A Paradigm for Developing Better Measures of Marketing Constructs." *Journal of Marketing Research* 16(1), 64-73.

de Reuver, M. and H. Bouwman (2012). "Governance mechanisms for mobile service innovation in value networks." *Journal of Business Research* 65(3), 347-354.

DeVellis, R. F. (2012). *Scale development : theory and applications*. Los Angeles: SAGE.

Diamantopoulos, A. and H. M. Winklhofer (2001). "Index Construction with Formative Indicators: An Alternative to Scale Development." *Journal of Marketing Research* 38(2), 269-277.

Eaton, B., S. Elaluf-Calderwood, C. Sørensen and Y. Yoo (2015). "Distributed tuning of boundary resources: the case of Apple's iOS service system." *MIS Quarterly* 39(1), 217-244.

Ghazawneh, A. and O. Henfridsson (2013). "Balancing platform control and external contribution in third-party development: the boundary resources model." *Information Systems Journal* 23(2), 173-192.

Goldbach, T., A. Benlian and P. Buxmann (2017). "Differential effects of formal and self-control in mobile platform ecosystems: Multi-method findings on third-party developers' continuance intentions and application quality." *Information & Management* 55(3), 271-284.

Gregory, R. W., R. Beck and M. Keil (2013). "Control balancing in information systems development offshoring projects." *MIS Quarterly* 37(4), 1211-1232.

Henderson, J. C. and S. Lee (1992). "Managing I/S Design Teams: A Control Theories Perspective." *Management Science* 38(6), 757-777.

Hilkert, D., A. Benlian, M. Sarstedt and T. Hess (2011). "Perceived Software Platform Openness: The Scale and its Impact on Developer Satisfaction." In: *The 32nd International Conference on Information Systems (ICIS)*. Shanghai, China.

Hinkin, T. R. (1998). "A Brief Tutorial on the Development of Measures for Use in Survey Questionnaires." *Organizational Research Methods* 1(1), 104-121.

Hoffmann, E. A. (2007). "Open-Ended Interviews, Power, and Emotional Labor." *Journal of Contemporary Ethnography* 36(3), 318-346.

Jansen, S., S. Brinkkemper and A. Finkelstein (2009). "Business Network Management as a Survival Strategy : A Tale of Two Software Ecosystems." *Ecosystems* 505(2), 34-48.

Jaworski, B. J. (1988). "Toward a Theory of Marketing Control: Environmental Context, Control Types, and Consequences." *Journal of Marketing* 52(3), 23-39.

Kirsch, L. J. (1997). "Portfolios of Control Modes and IS Project Management." *Information Systems Research* 8(3), 215-239.

Kirsch, L. J., D.-G. Ko and M. H. Haney (2010). "Investigating the Antecedents of Team-Based Clan Control: Adding Social Capital as a Predictor." *Organization Science* 21(2), 469-489.

Krausert, A. (2009). *Performance Management for Different Employee Groups*: Physica-Verlag Heidelberg.

MacKenzie, S. B., P. M. Podsakoff and N. P. Podsakoff (2011). "Construct Measurement and Validation Procedures in MIS and Behavioral Research: Integrating New and Existing Techniques." *MIS Quarterly* 35(2), 293-334.

Mähring, M. (2002). *IT Project Governance*. Doctoral Dissertation: Stockholm School of Economics.

Manikas, K. (2016). "Revisiting software ecosystems Research: A longitudinal literature study." *Journal of Systems and Software* 117(Supplement C), 84-103.

Moore, G. C. and I. Benbasat (1991). "Development of an Instrument to Measure the Perceptions of Adopting an Information Technology Innovation." *Information Systems Research* 2(3), 192-222.

Okoli, C. and K. Schabram (2010). "A Guide to Conducting a Systematic Literature Review of Information Systems Research." *Working Papers on Information Systems* vol. 10 no. 26, 1–51.

Ouchi, W. G. (1979). "A Conceptual Framework for the Design of Organizational Control Mechanisms." *Management Science* 25(9), 833-848.

Roberts, J. A., I.-H. Hann and S. A. Slaughter (2006). "Understanding the Motivations, Participation, and Performance of Open Source Software Developers: A Longitudinal Study of the Apache Projects." *Management Science* 52(7), 984-999.

Rossiter, J. R. (2002). "The C-OAR-SE procedure for scale development in marketing." *International Journal of Research in Marketing* 19(4), 305-335.

Sah, R. K. and J. E. Stiglitz (1986). "The Architecture of Economic Systems: Hierarchies and Polyarchies." *The American Economic Review* 76(4), 716-727.

Schlagwein, D., K. Conboy, J. Feller, J. M. Leimeister and L. Morgan (2017). "“Openness” with and without Information Technology: a framework and a brief history." *Journal of Information Technology* 32(4), 297-305.

Segars, A. H. (1997). "Assessing the unidimensionality of measurement: a paradigm and illustration within the context of information systems research." *Omega* 25(1), 107-121.

Snell, S. A. (1992). "Control Theory in Strategic Human Resource Management: The Mediating Effect of Administrative Information." *The Academy of Management Journal* 35(2), 292-327.

Thies, F., M. Wessel and A. Benlian (2016). "Effects of Social Interaction Dynamics on Platforms." *Journal of Management Information Systems* 33(3), 843-873.

Thies, F., M. Wessel and A. Benlian (2018). "Network effects on crowdfunding platforms: Exploring the implications of relaxing input control." *Information Systems Journal* forthcoming.

Tiwana, A. (2014). *Platform Ecosystems: Aligning Architecture, Governance, and Strategy*: Morgan Kaufmann Publishers Inc.

Tiwana, A. (2015). "Evolutionary Competition in Platform Ecosystems." *Information Systems Research* 26(2), 266-281.

Tiwana, A. (2015). "Platform Desertion by App Developers." *Journal of Management Information Systems* 32(4), 40-77.

Tiwana, A., B. Konsynski and A. A. Bush (2010). "Platform Evolution: Coevolution of Platform Architecture, Governance, and Environmental Dynamics." *Information Systems Research* 21(4), 675-687.

Vlasic, A. and P. Yetton (2004). "Effective Project Control: Insights from the Australian Construction Industry." In: *The 8th Pacific Asia Conference on Information Systems (PACIS)*. Shanghai, China.

Wareham, J., P. B. Fox and J. L. C. Giner (2014). "Technology Ecosystem Governance." *Organization Science* 25(4), 1195-1215.

Webster, J. and R. T. Watson (2002). "Analyzing the past to prepare for the future: writing a literature review." *MIS Quarterly* 26(2), 13-23.

Wessel, M., F. Thies and A. Benlian (2017). "Opening the floodgates: the implications of increasing platform openness in crowdfunding." *Journal of Information Technology* 32(4), 344-360.

Wiener, M., A. Cram and U. Remus (2017). "The View from the Top - How Senior Executives exercise Control over Information Systems Projects to enhance Performance." In: *The 25th European Conference on Information Systems (ECIS)*. Guimarães, Portugal. 1423-1438.