Junior Management Science 6(4) (2021) 745-756

Junior Management Science

journal homepage: www.jums.academy

Factors Influencing Developers' Acceptance of Native Development Environments: An Expansion of the Technology Acceptance Model

Nadja Ganter

ESB Business School Reutlingen

Abstract

Mobile operating systems opening up their development environments to third-party developers evoke skyrocketing supply of mobile applications. This study investigates factors influencing the acceptance of third-party developers to adopt native development environments for creating mobile apps. Based on an extensive literature review, the author proposes a research model, built on the technology acceptance model created by Davis. Through the data collected from an online questionnaire completed by third-party app developers, the model was tested using structural equation modeling. Findings indicate a significant effect of the proposed constructs: self-efficacy, output quality, subjective norm, perceived enjoyment, perception of external control, developer community and training. This analysis confirms that the intention to use a native development environment is significantly affected by perceived ease of use and perceived usefulness, whereby the effect size of perceived usefulness is noticeably stronger. Managerial implications suggest to promote high usefulness rather than primarily focusing on an easy to use interface.

Keywords: Native development environment; third-party develope; mobile app development; technology acceptance.

1. Introduction

The launch of smartphones and the related growth of mobile applications (hereafter, apps) revolutionized how the mobile phone industry operates. The key players are not only smartphone vendors, but also operating system providers. Apple's iOS and Google's Android facilitate mobile app distribution for third-party developers via an app market. It is essential to respond to the fast-changing customers' demand for volume and diversity of mobile apps as this is the main value channel for mobile phone end-users. Since operating system providers seek to provide the best app solution for all foreseeable needs, external contributions becomes crucial. Leveraging innovation by not only using internal resources but also relying on external ideas has been gaining popularity during the last few years. Chesbrough and Appleyard (2007) even argue this strategy to be an essential determinant of success. As a response, external developer platforms have emerged as an approach to accommodate external contributions. Thereby, a global network of highly skilled third-party developers connect and develop complement products or services on top of the organization's core set of resources (Parker, Van Alstyne, & Jiang, 2017).

Opening up the application development environment to third-party developers and allowing them distributing their apps via the respective app market, enable operating system providers to reduce their cost of development and, at the same time, leverage an enormous pool of innovation (Ghazawneh, 2012). The development environments hosted by the operating system providers itself are called native development environments (hereafter, NDEs). Among other emerging development environments for mobile apps, NDEs are the default tools and facilitate close integration to the end device. Encouraging third-party developers to create mobile apps on the NDE is essential to increase the value of the smartphone (Cusumano, 2010). Drastic consequences can be witnessed by the decline of Nokia and Blackberry that missed to provide a diversity of mobile apps (Goldbach, Kemper, & Benlian, 2014).

Despite this being an area of considerable interest, there is missing literature on why third-party developers accept an NDE for developing mobile apps. Researchers urge the examination of third-party developers (Basole & Karla, 2011) and to search related literature regarding social factors that

DOI: https://doi.org/10.5282/jums/v6i4pp745-756

may influence developers in other contexts (Steglich et al., 2019). Although previous investigations discuss some factors influencing third-party developers (Hilkert, Benlian, & Hess, 2010; Lee, Kim, & Hong, 2016), few studies quantify the actual impact on the usage behavior. This analysis explores this gap and examines possible factors through an in-depth literature review and validates them quantitatively.

The prominent technology acceptance model (from now on TAM) by Davis serves as a basis for the proposed research model. TAM evaluates the users' acceptance and hence the usage of a technology by assessing the perceived usefulness and the perceived ease to use (Davis, 1989; Davis, Bagozzi, & Warshaw, 1989). Over the years, TAM has been widely used and adapted in various fields by adding external constructs. A modified TAM, adopted to NDEs, is proposed while addressing the following research question:

> Which factors are influencing developers' acceptance of native development environments?

The purpose of this analysis is to complement and further expand existing literature on external contribution focusing on mobile app development on NDEs, as well as to validate and to expand TAM. The remainder of this paper is structured as follows. First, the characteristics of mobile app development are described as an essential background for the remaining paper. Second, an in-depth literature review is conducted to formulate concrete hypotheses. The methodology is described in the section that follows. Subsequently, the data analysis from the questionnaire responses is presented. The study closes with a discussion on managerial implications, limitations, and suggestions for further research as well as a precise conclusion.

2. The Characteristics of Mobile App Development

Mobile apps are small software programs running on mobile devices that perform a wide variety of tasks (Taylor, Voelker, & Pentina, 2011). To facilitate the external development of apps, mobile operating system providers publish resources like application programming interfaces (APIs) and software development kits (SDKs) (Ghazawneh, 2012), also referred to as technical boundary resources (Bianco, Myllarniemi, Komssi, & Raatikainen, 2014). APIs are code-based specifications to access the core assets of the platform (Ghazawneh, 2012), and SDKs are program resources, which help developers to program, test, and maintain the applications (Bianco et al., 2014). The development of apps can take place on native development environments provided by the operating system providers itself or on non-native environments such as web-based or cross-platform environments. Non-native development environments facilitate to develop an app once and run it on several platforms without writing it in different programming languages. The app is then published via the app market of the operating system providers. However, non-native development environments often lack in features and compatibility with the operating systems. By contrast, native development environments are tightly integrated and work with the latest and native features (Amatya & Kurti, 2014). Through the homogenous development tools, optimal app development can be achieved and is thus preferred by most developers. Therefore, this analysis will primarily focus on native development environments as default tool to develop mobile apps. While various operating systems have emerged, Google's Android and Appel's iOS dominate the market with a market share of 74.13% and 24.79% respectively (StatCounter, 2020). Due to the supremacy of these two operating systems, only their respective NDEs are considered for relevance reasons.

Coming from the music and personal computer industry, Apple disrupted the mobile industry by introducing iOS and the App Store (Boudreau & Lakhani, 2009). Today, 12 years later, 1.8 million apps are available for the end-users (Clement, 2020). The goal of opening up the development environment to third-party developers (while initially being closed to a small team of in-house developers) was to increase the cross-sales of its high-margin products by providing complements (Ghazawneh, 2012). To develop apps for iOS, third-party developers register on a fee-based NDE and get thereby access to the technical boundary resources. Apps are built using the programming languages objective-C or swift using Xcode. Apple will review the app before publishing it via the App Store. For paid apps, Apple keeps 30% share of the app sales, while the third-party developers can keep 70%. By shifting to a more dynamic and flexible approach, Apple triggered contributions by third-party developers to a wide variety of apps (Pisano & Verganti, 2008).

Google followed the successful strategy of Apple by introducing Android OS first on HTC and later for various mobile device manufacturers. Android uses an open-source model, to allow third-party developers to download SDKs for free and without registration (Boudreau & Lakhani, 2009). Native apps for Android are written in Java or Kotlin and commonly built using Android Studio. The application approval process before introducing the app to the market (via Google Play Store) is relatively simple and fully automated. Google charges developers a registration fee to access Google Play Store and later (similar to Apple) keeps 30% of unit sales (Sadi, Dai, & Yu, 2015). Today 2.5 million apps are available for the end-users of Android devices (Clement, 2020).

3. Literature Review and Hypotheses Development

3.1. Definition of the Research Process

For building a research model on factors influencing developers' acceptance of NDEs, this research builds on the established technology acceptance model (TAM). Thus, in a first step, most prominent TAM studies were scanned to analyze possible constructs. Based on a Web of Science and Google Scholar search, four primary studies defining the main TAM constructs were selected (summarized in Appendix A1).

In a second step, a thorough literature review, the main TAM constructs were analyzed for relevance in the context of mobile app development on NDEs. Thereby, the approach recommended by Webster and Watson (2002) was applied. First, a database-driven keyword search using the EBSCO Business Source Complete database was conducted. The following search string was used: "AB ("develop* environment" or "ios" or "android" or "mobile ecosystem" or xcode or android-studio) AND AB ("third-party developer*" or "external developer*" or "app* developer*") AND TX ("useful*" or "ease of use" or "easy to use" or "subjective norm" or "job relevance" or image or "output quality" or "result demonstrability" or "self-efficacy" "external control" or anxiet* or playful* or enjoy* or "objective usability")". The author considered only publications in peer-reviewed journals from the years 2007 to 2020. This resulted in 97 papers that were evaluated for their relevance by scanning the abstracts and identifying 11 articles that were considered for full reading. Secondly, further research was revised by reviewing articles references. Lastly, further literature was identified by reviewing articles citing the previous selected literature. By scanning the abstracts of articles identified in step two and three, ten further studies were classified as potentially relevant. Finally, out of the 21 potentially relevant studies, 6 were selected as applicable.

As the call for research by Basole and Karla (2011) and Steglich et al. (2019) suggest, the literature on third-party developers in a related context was also considered. Hilkert et al. (2010) confirm that third-party developers creating complementary applications for different software-platforms, are influenced by similar factors. Thus, the author selected studies examining third-party developers' acceptance of different development environments for developing complementary apps (summarized in Appendix A2).

Identified literature serves to define external constructs while also formulating concrete hypotheses and proposing a research model.

3.2. Technology Acceptance Model

TAM was first introduced by Davis and evaluates the users' acceptance of technologies by assessing usage behavior (Davis, 1989). Usage behavior (UB) refers to how often a system is used, measured as self-reported current usage (Davis, 1989). Based on the theory of reasoned action by Fishbein and Ajzen (1977), TAM introduces "perceived usefulness (PU)" and "perceived ease of use (PEOU)" as the two constructs influencing the intention to use a technology, which has been linked to subsequent usage behavior (Davis, 1989). PU is defined as "the degree to which a person believes that using a particular system would enhance his or her job performance" (Davis, 1989), whereas PEOU is defined as "the degree to which a person believes that using a particular system would be free of effort" (Davis, 1989). The definition of intention to use (IU) a system is based on Fishbein's and Ajzen's definition "the strength of one's intention to perform a specified behavior" (Fishbein & Ajzen, 1977).

Furthermore, TAM theorizes that perceived usefulness is also influenced by perceived ease of use, as the easier a technology to use, the more useful it can be (Venkatesh, 2000). TAM originally included a mediator, namely attitude, but as later studies confirm the accuracy of omitting this construct (Davis, 1989; Venkatesh, 2000; Venkatesh & Davis, 2000), this analysis will proceed to do so. TAM explains around 40% of the variance in intention to use and usage behavior and thus compares favorably with alternative models like the theory of planned behavior or theory of reasoned action (Venkatesh & Davis, 2000). TAM was intended to apply to different settings by identifying external constructs influencing the users of a system. Thereby, the effect of external constructs on intention to use is mediated by perceived usefulness and perceived ease of use (Venkatesh & Davis, 2000). Figure 1 illustrates TAM as the basis for the proposed research model.

The rationale for using TAM rather than other models is that it has been validated many times, and it remains flexible regarding application in different technology settings (Venkatesh & Davis, 2000). Furthermore, ease of use and usefulness are found to be essential concepts when analyzing the acceptance of development environments (Koch & Kerschbaum, 2014; Lee et al., 2016). Based on these findings, the author expects that general causalities found in TAM are also applicable in the context of this study. To verify this assumption, the following hypotheses are proposed:

H1: Intention to use NDEs positively influences usage behavior on NDEs

H2: Perceived usefulness positively influences intention to use NDEs

H3a: Perceived ease of use positively influences intention to use NDEs

H3b: Perceived ease of use positively influences perceived usefulness of NDEs

3.3. External Constructs within the NDE Context

For identifying external constructs, those defined by Venkatesh (2000) and Venkatesh and Davis (2000) serve as a basis. The author analyzed each construct for relevance by searching in NDE related literature. Venkatesh and Davis (2000) identified constructs influencing perceived usefulness. The following are adopted to this study and described in greater detail in this section: subjective norm, job relevance, image, and output quality. Furthermore, Venkatesh (2000) defined constructs influencing perceived ease of use, of which the following three are adopted to this study: self-efficacy, perception of external control, and perceived enjoyment. In addition, two constructs, not defined by the underlying TAM studies, are introduced and included in the proposed research model, as they were identified as important by analyzing NDE related literature. Constructs defined by Venkatesh and Davis (2000) and Davis (2000), which are

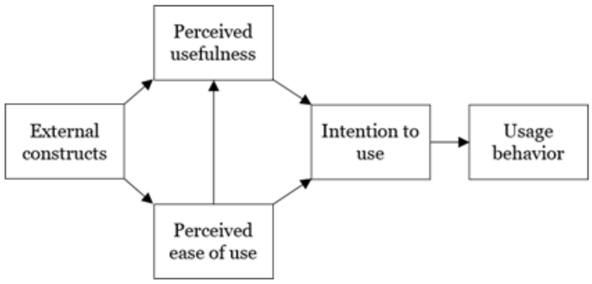


Figure 1: TAM based on Venkatesh and Davis (2000).

not mentioned in any NDE related literature, are excluded from this study (overview in Appendix A3).

Subjective norm (SN), an individual's perception that people who are important to oneself think one should or should not use a specific technology, influences perceived usefulness (Venkatesh & Davis, 2000). Hertel, Niedner, and Herrmann (2003) identify subjective norm as the "reactions of significant others" as one of the main motivational factors for third-party developers. The motivation to contribute to a development environment is higher the more positive in proportion to the expected reaction of e.g., family members, friends, or colleagues (Hertel et al., 2003). Based on these findings, the author proposes following hypothesis:

> H4: Subjective norm positively influences perceived usefulness of NDEs

Image (IM), the degree to which using a technology is perceived to enhance one's status in one's social system, influences the perceived usefulness (Venkatesh & Davis, 2000). Several studies confirm that developing apps can establish reputation and signal talent among their own community (Hilkert et al., 2010; Steglich et al., 2019) or even potential employers (Koch & Kerschbaum, 2014). The gain in reputation motivates third-party developers to contribute through mobile app development (Steglich et al., 2019). Thus, the author expects that the following hypothesis holds:

> H5: Image positively influences perceived usefulness of NDEs

Job relevance (JR), an individual's perception regarding the degree to which a system is relevant for one's job, determines the perceived usefulness (Venkatesh & Davis, 2000). Some developers are hired for specific app development tasks and thus rely on the usage of NDEs for their salary (Hilkert et al., 2010; Koch & Kerschbaum, 2014). Therefore, the perception regarding the financial return is, in this study, associated with job relevance. The research of Lee et al. (2016) indicates that an adequate financial return is a main attraction factor on mobile app development environments. Based on these findings, the author expects that the following hypothesis holds:

> H6: Job relevance positively influences perceived usefulness of NDEs

Output quality (OQ), the degree to which a person believes that the system produces good work, influences the perceived usefulness (Venkatesh & Davis, 2000). A comparison of development environments for developing mobile apps reveals high quality of apps developed on NDEs as they facilitate neat integration (Huy & Van Thanh, 2012). Furthermore, developers perceive NDEs as supportive when striving for best developing performance (Steglich et al., 2019). Therefore, the author proposes the following hypothesis:

> H7: Output quality positively influences perceived usefulness of NDEs

Developer community (DC), the degree to which an individual benefits from the size and engagement of the developer community using the same NDE, is assumed to influence perceived usefulness. Although not defined as an external construct by Venkatesh and Davis (2000) and Venkatesh (2000), the developer community must also be considered in this study's context. Steglich et al. (2019) identified that from a developers' perspective, a large developer community is perceived as a primary advantage of NDEs. The discussions and forums within the community can be exciting and useful for third-party developers (Steglich et al., 2019). Koch and Kerschbaum (2014) confirm that in choosing which ecosystem to join, the size of the developer community increases the NDEs attractiveness. Based on these findings, DC is argued to be a crucial external construct and will be tested in this study. The author suggests the following hypothesis:

> H8: Developer community *positively influences perceived usefulness of NDEs*

Training (TR), learning facilities to enhance developers' skills and know-how provided by the NDE, is a main motivational driver for third-party developers (Koch & Kerschbaum, 2014). When introducing external constructs of TAM, Venkatesh and Davis (2000) explicitly called for future research examining the effect of training. NDEs offer online training material and training events to support efficient development. This training attracts third-party developers as they acquire specific skills that can also be used elsewhere (Hilkert et al., 2010; Steglich et al., 2019). Lee et al. (2016) found that developers are more likely to accept an NDE if training is offered. Following hypothesis is proposed:

H9: Training positively influences perceived ease of use of NDEs

Self-efficacy (SE), the degree to which an individual believes that one has the ability to use a system, influences the perceived ease of use (Venkatesh, 2000). Hertel et al. (2003) identify "a high sense of personal self-efficacy" as an essential factor influencing third-party developers. When a developer can use an NDE without any help, the motivation to contribute is higher (Hertel et al., 2003). Based on this finding, the author proposes the following hypothesis:

H10: Self-efficacy positively influences perceived ease of use of NDEs

Perception of external control (PEC), the degree to which an individual believes that organizational and technical resources exist to support the use of the system, influences perceived ease of use (Venkatesh, 2000). From a third-party developers' perspective, NDEs must provide a variety of technical resources such as SDKs and APIs (Lee et al., 2016). The toolkit quality can be identified as one of the essential factors in choosing a development environment (Koch & Kerschbaum, 2014). Furthermore, the organizational resources, and thereby the platform openness, is critical to facilitate the usage of the NDE (Lee et al., 2016) and also serves as a decision criteria when choosing a development environment. Therefore, the following hypothesis is introduced:

H11: Perception of external control positively influences perceived ease of use of NDEs

Perceived enjoyment (PE), the degree to which the usage of a system is perceived to be enjoyable, influences perceived ease of use (Venkatesh, 2000). Enjoyment during the development process is a crucial factor for third-party developers (Steglich et al., 2019). Koch and Kerschbaum (2014) found that the intellectual stimulation of the innovation process itself is a primary reason for developers to join a smartphone ecosystem. The following hypothesis will be tested:

H12: Perceived enjoyment positively influences perceived ease of use of NDEs

4. Methodology

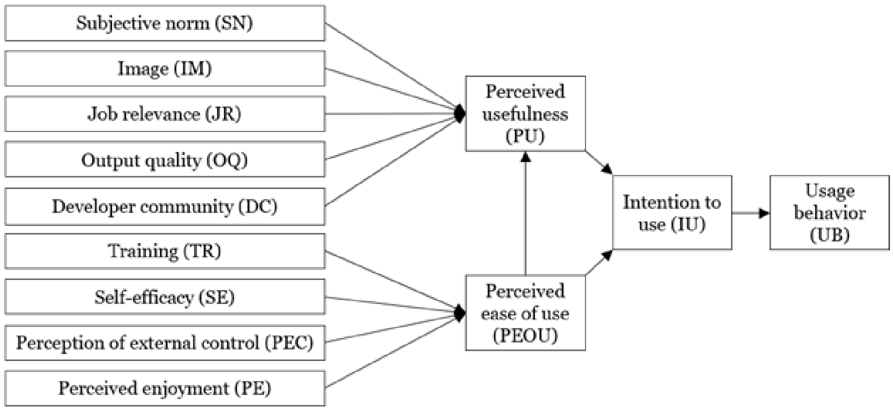
As outlined above, existing literature suggests several factors influencing third-party developers' acceptance of NDEs. Based on these findings, the author proposes a research model, illustrated in Figure 2.

To test the proposed research model (Figure 2), an online survey (Appendix B1) was conducted and the resulting data analyzed. A quantitative study was chosen, as previous research already suggests concrete constructs and call for quantitative validation (Lee et al., 2016). Furthermore, it facilitates to include a higher number of variables and a more appropriate generalization compared to qualitative studies (Wagner & Hering, 2014). Combining a qualitative literature review with quantitative research emphasizes high quality of social investigation (Zanón & Paz, 2009).

An online survey has several advantages compared to other survey methods. Respondents can be reached over long distances and across borders at the same time, which can counteract the methodological effect of different survey times. Furthermore, without the interviewer, the so-called "interviewer effect" is eliminated. The main disadvantage is that online surveys often address a younger (<60 years) and male-dominated sample (Wagner & Hering, 2014), which in this study can be neglected as this corresponds to the target group.

The questionnaire comprised 45 items, measuring 14 constructs on a five-point Likert-scale (constructs and respective items are listed in Table 2). Although seven-point Likert-scales are standard, it was decided to use five points to facilitate a neat design of the questionnaire on smartphones. Questionnaire items were assembled based on the examples of underlying TAM studies (Davis, 1989; Davis et al., 1989; Venkatesh, 2000; Venkatesh & Davis, 2000). Furthermore, closed-ended questions on socio-demographics were asked to compare to prior third-party developers statistics and thus verify the sample adequacy. Additionally, it was asked which NDE is used to test the suitability of compiling the responses of the two groups (developing for Android or iOS). As suggested by Straub (1989), the author conducted pretests among five NDE users to ensure the right selection of constructs and test comprehension of the questionnaire items. Based on the feedback from the pretests, a few minor changes were made to improve the validity and understanding. The author published the link of the questionnaire via the following distribution channels: social media groups, mailing lists of informatics institutes, mobile app development companies, developer forums, and private contacts triggering the snowball principle (see Appendix B2 for more details).

The resulting data was analyzed based on the proposed research model using the partial least square (PLS) structural equation modeling (SEM) technique, with SmartPLS 3.2.1. SEM enables the specification and testing of complex path models with multiple latent constructs (Hair, Hult, Ringle, & Sarstedt, 2014). There are mainly two types of SEM: Factor-based covariance SEM (CB-SEM) and partial least square SEM (PLS-SEM) (Hair et al., 2014). The rationale for choosing PLS-SEM is, amongst others, its adequacy for testing new relationships and that it avoids factor indeterminacy (Chin, Diehl, & Norman, 1988). Further reason to choose PLS is the minimal demands on measurement scales (Hair et al., 2014). Thus, the ongoing debate on whether Likert-scales are nominal or ordinal (Boone & Boone, 2017) can be ignored, and no assumptions of multivariate normality is required (Hair et al., 2014). Through PLS-SEM, working with a medium sample size is facilitated, and measurement errors in exogenous variables are treated more appropriately when compared to other methods (Chin et al., 1988). While analyzing the data, the guidelines by Avkiran and Ringle (2018), Hair et al. (2014) and Henseler, Hubona, and Ray (2016) were followed.



Note: Rectangles represent proposed constructs. Arrows in-between represent the proposed relationship and hypothesis.

Figure 2: Proposed Research Model

5. Data Analysis and Results

5.1. Sample

In total, 163 participants completed the questionnaire, of which 141 responses were used for the analysis. Out of all responses, 22 had to be eliminated because they were incomplete (completion rate below the cut-off of 85% as recommended by Hair et al. (2014)) or lacked prior experience in mobile app development on the NDEs of iOS or Android (marked "others" at the control-question which NDE they are using). The sample can be identified as adequate, as it corresponds to previous statistics on mobile app developers (Evans Data Corporation, 2017; Jet Brains, 2019). The average age of the participants is slightly younger (30 years) compared to prior statistics (36 years) (Evans Data Corporation, 2017) but still in an acceptable range. Of the 141 valid respondents, 83% are male and only 17% are female. The predominance of male participants reflects the target group (Evans Data Corporation, 2017). Although more than 50% of the respondents are from Europe, the geographic coverage can be considered as expansive as developers from all over the world participated. The majority of mobile app developers (52.5%) create mobile apps on the NDE provided by Android, significantly less on the NDE of iOS (32.6%), and only a few (14.9%) use both development environments. On average, participants have 3.4 years of experience in developing mobile apps on the respective NDE. Table 1 highlights basic socio-demographic information on the participants.

5.2. Multigroup Analysis

The author conducted a multigroup analysis (MGA) to test the adequacy of combining the questionnaire responses of developers using the NDE by Android with those by iOS. An MGA examines the differences in path coefficients of two sample groups (Hair et al., 2014). The MGA test highlights p-values greater than 0.1 for all path coefficients (detailed results in Appendix C2). Therefore, it can be concluded that the responses of the two groups are not significantly different. Hence, questionnaire responses are jointly analyzed.

5.3. Instrument Validation

Before conducting detailed analysis, validity and reliability were tested. Convergent validity (alignment of repeated measurement of the same construct) was tested with the average variance extracted (AVE). AVE measures the amount of variance captured by the constructs in relation to the amount of variance that is attributed to the measurement error (Avkiran & Ringle, 2018). The constructs have AVEs between 0.57 and 0.86, while the cut-off for good validity is 0.5 (Avkiran & Ringle, 2018). Hence adequate validity is identified (AVEs indicated in Table 2).

To assess the discriminant validity (distinct measure of different concepts) Fornell-Larcker criterion and Heterotrait-Monotrait ratio (HTMT) were conducted (detailed results in Appendix C3). Fornell-Lacker criterion requires that the square root of the AVE exceeds the correlation with other constructs (Avkiran & Ringle, 2018). This criterion was fulfilled by all constructs, besides the construct IU. Hence, HTMT ratio was analyzed additionally to test if this construct can still be identified as valid. Generally, Heterotrait correlations should be smaller than Monotrait correlations (Avkiran & Ringle, 2018). In the given dataset HTMT ratios are between 0.43 and 0.83 and thus fulfill the criterion for proper discriminant validity (Henseler et al., 2016).

Furthermore, two reliability tests were carried out to ensure internal consistency and indicator reliability. According to Avkiran and Ringle (2018) excellent indicator reliability is achieved when loadings are higher than 0.7. Likewise, loadings between 0.4 and 0.7 can be accepted when loadings of remaining items are greater than 0.7 (Avkiran & Ringle, 2018). In the given data set, all constructs but UB show indicator loading factors between 0.72 and 0.96 and thus illustrate excellent reliability (Table 2). The construct UB reveals one loading factor slightly below 0.7. However, the remaining indicator reveals a loading factor of 0.90. Thus, the indicators can still be considered as very reliable. Internal consistency was examined by the composite reliability coefficient (CR). Generally, a CR of 0.7 is considered as a threshold for

Table 1: Questionnaire Participants

Variables		Count	Percentage
Age	Average in years	30	N/A
Gender	Female	24	17%
	Male	117	83%
Continent	Africa	5	3.50%
	Asia	27	18.80%
	Europe	77	53.50%
	North-America	31	21.50%
	South-America	1	0.70%
NDE provider	Android	74	52.50%
	iOS	46	32.60%
	Both	21	14.90%
Experience	Average in years	3.4	N/A

Note: N=141

adequate internal consistency (Avkiran & Ringle, 2018). The given dataset reveals CRs greater than 0.7 (Table 2). Therefore, the data suggest good internal consistency.

Altogether, the reliability and validity of the construct measures used in this study received strong support from conducted tests. Table 2 highlights the measured items and the respective reliability and validity test results as well as the mean and standard deviation (SD) range for each construct.

5.4. Descriptive Measures

To get a first impression of the constructs, mean (M) and standard deviation (SD) for each item has been calculated. As all items of the same construct indicate contiguous descriptive measures, only the range per construct is indicated (in Table 2). Overall, participants of the questionnaire are rather heavy users of the respective NDE (M = 3.5–3.6; SD = 1.2–1.3; while 5 being the highest usage and 1 lowest usage). Generally, the NDE is perceived as useful (M = 3.8–3.9; SD = 1.0–1.1) and easy to use (M = 3.6–3.7; SD = 1.0–1.1). Amongst all external constructs, the highest means can be identified for the constructs developer community (M = 3.9–4.0; SD = 0.9–1.1) and perception of external control (M = 3.9–4.2; SD = 0.9–1.1). This can be interpreted as that the developer community of the NDE provides high value to the developers and that the resources provided by the NDE are generally perceived as adequate. Lowest means are reported for image (M = 3.3–3.4; SD = 1.0–1.1) and job relevance (M = 3.3–3.5; SD = 0.9–1.1). Therefore, it could be expected that the majority of developers neither think that they enhance their image when using the NDE nor use the NDE due to salary importance. Means of the other constructs range from 3.4 to 3.7 (SD = 0.9–1.1). Interpretation of means alone should be handled with caution and will therefore only be used as a first impression.

5.5. Model Fit

Unlike in CB-SEM there is no global goodness-of-fit measure available in PLS-SEM (Henseler et al., 2016). Yet, there is a model-quality indicator based on how close the predicted values of the dependent variables are to the observed values. This criterion is called the standardized root mean square residual (SRMR). SRMR is the square root of the sum of the squared differences between the model-implied and the empirical correlation matrix (Henseler et al., 2016). A value of 0.00 indicates a perfect fit, and the cut-off value is 0.08 (Henseler et al., 2016). The proposed research model highlights an SRMR of 0.058. Therefore, the model indicates high quality.

5.6. Structural Equation Modeling

Hypotheses are tested through the size and significance of path coefficients. Additionally, the research model is evaluated based on the coefficient of determination (r^2). As recommended by Avkiran and Ringle (2018), consistent bootstrapping (with 5000 bootstrap subsamples) in a two-tailed test with a 5% significance level is used. Path coefficients can range from −1 to +1, while values closer to absolute 1 reflect strong positive paths, and values closer to 0 indicate weak positive paths (Avkiran & Ringle, 2018).

The path coefficient of intention to use on usage behavior is very strong and highly significant ($\beta = 0.73$, $p < 0.01$). Intention to use is positively affected by perceived usefulness ($\beta = 0.63$, $p < 0.01$) and perceived ease of use ($\beta = 0.29$, $p < 0.05$), whereby PU has a stronger influence than PEOU. PEOU additionally affects PU ($\beta = 0.25$, $p<0.01$). Perceived usefulness is positively and significantly influenced by subjective norm ($\beta = 0.28$, $p<0.05$), output quality ($\beta = 0.30$, $p < 0.01$), and developer community ($\beta = 0.15$, $p < 0.1$), while the effect of job relevance ($\beta = 0.01$, $p > 0.1$) and image ($\beta = 0.07$, $p > 0.1$) is not significant. Perceived ease of use is positively and significantly influenced by training ($\beta = 0.24$, $p < 0.05$), self-efficacy ($\beta = 0.43$, $p < 0.05$), and perceived enjoyment ($\beta = 0.30$, $p < 0.05$). The effect of perception of external control on perceived ease of use ($\beta = -0.03$, $p > 0.1$) is not significant. Overall, 10 out of the

Table 2: Instrument Validation

Construct	Items	Mean (SD)	Loading	AVE	CR
Usage behavior (UB)	To which extent do you use the NDE to develop mobile apps? (I exclusively use the NDE/ I mainly use the NDE/ I sometimes use the NDE / I rarely use the NDE/ I only use the NDE if it's unavoidable)	3.5 - 3.6	0.9	0.64	0.78
	I usually develop mobile apps on the NDE*	(1.2 - 1.3)	0.68		
Intention to use (IU)	I aim to use the NDE often*	3.4 - 3.7	0.79	0.57	0.73
	I intend to be a heavy user of the NDE*	(1.1 - 1.2)	0.72		
Perceived usefulness (PU)	The NDE improves my app development performance*	3.8 - 3.9	0.83	0.75	0.9
	Using the NDE increases my productivity*	(1.0 - 1.1)	0.82		
	The NDE is useful for me*		0.93		
Perceived ease of use (PEOU)	I find it easy to use the NDE*	3.6 - 3.7	0.92	0.77	0.91
	Learning how to use the NDE was easy for me*	(1.0 - 1.1)	0.87		
	It is easy to become skillful at the NDE*		0.84		
Subjective norm (SN)	People I learn from, think I should use the NDE*	3.5 - 3.6	0.91	0.81	0.93
	My peers support the use of the NDE*	(1.1)	0.92		
	My social environment supports the use of NDEs*		0.86		
Image (IM)	Through developing on the NDE I signal skills and competences*	3.3 - 3.4	0.94	0.79	0.92
	I enhance my reputation through developing apps on the NDE*	(1.0 - 1.1)	0.82		
	People using the NDE receive good reputation*		0.91		
Job relevance (JR)	The usage of the NDE is part of my job*	3.3 - 3.5	0.91	0.71	0.88
	The monetary reward through the app development is important for me*	(0.9 - 1.1)	0.75		
	In my job the NDE is relevant*		0.87		
Output quality (OQ)	The NDE enables me to develop good apps*	3.6	0.94	0.81	0.93
	I rate the results from the NDE to be excellent*	(0.9 - 1.1)	0.84		
	The quality of the app I get from the NDE is high*		0.91		
Developer community (DC)	I benefit from other developers using the same NDE*	3.9 - 4.0	0.96	0.84	0.94
	A community of other developers using the same NDE is valuable for me*	(0.9 - 1.1)	0.89		
	I enjoy being part of the developer community*		0.9		
Training (TR)	Through the NDE I improve my developing skills*	3.6 - 1.0	0.92	0.83	0.93
	Learning facilities (e.g., tutorials and learning guides) of the NDE helps me getting better in app development*	(1.0)	0.93		
	The NDE provides training in app development*		0.86		
Self-efficacy (SE)	I am able to use the NDE, even if there is no one telling me how to use it*	3.5 - 3.7	0.73	0.67	0.86
	For me, it's intuitive to use the NDE*	(1.0 - 1.1)	0.9		
	I have the necessary skills for using the NDE*		0.81		0.86
Perception of external control (PEC)	On the NDE sufficient platform-specific SDKs are provided*	3.9 - 4.2	0.95	0.86	0.92
	I have the resources necessary to use the NDE*	(0.9 - 1.1)	0.9		
Perceived enjoyment (PE)	The actual process of using the NDE is enjoyable*	3.4 - 3.7	0.91	0.75	0.9
	I have fun developing apps on the NDE*	(1.0)	0.89		
	I enjoy developing apps on the NDE*				

Note: *measured on a five-point Likert-scale from 1 = totally disagree to 5 = totally agree; SD = standard deviation; CR = composite reliability; AVE= average variance extracted

13 proposed path coefficients are significant and support the respective hypothesis. Figure 3 illustrates the model results indicating the path coefficients and coefficient of determination (r^2). The model as displayed by SmartPLS is attached in Appendix C1.

The most commonly used measure to evaluate a structural model is the variance explained, also called the coefficient of determination (r^2) (Hair et al., 2014). While r^2 values range from 0 to 1, a higher value implies a higher level of predictive accuracy (Hair et al., 2014). According to Chin et al. (1988), constructs with an r^2 above the cut-off of 0.33 have moderate predictive power, while an r^2 exceeding the threshold of 0.67 means substantial predictive power. The proposed research model indicates for UB an r^2 of 0.53. This translates to around 53% of usage behaviors can be explained with the proposed model. IU indicates an r^2 of 0.74, which means that PU and PEOU explain about 74% of the variance in IU. The predictive power of PU and PEOU is 79% and 73% respectively, values which indicate substantial predictive power according to the thresholds of Chin et al. (1988).

For even more sophisticated results, the author then analyzed the effect size (f^2) of each hypothesis. The f^2 coefficient examines how much unexplained variance compensates for the r^2 change (Hair et al., 2014). Effect size values of 0.02, 0.15 and 0.35, respectively, represent small, medium, and large effects of the exogenous latent variable (Hair et al., 2014). Within the research model, IU on UB has the largest effect size ($f^2 = 1.12$). Noticeably is the larger effect of PU on IU ($f^2 = 0.73$), compared to PEOU on IU ($f^2 = 0.15$). In the following, external constructs and respective effect size are listed, here organized by size, beginning with the highest: self-efficacy ($f^2 = 0.21$), output quality ($f^2 = 0.16$), subjective norm ($f^2 = 0.16$), training ($f^2 = 0.11$) and developer community ($f^2 = 0.06$). In contrast, image, job relevance, and perception of external control are not considered, as hypotheses are not supported, and f^2 is very low. Table 3 summarizes all proposed hypotheses with respective results, corresponding f^2 and interpreted effect size.

6. Discussion and Conclusion

6.1. Overview

This analysis represents the first attempt to apply TAM in the context of native development environments for developing mobile apps. Generally, it is perceived as an appropriate underlying model as the general causalities found in previous TAM studies hold in this context. The proposed hypotheses based on the original TAM by Davis (H1, H2, H3a, H3b, H4) are all supported in this analysis.

The research model explains about 53% of the variance in usage behavior and 74% of the variance in the intention to use. Thus, it compares favorably to previous TAM studies, which explain about 40% of the variance in UB and IU (Venkatesh & Davis, 2000). Consistent with previous TAM studies, the effect of PU on IU (H2) is significantly higher than the effect of PEOU on IU (H3a).

Furthermore, perceived usefulness can be well predicted by the proposed external constructs. As identified by Venkatesh and Davis (2000), subjective norm (H4) and output quality (H7) significantly influence perceived usefulness. However, the constructs image (H5) and job relevance (H6) could not be validated as an influential factor in this analysis. The additionally proposed construct developer community, based on the studies by Koch and Kerschbaum (2014) and Steglich et al. (2019), reveals as a relevant construct.

Moreover, self-efficacy (H10) and perceived enjoyment (H12), as defined by Venkatesh (2000), significantly influence developers' perceived ease of use of the NDE. The perception of external control (H11) cannot be validated within this analysis. The additionally proposed construct training (H9) reveals to influence developers acceptance of NDEs (Hilkert et al., 2010; Lee et al., 2016; Steglich et al., 2019).

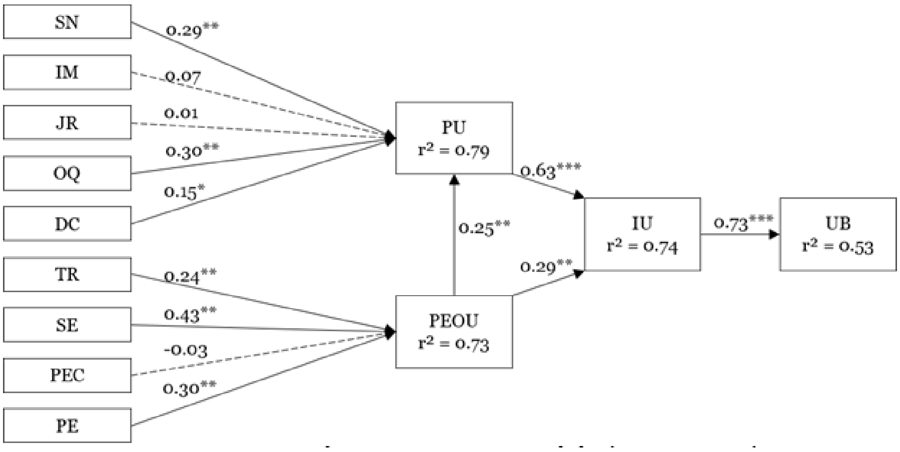
6.2. Managerial Implication

This study provides various novel contributions for the following interest groups. First of all, this study is highly valuable for the current NDE providers Android and iOS to identify improvement opportunities. Second, future NDE providers, not yet having a significant market share, can get valuable insights by looking at the more prominent players. Regardless of being a current or a future NDE provider, it is deemed beneficial to understand the factors that influence third-party developers' usage behavior to provide an attractive and developer-focused environment. Furthermore, this study also provides implications for non-native development environment providers. Emerging cross-platform environments also try to attract third-party developers and can surely benefit from this analysis. Likewise, emerging developer platform providers in all kinds of different industries have a keen interest in understanding third-party developers' motivation. Based on the analysis, the author suggests the following implications:

First, an interesting point is that perceived usefulness has a more substantial influence on the intention to use than perceived ease of use. Although proven otherwise, many managers still believe that ease of use is key to success and seem to primarily focus on the design of a platform (Chin et al., 1988). Yet, the analyzed data indicates that although PEOU is important, PU is an even more important determinant of developers' intention to use an NDE. Thus determinants of PU should not be underestimated as ease of use will not compensate for an NDE that is not perceived as useful.

Second, based on the strong effect of output quality, it is advisable to promote the high quality of apps developing on a certain development environment. Especially NDE providers can take advantage of the high standards and integrality of native apps.

Third, it is suggested to provide specific learning facilities, which are fun, highlight the training effect and thereby increase the self-efficacy. Concrete projects might be hackathon events or video tutorials. In doing so, the following three



Note: $***p < 0.001$, $**p < 0.01$, and $*p < 0.05$. UB = Usage behavior, IU = Intention to use, PU = Perceived usefulness, PEOU = Perceived ease of use, SN = Subjective norm, IM = Image, JR = Job relevance, OQ = Output quality, DC = Developer community, TR = Training, SE = Self-efficacy, PEC = Perception of external control, PE = Perceived enjoyment.

Figure 3: Model Results based on PLS-SEM.

Table 3: Results of Hypotheses and Effect Size

Hypothesis		Result	f^2	Effect size
H1	IU -> UB	Supported	1.12	High
H2	PU-> IU	Supported	0.73	High
H3a	PEOU -> IU	Supported	0.15	Moderate
H3b	PEOU -> PU	Supported	0.15	Moderate
H4	SN -> PU	Supported	0.16	Moderate
H5	IM -> PU	Not supported	0.01	-
H6	JR -> PU	Not supported	0.00	-
H7	OQ -> PU	Supported	0.16	Moderate
H8	DC -> PU	Supported	0.06	Small/ Moderate
H9	TR-> PEOU	Supported	0.11	Small/ Moderate
H10	SE-> PEOU	Supported	0.21	Moderate/ High
H11	PEC -> PEOU	Not supported	0.00	-
H12	PE -> PEOU	Supported	0.10	Small/ Moderate

factors, influencing developers, can be triggered at the same time: self-efficacy, training and perceived enjoyment.

Lastly, the high mean and significant effect of the developer community is worth mentioning. It is advisable to facilitate an engaging exchange amongst developers. Promoting a lively community is deemed beneficial to attract more developers to join.

6.3. Limitations and Further Studies

There are certain limitations in this analysis that create scope for future research. First, the literature review is mainly based on TAM studies. Even though TAM is an established model and has been tested many times, one may raise the concern that other models could also be used to test the usage of a technology. This limitation is linked to the fact that there might be missing constructs that could not be identified through this study's literature review. For example, TAM does not measure any direct effects on usage behavior. While the intention to use an NDE can very well be explained with PU and PEOU and respective constructs from theory, the predictive power of UB is lacking. Third-party developers might heavily use the NDE based on a group or management decision despite not finding it useful or not easy to use. Future research needs to test further constructs based on different models and also examine direct effects on usage behavior.

Additionally, collinearity and indirect effects of external constructs should be tested to reveal the overall effect of one construct. Finally, the limitation of the sample needs to be mentioned. Like in previous TAM studies, the questionnaire was conducted with a convenience sample in a contrived setting, which might go along with certain restrictions.

Besides the mentioned areas for improvement, this study identifies further areas of potential research. As identified in this study, the effect of self-efficacy, subjective norm, and output quality is undeniable and should be further examined.

Specific training methods could be compared to identify concrete management implications. Also, the potential of the developer community should be studied.

6.4. Conclusion

The objective of this paper was to analyze factors influencing developers' acceptance of native development environments for developing mobile apps, based on TAM. Through an extensive literature review, possible factors were identified, and then tested quantitatively using PLS-SEM. The research is consistent with the findings of previous TAM studies (Davis, 1989; Venkatesh, 2000; Venkatesh & Davis, 2000) that perceived usefulness and perceived ease of use positively influence the intention linked to the usage of the system. One of the most significant finding is that the impact of perceived usefulness on intention to use an NDE, compared to perceived ease of use, is noticeably higher. Therefore, managers should not merely focus on a neat interface, but also promote the great usefulness. Additionally, findings indicate a significant effect of output quality, subjective norm and developer community on the perceived usefulness of an NDE. The perceived ease of use of an NDE is significantly influenced by self-efficacy, training and perceived enjoyment.

The present study adds a relevant and novel contribution to the engaging research field of technology acceptance while also proposing concrete managerial implications for development environment providers.

References

Amatya, S., & Kurti, A. (2014). ICT Innovations 2013. Advances in Intelligent Systems and Computing. In V. Trajkovik & M. Anastas (Eds.), (pp. 219–229). Heidelberg: Springer.

Avkiran, N., & Ringle, C. (2018). *Partial Least Squares Structural Equation Modeling: Recent Advances in Banking and Finance*. Springer International Publishing.

Basole, R. C., & Karla, J. (2011). On the evolution of mobile platform ecosystem structure and strategy. *Business and Information Systems Engineering*, *3*(5), 313–322.

Bianco, V. D., Myllarniemi, V., Komssi, M., & Raatikainen, M. (2014). The role of platform boundary resources in software ecosystems: A case study. *11th Working IEEE/IFIP Conference on Software Architecture (WICSA), Sydney, Australia, April 7-11*, 11–20.

Boone, H. N., & Boone, D. A. (2017). Analyzing Likert Data. *The Journal of Extension*, *50*(2), 1456–1466.

Boudreau, K. J., & Lakhani, K. R. (2009). How to Manage Outside Innovation. *MIT Sloan Management Review*, *50*(4), 69–76.

Chesbrough, H. W., & Appleyard, M. M. (2007). Open Innovation and Strategy. *California Management Review*, *50*(1), 57–76.

Chin, J. P., Diehl, V. A., & Norman, K. L. (1988). Development of an Instrument for Measuring User Satisfaction of the Human-Computer Interface. *Proceedings of the SIGCHI conference on Human factors in computing systems*, 213–218.

Clement, J. (2020). Number of apps available in leading app stores 2019, Appfigures. *VentureBeat. Statista*. Retrieved 2020-02-05, from https://www.statista.com/statistics/276623/number-of-apps-available-in-leading-app-stores/

Cusumano, M. (2010). Technology strategy and management the evolution of platform thinking. *Communications of the ACM*, *53*(1), 32–34.

Davis, F. D. (1989). Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology. *MIS Quarterly*, *13*(3), 319–340.

Davis, F. D., Bagozzi, R. P., & Warshaw, P. R. (1989). User Acceptance of Computer Technology: A Comparison of Two Theoretical Models. *Management Science*, *35*(8), 982–1003.

Evans Data Corporation. (2017). *Software Developers Getting Younger, More Gender Diverse.* Retrieved 2020-04-06, from https://evansdata.com/press/viewRelease.php?pressID=253

Fishbein, M., & Ajzen. (1977). Belief, Attitude, Intention and Behavior: An Introduction to Theory and Research. *Contemporary Sociology*, *6*(2), 244–288.

Ghazawneh, A. (2012). Towards a Boundary Resources Theory of Software Platforms. *Jönköping International Business School*.

Goldbach, T., Kemper, V., & Benlian, A. (2014). Mobile application quality and platform stickiness under formal vs. self-control-Evidence from an experimental study. *International Conference on Information Systems*.

Hair, J. F., Hult, G. T. M., Ringle, C. M., & Sarstedt, M. (2014). *A primer on partial least squares structural equation modeling (PLS-SEM)*. Thousand Oaks, California: SAGE Publications Inc.

Henseler, J., Hubona, G., & Ray, P. A. (2016). Using PLS path modeling in new technology research: Updated guidelines. *Industrial Management and Data Systems*, *116*(1), 2–20.

Hertel, G., Niedner, S., & Herrmann, S. (2003). Motivation of software developers in open source projects: An Internet-based survey of contributors to the Linux kernel. *Research Policy*, *32*(7), 1159–1177.

Hilkert, D., Benlian, A., & Hess, T. (2010). Motivational drivers to develop apps for social software-platforms: The example of Facebook. *AMCIS 2010 Proceedings*, *86*, 1990–2000.

Huy, N. P., & Van Thanh, D. (2012). Developing Apps for Mobile Phones. *7th International Conference on Computing and Convergence Technology (ICCCT), IEEE*, 907–912.

Jet Brains. (2019). *The State of Developer Ecosystem 2019.* Retrieved 2020-04-06, from https://www.jetbrains.com/lp/devecosystem-2019/

Koch, S., & Kerschbaum, M. (2014). Joining a smartphone ecosystem: Application developers' motivations and decision criteria. *Information and Software Technology*, *56*(11), 1423–1435.

Lee, S. M., Kim, N. R., & Hong, S. G. (2016). Key success factors for mobile app platform activation. *Service Business*, *11*(1), 207–227.

Parker, G., Van Alstyne, M., & Jiang, X. (2017). Platform ecosystems: How developers invert the firm. *MIS Quarterly: Management Information Systems*, *41*(1), 255–266.

Pisano, G. P., & Verganti, R. (2008). Which Kind of Collaboration Is Right for You? *Harvard Business Review*, *86*(12), 78–86.

Sadi, M. H., Dai, J., & Yu, E. (2015). Designing Software Ecosystems: How to Develop Sustainable Collaborations? *Advanced Information Systems Engineering Workshops: CAiSE 2015 Workshops, LNBIP 215A*, 161–173.

StatCounter. (2020). *Mobile operating systems' market share worldwide from January 2012 to December 2019.* Statista. Retrieved 2020-02-06, from https://www.statista.com/statistics/272698/global-market-share-held-by-mobile-operating-systems-since-2009/

Steglich, C., Marczak, S., De Souza, C. R. B., Guerra, L. P., Mosmann, L. H., Figueira Filho, F., & Perin, M. (2019). Social aspects and how they influence mseco developers. *IEEE/ACM 12th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE)*, 99–106.

Straub, D. W. (1989). Validating Instruments in MIS Research. *MIS Quarterly*, *13*(2), 147–169.

Taylor, D., Voelker, T., & Pentina, I. (2011). Mobile Application Adoption by Young Adults: A Social Network Perspective. *Mobile Application Adoption by Young Adults: A Social Network Perspective*, *6*(2), 60–71.

Venkatesh, V. (2000). Determinants of Perceived Ease of Use: Integrating Control, Intrinsic Motivation, and Emotion into the Technology Acceptance Model. *Information Systems Research*, *46*(2), 186–204.

Venkatesh, V., & Davis, F. D. (2000). A Theoretical Extension of the Technology Acceptance Model: Four Longitudinal Field Studies. *Management Science*, *46*(2), 186–204.

Wagner, P., & Hering, L. (2014). Handbuch Methoden der empirischen Sozialforschung. In Baur & J. Blasius (Eds.), (pp. 661–673). Springer Fachmedien Wiesbaden.

Zanón, M. A., & Paz, D. P. (2009). *Disseny d'enquestes per a la investigació social*. Universitat de Barcelona.