Comparison of Questionnaire Based and User Model Based Usability Evaluation Methods

Meng Li [1,2]✉, Armagan Albayrak [1], Yu Zhang[2], Daan van Eijk[1] and Zengyao Yang[2]

[1] Delft University of Technology, Landbergstraat 15, 2628CE Delft, Netherlands
[2] Xi'an Jiaotong University, Xianning Road 28, 710029 Xi'an, China
m.li-4@tudelft.nl

Abstract. The usability now serves as a fundamental quality of a computational device, e.g. smartphone. Moreover, the smartphone has firmly embedded into our daily life as an indispensable part, so the context and style that user may interact with them are largely different from a decade ago. Nowadays, testing usability with end user has become a common sense. Thus, how valid a usability evaluation method could assess the 'extent to which a product can be used by specified users' (ISO 9241-11) to facilitate software design becomes an interesting question to explore.

In this research, three usability evaluation methods are compared. Among these methods, IsoMetrics is a standard questionnaire aiming at offer usability data for summative and formative evaluation; SUMI aims to assess quality of software product from end users perspective; User Model Checklist is a method based on user's cognition-motor chain in specific tasks. The coverage and amount of usability issues, user's effort of evaluation and software developer's feedback on evaluation result are compared under a simulated usability test on SMS function with a smartphone. The result indicate that User Model Checklist could cover 90.4% of the usability issues found by IsoMetrics and SUMI, while 26.3% usability issues found by User Model Checklist could not be covered by IsoMetrics and SUMI. Users put highest effort on accomplish IsoMetrics and lowest effort on User Model Checklist. Moreover, the feedbacks from the developers show that the User Model Checklist requires lower usability knowledge, offers clearer improvement points and supports detailed design better.

Keywords: Usability Evaluation Comparison, IsoMetrics, SUMI, User Model Checklist.

1 A Framework to Compare Usability Evaluation

1.1 The Multiple Definitions on Usability

ISO9241 part 10 is a worldwide applied usability standard and describes usability as a multi-factor conception, involving easy to learning, easy to use, system effectiveness, user satisfaction, as well as these factors are associated with the real environment for evaluation of specific goals [1].

Thence, usability is not a rigid concept, but an entirety composing of multiple factors, such as environment, user and tasks, and their mutual interaction. It focuses on evaluating whether software offers sufficient operation condition and operation guidance in specified context of use. It serves as an indicator for the quality of use of the software. Different types of software, for instance, mobile devices, website, *operation system* (OS) and applications proposed diversified definitions on usability [2] - [7].

Obviously, it is difficult to tell which usability evaluation method works more effectively if they have different criteria [8]. Before the various *Usability Evaluation* (UE) methods can be compared, firstly the factors influencing the effectiveness of these methods should be identified.

1.2 The Effectiveness of Usability Evaluation

In a study in 2003, Hartson et.al proposed following criteria for assessing the effectiveness of usability evaluation and referred to the methods that were used to evaluating military weapons systems:

- Fundamental criterion: detecting authentic usability errors;
- Practical criterion: Judging the authenticity of usability errors via combining standardized usability error checklists, expert's review and comment, and end user's review and comment altogether;
- Performance metrics: *comprehensiveness*, *validity*, *effect*, *reliability*, *utility* and *economy* [9];

These criteria could assess the effectiveness of usability evaluation, however the main problem is that these criteria are evaluator-centered not user-centered. Usability evaluation should adopt user's perspective to judge the quality of using authentically. The main goal of usability evaluation is gaining the knowledge about user's difficulties on interacting with a system, thus user-centered usability criteria is crucial. Furthermore, usually novice, average and skilled user have different criteria on usability, and they are different from international standard and usability experts. Therefore, this study firstly proposes a comparison framework on UE, which combines the perspective of end user, evaluator and software developer.

According to the authors of this study, the effectiveness of UE includes *authenticity*, *comprehensiveness*, and *utility* [10]. Authenticity stands for that UE authentically reflects the usability problems of users, including consistency with user's evaluation criteria, adopting real task and scenarios in testing. Comprehensiveness represents that UE can find product and system's usability problems as much as possible, and can cover every factors of usability as well. Utility means that UE feedbacks can result in modification suggestions or new designs. This paper focuses on following aspects regarding the comparison of UE methods:

- Firstly, for authenticity, analyzing the difference between UE's criteria and user's acceptance;
- Secondly, for comprehensiveness, comparing the number and distribution of usability errors pinpointed by different users with different UE methods;

- Thirdly, for utility, comparing UE results on satisfying the demands of the software developers.

Since utility concerns about applying UE feedbacks on software developing, this paper defines it in accordance with the international standard, and then validates its framework via a survey with software developers on their utility requirements of UE.

1.3 Defining Utility of Usability Evaluation according to Software Development Requirements

One purpose of comparing UE methods is to find out a method suits for different software development phase in existing project environment. This paper classifies the utility requirements on UE into three categories referring to ISO/TR 16982 [11]:

Evaluation environment: it means the need for UE at each stage of the development process in software engineering, and the constraints of UE from a specific project, for instance, project schedule, budget and product confidentiality.

Evaluation conditions: it refers the characteristics of users, tasks and products involving in UE.

Evaluation constraints: it indicates the UE prerequisites on personnel and devices, and expectation on UE results from developers and designers.

Since the utility requirements are critical to UE comparison, they needed to be validated with real demands from industry. Thus, this study surveyed 50 developers, who have experience on usability evaluation, covering different stages in software developing. They are asked to rate their requirements on UE using a 5-point Likert scale. Then, the utility requirements were modified according to the survey result, as shown in **Table 1.**

Table 1. Revised framework of demands on UE from software development projects.

Main Factor (Rate)	Sub-factor (Rate)	Description
Evaluation conditions (3.99)	Evaluation threshold (4.07)	The prerequisites need to be fulfilled, when conducting UE.
	Evaluation effect (4.05)	The expectation from design and coding on UE results
	Usability Experience (3.86)	Development project is familiar with UE technology.
Evaluation environment (3.35)	Software design (3.71)	When designing new products or functions, designers understand user's real ideas through UE.
	Software Improvement (3.30)	Needs from software UI and functional improvement on UE
	User Conditions (3.05)	Project restrictions on inviting (outside) user
Evaluation constraints (2.90)	UE Proficiency (3.06)	Proficiency in conducting UE
	Project Constraint (2.73)	The constraints from software development schedule, budget, and process on UE

Evaluation threshold, *Evaluation effect*, *Usability Experience*, *Software design* and *Software improvement* are the main utility requirements of UE from software developers. Thus, this paper suggests eight principles for the utility of UE methods:

- Low evaluation threshold: it suits users with different experience; be compatible with developing regulations; be able to compare across products;
- High evaluation effect: it can specifically identify user's usability difficulties and design defects; be helpful to improve functions and UIs;
- Low usability experience demand: it is easy to estimate UE budget; UE results are objective, specific and understandable for project members.
- Suitable for software design: it can assist logic design, new function evaluation and user-defined function design;
- Suitable for software improvement: it can define user's needs in-depth; suggest improvement possibilities; help detecting software flaws;
- Suitable for user conditions: it requires less end user participation; suites inner user testing;
- Low UE proficiency demand: it requires less experience and personal on UE team; needs easily available devices;
- Suitable for project constraints: it indicates more usability improvement chances with limited resources.

2 A Comparison Experiment on UE Methods

2.1 Objectives

According to the comparison principles proposed in section 1, this part focuses on design an experiment to compare the authenticity, comprehensiveness and utility of three UE methods. Therefore, this experiment needs to accomplish these comparisons shown as follows:

1) For authenticity, comparing assessment of novice, experienced and skilled users on UE methods;

2) For comprehensiveness, comparing the number and distribution of usability errors identified by these methods;

3) For utility, comparing assessment from software developers on a sample UE feedback;

2.2 Selection of UE methods

IsoMetrics inventory, *Software Usability Measurement Inventory questionnaire* (SUMI) and *User Model Checklist* (UMC) were chosen as UE methods to compare, since they consist validated usability checkpoints and include at least one user-centred approach.

IsoMetrics inventory. It is a standard questionnaire, aiming at offering information for iteration of software development. It developed by Osnabrück University and then validated in 1996, summarizing as:

1) The score used to measure the usability of the development process;

2) Specific information about faults and user-perceived attributes of faults;

3) The average weight of every perceived attribute of users in a particular type of system fault [12].

SUMI. It is a method measuring the software product quality from end user's perspective. It is a rigorously tested and proved questionnaire, thus acknowledge by ISO 9241 as a method for user satisfaction evaluating [13]. SUMI contains five usability factors:

Efficiency. It measures user perceived efficiency and mental load caused by Human-Computer Interaction.

Affect. It reflects user's general affectional response to software, such as like or dislike. It tests the feeling, the motivation and their specific experience from users, when they use a product.

Helpfulness. it means that user could sense the effect of information offered by the system.

Control. It indicates that to what degree user feels the software is under his/her control not reversely.

Learnability. It means that how confident user feels to start using the software and learning new functions.

User Model Checklist (UMC). Different elements in User Interface (UI), like icons, menus and controls, are the foundation of human-computer interaction in smartphone, because they are the carriers of affordance [14]. Thus, this method adopted UI elements as a basis to exam the usability of UI design. According to user's cognitive and motivational abilities for using Smartphone, this method established a framework of usability, including four types of user's needs, namely cognitive needs, motivation needs, needs for correction and needs for learning. The criteria of UMC are to what extent the design of UI elements meets with these needs.

The UMC method decomposes the UI elements into usability checkpoints in accordance with user's mental-task model [18]. Each UI element contains eight checkpoints, as the SMS icon shown in **Fig. 1**. The UMC consists of all checkpoints of UI elements in the subjective tasks.

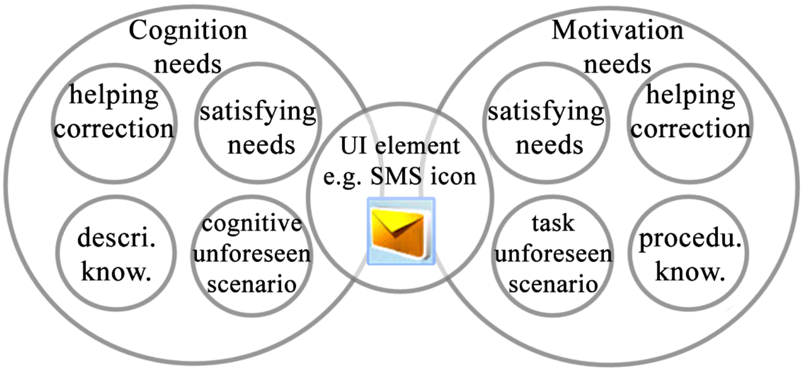

Fig. 1. Usability factors on a UI element based on User Model

Though IsoMetrics, SUMI and UMC have different usability factors, they share several common interests:

- User centred approach;
- Offering specific information about usability errors;
- Measuring usability for software developing;

These common interests are consistent with the effectiveness factors of UE methods.

2.3 Experiment Setting and Design

The Setting. The experiment simulated a usability testing of SMS tasks on a Smartphone, where the three selected UE methods were applied in a paired experiment design [15]. 30 participants joined the experiment and assessed their acceptance on each method after the test. The participants conducted the usability test in a quiet room, for this environment offered better observation for the evaluators and silence for the participants (see **Fig. 2.**).

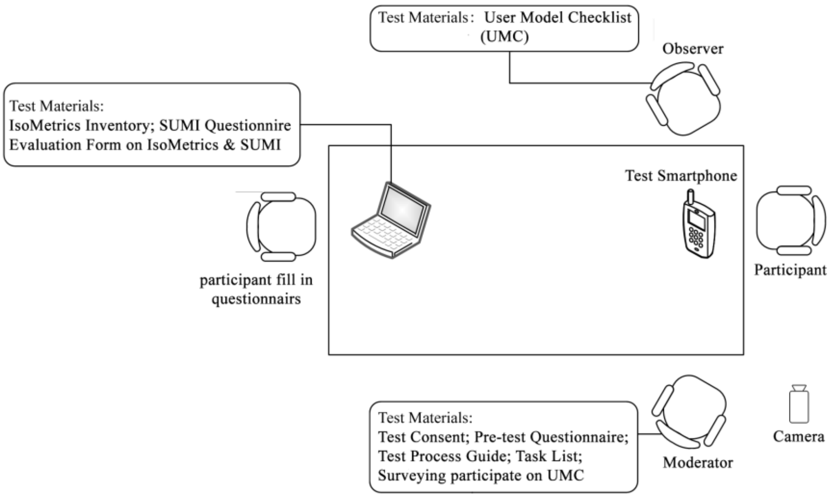
Test Materials： User Model Checklist (UMC)
Observer
Test Materials:
IsoMetrics Inventory; SUMI Questionnire
Evaluation Form on IsoMetrics & SUMI
Test Smartphone
Participant
participant fill in questionnairs
Test Materials:
Test Consent; Pre-test Questionnaire;
Test Process Guide; Task List;
Surveying participate on UMC
Moderator
Camera

Fig. 2. Test Setting Diagram

Experiment Process. The evaluation was composed of three steps:

Pre-test: the moderator interviewed participants about their habits and experience on Smartphone usage; then introduced the experiment, including the goal, process, methods in the evaluation to sign the inform consent. After that, the participants received a training session to learn the basic operation of the Smartphone (Nokia 6120C). The moderator discussed the goal and context of the subject tasks, so that the participants could use the test phone as in real world.

Testing: the participants accomplish the tasks under specified context, while the observer marks match/mismatch of the checkpoints on UMC via observing participant's operation. After each task, there was a short break, so that the two evaluators could discuss the uncertain marks of the checkpoints with the participant.

Post-test: after finish all tasks, the participants assessed their acceptance on the UMC via a questionnaire containing 20 questions based on five criteria. Next, the participants evaluated the usability of these tasks respectively via IsoMetrics inventory and SUMI on randomized order. Then, they assessed their acceptance on these two UE methods via an assessment scale. At the end, the evaluators reviewed the whole process with each participant and collected his/her feedbacks.

2.4 Participants

According to the SRK framework [16], different knowledge of usage leads to different types of usability errors. There are two types of knowledge related to using Smartphone: descriptive and procedural knowledge. The typical descriptive knowledge of Smartphone are understanding the basic concepts about Smartphone usage, and knowing how to compare across different Smartphones; the typical procedural knowledge are the number and scope of usage rules they mastered, and automated level of their usage.

These knowledges converted into six questions to divide the participants into three groups:

- Skilled users: they has used more than 5 Smartphones or familiar with all functions and short cut keys with more than 3 years' experience, and used more than 6 functions daily;
- Experienced users: they has used 2-3 phones with 1-3 years experience;
- Novice users: they has less than a half year experience or never used smartphone, or use less than 2 functions daily;

The detailed statistics on their Smartphone experience shows in **Table 2**.

Table 2. Statistics on participant's years of Smartphone usage

Years of smartphone usage	Novice	Average	Skilled	Total
<0.5-1	1	1	0	2
>1-3	5	7	2	14
>3	4	2	8	14
In total	10	10	10	30

3 Results on UE Comparison

3.1 User Acceptance of Different UE Methods

IsoMetrics. IsoMetrics consists of seven usability factors from ISO 9421 part 110, a revision from ISO 9241 part 10[17]. IsoMetrics includes 78 checkpoints, 76% of which the users find difficult to answer. The **Table 3** summarizes the high non-answerable checkpoints in IsoMetrics, including those that more than 30% participants cannot understand. In general, at least 30% participants cannot properly assessed 24.56% of usability checkpoints in IsoMetrics. Moreover, in two factors, namely *Suitability for individualization* and *Suitable for learning*, more than 80% of their checkpoints is high non-answerable.

Table 3. High non-answerable usability checkpoints of IsoMetrics

Factor	The number of the checkpoints over 30% user cannot answer	High non-answerable checkpoints ratio	Highest non-answerable rate for one statement
Suitability for individualization	5	83.33%	70.00%
Suitable for learning	2	100%	33.33%
Error tolerance	7	46.67%	53.33%
Self-descriptiveness	1	12.50%	40.00%
Suitable for the task	1	10.00%	36.67%
Controllability	1	11.11%	30.00%
Conformity with user expectations	0	0	26.67%

SUMI. It contains 50 usability checkpoints, which explain the usability errors from user's perspective. This study collected the number and the percentage of checkpoints that the participants could not answer in each usability factor (see **Table 4**). Because the participants either cannot understand them or does not encounter related situations in testing. The user acceptance shows that 74% of the usability checkpoints in SUMI are non-answerable for users. In general, 13.51% usability checkpoints in SUMI is high non-answerable, which has better user acceptance than IsoMetrics.

Table 4. High non-answerable usability checkpoints of SUMI

Factor	The number of the checkpoints over 30% user cannot answer	Percentage of high non-answerable checkpoints	Highest non-answerable rate of single statement
Efficiency	0	0	16.67%
Affect	0	0	3.33%
Helpfulness	2	25.00%	43.33%
Control	3	37.50%	66.67%
Learnability	0	0	23.33%

User Model Checklist. The participants scores their acceptance of UMC ranging from 3.48 to 3.93 with 5-points-Likert scale, which means the majority of users agree that UMC can identify their usability errors authentically. Detailed information shows in **Table 5.**

Table 5. User's assessment on UMC

Criteria	Score
User can understand the goal of evaluation.	3.93
User has thought flow during operation.	3.74
The evaluators do not disturb user.	3.63
User identified all his/her usability difficulties in UMC test.	3.56
User can understand the questions in UMC.	3.48

3.2

3.2 Usability Problems in Different User Groups

The Usability Errors Distribution across UE Methods. In this study, the checkpoints of the three UE methods are corresponded one by one. The UMC covers 90.4% of all usability errors found by IsoMetrics and SUMI, but 26.3% of errors identified by UMC is not covered by the other two methods. shows the overlapped and unique usability errors identified via these three methods.

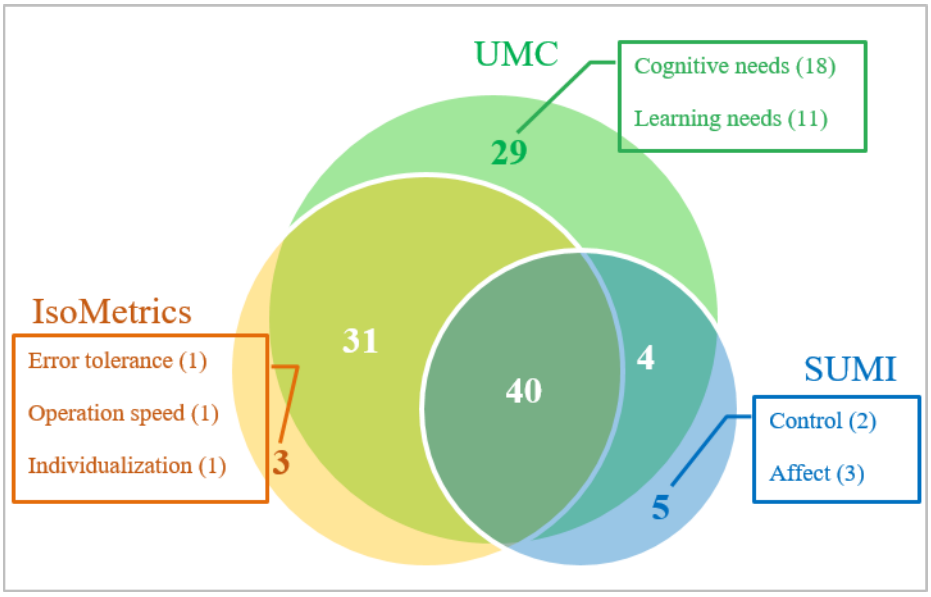

Fig. 3. Venn diagram on usability errors covered by three UE methods

Looking at the 40-shared checkpoints, the main usability factors are Suitable for the task, Suitable for learning, Controllability and User expectations, which together cover 82.5% shared checkpoints. In additional, UMC and IsoMetrics also share 31 checkpoints on, e.g. Error tolerance, Self-descriptiveness and Controllability. UMC and SUMI share 4 checkpoints on *Helpfulness*, *Learnability*, *Efficiency* and *Affect*. Detailed data refers to **Table 6**.

Table 6. The shared usability checkpoint across methods

All UE methods	IsoMetrics & UMC	SUMI & UMC
Suitable for the task(13)	Error tolerance (11)	Helpfulness (1)
Suitable for learning (8)	Self-descriptiveness (7)	Learnability (1)
Controllability (6)	Controllability (5)	Efficiency (1)
User expectations (6)	Individualization (4)	Affect (1)
Self-descriptiveness (4)	User expectations (2)	
Error tolerance (3)	Suitable for the task(2)	

How the 40-shared usability checkpoints connect across UE methods? The **Table 7** demonstrates this correspondence. The *Suitable for the task, Controllability* and *User expectations* in IsoMetrics are related to *Motivational needs* in UMC, and are partly connected to *Efficiency, Control* and *Affect*. The *Self-descriptiveness* and *Error tolerance* are related to *Cognitive needs* in UMC, and *Helpfulness* and *Control* in SUMI alike. IsoMetrics and SUMI share the same checkpoints on *Learnability*, which overlap evenly *Motivational* and *Cognitive needs* in UMC.

Table 7. Usability factors corresponding relationship across three UE methods

IsoMetrics	SUMI	UMC
Suitable for the task	Efficiency(8), Control(3), Helpfulness(2)	Motivational needs
Suitable for learning	Learnability(8)	Motivational needs (4), Cognitive needs(4)
Controllability	Learnability(8), Control(5)	Motivational needs
User expectations	Affect (4), Learnability(1), Efficiency(1)	Motivational needs
Self-descriptiveness	Helpfulness (4)	Cognitive needs
Error tolerance	Control (2), Helpfulness(1)	Cognitive needs

The unique usability checkpoints from IsoMetrics are on *Error tolerance, Operation speed* and *Individualization*, and SUMI emphasizes the *Control* and *Affect* of users during operating. Whereas, UMC can accurately pinpoints affordance deficiency of specific UI element on understanding (19), knowing (4), memorizing (2), detecting (2), recognize (1) and recall (1), which are critical for the cognition and motivational learning of users.

Comparison on User Types and the number of Usability Errors. This section compares the number of usability errors identified by the three UE methods. It examines whether there are significant differences between IsoMetrics, SUMI, and UMC, on the number of usability errors identified per user and the total number of usability errors, via within-user paired T test.

Cumulative total amount of usability errors. The total amount of usability errors identified respectively by these three UE are **103** in UMC, **74** in IsoMetrics and **49** in SUMI. With the increase in the number of users, the cumulative usability error amount of each UE method grows as shown in line chart Error! Reference source not found.. For UMC and SUMI, the first 10 users identify near 90% of the usability errors. Similarly, in IsoMetrics, 8 novice users contribute 93% usability errors.

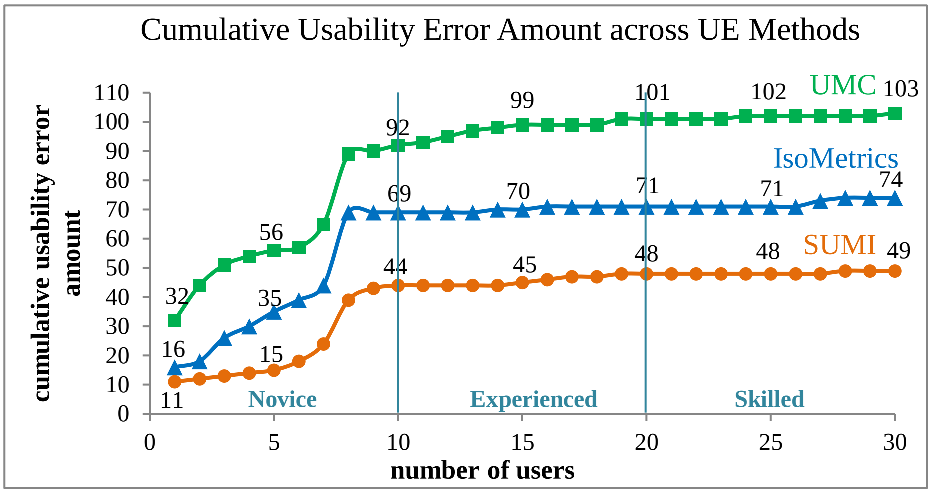

Fig. 4. Cumulative usability errors across three UE methods

The correlations across different methods are significantly, according to paired T test shown in **Table 8**. This indicates that these three UE methods are consistent in the overall effectiveness of usability errors identification. The paired sample t-test also shows that the difference on the cumulative number of identified usability errors between the IsoMetrics, SUMI, and UMC is very significant (P<0.0001).

Table 8. Correlation of the cumulative total number of usability errors

#	Variables	Pearson R	Significance
Pair1	UMC & IsoMetrics	0.97	8.13E-19
Pair2	UMC & SUMI	0.97	2.24E-19
Pair3	IsoMetrics & SUMI	0.99	8.77E-24

The total number of usability errors discovered by a single user. The number of usability issues discovered by a single user via three UE methods shows in **Table 9**. The **Table 10** demonstrates that the number of usability errors identified per user across three UE methods is significantly related, indicating that the three methods are consistent in evaluating usability for an individual user.

Table 9. The average number of usability errors discovered by one user

Methods	Min.	Max.	Mean	STD. DEV.
UMC	15	67	30.57	10.97

IsoMetrics	0	52	12.83	10.16
SUMI	1	41	11.1	8.86

Table 10. The correlation between three methods on usability error number for one user

#	Variables	Pearson R	Significance
Pair1	UMC & IsoMetrics	0.69	2.52E-05
Pair2	UMC & SUMI	0.78	2.85E-07
Pair3	IsoMetrics & SUMI	0.85	2.93E-09

There is a significant difference (P<0.01) between UMC and other two methods on mean usability error discovery, indicates that an individual user can significantly identify more errors via UMC than other two methods. Meanwhile, the usability error identified per user for IsoMetrics and SUMI are not significant different, indicates that they have similar efficiency on usability error identification.

3.3

3.3

3.3 Fulfillment of Developer's Requirements

The comparison on utility of UE methods consists of two parts, assessments from 30 experiment participants on *Evaluation threshold* and assessments from 50 software developers on *Evaluation effect*, *Usability experience demand*, *Suitable for software design* and *Suitable for software improvement*, via a sample result from the UE experiment.

*1) Evaluation threshold assessment.*After these UE, the users need to select the most difficult and the easiest evaluation method, as shown in **Table 11**. The users choose IsoMetrics as the most difficult UE method to accomplish, whereas the UMC is the easiest one to conduct. This significant difference (P<0.001) on difficulty of conducting UE across methods indicate the UMC has lower evaluation threshold.

Table 11. User's assessment on the difficulty degree of three UE methods

Degree of difficulty	User types	IsoMetrics	SUMI	UMC
Easiest	Novice	0	0	10
	Experienced	0	2	8
	Skilled	0	1	9
	Total	0	3	27
Most difficult	Novice	9	1	0
	Experienced	9	1	0
	Skilled	5	5	0
	Total	23	7	0

The attitudes from different users group on the easiest and most difficult UE methods is consist (P>0.05 in one-way ANOVA). According to the statistic on user's attitude, the evaluation threshold gradient on the three UE methods is:

IsoMetrics (-0.77) > SUMI (-0.13) > UMC (0.9)

2) Evaluation effect comparison. The requirements on the evaluation effect include objectiveness of UE and the helpfulness for the design of Help documents. Software developers have very significant different attitude among these methods (P< 0.0005 in Chi-square test), where UMC offers feedbacks that are more objective and facilitates Help document design better (see **Table 12**).

Table 12. Statistics on evaluation effect assessment

Evaluation effect	IsoMetrics	SUMI	UMC
Objectiveness of UE	4	6	40
Helpfulness for Help documents design	6	15	29

3) Usability experience demand comparison. Usability experience demand mainly assesses whether UE results are easy to understand and clearly reflect the user's actions. Software developers agree that the usability feedbacks from UMC are easier to understand and they demonstrate the usability errors more specifically (P< 0.0005 in Chi-square test), see **Table 13**. Therefore, UMC suites the UE team who has less usability experience.

Table 13. Comparison on Usability experience demand

Usability experience demand	IsoMetrics	SUMI	UMC
Evaluation results are easy to understand	4	11	35
Clearly reflect user's actions	2	7	41

3) Suitable for software design comparison. Suitable for software design includes contributing to concept design and helping with detailed design. According to software developer's assessment, UMC contributes to concept design more effective with large significance (P< 0.0001 in Chi-square test). Besides, it also significantly offers better helps for detailed design (P< 0.001 in Chi-square test), as shown in **Table 14** .

Table 14. Comparison on Suitable for software design

Suitable for software design	IsoMetrics	SUMI	UMC
Contribute to concept design	2	7	41
Help with detailed design	11	12	27

5) Suitable for software improvement comparison. Suitable for software improvement concerns about Helps with design improvement and Contribute to bug detection. Software developers find the UMC feedbacks support design iteration greater (P<

0.0001 in Chi-square test). Moreover, the UMC can also more precise detect bugs with significance (P< 0.0005 in Chi-square test), as shown in **Table 15**.

Table 15. Comparison on Suitable for software improvement

Suitable for software improvement	IsoMetrics	SUMI	UMC
Helps with design improvement	5	10	35
Contribute to bug detection	8	12	30

4 Discussion

4.1 The Factors Influencing UE Effectiveness

This section reviews the main factors lead to low effectiveness of UE, referring to authenticity, comprehensiveness and utility, like unauthentic evaluation, limited usability feedbacks and high evaluation threshold.

Authenticity. The main source of unauthentic usability feedback lies in the non-understandable questions. A typical example is from IsoMetrics that two usability factors containing 8 checkpoints offers invalid feedback. Because 30% to 70% users reported, they cannot answer 7 of them for they either not relate to testing tasks or are not understandable. It matches with the user’s assessment on these UE methods. Similar situations happened also with SUMI and UMC during the experiment.

Comprehensiveness. A main constrain for the standardized usability questionnaires, like IsoMetrics and SUMI, exists in their fixed checkpoints. It is obvious that UMC could cover more usability errors with boarder rang, because it is highly connected to the UI elements within the testing tasks. However, it needs combine checkpoints on *affection*, *user control* and *individualization* to identify usability errors as much as possible.

The generalized usability checkpoints with structural factor framework, it increases the reliability of the evaluation. Thus, they serve as benchmarks for usability, especially for across systems evaluation. Consequently, this sacrifices the accuracy on feedbacks and the fitness to specific tasks.

Though the experiment tried to include some distractors in real context, like a coming call during the text messaging, these three UE methods only slightly consider the unforeseen scenarios. They are quite common in real using context, especially under higher mobile environment.

Utility. Even though the UMC shows significant higher competences to meet the demands of software development, the case in UE practices in specific project is much more complex than this simulated UE testing. IsoMetrics needs fewer participants to find 90% of all usability errors; it can be viewed as it has higher evaluation efficiency. Thus, it fit the project constraints principle better.

Moreover, the comparison of utility is only constraint on five more objective principles, because the *user conditions*, *UE proficiency* and *project constraints* vary project after project. Thence, a rational way to compare UE methods with them is qualitative analysis in accordance with specific project setting.

4.2 Connection across Three UE Methods

The relation between authenticity, comprehensiveness and utility. Actually, the three dimensions of UE effectiveness, authenticity, comprehensiveness and utility are highly related. Because when the usability checkpoints in a UE method are not clearly defined and not task-oriented, they will easily lead to misunderstanding in users. Consequently, it will produce less authentic responses from users, and deliver an incomplete data set on usability feedbacks. Likewise, its feedbacks are also difficult to understand for the software developers, who have less usability experience.

Thereupon, clear stated and answerable usability checkpoint in-line with specific testing task and context, it can generate more authentic usability feedbacks. In addition, it can also help software developers understand the usability errors more accurate, thereafter propose concrete design improvement.

The shared usability factors across UE methods. The *Suitable for the task*, *Controllability (Control)*, *User expectations*, *Efficiency* in IsoMetrics and SUMI exam that to what extent user can smoothly apply the usage rules of Smartphone in their tasks. They are the procedural knowledge that connects to user's *Motivational needs*. The *Affect* checkpoints are about the affectional reaction aroused in using process, which is influenced by motivational needs of users.

The checkpoint in *Self-descriptiveness*, *Error tolerance, Helpfulness*, *Control* of IsoMetrics and SUMI are the UI elements convey the information about operational status of Smartphone, such as OS breakdown, function introduction or input status. Users need to perceive this information and understand them correctly. They related to *Cognitive needs* in using Smartphone.

IsoMetrics and SUMI share the same checkpoints on *Learnability*, which overlap evenly *Motivational* and *Cognitive needs* in UMC. Because from a novice user to an experienced one, the user need to learn and remember the use rules, meanwhile detect and recognize the system status via UI elements.

4.3 Redefine User's Experience Level.

The experiment found that the user's experience in using Smartphones has two types: task experience and operation experience. The task experience includes general knowledge on Smartphone usage, like knowing the basic concepts related to operating a mobile phone, understanding the operating rules, remembering the shortcut actions, and being familiar with the common system functions in Smartphones. Operational experience refers operation skills on specific UI elements and Smartphone under specific context, including familiarity with specific Smartphone OS, ability to adapt different Smartphones and automation of operations. Due to the prevalence of Smartphone [19][20] in living and working context, nowadays, the task experience of smartphone is more and more become a part of user's daily common sense.

Accordingly, this study re-categorizes users into four types: experienced-novice, experienced, skilled-experienced and skilled (see **Table 16**). The re-categorizing shows more than a half of the users are experienced-novice, and then become skilled-experienced fast. The further work needs to be done on designing a matrix with task-operation experience and procedural-descriptive knowledge as axes to define user's experience level more accurate.

Table 16. Re-categorizing of users

Task	Operation	Original User Type	Percentage
Experienced	Novice	Novice 9, Experienced 7	53.3%
Experienced	Experienced	Novice 1, Experienced 1	6.7%
Skilled	Experienced	Experienced 3, Skilled 5	30.0%
Skilled	Skilled	Skilled 6	20%

5 Conclusion

This paper proposed effectiveness as a general measurement for UE methods comparison, which exams the UE methods via *Authenticity*, *Comprehensiveness* and *Utility*. Specifically, this study proposed eight principles on UE utility based on ISO/TR 16982 and the utility demands from software developers, which are:

- Low evaluation threshold;
- High evaluation effect;
- Low usability experience demand;
- Suitable for software design;
- Suitable for software improvement;
- Suitable for user conditions;
- Low UE proficiency demand;
- Suitable for project constraints.

In general, User Model Checklist holds better user acceptance, offers richer and more accurate usability feedbacks, and fits smoother in software developing. Its flexibility and high-task oriented characters enable software developers adopt it in different developing stages. The revised UMC should contain both UI element checkpoints and those on *affection*, *user control* and *individualization*.

IsoMetrics and SUMI are validated reliable UE tools based on well-acknowledged usability international standard, which serves as usability benchmarks for various systems. They both suite formative and determined usability evaluation, aiming at issuing general and qualitative usability level rather than pointing out specific design improvement. IsoMetrics benefits from its lager checkpoint pool and smaller testing sample, while SUMI has higher user acceptance and medium evaluation threshold.

This study also found that the user's experience contains two componen*ts: task experience* for general using, and *operational experience* related to specific device and context. Thus, this paper suggests developing an experience matrix to identify user's experience level more precise, which applies task-operation experience and procedural-descriptive knowledge as two dimensions.

References

1. Macleod, M., Bowden, R., Bevan, N., & Curson, I.: The MUSiC performance measurement method. Behaviour & Information Technology, 16(4-5), 279-293(1997).
2. Nokia Corporation.: Series 60 Developer Platform 2.0: Usability Guidelines For Enterprise Applications. http:// www.forum.nokia.com/usability (2004).
3. Weiss, S.: Handheld usability. John Wiley & Sons, Chichester (2003).
4. Nielsen, J.: Usability 101: Introduction to Usability. https://www.nngroup.com/articles/usability-101-introduction-to-usability (2012).
5. Nielsen, J.: Usability engineering. Elsevier, London (1994).
6. Lathan, C. E., Newman, D. J., Sebrechts, M. M., & Doarn, C. R.: Evaluating a Web-Based Interface for Internet Telemedicine. NASA, Washington (1997).
7. Apple Inc.: Human Interface Guideline. https://developer.apple.com/ios/human-interface-guidelines/overview/themes/ (2018)
8. Hornbæk, K.:. Current practice in measuring usability: Challenges to usability studies and research. International journal of human-computer studies, 64(2), 79-102(2006).
9. Hartson, H. R., Andre, T. S., & Williges, R. C.: Criteria for evaluating usability evaluation methods. International Journal of Human-Computer Interaction, 15(1), 145-181 (2003).
10. Li, L.S.: Design Investigation. China Architecture & Building Press, Beijing (2007).
11. International Organization for Standardization: ISO/TR 16982:2002(E) Ergonomics of human-system interaction—Usability method supporting human-centred design. ISO, Switzerland (2002).
12. Gediga, G., Hamborg, K. C., & Düntsch, I.: The IsoMetrics usability inventory: an operationalization of ISO 9241-10 supporting summative and formative evaluation of software systems. Behaviour & Information Technology, 18(3), 151-164 (1999).
13. Kirakowski, J., & Corbett, M.: SUMI: The software usability measurement inventory. British journal of educational technology, 24(3), 210-212 (1993).
14. Follmer, S., Leithinger, D., Olwal, A., Hogge, A., & Ishii, H.: inFORM: dynamic physical affordances and constraints through shape and object actuation. In Uist, Vol. 13, (2013).
15. Solso, R. L., & Johnson, H. H.: An introduction to experimental design in psychology: A case approach, Third Edition. Harper & Row, Publishers, Inc., New York (1989).
16. Rasmussen, J.: Skills, rules, and knowledge; signals, signs, and symbols, and other distinctions in human performance models. IEEE transactions on systems, man, and cybernetics, (3), 257-266 (1983).
17. International Organization for Standardization: ISO 9241-110:2006 Ergonomics of human-system interaction -- Part 110: Dialogue principles. ISO, Switzerland (2006)
18. Li, L.S.: Human Computer Interface Design. Science Press, Beijing (2004).
19. Berenguer, A., Goncalves, J., Hosio, S., Ferreira, D., Anagnostopoulos, T., & Kostakos, V.: Are Smartphones Ubiquitous?: An in-depth survey of smartphone adoption by seniors. IEEE Consumer Electronics Magazine, 6(1), 104-110 (2017).
20. Lee, H., Ahn, H., Nguyen, T. G., Choi, S. W., & Kim, D. J.: Comparing the self-report and measured smartphone usage of college students: a pilot study. Psychiatry investigation, 14(2), 198-204 (2017).