The Journal of Systems and Software 83 (2010) 2045–2058

Contents lists available at ScienceDirect

The Journal of Systems and Software

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

Measuring effectiveness of HCI integration in software development processes

Anirudha Joshi [a,*], N.L. Sarda [a,1], Sanjay Tripathi [b,2]

[a] IIT Bombay, Mumbai 400076, India
[b] Tech Mahindra Ltd., Pune 411004, India

ARTICLE INFO

Article history:
Received 12 June 2009
Received in revised form 1 March 2010
Accepted 3 March 2010
Available online 15 June 2010

Keywords:
HCI-SE integration
Usability goals
Metrics

ABSTRACT

Integrating human–computer interaction (HCI) activities in software engineering (SE) processes is an often-expressed desire. Two metrics to demonstrate the impact of integrating HCI activities in SE processes are proposed. Usability Goals Achievement Metric (UGAM) is a product metric that measures the extent to which the design of a product achieves its user-experience goals. Index of Integration (IoI) is a process metric that measures the extent of integration of the HCI activities in the SE process. Both the metrics have an organizational perspective and can be applied to a wide range of products and projects. An attempt has been made to keep the metrics easy to use in the industrial context. While the two metrics were proposed mainly to establish a correlation between the two and thereby demonstrate the effectiveness of integration of HCI in SE processes, several other applications seem likely. The two metrics were evaluated in three independent studies: a classroom-based evaluation with two groups of students, a qualitative feedback from three industry projects, and a quantitative evaluation using 61 industry projects. The metrics were found to be useful, easy to use, and helpful in making the process more systematic. Our studies showed that the two metrics correlate well with each other and that IoI is a good predictor of UGAM. Regression analysis showed that IoI has a somewhat greater effect on UGAM in projects that use the agile process model than the waterfall process and in the projects that are executed as a contracted software development service than in the projects in product companies. UGAM also correlated well with the traditional usability evaluations.

1. Introduction

Large contracted software development companies with tens of thousands of employees are often involved in a wide variety of software development projects. Managers of user experience (UX) groups in such companies need to track progress of each project and ensure the quality of deliverables. They are often required to juggle across projects a limited resource – the time of their best UX professionals. While there are numerous usability metrics to evaluate specific projects, only a few of them allow organizations to track progress across projects easily. A product metric called Usability Goal Achievement Metric (UGAM) that measures the extent to which the design of the product achieves its user-experience goals is proposed. The objective is to provide a summary measure of the user experience of a product that is independent of the domain, context of use, platform, or the software development process, so that the manager is able to make judgments across projects.

Another challenge faced by UX groups is integrating HCI in established SE processes. The field of HCI has a large amount of literature on user-centred design methods, techniques, and processes (Beyer and Holtzblatt, 1998; Cooper and Riemann, 2003; Mayhew, 1998; Nielsen, 1993; Shneiderman, 2004, etc). These proposals are excellent demonstrations of how user-centred design can result in improved user-experience design. Unfortunately, major gaps between HCI and SE continue to exist in academics, literature, and industrial practice. The IFIP Working Group 2.7/13.4 on User Interface Engineering remarks that 'there are major gaps of communication between the fields of HCI and SE: the architectures, processes, methods and vocabulary being used in each community are often foreign to the other community' (IFIP, 2004). For example, while SE literature admits that communication with the customer is an unsolved problem, even recent editions of standard text books on software engineering such as (Kroll and Kruchten, 2003) and (Pressman, 2005) do not suggest employing established user study techniques like (Bevan, 2008) or (Beyer and Holtzblatt, 1998) during communication. Example projects shown in (Kroll and Kruchten, 2003; Pressman, 2005) seem to take HCI design lightly, prematurely and without following any process. A detailed critique of SE literature from an HCI perspective was presented in (Joshi, 2006). There have been several proposals to integrate HCI in SE process models (for example, Costabile, 2001; Göransson et al.,

* Corresponding author. Tel.: +91 9820345569; fax: +91 2225767803.
E-mail addresses: anirudha@iitb.ac.in (A. Joshi), nls@iitb.ac.in (N.L. Sarda), tripsanjay@gmail.com (S. Tripathi).
[1] Tel.: +91 9820120045.
[2] Tel.: +91 9922963298.

0164-1212/$ – see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.jss.2010.03.078

2003; Joshi and Sarda, 2007; Pyla et al., 2003) but none has become popular in the industry. One reason for this could be the concerns about return on investments. Though there is plenty of evidence of the a return on investment of usability activities in general (Bias and Mayhew, 2005), there is no direct evidence that shows that better integration of HCI activities in SE processes leads to better products at smaller costs.

A process metric called Index of Integration (IoI) that measures the extent to which HCI activities are integrated with SE processes is the second proposal. Contracted software companies often promise a certain quality of user experience and a certain level of process compliance to clients. UX managers managing several projects need summary measures to ensure process compliance and quality of deliverables. A correlation between a product metric and a process metric can also demonstrate the return on investment on integration of HCI with SE – if a higher process metric consistently leads to a higher product metric, it makes sense to invest in better integration of HCI with SE.

The next section gives an overview of related work in HCI metrics. Sections 3 and 4 describe the proposals for two metrics – Usability Goals Achievement Metric (UGAM) and Index of Integration (IoI), respectively. These two metrics were first introduced in (Joshi and Tripathi, 2008). Sections 5 and 6 describe three studies that evaluated the two metrics and their findings. Section 7 draws conclusions from the study.

2. Metrics in HCI

Metrics are thoroughly discussed in software engineering literature. Fenton and Pfleeger (Fenton and Pfleeger, 2002a) describe measurement as "the process by which numbers or symbols are assigned to attributes of entities in the real world in such a way as to describe them according to clearly defined rules." Pressman (Pressman, 2005) highlights the subtle difference between measurement and metrics – measurement occurs as the result of collection of one or more data points, while a metric tries to relate the measures in some way. IEEE Standard Glossary (IEEE, 1993) defines a metric as "a quantitative measure of the degree to which a system, component, or process possesses a given attribute".

Though the word 'metric' is seldom used in practice of usability, several measures are often used to evaluate the user-experience quality of products. Seconds taken to withdraw money from an ATM, the number of keystrokes required to enter a word in a complex script, the number of errors made while carrying out a banking transaction, or the percentage of users that abandon the shopping cart on checkout are examples of quantitative measures of the user experience afforded by a product. However, none of these is a summary measure that can be used for an apple-to-apple comparison across projects that vary in domain, platform, and context. While several research papers discuss metrics related to usability and HCI, this paper only focuses on those that give a summary measure.

There have been several attempts to devise a summary measure to identify an overall measure of perceived usability. Many of these are based on users' post-session rating in response to a questionnaire. System Usability Scale (SUS) (Brooke, 1996) uses 10 statements to which users agree or disagree on a 5-point scale. The metric is calculated by aggregating scores for the 10 statements and scaling the result to 100. Similar in approach, though a bit different in details are QUIS (Chin et al., 1988) and SUMI (Kirakowski and Corbett, 1993).

Lin et al. (1997) propose the Purdue Usability Testing Questionnaire based on eight HCI considerations to derive a single weighted average score for usability. While the approach does lead to a single usability score, the selected eight considerations (compatibility, consistency, flexibility, learnability, minimal action, minimal memory load, perceptual limitation, and user guidance) seem to be a mix of usability goals, and heuristics that achieve those goals. Secondly, the weight for parameters is to be assigned by the evaluator during the evaluation (without consulting other stakeholders). Thirdly, the eight considerations listed and the questions listed under each of them seem to be limiting and do not leave room for context-specific goals (for example, "user should be able to do it right the first time").

McGee (2004) derives a single usability scale across users by including additional reference tasks. However, McGee does not suggest how to derive a single measure for usability from measures for the different tasks. Further, this work is completely dependent on the technique of usability evaluation. This is not always practical in a global, contracted software company striving to move up the HCI maturity ladder. The other limitation of this method is that it relies only on the perception of users and ignores perspectives of other stakeholders, particularly the goals of business stakeholders.

There have been other attempts to capture user performance into a single metric. Lewis (1991) used a rank-based system of assessing competing products based on user's objective performance measure and subjective assessment. The metric is useful for a relative comparison between like products with similar tasks but it does not result in a measure that can be used across unlike products.

Sauro and Kindlund (2005) proposed a 'single, standardised, and summated' usability metric for each task by averaging four standardised values for task time, errors, completion, and satisfaction. However, each of these is assigned an equal weight. Tasks, domains, users, contexts, and platforms vary a lot and it does not make sense to give equal weight in all the contexts. Moreover, the metric ignores some aspects such as learnability, and ease of use, which might be important in some contexts.

Literature also has issues with metrics in usability. Gulliksen et al. (2008) are critical of measurement in the area of usability and user experience. They are concerned that given the difficulties in measurement, people may measure only those aspects that are easy to measure, and these measures may turn out to be "eternal truths". They find that organisations face problems in interpreting metrics, drawing conclusions, and turning them into action items. Moreover, once something is numerically assessed, everyone may focus on the measurement alone and ignore the complexity behind the work situation. However, Gulliksen et al. are not totally opposed to measurement per se, but are merely pointing to limitations that a measurement induces.

Hornbæk and Law (2007) argue against the validity of a single, summative measure because it either relies solely on users' perceptions or is arbitrary in including or excluding constituent usability parameters from its summation procedure. They found medium to low correlations among efficiency, effectiveness, and satisfaction across many projects and they argue that attempts to reduce usability to one measure (for example, Sauro and Kindlund, 2005; McGee, 2004) therefore lose important information.

While some information will always be lost in a summative measure, such a measure could still be useful. Goals are an important way for the stakeholders to express the desired user experience in the design. For example, in an application for a call-centre agent, efficiency and effectiveness may be important, and if the application allows the user to complete successfully a large number of calls with minimal fatigue, it ought to be recognised as successful. On the other hand, for a computer game, the player's satisfaction may be of utmost importance, and if the game manages to make gamers happy (even) at the cost of efficiency or effectiveness, so be it – the game ought to be considered successful. If scores for efficiency, effectiveness, and satisfaction of these two products are merely aggregated in a metric, these scores may not correlate as Hornbæk and Law predict, and both the call-centre application and

the game might show up as mediocre applications. On the other hand, if each goal were assigned a weight, the weighted average of the scores against those goals would indicate that the designs did well in achieving the set target. The proposal for UGAM attempts to do precisely that.

Measuring the wider notion of user experience (as opposed to usability) is a relatively new and more difficult concept in HCI and is attracting the attention of the academia as well as the industry. Usability parameters are typically related to the processing of information or completion of tasks. However, affective reactions and emotional consequences play an important role in the overall user experience (Mahlke, 2005). In some product contexts, visceral, behavioural, and reflective elements (Norman, 2004), aesthetics (Tractinsky et al., 2000), enjoyment (Jordan, 2000), and creativity (Swallow et al., 2005) may need to be considered.

However, it is difficult to characterise user experience in terms of metrics, goals, or even a definition. McCarthy and Wright (2004) look at people's experience in terms of 'felt life'. People actively construct their experiences and that experience of each person is unique, rich, and difficult to communicate, let alone predict, design, or measure. In a recent (2009) survey of 275 respondents from the user-experience profession, Law et al. (2009) concluded that the concept of user experience is dynamic, context-dependent, and subjective. While it is difficult to define user experience or measure it universally, within the specific context of a project, it is possible for a design team to agree about the issues that would affect users' experience the most. We hope to leverage this ability to capture in UGAM the subtle nuances of user experience beyond usability.

None of the summary metrics mentioned above measure the experience of a product with reference to all the user and business goals relevant to a product. Many are too complex to compute practically on an on-going basis in the industrial practice. Most lack the flexibility required to serve the needs of a wide variety of projects or to mature with a UX group.

While there have been several attempts to define product metrics to measure the usability and user experience, there seem to be no proposal that attempts to measure the quality of HCI design process followed or the integration of HCI activities with SE processes.

3. Usability Goals Achievement Metric

Fenton and Pfleeger (2002b) emphasise the importance of goals in a metric: "a measurement program can be more successful if it is designed with the goals of the project in mind". User-experience goals are very important in driving the design of interactive products. They help speed up the design process, make the design activity more tangible and help evaluate the design. User-experience goals can be understood easily, even by non-UX-professionals, and they have a significant overlap with business goals. Stakeholders outline the user-experience goals and UX professionals fine-tune them based on their knowledge and findings from user studies. User-experience goals are (and should be) available early in a project – another plus when it comes to metric calculation in a practical situation.

We propose Usability Goals Achievement Metric (UGAM), a product metric that measures the quality of user experience. UGAM is product metric on a scale of 0–100, where 100 represents the best user experience possible and 0 represents the worst. The motivations are:

- to measure the user experience of a product in reference to its user-experience goals;
- to develop a flexible metric that can be applied across a variety of projects, irrespective of domain, context, platform, process model, and usability technique;
- to develop a flexible metric that will mature with the organization;
- to compute the metric with minimal additional costs and efforts.

UGAM consists of the following conceptual elements:

- **Goals**: High-level user-experience goals guide the design of interactive systems.
- **Goal Parameters**: Each high-level user-experience goal is broken down into a set of parameters that help a designer to achieve and measure the achievement of the higher-level goal in a direct manner. For example, parameters for learnability could be: options/data/information should be easy to find, user should take little time to learn, user should be able to learn on one's own, the product should be consistent with its earlier version, etc.
- **Weight**: Each goal parameter has a weight between 0 and 5 where 0 indicates that the goal is not relevant, 1 indicates that the activity is somewhat relevant, 2 indicates the typical importance (the hygiene factor), 3 indicates the goal parameter is more important than usual, 4 indicates that it is very important and 5 represents that it is extremely important.
- **Score**: Each goal parameter has a score between 0 and 100, where 0 represents the worst possible user-experience design on account of that parameter, 25 represents that the design is quite bad, though not the worst, 50 represents an undecided state, 75 represents that the design was good enough, though not exceptional and 100 represents the best possible user-experience design rated against the goal parameter. A parameter may be assigned a score either by directly linking it to user performance (for example, *percentage of users who could find all the critical options*), or by using an assessment by the evaluators from a qualitative usability evaluation (for example, *after a think-aloud test, how confused were the users about the conceptual model?*), or simply by reviewers' rating (for example, *after a heuristic evaluation*).
- **Guidelines**: The purpose of the guidelines is to help evaluators assign a score to parameters. Guidelines let the goal-setters express themselves better and interpret goals for the context of a project – for example, "*The goal parameter 'Consistency with earlier version' means all frequent and critical tasks from earlier version are unchanged.*"

  Further, guidelines tell the evaluators how to assign scores. For example, "The interface clearly communicates the correct conceptual model. Strongly agree = 100, weakly agree = 75, neutral = 50, weakly disagree = 25, and strongly disagree = 0".

  Guidelines could also directly link scores to specific performance measures in a usability test. For example: "The goal parameter 'User should take little time to learn' is evaluated on the basis of average time taken to learn to perform benchmark tasks without errors, as follows: less than 15 min = 100; 15 min to 1 h = 75; 1–2 h = 50; 2–8 h = 25; more than 8 h = 0."

  Another example: "The goal parameter 'Product should not induce errors' is evaluated on the basis of the number of design-induced errors reported: 0 errors = 100; 1–4 errors = 75; 5–10 errors = 50; 11–20 errors = 25; 21 or more errors = 0."

Though expressing user-experience goals is a common activity in HCI design, there is no standard way of doing it. There are many ways to describe high-level user-experience goals. For example, ISO 9126-1 describes usability in terms of understandability, learnability, operability, and attractiveness (IOS, 2001). ISO 9241 on the other hand defines usability as the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use (IOS, 1997). Shneiderman (Shneiderman, 2004) describes goals for user interface design in terms of five human factors central to evaluation: time to learn, speed of performance, rate of errors by

Table 1
An example of UGAM calculation.

Goals and goal parameters	Weight	Goal parameter score	Goal score	UGAM score
Learnability			69.44	
Findability: options/data/information should be visible/easy to find	4	75		
User should take less time to learn: (e.g. in <10 min, in <2 h practice, in <2nd attempt)	3	50		
Users should be able to learn on their own	4	50		
Product should be internally consistent	5	100		
Product should be consistent with other products, older methods/past habits of users	2	50		
Product should be consistent with the earlier version	0	0		
User should remember/retain critical, but infrequent tasks	0	0		
Speed of use			65.00	
User must be able to do the primary task/the most frequent tasks quickly, easily, at all times	0	0		
User should be able to navigate quickly and easily	4	75		
Product should not load user's memory/product should not put cognitive load on a user	2	75		
Flexibility: user should control the sequence of tasks	1	50		
User should be able to complete frequent/critical tasks in specific time/no. of steps/in less efforts	0	0		
Product should be personalised for the user automatically	0	0		
Product should be localised for specific market segments	3	50		
Users should be able to customise the product for themselves	0	0		
Ease of use			50.00	
Interface should clearly communicate the conceptual model	2	25		
Intuitiveness: User should be able to predict the next step/task	3	50		
No entry barrier: user must be able to complete critical tasks	0	0		
Product should require no unnecessary tasks	2	75		
Product should automate routine tasks/minimise user task load	0	0		
Product should be always on, always accessible	0	0		
Ease of communication			75.00	
Information architecture: Information should be well aggregated, well categorised, well presented	3	75		
Communication should be clear/user should easily understand text, visuals	4	75		
Error-free use			63.89	
Product should give good feedback/display its current status	3	75		
Product should not induce errors	3	75		
Product should tolerate user's errors/forgiving interface/should prevent errors	1	25		
Product should help user recover from errors/help users troubleshoot problems	2	50		
Subjective satisfaction			75.00	
User should feel in control of the product/behavioural appeal	3	75		
User should feel emotionally engaged with product/brand/product should be fun/reflective appeal	1	0		
User should find the product aesthetically appealing/product should have a visceral appeal	3	100		
UGAM score				66.81

users, retention over time, and subjective satisfaction. Nielsen's list of goals is similar (Nielsen, 1993). Bevan (2008) summarises other standards and several other ways of organising usability measures at the user interface level as well as at the system level.

A suggested list of goals and goal parameters is shown in Table 1. We reported how we arrived at precisely this list elsewhere (Joshi, 2009a) and summarise it below. It must be highlighted that this is not a prescribed, exclusive set. Designers and stakeholders are free to use any of the sets mentioned above or to derive additional, relevant parameters that express their goals better – in theory, these could be drastically different from the ones we suggest. Goals and parameters could be added, removed, or combined according to the context of the project, the needs of the users, the vision of the stakeholders, and to fit the terminology with which the product development team is comfortable. The initial list is meant to give users a starting point, while the flexibility is meant to allow the metric to mature with the experience of the organization using UGAM.

Goals from Shneiderman (2004) and Nielsen (1993) were adopted as a starting point for our "bird-level" user-experience goals and expressed them as learnability, speed of use, error-free use, and subjective satisfaction. 'Ease of use' and 'ease of communication' were added to the list. Under each goal, a list of goal parameters was added. This list was improved through formative evaluations by using it for 15 projects. During and after each formative evaluation, the goal parameters were added, re-worded, split, merged, or regrouped to fit the contexts. Still, goal parameters were kept general enough to apply to a wide range of products and expressive enough to suit the needs of individual products. Summative evaluations were done by setting goals for 49 industry projects to evaluate if the list helped in setting goals in industry projects. Participants found the goal parameters to be useful and systematic, and said that they would use them in their future projects (Joshi, 2009a).

Shneiderman states that 'a clever design for one community of users may be inappropriate for another community' and 'an efficient design for one class of tasks may be inefficient for another class' (Shneiderman, 2004, p. 14). Weights express the relative importance of goals and parameters in the context of a project. For example, a product meant to be used several times a day by a call-centre agent is likely to have higher weight for 'speed of use'. A one-time use product like a website for visa application for a tourist might insist on learnability and error-free use. On the other hand, a life-critical product to be used in an operation theatre is likely to rate highly error-free use and may sacrifice learnability.

The stakeholders assign the weight to set the context of the project. Goal-setters should be aware that while it may be tempting to set a high weight to each goal parameter, it might not be necessary, practical, or even possible to achieve such a design. The weights should reflect the priorities of the business, the stakeholders, and the users. The weight would also help prioritize usability evaluation activity – the goals and parameters with the highest weight must be evaluated more thoroughly, while the goals with lower weights could be perhaps evaluated by a discount method.

The process for computing UGAM for a product involves the following steps:

- **Goal Setting:** Early in the project, typically just after user studies but before design, an HCI professional and stakeholders identify goals and parameters for each goal, assign weight to each goal parameter, and decide evaluation guidelines for the parameters.
- **Scoring:** Immediately after each usability evaluation, one or more independent HCI professionals assign a score to each parameter of each goal. The usability evaluation could be either user-based (for example, a usability test) or review-based (for example, a heuristic evaluation).
- **UGAM Calculation:** UGAM is the sum of the weighted average of the scores of all goals. UGAM = $\sum(W_p \times S_p)/\sum W_p$ where W_p is the weight of the goal parameter p and S_p is the score of the goal parameter p.

The guidelines described above are used for scoring. Scores of some of the parameters can be linked directly to the findings of the usability evaluations (for example, percentage of users who did not make errors while doing benchmark tasks or percentage of users who thought the product was engaging). Other parameters may not be so easily linked numerically (for example, conceptual model confusions discovered during a think-aloud test or problems identified during a heuristic evaluation). In such a case, evaluators consider the guidelines and their own experience to arrive at a score for each parameter. If there are multiple evaluators, a simple average across evaluators is deemed to be the score for a given parameter. First, multiple evaluators assign scores independently. If there is a significant variation in their scores, the evaluators discuss the parameter and have the opportunity to converge their scores before the average is calculated.

Table 1 shows an example of UGAM calculation for an industrial project. The team first allocated weights to each goal parameter (shown in column 1). The evaluators then assigned a score for each goal parameter (shown in column 2). The weighted average for the goal parameter scores under each goal resulted in the score of that goal (shown in column 3) and the weighted average of all goal parameter scores resulted in the UGAM score (shown in column 5). A weighted average of only goal parameters under one high-level goal gives the score against that goal (shown in column 4). In this example, the team could see that by focussing on the high-weighted but low-performing goals (for example, ease of use) and goal parameters (for example, users should be able to learn on their own), they could improve their UGAM score significantly.

UGAM is a summary measure, but it also creates a profile of the user experience that was afforded and allows a drill-down to goal and goal parameter level performance. UGAM can be used to track the changes in the overall user experience delivered by a product across versions. Managers could use UGAM to track the performance of several projects at a time and to identify the black sheep among projects and plan on assigning their best people to those projects early. In case of applications with multiple user profiles, separate UGAM should be calculated for each profile since their experience with the product is likely to be different. Calculation of UGAM could be a part of every usability evaluation of the project, but it is recommended that it should certainly be a part of the final usability evaluation, beyond which no design changes are planned.

3.1. Comparing UGAM to traditional usability studies

An evaluation was done in a classroom setting with the help of two groups of students. The main goal of this experiment was to evaluate if UGAM scores are comparable with the way a traditional usability evaluator rates a product after conducting a traditional usability evaluation – in other words, to validate UGAM. The other goals were to check if UGAM can be calculated in a small amount of time and whether people could learn to calculate UGAM with minimal training.

3.1.1. The products

Three products were evaluated in three categories – three SMS input methods for Hindi, three CD-writing applications and three cricket websites – a total of nine products. The three SMS input methods for Hindi were based on currently available phones of Nokia, Sony Ericson, and a new system. The targeted audience was Hindi-speaking users who had studied between standards 7th to 10th of schooling, who had been using a mobile phone from 6 months to 2 years, and who had never learnt to input Hindi text in any system. The evaluators were asked the following key questions: Which system is the easiest for the beginner? Which encourages self-learning? Which is the best after practice? Overall, which system is the best? Are there any specific problems and suggestions for improvement of in each design?

The three CD-writing products that were evaluated were Roxio, Nero, and Windows Vista native CD-writing application. The chosen user segment consisted of middle-aged, medium tech-savvy users who wished to take regular backups of their work without any help.

The three cricket websites were cricinfo.com, cricketnext.com, and cricbuzz.com. The chosen user segment consisted of office-going, medium tech-savvy users who were cricket fans, who played cricket, who always followed matches of their country and occasionally followed matches between other countries, who had access to the Internet, and who used the internet for occasional searches, but had never visited the three sites in question.

3.1.2. The teams and evaluation methods

The experiment was carried out with the participation of two student teams working independently. One team comprised of nine students doing their masters in interaction design who were attending a course on usability evaluation (the IxD team). The other team comprised of 34 computer science and engineering students – a mix of masters and fourth year undergraduate students – who attended an introductory course on human–computer interaction (the CS team). A persona description of targeted users and a scenario were provided to the evaluators in each case.

The IxD Team

The IxD team was the control group and represented the traditional usability evaluation techniques. These students had an aptitude for interaction design and had previously attended several courses related to design, including courses on interaction design and user studies. This exercise was done during a course on usability evaluation. This team was divided into three groups. Each group evaluated one product from each category using three different usability evaluation techniques (as a part of the course).

The Hindi SMS input products were evaluated using a **performance test**. This was the most rigorous of the three tests. Each group recruited five users. Each user was given a 5-min orientation of the phone model. Then the user was shown 18 cards one by one. Each card had one Hindi word written on it. The words were sequenced in the increasing order of difficulty. The user was asked to input the words on the phone. The user was encouraged to type each word without help. If the user could figure out how to type the word, the task was considered 'successful without help'. If the user could not type the word on his own, help was provided. If the user could type the word with help, the task was considered 'successful with help'. If the user gave up or could not type the word in spite of help, it was counted as an unsuccessful task.

The three CD-writing applications were evaluated with the help of a discount usability evaluation method of **heuristic evaluation**.

Each product was reviewed by a sub-group of three evaluators. Each person in the group evaluated the product independently and listed out usability problems. The three evaluators then met for a debriefing session and combined their lists. They also came up with suggestions to improve the design.

Three cricket websites were evaluated by the technique of **think-aloud test**. Each sub-group recruited four to six users. Each sub-group ensured that it had users in a wide age range. Each user was given a set of tasks and was asked to think aloud while trying to perform the task. Tasks included finding the latest score of an ongoing cricket match, reviewing the summary of a match that got over recently, finding the next match that you can see with friends at a nearby stadium or on TV, finding the rules of the (then new) league cricket match series, and finding and comparing statistical data of two cricket players. As each user performed the tasks, their problems, confusions, frustrations, and comments were noted.

Each evaluation was spread over 6 days (4 working days and 2 days of a weekend for buffer time). Though each product was evaluated by a sub-group, the sub-groups worked together, shared their work often, agreed on common goals for products within a category, and had a common test protocol and a common test setup. The sub-groups recruited users together and presented findings to each other on the last day of each evaluation. After the presentations, all the students were asked to rate each product on a scale of 0–10 based on the findings, where 0 represented the worst user experience and 10 represented the best.

The CS Team

The CS team was the test group and only used UGAM for evaluation. The CS team students came from a computer science discipline and they were only half-way through their first course on HCI. They were taught the technique of calculating UGAM in a 1.5-h long session and did the actual UGAM evaluation in the next session, also lasting 1.5 h (though some groups took up to 2.5 h to complete).

This team computed UGAM for the same nine products described above and against the same briefs. The 34 students were divided into nine groups. Each group was asked to evaluate one product. Each group worked independently, without sharing any material such as goals, parameters, weights, or scores with other groups. Overall, the CS team had much shorter time to complete the assignment and could carry out only a brief inspection of the product. In all respects viz. background in interaction design, knowledge of usability evaluation techniques, time available for evaluation, access to users, interaction with other groups, and sharing findings about other product evaluations, the CS team was at a disadvantage compared to the IxD team.

3.1.3. Findings

Tables 2–4 summarize the ratings by the IxD team and UGAM scores by the CS team. Table 2 also lists the findings from the performance test – the task completion rates of users without and with help.

To determine the relationship between the ratings by IxD team and UGAM scores by the CS team for the nine products, two-tailed Pearson's correlation was performed. It was found that there is a positive correlation $r = 0.68$, $p = 0.04$ between ratings by IxD team ($M = 5.44$, SD = 1.55, $N = 9$) and UGAM ($M = 65.71$, SD = 11.12, $N = 9$). UGAM predicted 46% of the variance in the ratings by IxD team ($r^2 = 0.46$).

Within product categories, the ranking by IxD team ratings and by UGAM scores tallied in the categories for Hindi text input (performance test) and CD-writing software (heuristic evaluation). In the case of Hindi text input (performance test), UGAM scores also tallied with the ranking, by the results of the performance test. On the other hand, for the cricket websites (think-aloud protocol), only the bottom ranked product tallied, while the two top-rated sites were exchanged. This is perhaps because the first two product categories are goal-driven and task-oriented and the goals in Table 1 were expressive enough. Cricket websites are information spaces to be explored at leisure, goals and tasks are not so clear, and evaluations may tend to be more subjective. Perhaps the difference can also be attributed to the different techniques used by the IxD team. Even so, the top two sites (cricinfo.com and cricketnext.com) got close scores in both – the rating and UGAM. Further, the standard deviation of the rating for cricinfo.com was unusually high, pointing to disagreement within the IxD team rating.

It can be concluded that despite significant constraints (background in interaction design, knowledge of usability evaluation techniques, time available for evaluation, access to users, interaction with other groups, and sharing findings about other product evaluations), the team using UGAM could reasonably mimic ratings by traditional usability evaluators.

Table 2
Performance test findings, ratings by IxD team and UGAM scores by CS team for Hindi text input on mobile phones.

	New system	Nokia	Sony Ericson
Findings by IxD team (average number of words typed by a user, number of users = 5, total words attempted = 18)			
Without help	9.6 (53%)	8.0 (44%)	3.4 (19%)
With help	8.0 (44%)	2.4 (13%)	4.2 (23%)
Total	17.6 (98%)	10.4 (58%)	7.6 (42%)
Rank	1	2	3
Ratings by IxD team (control group, scale 0–10)			
Average rating	7.6	6.5	3.7
SD	0.8	0.8	0.8
Rank	1	2	3
Ratings by IxD team (control group, scale 0–10)			
UGAM score	69.5	67.5	48.9
Rank	1	2	3

Table 3
Heuristic evaluation ratings by the IxD team and UGAM scores by the CS team for CD-writing applications.

	Roxio	Nero	Vista
Ratings by IxD team (control group, scale 0–10)			
Average rating	6.5	4.8	3.6
SD	0.9	1.2	1.1
Rank	1	2	3
UGAM by CS team (test group, scale 0–100)			
UGAM score	80.5	75.8	60.2
Rank	1	2	3

Table 4
Ratings by the IxD team after think-aloud tests and UGAM scores by the CS team for cricket websites.

	cricinfo.com	cricketnext.com	cricbuzz.com
Ratings by IxD team (control group, scale 0–10)			
Average rating	6.6	6.1	3.6
SD	2.4	0.6	0.5
Rank	1	2	3
UGAM by CS team (test group, scale 0–100)			
UGAM score	63.1	75.6	50.5
Rank	2	1	3

4. Index of Integration

We conceive Index of Integration (IoI) as an empirical process metric, nominally on a scale of 0–100, where 100 represents the best possible integration of HCI activities in the software development activities and 0 represents the worst with respect to a prescribed process model.

Table 5
An example of IoI calculation for the integration of HCI activities in the waterfall model prescribed in Joshi and Sarda (2007).

Phases and HCI activities	Recommended weights	Assigned weights	Activity score	Phase score	IoI score
Communication				50.00	
Contextual user studies and user modelling, competitive product analysis	3–4	4	25		
Ideation with a multidisciplinary team (HCI, tech, biz)	2	3	50		
Product definition/information architecture/wireframes with a multidisciplinary team (HCI, tech, biz)	1–3	3	50		
Usability evaluation (formative) and refinement of product definition	1–3	4	75		
Modelling				75.00	
Detailed UI prototyping	4–5	5	75		
Usability evaluation (formative) and refinement of prototype	4–5	4	75		
Construction				78.57	
Development support reviews by usability team	3	4	100		
Usability evaluation (summative)	1–3	3	50		
IoI score					64.17

The IoI metric consists of the following conceptual elements:

- **Software Engineering Phases:** These are the broad phases as described in the software engineering process models as described. For example, see Pressman (2005).
- **HCI Activities:** HCI activities are prescribed for each phase of the software engineering process model. These could be organisation-specific or based on a published recommendation. In this paper, the activities prescribed in Joshi and Sarda (2007) and Joshi (2009b) have been used as examples.
- **Weight**: Each HCI activity is given a weight on the scale of 0–5 where 0 indicates that the activity is not relevant, 1 indicates the activity is somewhat relevant, 2 indicates the activity is typically important, 3 indicates the activity is more important than usual, 4 indicates that the activity is very important and 5 indicates that the activity is extremely important in the context of the project.
- **Score**: Each activity has a score associated with it. The score is given on a rating of 0–100, where 100 represents the best case situation, i.e. the activity was done in the best possible manner, with the highest fidelity, in the most appropriate phase of software development and with the best possible deliverables; 75 represents that the activity was somewhat toned down, but was still well-timed and well-executed; 50 represents that the activity was done with some shortcuts or perhaps was not timed well; 25 represents that the activity was done with many shortcomings; and 0 represents the worst case situation where the activity was not done at all.
- **Activity evaluation guidelines:** These spell out considerations that help the evaluation of each activity. Guidelines may define the techniques used to carry out activities, the skill and experience levels of the people doing the activities, the deliverables and other parameters that affect the fidelity of the activity. For example, following are the guidelines for the activity of 'contextual user studies and user modelling, competitive product analysis' in Table 5:
  1. Both organizational data gathering and user studies are done before requirements are finalized.
  2. User studies are done in the context of the users by the method of contextual inquiry.
  3. User studies are done with at least 20 users in each profile.
  4. User studies are done by people with experience in user studies in a similar domain of at least 2 projects.
  5. The findings including user problems, goals, opportunities, and constraints are analyzed, documented, and presented in an established user modelling methodology such as personas, work models, affinity diagram, etc.
  6. Competitive/similar products and earlier versions of the products are evaluated for potential usability problems, at least by using discount usability evaluation methods such as heuristic evaluation, and are benchmarked.
  7. User-experience goals are explicitly agreed upon before finalizing requirements.

  100 = All the above are true, the activity was performed exceptionally well, 75 = At least 5 of the above are true, including point 7, or all the above are true, but point 3 had fewer than 20 users per profile, the activity was performed reasonably well, 50 = At least 3 of the above are true, including point 7, the activity was done with some shortcuts and/or perhaps was not timed well, 25 = Only 2 of the above are true, the activity was done poorly with many shortcomings, 0 = None of the above are true, the activity was not done.

Detailed guidelines for evaluating all the HCI activities listed in Table 5 have been created and are available online (Joshi, 2009b).

The process for computing IoI for a project has the following steps:

- **Company HCI Process Prescription:** The leaders in the HCI group in an organisation prescribe the HCI activities to be carried out in a particular phase of SE process, the expected deliverables from each activity, suggested weights for each activity and suggested activity evaluation guidelines. As it often happens, an organisation may follow not one SE process, but several. In that case, the HCI activities need to be integrated with each SE process. The leaders also suggest a weight for each HCI activity and the guidelines to score each activity. Second column of Table 5 summarises recommended weights for HCI activities for the waterfall model based on Joshi and Sarda (2007) and Pressman (2005) and our interaction with UX team leaders from two companies.
- **Project HCI Process Definition:** After getting a project brief and after understanding the domain, the users, and the project context, a UX professional fine-tunes the weights for the prescribed HCI activities. He/she should consult colleagues in the UX team and development team, and business stakeholders before finalizing the weights. For example, if the domain or users are unknown to the UX team, it may be very important to do 'contextual user studies and user modelling, competitive product analysis' in the communication phase (weight = 4). On the other hand, if the UX

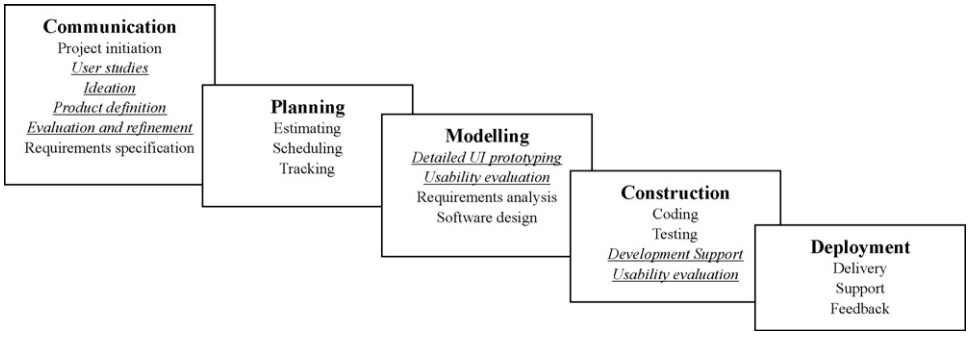

Fig. 1. Integration of HCI activities with the phases of the waterfall model (Joshi and Sarda, 2007). The HCI activities corresponding to each phase have been underlined. The other activities are from Pressman (2005).

team is already very familiar with the context and the domain and if they have a lot of experience designing similar products, it may be not so important (weight = 2).

- **Process Evaluation:** After the project is over, a group of independent UX professionals review the HCI activities, evaluate them for process compliance, and give a score for each activity on a scale of 0–100. In case of multiple evaluators, an average across evaluators is deemed to be the score.
- **IoI Calculation**: The metric is found by computing the weighted average of the scores of all activities: $\text{IoI} = \sum(W_a \times S_a / \sum W_a)$, where W_a is the weight for a particular HCI activity, S_a is the score (from 0 to 100) for that activity. In case there is a lot of divergence in scores of a particular HCI activity, the activity is discussed and reviewers are given a chance to change their score before an average is taken.

Software engineering phases have been extensively described in literature. For example, the phases of the waterfall process model are Communication, Planning, Modelling, Construction and Deployment (Pressman, 2005). On the other hand, no widely accepted process model to integrate HCI activities for the given SE phases has emerged so far. However, there have been a few proposals. Costabile (2001) suggests a way for integrating HCI activities with the waterfall model. Göransson et al. (2003) suggest the creation of a new discipline called usability design for integrating HCI activities with Rational Unified Process (RUP).

The use of IoI with the proposals for integrating HCI activities in the waterfall process model (Joshi and Sarda, 2007) is demonstrated. It is suggested that the Communication phase should have contextual user studies and user modelling, competitive product analysis, ideation with a multidisciplinary team, product definition/information architecture/wireframes with a multidisciplinary team, formative usability evaluation, and refinement of product definition. The Modelling phase of the waterfall model should include detailed user interface prototyping, formative usability evaluation of the user interface, and refinement of prototype. The usability team should support the development team by conducting reviews during the Construction phase and when a reasonably high-fidelity version is available, a summative usability evaluation should be done. Fig. 1 summarizes the HCI activities suggested for the waterfall model phases based on the recommendations made above. The underlined activities in Fig. 1 are the suggested HCI activities while the other activities are the regular SE activities suggested by Pressman (2005).

Table 5 shows calculation of IoI for an example project in an industrial context. First, a senior UX professional from the company adopted the HCI activities, activity weights, and evaluation guidelines that the company should be following based on the recommendations. The second column of Table 5 contains these weights. Then a project that had recently ended was selected. The project manager and the UX professionals working on the project fine-tuned the weights for the project context. The third column of Table 5 contains these weights. A group of reviewers comprising of some project insiders and outsiders reviewed and rated the HCI activities based on the guidelines, and its IoI was calculated. The fourth column of Table 5 contains the scores assigned to each

Table 6
An example of IoI calculation for the integration of HCI activities in the agile model prescribed in Joshi (2009b).

Phases and HCI activities	Recommended weights	Assigned weights	Activity score	Phase score	IoI score
Before iterations start				60.71	
Contextual user studies and user modelling, competitive product analysis, ideation	4	4	50		
Product definition/IA/wireframes with a multidisciplinary team, evaluation	3	4	50		
Detailed UI prototyping for 1st iteration	5	3	100		
Usability evaluation (formative) and refinement of prototype for 1st iteration	3	3	50		
During iterations				56.67	
Detailed UI prototyping for the next iteration	4	4	100		
Usability evaluation (formative) and refinement of the prototype for the next iteration	2	4	25		
Development support reviews by usability team for the current iteration	3	3	50		
Usability evaluation (summative) of the earlier iteration	1	4	50		
After the last iteration				75.00	
Usability evaluation (summative) of release version	3	4	75		
IoI score					60.61

activity by the reviewers. Weighted average scores for HCI activities for each phase is presented in column five and the weighted average for all activities, i.e. the IoI, is presented in column six. In this example, it is clear that the biggest process improvements in this project were required in the communication phase. By comparing the IoI scores, phase scores, and activity scores in several projects, the team can deduce where it needs to be putting in most of its efforts in process improvement.

Similarly, a proposal to integrate HCI activities into the agile processes and developed corresponding guidelines (Joshi, 2009b) was developed. The proposal has four main ideas. First, HCI activities that are difficult to fit in a typical agile iteration (for example, detailed user studies, ideation and exploration of the product definition) ought to happen in the project before the agile iterations begin. At times, this could be an independent project. Secondly, HCI activities should be iterative and the iterations should synchronise with the software development iterations, matching the heartbeat of the project. However, HCI activities should stay ahead of the development team by at least one iteration so that the interface of the software being developed for each iteration has been prototyped and evaluated for usability before it is actually built. Thirdly, the HCI team should closely coordinate with the software development team during the development on a day-to-day basis. This 'development support' will ensure that the user interface is implemented as close to its original intent as possible with minimal documentation. Fourthly, a rigorous summative usability evaluation should be carried out after a few iterations. Feedback coming from formative and summative evaluations should be constantly fed back into future iterations to improve the design.

Table 6 shows calculation of IoI for an example project that uses agile methods based on this proposal.

5. Industry feedbak on UGAM and IoI

First, a qualitative evaluation was done to get preliminary feedback from practitioners on the two metrics. The goal of this evaluation was to try the two metrics in an industrial context, to get feedback on whether such metrics are necessary and useful, and to collect qualitative feedback to improve the metrics. We also wanted to measure the time it would take to compute the metrics in real-life situations.

UGAM and IoI were computed retrospectively for three projects in two large contracted software development companies. In each case, the metrics computation was done by HCI professionals from the project, independent HCI professionals, and project stakeholders. These stakeholders were familiar with the software artefact that was developed and the process that was followed. At the end of the metrics computation, feedback was taken from participants about the metrics.

It typically took about 3 h to compute both IoI and UGAM for each project. The time included explaining the two metrics, weight assignment, and scoring. This seemed to be the optimum time; longer meetings were difficult to schedule. The projects performed similarly in IoI and UGAM scores – the one project that had a high UGAM value also had a high IoI value. Participants, particularly project stakeholders, were at home with the activity of metric calculation. To them, the activity seemed to bring HCI closer to SE. It seemed to create lot of buy-in for HCI activities from the project stakeholders. One project stakeholder said, "*I never thought we could think so much [about user experience]*". The activity seemed to be more successful in projects where several stakeholders from the project participated as it stimulated discussion among stakeholders. While the participants appreciated the organizational perspective, the metrics seemed of less use to the projects as the projects were already over. Participants suggested that metrics should be calculated mid-way through a project, while course correction was still possible.

Specifically, UGAM seemed to help the HCI designers and project stakeholders to make goals explicit. One HCI designer remarked, "*Had we done this earlier, I would have known where to focus*". The teams adjusted weight to suit goal parameters to their project – they confirmed that this flexibility is indeed desirable. Though parameter evaluation guidelines for UGAM helped, more details were desired. Giving examples of HCI goals (learnability, ease of use, etc.) helped participants to set goal parameters and weights. One stakeholder remarked, "*Without these inputs it would have been difficult to [assign weight and scores]*".

In case of a few UGAM parameters, divergent scores emerged for some parameters in each project. Usually, variations were observed in parameters for which the evaluation guidelines were not understood well or were interpreted differently by evaluators. In such cases, it was felt that it was better to let the participants discuss the parameter and change ratings to converge scores if they so desire. Reducing the number of steps in scoring a parameter (for example, 0–25–50–75–100) and assigning a meaning to each step helped reduce the variation in the scores. More detailed guidelines would help in reducing divergence further.

Computing IoI was useful for project stakeholders as they could see the importance of HCI activities in the SE context. The HCI activities integrated in SE process models were acceptable as suggested. Though they were explicitly asked, none of the project stakeholders wanted changes in the prescribed HCI activities, their weights, or the evaluation guidelines. An important feedback was a need for process models specifically targeted to redesign projects. Process models typically discuss new product development. Given that many industry projects are of the type "*next version of X*", process models must be specifically adapted for them.

Walking through the activity evaluation guidelines helped in scoring, as not all stakeholders were aware of all the HCI activities. It was felt that IoI should be computed before computing UGAM as this minimizes bias. The metric descriptions presented in this paper are a result of iterative modifications that reflect the feedback and lessons learnt.

6. Correlating UGAM and IoI

A final evaluation was done with two groups of industry participants. The purpose of this evaluation was to explore the quantitative relationship between IoI and UGAM in industrial projects. We wanted to explore whether IoI, a process metric, has any effect on UGAM, a product metric.

Study A

The first group attended a training programme – a 9-day professional course on human–computer interaction design conducted by one of the authors. Participants came from mixed educational backgrounds such as graphic design, product design, web design, user interface design, e-learning, engineering, product management, and ergonomics. Industry experience of participants varied between 2 and 15 years but HCI-related experience of participants varied from only 1–3 years. Many participants had an aptitude for design and a few of them even had formal design education.

A study to correlate UGAM and IoI was conducted during the course (Study A). Before the study started, the participants had attended four and a half days of the course during which they had learnt about conducting and analyzing contextual interviews, affinity diagrams, personas, conceptual models, layers of user experience, design process, and user-experience goals. After this, participants were invited to participate in the UGAM and IoI calculation for a project on which they had worked professionally.

Participants were then taught the method of calculating UGAM and IoI. During UGAM calculation, participants were walked through each goal parameter and were shown examples to explain its meaning. Similarly, during IoI calculation, they were walked through each HCI activity and its implication for the SE process. Participants were informed that it was optional to submit the project data and they could remain anonymous in their submission.

Study B

A second, more controlled study was done with participants who had more experience and formal background in HCI (Study B). Before the study, a 1-day tutorial on user-experience metrics was announced on a mailing list of HCI professionals in India. The tutorial was conducted five times on weekends in five different cities. It gave a general overview of several user-experience metrics, including UGAM and IoI. The tutorial was free to attend.

The tutorial was open only to participants who had a few years of experience in HCI or a related area. The participants filled out a registration form for attending the tutorial and those who had no experience in HCI were screened out. These participants had an overall average experience of 7 years and HCI-related experience of 3.5 years.

After the tutorial, participants were chosen randomly and were invited to participate in the study. Each participant who agreed was interviewed in two or three sessions. The first session was a short briefing session over the phone, during which the participant was told the purpose and the procedure of the study. Each participant was requested to contribute two projects. The participant was encouraged to contribute not only projects that he thought went well, but also projects that he thought did not go so well. The participant was also requested to report the nature of the project – whether it was a contracted software development service for a client or a project in a product company. The participant was promised confidentiality and was invited to participate in the second session. The second session was scheduled at a time convenient to the participant.

At the start of the second session, the participant was reminded about the purpose and the procedure. He was handed over printed forms for UGAM and IoI calculation and was asked to fill it out while constantly thinking aloud as he answered each question. The participant was not left to himself, but was walked through the forms as he filled them out. Each question was verbally explained and the participant was encouraged to ask questions. Clarifications and examples were given where necessary. This was done to ensure that he understood each question clearly. Many second sessions were scheduled as face-to-face meetings. Later, some second sessions were also scheduled over phone. For sessions conducted over phone, the participants were sent a soft copy of the forms in advance. A third session similar to the second one was scheduled if necessary.

6.1. Findings

Thirty-five participants participated in Study A, of which 21 participants submitted their data for both UGAM and IoI. 141 participants registered for the 5 tutorials in Study B. Of these, 83 participants qualified and attended the tutorials. Of these, 33 participants were requested to participate in the study. Most agreed, but only 23 could be scheduled. Between them, Study B participants contributed 40 projects. The participants came from a wide variety of companies including four large (25,000+ employees each) contracted software development companies, four relatively smaller contracted software development companies, four multi-national companies with large product development centres in India, one large, internationally popular internet company, and five smaller product development companies.

Table 7
Raw UGAM and IoI scores for 61 projects in two studies.

IoI	UGAM	Process	Type
92.39	69.93	Waterfall	Service
89.29	80.90	Waterfall	Product
80.43	71.43	Waterfall	
79.63	85.39	Waterfall	Service
79.31	80.65	Agile	Service
78.26	86.67	Waterfall	
77.17	71.40	Waterfall	
76.09	73.61	Waterfall	
75.83	74.68	Waterfall	Service
73.75	71.70	Waterfall	Product
72.00	75.33	Waterfall	Service
70.95	68.67	Waterfall	Service
69.17	77.32	Agile	Product
69.17	77.32	Agile	Product
67.31	63.51	Waterfall	Service
67.24	68.00	Agile	Product
66.30	64.08	Waterfall	
66.18	85.14	Agile	Service
65.63	68.66	Waterfall	Service
65.38	75.73	Waterfall	Service
65.22	70.71	Waterfall	
64.58	75.63	Waterfall	Product
64.17	66.81	Waterfall	Service
64.13	58.59	Waterfall	
63.39	78.78	Waterfall	Product
63.00	61.44	Waterfall	Service
62.96	80.16	Waterfall	Product
62.96	74.39	Agile	Product
61.46	68.88	Waterfall	Product
61.00	70.79	Waterfall	Service
60.61	70.99	Agile	Service
58.70	67.56	Waterfall	
58.33	70.74	Waterfall	Service
57.61	78.19	Waterfall	
57.50	74.65	Waterfall	Service
56.00	66.67	Waterfall	Product
55.47	77.22	Agile	Service
55.43	77.59	Waterfall	
55.36	62.26	Waterfall	Product
54.57	65.18	Waterfall	
51.92	56.32	Waterfall	Service
51.09	57.77	Waterfall	
50.00	65.41	Waterfall	Service
50.00	58.33	Waterfall	
50.00	52.96	Waterfall	
50.00	44.20	Waterfall	
46.74	65.91	Waterfall	
46.55	48.42	Agile	Service
46.05	42.47	Waterfall	Service
45.83	61.43	Waterfall	Service
41.30	46.52	Waterfall	
38.54	55.36	Agile	Service
38.04	57.39	Waterfall	Product
38.04	53.05	Waterfall	
37.50	59.23	Agile	Service
36.96	33.42	Waterfall	
35.87	44.21	Waterfall	
33.93	46.55	Waterfall	Service
31.00	47.30	Waterfall	Service
29.35	54.58	Waterfall	
26.85	56.92	Waterfall	Product

A combined analysis of the 61 projects from Study A and Study B is presented below.

Table 7 lists the raw data for the UGAM and IoI scores for the 61 projects. It also lists the type of process model followed. Out of the 61 projects, 50 projects reported following the waterfall model, while 11 projects reported following agile process models. Of the 40 projects in Study B, 26 were carried out as part of a contracted software development service for a client while 14 were projects in product companies (this data was not collected in Study A).

Normal P–P plot (Fig. 2) drawn for UGAM and IOI, shows that assumptions of normality are not grossly violated for either metric.

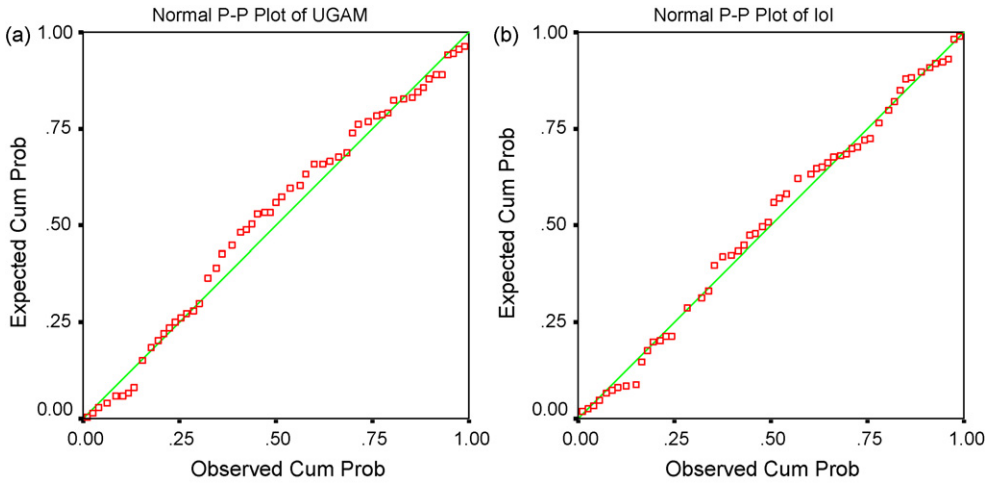

Fig. 2. Normal P–P plot for User Experience Goal Achievement Metrics (a) and for Index of Integration (b) ($N = 61$).

Table 8
Model summary for UGAM regressed on IoI.

	Overall ($N = 61$)	Waterfall ($N = 50$)	Agile ($N = 11$)	Service ($N = 26$)	Product ($N = 14$)	Service Waterfall ($N = 19$)	Product Waterfall ($N = 10$)
R	0.752	0.753	0.809	0.747	0.825	0.769	0.841
R^2	0.566	0.567	0.655	0.559	0.681	0.591	0.707
Adjusted R^2	0.558	0.558	0.617	0.540	0.654	0.567	0.671
Standard error of the estimate	7.9209	7.9012	7.1171	7.9147	4.7463	7.2775	5.1896
Change statistics							
R^2 change	0.566	0.567	0.655	0.559	0.681	0.591	0.707
F change	76.862	62.924	17.084	30.370	25.617	24.607	19.326
df1	1	1	1	1	1	1	1
df2	59	48	9	24	12	17	8
Significance F change	0.000	0.000	0.003	0.000	0.000	0.000	0.002

Therefore, to determine the relationship between UGAM and IOI, two-tailed Pearson's correlation was performed. It was found that there is a significant positive correlation ($r = 0.752$, $p < 0.0005$ two-tailed) between UGAM ($M = 65.82$, SD = 11.92, $N = 61$) and IOI ($M = 58.38$, SD = 14.93, $N = 61$).

A simple linear regression was performed to determine if IoI significantly determines the scores of UGAM. A significant model emerged (Table 8), with predictor IoI accounting for 56% of the variance in UGAM (adjusted $r^2 = 0.56$), which was highly significant ($F = 76.862$, $p < 0.0005$). The coefficients (Table 9) show that IoI ($\beta = 0.601$, $p < 0.0005$) demonstrated significant effects on UGAM. The t-statistic for the slope was also significant $t = 8.767$, $p < 0.0005$. The 95% confidence interval for IoI β varies from 0.463 to 0.738. Thus, it could be concluded that there was a positive significant relationship between IoI and UGAM.

UGAM and IoI were analysed using ANOVA in means (Table 10). A small significance value indicates a linear relationship between UGAM and IoI ($F = 69.94$, $p < 0.0005$). Thus, the expected value of UGAM can be represented by the following equation:

$$\text{UGAM} = 0.601 \times \text{IoI} + 30.763$$

It could be concluded that the relationship between IoI and UGAM is strong, positive, and linear. This is also evident in the curve plotted between observed IOI and UGAM drawn against theoretical linear distribution between the two variables (Fig. 3).

Table 9
Regression coefficients for UGAM regressed on IoI.

			Overall ($N = 61$)	Waterfall ($N = 50$)	Agile ($N = 11$)	Service ($N = 26$)	Product ($N = 14$)	Service Waterfall ($N = 19$)	Product Waterfall ($N = 10$)
(Constant)	Unstandardised coefficients	B	30.763	30.992	29.270	32.245	43.796	31.691	43.958
		Standard error	4.126	4.409	10.172	6.365	5.554	7.031	6.131
	t		7.457	7.030	2.878	5.066	7.885	4.508	7.169
	Significance		0.000	0.000	0.018	0.000	0.000	0.000	0.000
	95% confidence interval for B	Lower bound	22.508	22.128	6.260	19.109	31.695	16.858	29.819
		Upper bound	39.018	39.856	52.280	45.382	55.898	46.524	58.097
IoI	Unstandardised coefficients	B	0.601	0.582	0.693	0.576	0.445	0.559	0.439
		Standard error	0.068	0.073	0.168	0.104	0.088	0.113	0.100
	t		8.767	7.932	4.133	5.511	5.061	4.961	4.396
	Significance		0.000	0.000	0.003	0.000	0.000	0.000	0.002
	95% confidence interval for B	Lower bound	0.463	0.434	0.314	0.360	0.254	0.321	0.209
		Upper bound	0.738	0.729	1.072	0.791	0.637	0.796	0.669

Table 10
ANOVA in means table for UGAM × IoI ($N = 61$).

		Sum of squares	df	Mean square	F	Significance
Between groups	(Combined)	7972.475	52	153.317	2.224	0.115
	Linearity	4822.351	1	4822.351	69.943	0.000
	Deviation from linearity	3150.124	51	61.767	0.896	0.632
Within groups		551.572	8	68.947		
Total		8524.048	60			

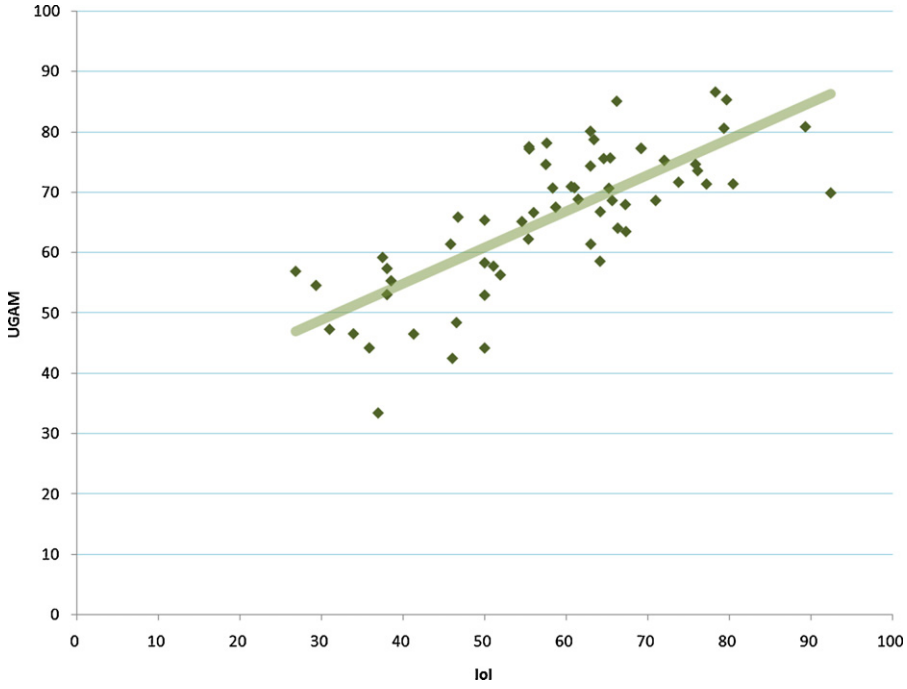

Fig. 3. IOI vs. UGAM scatter plot against linear curve ($R^2 = 0.56$).

Tables 8–10 report further findings from regression analyses on subsets of the data only considering the projects that followed:

- the waterfall models ($N = 50$),
- the agile process models ($N = 11$),
- the known contracted service projects ($N = 26$),
- the known product company projects ($N = 14$),
- the known contracted service projects that followed the waterfall model ($N = 19$),
- the known product company projects that followed the waterfall model ($N = 10$).

All models were significant (Table 8) and consistently returned significant positive coefficients, with all pairs of the 95% confidence intervals for IoI β positive (Table 9). Thus, it could be concluded that IoI and UGAM can be used by software projects that use either waterfall model or agile model and by organisations involved in contracted service projects as well as in product development.

It is interesting to note that the constant seems to be somewhat higher and the coefficient of IoI is somewhat lower for the two subsets of product company projects compared to the two subsets of services company projects (Table 9). This implies that process integration is somewhat less important in product companies than in service companies – possibly because there are other factors at play in product companies (such as high stakeholder involvement, immediacy of business impact of poor usability or simply high levels of skills, talent, and experience).

Similarly, it is interesting to note that agile projects seem to have a higher coefficient of IoI compared to that of waterfall projects – implying a higher impact that process integration can have on the quality of user experience in agile projects.

7. Conclusions and discussion

We proposed a metric to measure the quality of the product against its usability goals (UGAM) and another metric to measure the level of integration of HCI activities in SE process models (IoI). The metrics were evaluated in three different ways.

A classroom-based experiment was conducted to evaluate UGAM. The experiment showed that even in adverse conditions, UGAM could predict the user experience of a product in a way similar to ratings of evaluators after traditional usability studies for goal-driven, task-oriented products. UGAM did not predict as accurately the ratings of what could be a leisurely exploration of information spaces, though it was not completely off the mark.

Qualitative feedback on UGAM and IoI from industrial participants showed that the metrics helped the participants and they were positively inclined to use them. UGAM and IoI were found to be useful and practical in evaluating products and processes. There was a lot of buy-in from project stakeholders to calculate metrics, as they wanted to track and control the user-experience design of the product. They were happy to discover that the metrics were lightweight and not dependent on specific usability methods.

A third evaluation of UGAM and IoI on 61 industry projects showed a strong positive linear correlation between IoI and UGAM

in all projects. It was also found that IoI has a somewhat greater effect on UGAM in projects that use the agile process model rather than the waterfall process model and in projects that are executed as a contracted software development service rather than in projects in product companies.

It is important to discuss the limitations of the studies and evaluations. The evaluation discussed in Section 3.1 was carried out by students of interaction design and computer science in the classroom. It is quite possible that assessments of professionals are different. Further, the conclusions drawn are based on the evaluation of only nine products. It would be interesting to follow up with more detailed assessments by professionals with more products.

An important limitation of the two studies presented in Section 6 was that UGAM and IoI calculation were done by only one project member from each project, while ideally one should calculate the two metrics by averaging inputs from several members, both internal and external. Perhaps another limitation is that the study was done only in one country – India. It could be that the practices in the IT industry in India are different from those in the other countries. For example, it is quite common to find projects using the Rational Unified Process model (RUP) in Europe, while in India the most popular process model is waterfall, followed by agile processes. It is interesting to note that not a single project that uses RUP was found. However, within this limitation, we believe we got a wide representation of projects, including services, products, and projects following the waterfall and agile process models. Additional studies are required in other geographies to generalise the results, but we do not expect significantly different results.

It is important to discuss the limitations of the two metrics. As discussed above, there are arguments against summary measures and both UGAM and IoI are summary measures. Yet, summary measures are useful in many contexts. As Gulliksen et al. (2008) argue, decision makers often want measures to base their decisions upon and while this could be a risk in the context of usability, it may also be a good opportunity. Summary measures are particularly useful for comparisons across projects. Such comparisons can help the team understand what works and what does not and improve the performance year-on-year. UGAM and IoI have an organizational perspective and allow comparison across projects. In addition, UGAM and IoI are different from the summary measures cited above. Both are based on weighted averaging of self-set targets, which ensures that important issues count for more. Both create a profile and allow a drill-down to constituent components, which point to specific areas where corrective action might be taken.

Perhaps the most important limitation of UGAM comes from the ephemeral nature of 'user experience'. Any attempt to embody such an abstract phenomenon numerically is bound to be subjective and measures are open to interpretation. Yet, with UGAM, you get what you set. UGAM is not meant to measure the entire abstract notion of user experience, just the level to which the designers achieve the goals they set for themselves. Designers set goals according to their understanding of the context, user needs, and business goals. If their judgements (or judgements of the stakeholders who influenced them) are erroneous to an extent, UGAM would not reflect the true user experience to that extent.

We believe that in spite of these limitations, UGAM is useful. UGAM shows the extent to which targeted user-experience goals are achieved in a project. It was found that breaking up abstract notions of user experience into specific goals and parameters helped evaluators focus on one issue at a time and reduced the subjectivity in measurement. Linking parameter scores to performance metrics, making the evaluation criteria explicit, and averaging across several evaluators further reduced the subjectivity in judgment.

The main limitation of IoI is that it does not measure the absolute process quality; rather how compliant a project was to the prescribed process. There are no widely accepted process models that integrate HCI with SE processes today. Yet, IoI in conjunction with UGAM and other product metrics may be used to verify the effectiveness of new and current process model proposals. If the product metrics and IoI are correlated (as was the case in our proposals to integrate HCI activities in waterfall and agile processes), the new process proposal should be acceptable. On the other hand, if the UGAM and IoI do not show a correlation, it questions the efficacy of the prescribed process models.

In future, we plan to use metrics prospectively throughout the duration of projects and demonstrate their usefulness during the project. We will be building more elaborate tools and guidelines to improve the consistency of weights and scores. We also propose to do additional validations of the two metrics in experimental and industrial situations.

Acknowledgments

We thank Pramod Khambete, Ved Prakash Nirbhay, Deepak Korpal and other participants from Tech Mahindra and Atul Manohar, Aniruddha Puranik and other participants from Satyam for helping us evaluate and improve the early versions of the metrics. We thank students of courses CS 708 and IN 604 for the year 2008 and participants of Monsoon Course on HCI 2008 for participating in our experiments. We thank Prof. U.A. Athavankar, Prof. Umesh Bellur, and Prof. S. Sudarshan of IIT Bombay for their suggestions in developing the two metrics.

References

Bevan, N., 2008. Classifying and selecting UX and usability measures in law. In: Bevan, N., Christou, G., Springett, M., Lárusdóttir, M. (Eds.), International Workshop on Meaningful Measures: Valid Useful User Experience Measurement, pp. 13–18.

Beyer, H., Holtzblatt, K., 1998. Contextual Design: Defining Customer Centered Systems. Morgan Kaufman.

Bias, R., Mayhew, D. (Eds.), 2005. Cost-Justifying Usability, Second Edition: An Update for the Internet Age. Morgan Kaufmann.

Brooke, J., 1996. SUS: a Quick and Dirty Usability Scale. In: Jordan, P.W., Thomas, B., Weerdmeester, B.A., McClelland (Eds.), Usability Evaluation in the Industry. Taylor & Francis.

Chin, J.P., Diehl, V.A., Norman, K.L., 1988. Development of an instrument measuring user satisfaction of the human–computer interface. In: ACM CHI, pp. 213–218.

Cooper, A., Riemann, R., 2003. About Face 2.0 the Essentials of Interaction Design. Wiley.

Costabile, M.F., 2001. Usability in the software life cycle. In: Chang, S.K. (Ed.), Handbook of Software Engineering and Knowledge Engineering, vol. 1. World Scientific, pp. 179–192.

Fenton, N.E., Pfleeger, S.L., 2002a. Software Metrics – A Rigorous and Practical Approach. Thomsan Brooks/Cole, p. 5.

Fenton, N.E., Pfleeger, S.L., 2002b. Software Metrics – A Rigorous and Practical Approach. Thomsan Brooks/Cole, p. 84.

Göransson, B., Lif, M., Gulliksen, J., 2003. Usability design – extending rational unified process with a new discipline. In: International Workshop on Interactive Systems Design, Specification, and Verification.

Gulliksen, J., Cajander, A., Eriksson, E., 2008. Only figures matter? – If measuring usability and user experience in practice is insanity or a necessity in law. In: Bevan, N., Christou, G., Springett, M., Lárusdóttir, M. (Eds.), International Workshop on Meaningful Measures: Valid Useful User Experience Measurement, pp. 91–96.

Hornbæk, K., Law, E., 2007. Meta-analysis of correlations among usability measures. In: Proceedings of CHI, pp. 617–626.

IEEE, 1993. IEEE Standard Glossary of Software Engineering Terminology, IEEE.

IFIP Working Group 2.7/13.4 on User Interface Engineering, 2004. Bridging the SE & HCI Communities, http://www.se-hci.org/bridging/index.html (accessed August, 2008).

International Organization for Standardization, 2001. ISO/IEC 9126-1:2001 Software Engineering – Product Quality.

International Organization for Standardization, 1997. ISO 9241-1:1997 Ergonomic requirements for office work with visual display terminals (VDTs).

Jordan, P.W., 2000. Designing Pleasurable Products. Taylor & Francis.

Joshi, A., 2006. HCI in SE process literature. In: Indo-Dan HCI Research Symposium, IIT Guwahati.

Joshi, A., Sarda, N.L., 2007. HCI and SE: towards a 'truly' unified waterfall process. In: HCI International '07.
Joshi, A., Tripathi, S., 2008. User experience metric and index of integration: measuring impact of HCI activities on user experience. In: First International Workshop on the Interplay between Usability Evaluation and Software Development (I-USED), vol. 407. CEUR.
Joshi, A., 2009a. Usability goals setting tool. In: 4th Workshop on Software and Usability Engineering Cross Pollination: Usability Evaluation of Advanced Interfaces, Interact.
Joshi, A., 2009b. Index of Integration, http://www.idc.iitb.ac.in/~anirudha/ioi.htm (accessed November 5, 2009).
Kirakowski, J., Corbett, M., 1993. SUMI: the Software Usability Measurement Inventory. British Journal of Educational Technology 24 (3), 210–212.
Kroll, P., Kruchten, P., 2003. The Rational Unified Process Made Easy. Pearson Education.
Law, E., Roto, V., Hassenzahl, M., Vermeeren, A., Kort, J., 2009. Understanding, scoping and defining user experience: a survey approach. In: Proceedings of CHI, pp. 719–728.
Lewis, J., 1991. A rank-based method for the usability comparison of competing products. In: Human Factors and Ergonomics Society 35th Annual Meeting, pp. 1312–1316.
Lin, H., Choong, Y., Salvendy, G., 1997. A proposed index of usability: a method for comparing the relative usability of different software systems. Behaviour & Information Technology, pp. 267–278.
Mahlke, S., 2005. Understanding users' experience of interaction. In: Marmaras, N., Kontogiannis, T., Nathanael, D. (Eds.), Proceedings of EACE '05, pp. 243–246.
Mayhew, D., 1998. The Usability Engineering Lifecycle: A Practitioner's Handbook for User Interface Design. Morgan Kaufmann.
McCarthy, J., Wright, P., 2004. Technology as Experience. The MIT Press.
McGee, M., 2004. Master usability scaling: magnitude estimation and master scaling applied to usability measurement. In: Proceedings of CHI'00. ACM Press, pp. 335–342.
Nielsen, J., 1993. Usability Engineering. Morgan Kaufmann.
Norman, D.A., 2004. Emotional Design: Why We Love (or Hate) Everyday Things. Basic Books.
Pressman, R., 2005. Software Engineering – a Practitioner's Approach, 6th ed. McGraw-Hill.
Pyla, P.S., Pérez-Quiñones, M.A., Arthur, J.D., Hartson, H.R., 2003. Towards a model-based framework for integrating usability and software engineering life cycles. In: Interact 2003 Workshop on "Closing the Gaps: Software Engineering and Human Computer Interaction", pp. 67–74.
Sauro, J., Kindlund, E., 2005. A method to standardize usability metrics into a single score. In: CHI '05, pp. 401–409.
Shneiderman, B., 2004. Designing the User Interface, Strategies for Effective Human–Computer Interaction, 4th ed. Addison Wesley.
Swallow, D., Blyth, M., Peter, W., 2005. Grounding experience: relating theory and method to evaluate the user experience of smart-phones. In: Proceedings of the 2005 Annual Conference on European Association of Cognitive Ergonomics, pp. 91–98.
Tractinsky, N., Katz, A.S., Ikar, D., 2000. What is beautiful is usable. Interacting with Computers 13, 127–145.

Anirudha Joshi is an interaction designer and associate professor in the Industrial Design Centre, IIT Bombay. His research interests include integration of HCI with software engineering and interaction design for development.

NL Sarda is professor and former head of the Computer Science and Engineering department of IIT Bombay and former dean academic programmes of IIT Bombay. His main research interests are in the areas of database systems and software engineering. He is associated with many organisations as a consultant, offering advice in planning, selection, implementation and evaluation of information technology solutions.

Sanjay Tripathi holds PHD in HCI and Masters in technology and management from Flensburg University, Germany. Currently he is working as scientist at the ABB Corporate Research Bangalore center. Prior to ABB, he worked at large software service provider company in India and major part of his professional life being spent in various industry/academia research projects in Germany.