



HAL
open science

Are Software Developers Just Users of Development Tools? Assessing Developer Experience of a Graphical User Interface Designer

Kati Kuusinen

► **To cite this version:**

Kati Kuusinen. Are Software Developers Just Users of Development Tools? Assessing Developer Experience of a Graphical User Interface Designer. 6th International Conference on Human-Centred Software Engineering (HCSE) / 8th International Conference on Human Error, Safety, and System Development (HESSD), Aug 2016, Stockholm, Sweden. pp.215-233, 10.1007/978-3-319-44902-9_14 . hal-01647699

HAL Id: hal-01647699

<https://inria.hal.science/hal-01647699>

Submitted on 24 Nov 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Are Software Developers just Users of Development Tools? Assessing Developer Experience of a Graphical User Interface Designer

Kati Kuusinen

University of Central Lancashire, Preston, Lancashire, PR1 2HE, UK
kkuusinen@uclan.ac.uk

Tampere University of Technology, Tampere, Finland

Abstract. Software developers use software products to design and develop new software products for others to use. Research has introduced a concept of developer experience inspired by the concept of user experience but appreciating also the special characteristics of software development context. It is unclear what the experiential components of developer experience are and how it can be measured. In this paper we address developer experience of Vaadin Designer, a graphical user interface designer tool in terms of user experience, intrinsic motivation, and flow state experience. We surveyed 18 developers using AttrakDiff, flow state scale, intrinsic motivation inventory and our own DEXI scale and compare those responses to developers' overall user experience assessment using Mann-Whitney U test. We found significant differences in motivational and flow state factors between groups who assessed the overall user experience either bad or good. Based on our results we discuss the factors that construe developer experience.

1. Introduction

Software development, especially the design and development of the graphical user interface (GUI) strongly benefits from the developers' ability to emphasize with the user and understand user needs. Simultaneously, it requires numerous technical skills to create GUIs. Traditionally GUI development has been done manually via application programming interfaces (APIs) which generally are interfaces to components often represented as libraries. Thus, when using APIs, developers program the GUI by calling required widgets and components via the API and giving parameters for them manually. The resulting GUI is only visible after it is runnable. Often in such approach, GUI is first designed and drawn by UI (user interface) designers separate from front-end developers. The design is often communicated to developers as ready-made static images of the future GUI, which might not be fully implementable as such [10], [11], [25].

Considering notation, we will use “*GUI designer*” throughout the paper to refer to software and “*UI designer*” to refer to a person.

Another approach for designing and developing GUIs is to use a GUI designer which often is a WYSIWYG (what you see is what you get) editor that allows the designer/developer to produce GUIs by dragging and dropping UI elements on screen layout. This approach can, for instance, help those UI designers who are less experienced in programming to produce more implementable design and developers would only need to refactor the produced code [23]. Or it could help a developer somewhat knowledgeable in user experience (UX) issues to produce design that a UX specialist could only review to ensure the quality of use.

Although qualities of both software developers and development work have been studied, developers have rarely been seen as users of development tools in the research. As developers are users of, for instance, GUI designers, all that is true to any user according to UX definitions (e.g. [19]), should apply also to developers. However, the dualistic role of the developer both as user of development tools and designer of new software products makes the developer special: Besides being users of GUI designers, developers should be able to understand the human user to be able to fulfill their needs with the GUI under development. A concept of developer experience (DX) has been suggested to address the particularities to software development [9]. The concept of DX is influenced by the concept of UX [9]. Moreover, DX consists of aspects related to cognition, affect, and conation and understanding of the concept should help practitioners in improving project environments with respect to developers' perceptions and feelings [9].

This paper has two main contributions. First, we present our DEXI scale developed for assessing the particularities of DX. Then we explore which factors of UX, flow state experience and intrinsic motivation correlate with developers' overall UX assessment regarding the assessed GUI designer (Vaadin Designer, <https://vaadin.com/designer>) thus contributing towards increased understanding of developer experience. Finally, we discuss these findings in relation to both related research and our previous findings on DX of integrated development environments.

The rest of this paper is organized as follows. Section 2 introduces background and related work. Section 3 describes the survey study method including description of the procedure and participants. Section 4 presents survey results. Section 5 discusses the results and section 6 discusses the validity and possibilities for future work. Finally, section 7 gives concluding remarks to the paper.

2. Background

Traditional GUI development has based on desktop screen model born in 1980's [31]. The model, based on windows and menus on a desktop computer screen used with a physical keyboard and a mouse was dominant from 1980's to around millennium when handheld devices started to become more common [31]. In fact, the desktop GUI generally still works in a similar fashion it was introduced in the 1980's. Mobile devices, however, have introduced new design challenges with their small-sized screens, virtual keyboards on touch screens, voice control mechanisms etc. [24], [31]. In addition, interoperability demands of cross-platform environments introduce

additional requirements for GUI design: The variety of platforms and operating systems has never been as massive as it is today. Furthermore, embedded systems and internet of things bring totally new kind of considerations on the research field of human-computer interaction.

2.1. GUI Designers

Commonly used GUI designers, sometimes called also GUI builders, include Eclipse Window Builder, Flex Builder, JetBrains Swing GUI designer, Netbeans Swing GUI Builder, Sencha Architect, and Vaadin Designer. GUI designers offer a visual interface to the underlying component and widget libraries. Therefore, designing the UI with GUI designer software can help to mitigate the problem with unimplementable design often seen when UX designers produce UI design as static UI images that developers need to interpret and amend for implementation. Such problems have been reported, for instance in [10], [11], [25].

In general, GUI designers help reduce the number of lines of code developers need to write for the GUI since those tools generate part of the code [31]. They also make GUI development faster [31] and can help UI designers and developers cooperate better and the designer to produce more implementable design compared to the use of static UI images as design artefacts [22], [23]. In cross-platform environments where GUI needs to be designed for several platforms, creating design manually is time-consuming [31]. The variety, functionality, and use of GUI design tools has grown rapidly during the mobile era. Moreover, digital prototyping and design tools have replaced paper prototyping in many organizations because of their fastness to use, repeatability, and detailed look and feel [22], [23]. In addition, responsive design (design that scales based on the screen size of the device) makes the use of paper prototypes difficult as the designer cannot design fixed-sized screens and with paper prototypes it is hard to understand what will be visible for the user at once [23].

In addition to actual GUI designer software, there are numerous prototyping tools available for rapid prototyping before the GUI building. Those include, for instance, Balsamiq, InVision, and Pencil. These tools differ from GUI designers in that they are mostly intended for sketching and making mock-ups and prototypes instead of creating the production version of the UI.

2.2. User Experience

The standard definition of user experience (UX) is as follows: a “*person's perceptions and responses resulting from the use and/or anticipated use of a product, system or service*” [19]. Commonly, UX is understood as subjective, context-dependent, and dynamic [29]. It is affected by user's expectations, needs and motivation, system's characteristics such as purpose and functionality, and the context of use including physical, organizational and psychological aspects [18]. One of the most commonly referred models of UX, the hedonic-pragmatic model divides user experience into

hedonic or non-utilitarian dimension and pragmatic or instrumental dimension [15]. Hassenzahl [15] further divides hedonic quality into two sub dimensions of identification and stimulation while instrumental quality contains mostly items related to usability and usefulness. Usability is often seen as a necessary precondition for good UX [16], [28].

2.3. Developer Experience

Developer experience (DX) is a recent concept addressing the experience of software development [9]. Whereas UX occurs in relation to the use of software, DX addresses the creation of new software. Consequently, UX does not cover all aspects of DX [8]: Developers use development tools in order to create software that provokes experience in the user. On the other hand, developers are users of development software and thus it can be argued that the concept of UX can be applied also to software developers. What makes the developer special, however, is the dualistic nature of being simultaneously both producer and user of software.

Fagerholm's concept of DX [8] presents that DX is the sum of cognitive, affective, and conative mental processes of developer in a context consisting of social and technical environments in which they interact with other developers and numerous technical artefacts (Figure 1).

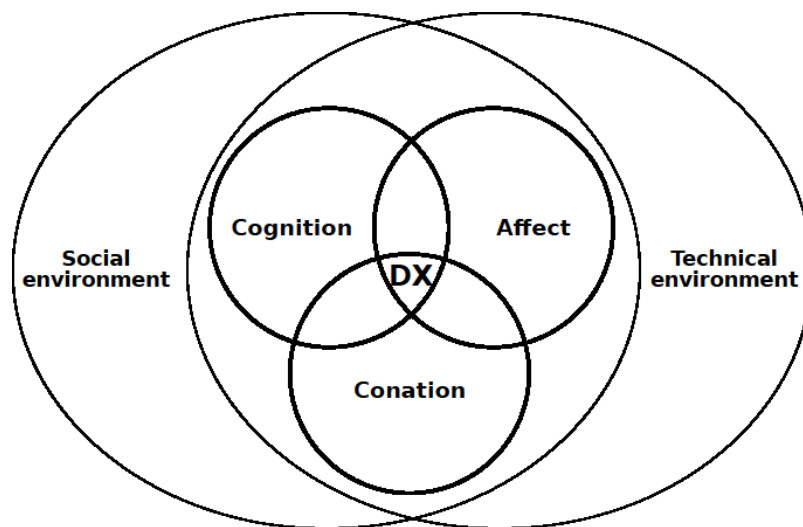


Figure 1. Concept of developer experience [8]

The concept of DX aims at providing an intuitive abstraction of the vast variety and quantity of human factors that influence developers and the outcomes of software development [9]. While UX considers the context of use, DX considers the context of development. DX includes also aspects beyond software tools, such as development processes, modeling methods, and other means of structuring development tasks.

Some of these aspects are embedded in tools such as GUI builders while others are part of organizational practices. The software development activity and environment differ in significant ways from other information-intensive activities and environments. For example, software development requires a nested understanding: developers use software installed on a computer system to build another software system that is to be used by users to accomplish their task in their particular domain providing them desired UX. Also, developers frequently configure and extend their tools, and are in effect continuously developing both the development environment and the end product at the same time.

2.4. Motivation and Flow State Experience in Software Development

One of the current influential theories of motivation is the self-determination theory from Deci and Ryan [7]. They distinguish between intrinsic and extrinsic motivations. Intrinsic motivation refers to engaging in a task because of it is inherently pleasurable and satisfying whereas extrinsic motivation refers to engaging in a task because of its outcomes, the task is used as a means to lead to the outcome [7].

Developer motivation is as an important factor in software development. Majority of studies on motivation in development context report that developers are distinct from other occupational groups with respect to motivation [1]. "The work itself" is the most commonly cited motivator, but there is a lack of detail regarding what aspects of the work is motivating, how motivational processes occur, and what the outcomes of motivating developers are [1], [12]. Investigations into affective aspects of software development show the importance of considering also affect. The presence and variation of developers' emotions over time has been documented [36]. Programming tasks are influenced by mood [21], and happiness has been found to have productivity benefits [14]. This underlines the importance of considering affective aspects both for purposes of well-being and outcomes.

Flow experience refers to a state of concentration so focused that it amounts to absolute absorption in an activity [4]. Applicable to both work and leisure [5], flow builds on intrinsic motivation and internal reward over the achievement rather than on external goal or recognition. Its effect can be characterized as being totally focused on a particular task at hand, so that the person becomes fully immersed in a feeling of energized focus, full involvement, and enjoyment in the process of the activity. While immersed, three conditions have to be met to achieve a flow state [6]:

1. One must be involved in an activity with a clear set of *goals and progress*;
2. The task at hand must have clear and immediate *feedback*;
3. One must have a good balance between the perceived *challenges* of the task at hand and their own perceived *skills*, so that there can be confidence in one's ability to complete the task at hand.

These three conditions of flow state experience are comparable with what the literature has found focal for software developers and development work. For example, clear goals, technically challenging work, knowing the purpose of tasks,

making good use of skills, and getting feedback are widely reported motivators of software developers [1], [35]. In addition, the core of software development work is writing the program code, which demands the ability to concentrate and work alone for several hours [2]. Both these motivators of developers and the nature of development work makes the concept of flow state experience interesting in the context of software development.

3. Method

Our research has an overall goal to increase understanding of DX. We aim to clarify how flow state experience, intrinsic motivation, and UX are intertwined in software development and with developer experience. This will enable improvement of development tools to better support developers' ability to experience flow in their work and to enhance developers' intrinsic motivation towards their work. Our hypothesis is that these factors make developers' work more enjoyable and increases their productivity. We have previously conducted a similar study in the context of integrated development environments [26].

This paper reports a survey study that was organized for two main reasons. 1) To examine the relationship of flow state experience, intrinsic motivation, and developer experience and 2) to evaluate the DX of a commonly used GUI designer called Vaadin Designer. The survey was organized together with Vaadin Ltd, the provider of the GUI designer under evaluation.

We present the following research question:

What kind of relation does developer experience and the ability of Vaadin designer to fulfill developers' needs have with flow state experience and intrinsic motivation?

3.1. Survey Contents

We conducted a survey study measuring developers' self-reported experiences considering a particular GUI designer. Our web survey consisted of the following four scales:

1. the Short Dispositional Flow State Scale (SDFS-2) [20] used in its entirety,
2. parts of the Intrinsic Motivation Inventory (IMI) [34] including questions related to interest/enjoyment, perceived competence, effort/importance, and perceived choice, and
3. the short version of the AttrakDiff-2 (SAD-2) [17] used in its entirety, and
4. our own Developer Experience Scale (DEXI) [26]. The scales, except DEXI, were selected since they are widely used and validated. They are also short enough to be combined in a survey.

In addition, we asked for overall UX assessment (OUX) and overall ability of the GUI designer to fulfill the needs of the respondent (NFS) as follows:

“*How would you rate the overall user experience of Vaadin Designer?*” using a seven-point Likert scale from “Bad” (1) to “Good” (7), and

“*How well does Vaadin Designer respond to your needs?*” using a seven-point Likert scale from “Not at all” (1) to “Completely” (7). All other measurements are compared to OUX assessment. NFS has no special role in this paper, it is used only as one of the independent variables.

Furthermore, we asked the following open-ended questions considering the use of GUI designers: “*In your opinion, what are the best qualities of Vaadin Designer?*”, “*How could Vaadin Designer better support your development work?*”, and “*Describe the ideal way of creating UI*”.

Finally, we also asked demographics including experience of software development in years, experience of the particular GUI designer (seen it in use, used it [once or twice, a few times, several times, regularly for less than a month, regularly for over a month but less than a year, regularly for over a year]), age, country the respondent was based in (list of countries provided), size of the organization they were working for (individual developer, micro organization (<10 employees), small (10-100), medium-sized (100-1000), large (over 1000 employees)), and the user group of the GUI designer they belonged (UI designer [they design the UI, someone else does the coding], UI developer [they produce the code], evaluator [typically architect or lead developer who assess the feasibility of different technologies for the organization], other).

Next we present the used scales in detail. We measured the frequency with which developers experience different dimensions of flow during software development activities by using the **Short Dispositional Flow State Scale (SDFS-2)** [20]. We used a 7-level Likert scale, ranging from 1 (never) to 7 (always). The instructions for the questions were “*Please evaluate how often (from 1 = never to 7 = always) you experience the following while you are doing development work with Vaadin Designer*”. The SDFS-2 measures nine dimensions of flow, each with one item (Table 1). In addition to the nine SDFS-2 items, an additional item was included to measure the experience of frustration: “*I feel frustrated*”.

Table 1. SDFS-2 scale. Dimensions of state of flow and related survey items [20]

Flow item	Survey question
Challenge-skill balance	I feel I am competent enough to meet the high demands of the situation
Action awareness	I do things spontaneously and automatically without having to think
Clear goals	I have a strong sense of what I want to do
Unambiguous feedback	I have a good idea while I am performing about how well I am doing
Concentration on task	I am completely focused on the task at hand
Sense of control	I have a feeling of total control
Loss of self-consciousness	I am not worried about what others may be thinking of me
Transformation of time	The way time passes seems to be different from normal
Autotelic experience	The experience is extremely rewarding

Intrinsic Motivation Inventory (IMI). Since the original IMI is long and partially repetitive, and shorter versions have been commonly used [30], we used a shortened version (Table 2). Utilized scale included selected items from the following IMI subscales: interest/enjoyment (the actual self-report measure of intrinsic motivation), perceived competence, effort/importance, and perceived choice. Framing of the question and assessment scale was according to IMI. Thus, the question was as follows: “*For each of the following statements, please indicate how true it is for you, using the following scale*”, and we utilized a seven-level scale ranging from 1 (not at all true) to 7 (very true).

Table 2. Selected subscales and survey items of IMI [34]

Subscale	Survey item
Interest/Enjoyment	I enjoy UI creation very much
	I think UI creation is a boring activity
	I enjoy using Vaadin Designer very much
Perceived competence	I am satisfied with my performance at UI creation
	I am pretty skilled in UI creation
	I am pretty skilled in using Vaadin Designer
Effort/Importance	It is important to me to do well in UI creation
Perceived choice	I use Vaadin Designer because I have no choice

DX and UX assessment. Both **DEXI** scale and **Short AttrakDiff 2** were used in the survey to assess DX or UX of the particular GUI designer tool. The wording from AttrakDiff [15] was used with both scales as follows: “*With the help of the word-pairs, please enter what you consider the most appropriate description for Vaadin Designer:*” The short version of AttrakDiff-2 (SAD-2) was used as such [17]. It contains four items (word-pairs) for both practical (PQ) and hedonic quality (HQ), and one for measuring goodness and beauty each (general quality GQ) (Table 3).

Table 3 Short AttrakDiff 2 [17]

Measurement	Word-Pairs
SAD_General	Bad – Good
SAD_General	Ugly – Beautiful
SAD_Practical	Confusing – Structured
SAD_Practical	Practical – Impractical
SAD_Practical	Unpredictable – Predictable
SAD_Practical	Simple – Complicated
SAD_Hedonic	Dull – Captivating
SAD_Hedonic	Stylish – Tacky
SAD_Hedonic	Cheap – Premium
SAD_Hedonic	Creative – Unimaginative

We formed our own DEXI scale for measuring additional items related to developers' UX, or DX. Thus, we aimed at construing a scale that would be relevant for software development. We selected word-pairs for DEXI from the dataset of a meta-study considering the often used UX items in research tailored for work-related systems [37], amongst concepts that have been used to describe DX [9], and amongst those characteristics of development platforms developers find beneficial [13]. We utilized the structure and wording of AttrakDiff in DEXI. We selected 5 items (DEXI_P) measuring pragmatic quality and 6 items (DEXI_H) measuring hedonic (non-utilitarian) quality. One item (DEXI_G) measured general quality (Table 4). DEXI scale was piloted with students in a code camp experiment and used in a study reported in [26].

Table 4 DEXI scale

Quality	Survey question	Source
DEXI_G	Recommendable – Not recommendable	[33], [37]
DEXI_P	Efficient – Inefficient	[13], [37]
DEXI_P	Flexible – Inflexible	[13]
DEXI_P	Easy to learn – Difficult to learn	[13], [37]
DEXI_P	Limited – Extensive	[13]
DEXI_P	Uninformative – Informative	[37]
DEXI_H	Motivating – Discouraging	[9], [37]
DEXI_H	Increases respect – Decreases respect	[37]
DEXI_H	Enjoyable – Unenjoyable	[37]
DEXI_H	Promotes creativity – Suppresses creativity	[2], [37]
DEXI_H	Engaging – Uninvolving	[2], [37]
DEXI_H	Brings me closer to others – Separates me from others	[2], [9], [13], [37]

3.2. Procedure

We organized an online survey to evaluate Vaadin Designer, a commonly used GUI designer. The survey was organized together with Vaadin Ltd, the provider of Vaadin Designer. The survey had a front page presenting informed consent statements adopted from World Health Organization's Informed consent form template for qualitative studies [38]. Participants were informed of the organizer, purpose, target respondents, contents, and confidentiality of the survey as well as of the expected completion time and treatment of the information they will provide. We instructed only those who have been using the GUI designer to respond to the survey, and to respond only once. The survey was available for the respondents for four weeks. However, no notifications were sent for the request of the company.

3.3. Respondents

In total, we got 20 responses of which 18 were valid. Organizations of the 18 respondents were as follows, individual developers: 4, micro businesses: 4, small businesses: 3, middle-sized businesses: 4, and large businesses: 3. Mean experience from software development was 10 years ranging from 1 to 25 (SD: 7 years). Three of the respondents considered themselves UI designers, 9 were UI coders, 4 were evaluators and 2 were full-stack developers. Regarding the experience of the assessed GUI designer, one respondent had only watched a demonstration or somebody else using the GUI designer, 4 had tried it once or twice, 4 had used it a few times, 4 several times, 2 had used it regularly for less than a month, and 3 had used it regularly for more than a month but less than a year. Given this, respondents can be considered experts in software development but novices in using the GUI designer. Respondents' mean age was 36, SD 10 years, range 21 to 59 years. The respondents were from all over the world: Germany: 4 respondents, India: 2, Spain: 2, US: 2, and 1 from each of the following: Belgium, Columbia, Ecuador, Hungary, Indonesia, Italy, Portugal, and Romania. Thus, 11 respondents were from European countries, 3 from Americas, and 3 from Asian countries.

3.4. Analysis

The design of data analysis was finalized after the data gathering to suit the small data set of 18 valid responses. Thus, we approach the research question “*what kind of relation does developer experience and the ability of Vaadin designer to fulfill developers' needs have with flow state experience and intrinsic motivation?*” from two angles with two statistical methods using SPSS. First we run a more robust statistical test, namely Mann-Whitney U test on the data and compare between those respondents who assessed their overall UX with Vaadin Designer high and those who assessed it low. Second, we run Kendall's Tau correlation analysis on the data to address which items in each scale correlate with the overall UX assessment. Both Mann-Whitney U test and Kendall's Tau correlation analysis are nonparametric methods suitable for smaller and non-normal datasets. In Kendall's Tau correlation analysis, we calculate the correlation between OUX and each item in SDFS-2 and IMI. In using Mann-Whitney U, we test for equality of means to compare responses of those developers who assessed OUX in the seven-point scale 3 or lower (close to “bad”) and those who assessed OUX 5 or higher (close to “good”). In the valid dataset, there were 8 respondents that considered the OUX of the GUI designer bad (ratings 1-3) and 9 who considered it good (ratings 5-7), thus one of the respondents assessed OUX with four, neutral, and this assessment was discarded from the analysis. We name the respondent groups “OUX_bad” (ratings 1-3) and “OUX_good” (ratings 5-7), respectively. Our null hypothesis for Mann-Whitney U is as follows:

H: “*The distribution of [each survey item] is the same across respondent groups OUX_bad and OUX_good.*”

The hypothesis states that respondents assessing OUX as bad (group OUX_bad) cannot be significantly separated from those assessing it good (group OUX_good) in their responses to DX, UX, intrinsic motivation, and flow state experience. If the hypothesis is not supported, it means that there is statistically significant difference in responses between those groups.

3.5. Impact of Demographics

We ran Kendall's Tau and Mann-Whitney U test between OUX and demographic variables where feasible. There were no significant correlations and the null hypothesis remained for all demographics: there were no statistically significant differences in the demographics between groups of OUX_good and OUX_bad, i.e. between those who assessed overall UX high (good) and those who assessed it low (bad).

4. Results

We present first the Mann-Whitney U test results, continue with Kendall's Tau correlation analysis and end the results section with answers to the open-ended questions.

4.1. Mann-Whitney U Test Results

Table 5 presents only statistically significant values of Mann-Whitney U test to save space. These are the items for which the null hypothesis is not supported. Thus, for all the items in Table 5, there is a statistically significant difference between those respondents who considered the overall UX of the GUI designer bad (1-3) and those who considered it good (5-7 on a seven-point scale from bad to good). It means these items can also be used to differentiate between those groups of respondents. For example, respondents who do things spontaneously and automatically without having to think also consider the overall UX good significantly more often than those who do not, and vice versa. These results indicate that it is important to support developer flow and prevent frustration to ensure good developer experience.

Participants who assessed UX of Vaadin Designer high enjoyed using the Designer more and considered themselves more skilled with the Designer significantly more often than participants who assessed UX low. Furthermore, they were more spontaneous in their tasks and felt frustration significantly less than low assessors. Thus, in this data set, flow state experience (challenge-skill balance in particular) and intrinsic motivation towards the tool had an impact on user experience assessment.

Of the UX items in SAD-2 and DEXI, the null hypothesis was not supported for 2 of 3 word-pairs measuring general quality, for 5 of 9 word-pairs measuring practical quality and 2 of 10 measuring hedonic quality. In DEXI scale 6 out of 12 word pairs (50%) were able to differentiate between OUX_bad and OUX_good whereas SAD_2 was able differentiate between those only with 3 word pairs out of the total 10 (30%). Moreover, no word-pairs measuring hedonic quality in Short AttrakDiff 2 were able to differentiate OUX_good and OUX_bad whereas two word-pairs, namely motivating – discouraging and enjoyable – unenjoyable in DEXI were. However, larger datasets are required for further evaluation of the scales.

Table 5. Statistically significant Mann-Whitney U test results between respondent groups OUX_good and OUX_bad for NFS and all subscales (OUX, SDFS-2, IMI, SAD-2, DEXI). N = 17

Survey item per subscale	U	p	r
How well does Vaadin Designer respond to your needs?	3.50	<.001	0.77
I do things spontaneously and automatically without having to think	8.50	<.01	0.65
I feel frustrated	6.00	<.01	0.71
I enjoy using Vaadin Designer very much	8.50	<.01	0.65
I am pretty skilled in using Vaadin Designer	11.00	<.05	0.61
SAD_G Bad – Good	2.00	<.001	0.81
SAD_P Unpredictable – Predictable	12.00	<.05	0.58
SAD_P Practical – Impractical	7.00	<.01	0.69
DEXI_G Recommendable – Not recommendable	6.50	<.01	0.70
DEXI_P Easy to learn – Difficult to learn	10.00	<.05	0.62
DEXI_P Flexible – Inflexible	9.00	<.01	0.64
DEXI_P Efficient – Inefficient	5.50	<.01	0.73
DEXI_H Motivating – Discouraging	1.00	<.001	0.83
DEXI_H Enjoyable – Unenjoyable	6.00	<.01	0.71

4.2. Kendall's Tau Correlation Analysis

As expected, overall UX assessment (OUX) strongly correlated with need fulfillment (NFS), $r_{\tau} = .84$, $p = < .001$. This finding is in line with Hassenzahl's results [17]. Correlations between OUX and individual items in Short AttrakDiff 2 and DEXI scales are presented in Table 6. What is notable is that 9/12 (75%) of word-pairs in DEXI scale significantly correlated with OUX while only 4/10 (40%) of SAD2 had a significant correlation with OUX, all of the latter measuring pragmatic or general quality. Otherwise there is nothing surprising, the majority of the word-pairs correlate with OUX which can be anticipated as all the items are to address UX.

Intrinsic motivation. OUX had a significant correlation with both items that measure intrinsic motivation in relation to the assessed GUI designer, namely *I enjoy using Vaadin Designer very much* and *I am pretty skilled in using Vaadin Designer* (Table 7). The first mentioned measure interest/enjoyment while the second measures perceived competence, both related to the tool under evaluation. Other items had no significant correlations with OUX.

Table 6. Results of Kendall's Tau (r_{τ}) correlation analysis between overall UX (OUX) assessment and individual items of SAD2 and DEXI scales. Legend: SAD = Short AttrakDiff 2, G = general quality, P = practical quality, H = hedonic quality, n.s. = not significant. N = 18

Measure	Items	r_{τ}	p
SAD_G	Bad – Good	.770	<.001
SAD_P	Practical – Impractical	-.650	<.05
SAD_P	Confusing – Structured	.530	<.01
SAD_P	Unpredictable – Predictable	.470	<.05
SAD_H	Stylish – Tacky	-.330	n.s.
SAD_P	Simple – Complicated	-.310	n.s.
SAD_H	Dull – Captivating	.270	n.s.
SAD_G	Ugly – Beautiful	.240	n.s.
SAD_H	Creative – Unimaginative	-.130	n.s.
SAD_H	Cheap – Premium	.090	n.s.
DEXI_H	Motivating – Discouraging	-.780	<.001
DEXI_G	Recommendable – Not recommendable	-.750	<.001
DEXI_P	Efficient – Inefficient	-.680	<.001
DEXI_P	Flexible – Inflexible	-.630	<.01
DEXI_H	Enjoyable – Unenjoyable	-.620	<.01
DEXI_P	Easy to learn – Difficult to learn	-.570	<.01
DEXI_H	Increases respect – Decreases respect	-.440	<.05
DEXI_H	Promotes creativity – Suppresses creativity	-.430	<.05
DEXI_H	Engaging – Uninvolving	-.400	<.05
DEXI_H	Brings me closer to others – Separates me from others	-.360	n.s.
DEXI_P	Limited - Extensive	-.230	n.s.
DEXI_P	Uninformative – Informative	.170	n.s.

Flow state experience. Regarding flow state experience, OUX correlated with the following items. *I feel frustrated, the experience is extremely rewarding, the way time passes seems to be different from normal, I have a feeling of total control, I do things spontaneously and automatically without having to think, and I feel I am competent enough to meet the high demands of the situation* (Table 8). Thus, there was a significant correlation between OUX and perceived competence item both in IMI and SDFS-2 as well as between OUX and enjoyment in IMI and autotelic experience in SDFS-2. These findings are in line with our previous results considering integrated

development environments [26]; perceived competence and items assessing enjoyment were correlated in that study too.

Table 7. Results of Kendall's Tau (r_{τ}) correlation analysis between overall UX (OUX) assessment and individual items of IMI subscale. N = 18

Intrinsic motivation item	r_{τ}	p
I enjoy using Vaadin Designer very much	.560	<.01
I am pretty skilled in using Vaadin Designer	.490	<.01
I think UI creation is a boring activity	.350	n.s.
I use Vaadin Designer because I have no choice	-.290	n.s.
It is important to me to do well in UI creation	-.150	n.s.
I enjoy UI creation very much	.100	n.s.
I am satisfied with my performance at UI creation	.090	n.s.
I am pretty skilled in UI creation	.040	n.s.

Table 8. Results of Kendall's Tau (r_{τ}) correlation analysis between overall UX (OUX) assessment and individual items of SDFS-2. N = 18

Dispositional flow state scale item	r_{τ}	p
I feel frustrated	-.740	<.001
The experience is extremely rewarding	.570	<.01
The way time passes seems to be different from normal	.500	<.01
I have a feeling of total control	.490	<.05
I do things spontaneously and automatically without having to think	.460	<.05
I feel I am competent enough to meet the high demands of the situation	.420	<.05
I am completely focused on the task at hand	.290	n.s.
I am not worried about what others may be thinking of me	.260	n.s.
I have a good idea while I am performing about how well I am doing	.100	n.s.
I have a strong sense of what I want to do	.090	n.s.

4.3. Responses to Open-Ended Questions

Respondents considered the GUI developer makes the GUI creation process faster compared to the traditional method. They also liked that the GUI developer enables rapid feedback gathering on the actual working UI. They desired for full support on drag and drop feature in that the whole UI could be created simply by placing components on screen. Moreover, they asked for proper support for web UIs and responsive design and that the GUI designer would create proper code based on the UI design on selected programming language instead of HTML. Ideally, the system should also offer support for user involvement in the UI design process.

5. Discussion

5.1. Research Question Revisited

This paper addressed the following research question:

“What kind of relation does developer experience and the ability of Vaadin designer to fulfill developers’ needs have with flow state experience and intrinsic motivation?”

We addressed the research question first in terms of scale items’ ability to differentiate between respondent groups with low and high overall UX assessment (OUX_bad and OUX_good), respectively. The null hypothesis was rejected for flow state experience scale item *“I do things spontaneously and automatically without having to think”*, the additional item *“I feel frustrated”*, and intrinsic motivation inventory items *“I enjoy using Vaadin designer very much”*, and *“I am pretty skilled in using Vaadin Designer”* which means that on these items, there is statistically significant difference between responses of those developers assessing OUX good (OUX_good) and those that assess it bad (OUX_bad). Second, we addressed the research question with a correlation analysis. The same IMI scale items resulted from both analyses: *“I enjoy using Vaadin designer very much”*, and *“I am pretty skilled in using Vaadin Designer”* correlated with OUX assessment and could differentiate between OUX_bad and OUX_good. This result is in line with our previous finding: intrinsic motivation was a significant predictor of DX in [26]. A number of flow state scale items were correlated with OUX, including the two items that were able to differentiate between OUX_good and OUX_bad. Mann-Whitney U test is more powerful than correlation analysis and thus it is expected that the correlation analysis includes the results of the test.

5.2. Discussion on Related Research

We got similar results in two previous studies. In our study with integrated development environments, hedonic quality was on the borderline of being significant predictor of DX ($p = .05$) while DX could not be predicted from pragmatic or generic quality [26]. Moreover, in a study on work-related systems [27], we found that rather than the system being professional, respondents appreciated that the system made them feel professional about themselves. Similarly, in this study, system being creative was not associated with DX while the system making the developer feel creative was. In fact, many of the DEXI scale items measuring hedonic quality address the emotion the system evokes in the respondent rather than qualities of the system (motivating, promotes creativity, increases respect, and brings me closer to others). Furthermore, of word pairs measuring pragmatic quality, simple-complicated has been difficult to

interpret in work-related context and it does not correlate significantly with DX in this study. Our explanation is that simple systems generally cannot address all the complexity many work-related activities require. For instance, software development has often been described as complex whereas constructing physical buildings is only complicated [3]. Similarly, it is not straightforward to interpret the word pair ‘technical – human’ of AttrakDiff [15] in development context which is both technical and human.

‘Limited – extensive’ and ‘informative – uninformative’ were important for DX considering IDEs [26] while they were uncorrelated with DX in this study. Our explanation is that IDEs unlike GUI designers need to be extensive because software systems often are unique and base on new technologies and their combinations whereas GUIs often are conventional and base on design conventions while more interesting solutions are hidden from the user beneath the presentation layer. Moreover IDEs are large development environments whereas GUI designers are inherently much more limited systems.

Research on GUI designers and GUI design tools is scarce. There are papers on developed designs on new GUI tools but research on their usage on this century seems to be nonexistent. To our knowledge, Myers [31] is still the most extensive article on the topic. There was a workshop on “The Future of User Interface Design Tools” in CHI conference in 2005 [32] but the extended abstract nor the few citing papers do not contain articles on GUI designers or developer work on GUI creation. Also research on developer experience is still scarce.

Figure 2 summarizes our statistical test results both from the study on GUI designers presented in this paper and on IDEs presented in [26]. Based on these studies, the most relevant factors of flow state experience and intrinsic motivation are enjoyment and reward the work offers.

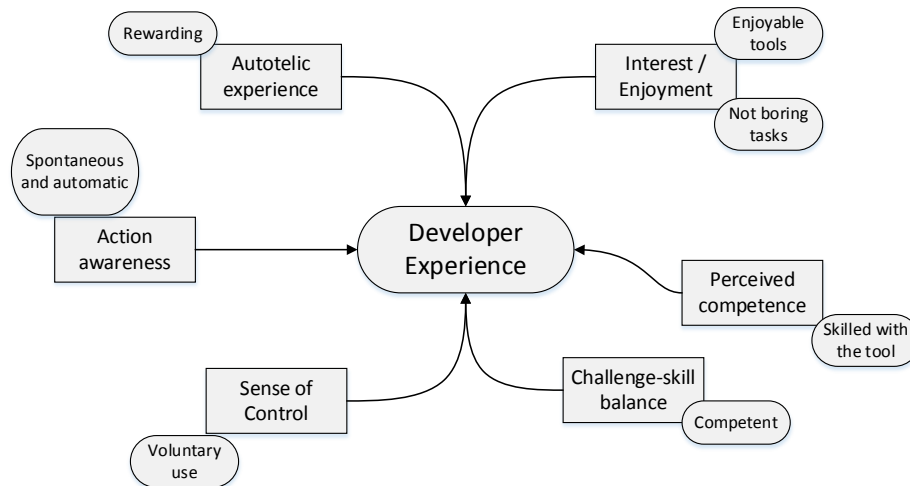


Figure 2. Most relevant factors of intrinsic motivation and flow state experience to developer experience.

Sense of control was a significant predictor of DX in [26]. Perceived competence and challenge-skill balance help the developer to manage with demanding tasks while working spontaneously and automatic without having to think is often understood as working under flow state experience. These are commonly mentioned motivation factors in related research [1] and our research connects them with development tools and developer experience of those tools. Advanced technologies motivate developers in their work [1] and development tools are often intertwined with utilized technologies.

6. Limitations and Future Work

The most obvious threat to validity is arguably the small respondent number, 18 software specialists. This is particularly critical for correlation analysis, especially considering the large number of connected variables. Small sample size can cause fluctuation in correlation values which makes the analysis less powerful and thus reduces the likelihood of reflecting a true effect. It also prevented us from using more powerful methods. For Mann-Whitney U test the respondent number is sufficient although the test would have benefitted from a larger sample size.

Further studies are needed to validate DEXI scale and to gain better understanding on the concept of DX. In addition, the population was diverse which can be seen both strengthening the results but also making the data incoherent. Our results however, are in line with our previous results considering another dataset on DX of integrated development environments [26] which is to increase the likelihood of reliability in this study.

The results cover only one GUI designer. Thus, larger respondent populations and users of several GUI designers ought to be studied. Moreover, to compare the DX of APIs and GUI designers in larger scale remains another interesting topic for future research. In addition, user groups such as UI/UX designers and UI/front-end developers and their needs could be studied separately or in comparison to increase the understanding of how designers and developers can better work together. DEXI scale could also be evaluated in the context of end user development and to compare developers' and end user developers' experiences. Finally, it could be interesting to study if there was a difference between developers' assessment of overall UX score and overall DX score and how those relate to UX and DEXI scales. To conclude, this study does not come without its limitations but it serves as one opening to further DX studies and it is also one of the scarce studies on GUI designer tools.

7. Conclusion

This paper presented a survey study on developer experience (DX) of a GUI designer tool. It introduced DEXI scale for assessing DX of development tools. We presented both quantitative and qualitative results on DX, measured with Short AttrakDiff 2

[17], a commonly used UX scale and DEXI scale specifically designed to assess DX. We found a statistically significant difference in responses between those developers who assessed the overall UX of the GUI builder good and those who assessed it bad considering both their flow state experiences and intrinsic motivation. Our results suggest that developer experience can be improved by fostering the developer's interest and enjoyment, offering rewarding experiences, supporting challenge-skill balance and feeling of competence. Finally, supporting action awareness and providing a sense of control are also key factors of good DX. Considering what software developers appreciate in GUI designer tools, we found that making the GUI design process faster and ensuring early feedback on the working version of the GUI under development were especially liked. Finally, the optimal way to create GUIs was often described as having full support on drag and drop and on responsive design, involving the user in the process, and the ability of the GUI designer tool to export the design into specific programming language instead of HTML code.

Acknowledgment

I want to thank all the survey respondents for their valuable contribution. I also want to thank Vaadin for making the study possible. I am grateful for my contact person in the company for their help in organizing the survey. This research has been conducted while I was working for Tampere University of Technology, Finland. My research has been supported by TEKES as part of the Need for Speed research program of DIGILE (Finnish Strategic Centre for Science, Technology and Innovation in the field of ICT and digital business).

References

1. Beecham, S., Baddoo, N., Hall, T., Robinson, H., and Sharp, H. (2008). Motivation in software engineering: A systematic literature review. *IST*, 50, 860–878.
2. Capretz, L. F., & Ahmed, F. (2010). Making sense of software development and personality types. *IT professional*, 12(1), 6-13.
3. Coplien, J. O., & Bjørnvig, G. (2011). *Lean architecture: for agile software development*. John Wiley & Sons.
4. Csikszentmihalyi, M. *Flow: The psychology of optimal experience*. Vol. 41. New York: HarperPerennial, 1991.
5. Csikszentmihalyi, M; LeFevre, J. Optimal experience in work and leisure. *Journal of Personality and Social Psychology*, Vol 56(5), May 1989, 815-822.
6. Csikszentmihalyi, M.; Abuhamed, S. & Nakamura, J. (2005), "Flow", in Elliot, A., *Handbook of Competence and Motivation*, New York: The Guilford Press, pp. 598–698.
7. Deci, E., & Ryan, R. M. (2012). *Self-determination theory. Handbook of theories of social psychology*, Los Angeles, SAGE. ISBN: 9780857029607
8. Fagerholm, F. (2015). *Software developer experience: Case studies in lean-agile and open-source environments*. Doctoral Thesis. Series of publications A, report A-2015-7. University of Helsinki. ISBN 978-951-51-1747-2.

9. Fagerholm, F., & Münch, J. (2012). Developer experience: Concept and definition. Proc. International Conference on Software and System Process (pp. 73-77). IEEE Press.
10. Ferreira, J., Sharp, H., Robinson, H. User experience design and agile development: managing cooperation through articulation work. *Software: Practice and Experience* 41(9), 2011, John Wiley & Sons, pp. 963-974.
11. Ferreira, J., Sharp, H., Robinson, H. Values and Assumptions Shaping Agile Development and User Experience Design in Practice. Proc. *XP 2010*, LNBIP 48, pp. 178-183, (2010).
12. Franca, A. C. C., Gouveia, T. B., Santos, P. C. F., Santana, C. A., and da Silva, F. Q. B. (2011). Motivation in software engineering: A systematic review update. Proc. Evaluation and Assessment in Software Engineering (EASE), pages 154-163
13. Gass, O., Meth, H., & Maedche, A. (2014). PaaS Characteristics for Productive Software Development: An Evaluation Framework. *IEEE Internet Computing*, 18(1), 56-64.
14. Graziotin, D., Wang, X., Abrahamsson, P., Software Developers, Moods, Emotions, and Performance, *IEEE Software*, vol. 31, no. 4, pp. 24-27, July-Aug., 2014
15. Hassenzahl, M. The Interplay of Beauty, Goodness and Usability in Interactive Products. Proc. HCI. Lawrence Erlbaum Associates, 19, 4 (2004), 319-349.
16. Hassenzahl, M. (2008). User experience (UX): towards an experiential perspective on product quality. Proc. 20th International Conference of the Association Francophone d'Interaction Homme-Machine (pp. 11-15). ACM
17. Hassenzahl, M., Diefenbach, S., & Göritz, A. (2010). Needs, affect, and interactive products—Facets of user experience. *Interacting with Computers*, 22(5), 353-362.
18. Hassenzahl, M. and Tractinsky, N. User experience - A research agenda. *Behaviour & Information Technology* 25 (2), 91-97. 2006
19. ISO 9241. Ergonomic requirements for office work with visual display terminals (VDTs) – Part 11: Guidance on usability. Genève, CH: International Organization for Standardisation. 1998
20. Jackson, S. A., Martin, A. J., & Eklund, R. C. (2008). Long and short measures of flow: The construct validity of the FSS-2, DFS-2, and new brief counterparts. *JSEP*, 30(5), 561.
21. Khan, I. A., Brinkman, W.-P., and Hierons, R. M. (2011). Do moods affect programmers' debug performance? *Cognition, Technology & Work*, 13(4):245-258
22. Kuusinen, K. BoB - A Framework for Organizing Within-Iteration UX Work in Agile Development. Will appear in Integrating User Centred Design in Agile Development in Cockton, G., Larusdottir, M.K., Gregory, P. and Cajander, Å, (Eds)
23. Kuusinen, K. (2015). Integrating UX work in agile enterprise software development. Doctoral thesis, Publication 1339, Tampere University of Technology
24. Kuusinen, K. and Mikkonen, T. On Designing UX for Mobile Enterprise Apps. Proc. Software Engineering and Advanced Applications (SEAA) 2014, pp. 221-228.
25. Kuusinen, K., Mikkonen, T., Pakarinen, S. Agile user experience development in a large software organization: good expertise but limited impact. Proc. Human-Centered Software Engineering (HCSE'12). Springer Berlin Heidelberg, 2012. 94-111.
26. Kuusinen, K., Petrie, H., Fagerholm, F., & Mikkonen, T. (2016, May). Flow, Intrinsic Motivation, and Developer Experience in Software Engineering. In International Conference on Agile Software Development (pp. 104-117). Springer International Publishing.
27. Kuusinen, K., Väättäjä, H., Mikkonen, T. and Väänänen, K. Towards understanding how agile teams predict user experience. Will appear in Integrating User Centred Design in Agile Development in Cockton, G., Larusdottir, M.K., Gregory, P. and Cajander, Å, (Eds)
28. Lallemand, C., Gronier, G., & Koenig, V. (2015). User experience: A concept without consensus? Exploring practitioners' perspectives through an international survey. *Computers in Human Behavior*, 43, 35-48.

29. Law, E., Roto, V., Hassenzahl, M., Vermeeren, A. and Kort, J. Understanding, scoping and defining user experience: a survey approach. In Proc. CHI'09, ACM (2009), 719-728.
30. McAuley, E., Duncan, T., & Tammen, V. V. (1989). Psychometric properties of the Intrinsic Motivation Inventory in a competitive sport setting: A confirmatory factor analysis. *Research Quarterly for Exercise and Sport*, 60, 48-58.
31. Myers, B., Hudson, S. E., & Pausch, R. (2000). Past, present, and future of user interface software tools. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 7(1), 3-28.
32. Olsen Jr, D. R., & Klemmer, S. R. (2005, April). The future of user interface design tools. In CHI'05 Extended Abstracts on Human Factors in Computing Systems (pp. 2134-2135). ACM.
33. Reichheld, F. F. (2003). The one number you need to grow. *Harvard business review*, 81(12), 46-55.
34. Ryan, R. M. (1982). Control and information in the intrapersonal sphere: An extension of cognitive evaluation theory. *Journal of Personality and Social Psychology*, 43, 450-461.
35. Sharp, H., Baddoo, N., Beecham, S., Hall, T., & Robinson, H. (2009). Models of motivation in software engineering. *Information and Software Technology*, 51(1), 219-233.
36. Shaw, T. (2004). The Emotions of Systems Developers: An Empirical Study of Affective Events Theory. *Proc. Computer Personnel Research: Careers, Culture, and Ethics in a Networked Environment, SIGMIS CPR '04*, pp. 124-126. ACM
37. Sundberg, H.-R. (2015). The importance of user experience related factors in new product development – Comparing the views of designers and users of industrial products. 23rd Nordic Academy of Management Conference, 12-14 August 2015, Copenhagen, Denmark
38. World Health Organization, Informed consent form template for qualitative studies. Available at: http://www.who.int/rpc/research_ethics/informed_consent/enTools