To appear in Proceedings of the International Conference on Software Engineering Theory and Practice (SETP-08), July 2009

An Effort Based Model of Software Usability

Oleg V. Komogortsev
Texas State University
San Marcos, TX 78666
ok11@txstate.edu

Carl J. Mueller
Texas State University
San Marcos, TX 78666
carl.mueller@txstate.edu

Dan Tamir
Texas State University
San Marcos, TX 78666
dt19@txstate.edu

Liam Feldman
Texas State University
San Marcos, TX 78666
lf1081@txstate.edu

Abstract

This paper presents a new effort based model for software usability and reports the results of a set of experiments performed to assess the validity of the model. The model is based on the notion that usability is an inverse function of effort. Physical and mental effort are obtained and inferred from logging physical activity and eye tracking. Using the new model, an objective metric of software usability that facilitates setting measurable requirements and enables the comparison of two or more implementations of the same application is developed. The experiments results show high correlation to learning theory models, strongly support the relationship of effort to usability and demonstrate that operability, learnability, and understandability of software systems can be measured using the effort based metric .

1 Introduction

Poor software quality can lead to execution errors, deviation from requirements and specs, substantial development cost overruns, and user dissatisfaction [9, 14, 22]. In a few extreme situations poor software quality can lead to death [14]. Usability, one of the components of software quality, is related to the operability, learnability, understandability, and level of satisfaction associated with a software system.

Software developers have a wide variety of tools for prototyping, inspecting, and testing software usability [21]. Considering the large number of user complaints about software usability, these techniques may not address the problem efficiently. Furthermore, the challenges presented by usability issues may not lie solely in the tools and techniques used in the development process. Physiological and psychological characteristics as well as sociological conditioning heavily influence software usability making it possibly one of the most subjective attributes of software quality. Usability evaluation requires observing a number of human subjects while engaged in using the system. Interpreting these observations necessitates adding a person skilled in psychological / cognitive evaluation to the testing team.

Many software engineers are not familiar with the factors influencing usability and are frequently uncomfortable with the entire topic. Furthermore, occasionally developers do not view usability testing as productive evaluation because these evaluations usually indicate an area where the subjects had problems and does not necessarily pinpoint a specific issue with the software. Being close to the project deadline can amplify this frustration of software engineers and engineering management with current usability evaluation procedures, especially when there is no way to determine how much time and effort are required to identify and modify the usability issues with the software. Because of the uncertainty and expense of usability evaluations, some managers are reluctant to include formal usability testing in their development plan. Instead of using testing, these managers prefer to rely on best practices, templates, and inspections to establish software usability.

One approach that may make software engineers more comfortable with the topic of usability is to recast it into terms and concepts that are familiar to the software engineering community. Investigating objective and engineering-based methodologies of evaluating software usability is the focus of this research.

The actual challenge of developing usable software may lie in the lack of a clear and concise understanding of what too many software engineers view as a fuzzy concept. Not all authorities on software quality provide a definition of usability. Some authorities recommend usability testing but only provide a checklist of things to investigate [4, 12, 17]; and these authorities are, for the most part, balancing between systems with "card input" and interactive systems. Most quality models [1, 10, 15, 18] provide a relatively consistent and concise definition of usability, but the attributes used to characterize the many facets of usability are not consistent. This research uses the characterization of usability provided in the ISO/IEC 9126 because it is one of the more recent quality models, it is an industry standard, and it provides a measurement system for each of their quality attributes and characteristics. This standard defines usability as "the capability of the software product to be understood, learned, used, and attractive to the user when used under specified conditions" [1], with the following characteristics: Understandability, Learnability, Operability, Attractiveness, and Compliance.

Understandability is the ability of a user to understand the capabilities of the software and its

suitability to accomplish specific goals. Traditionally, it is measured by providing the user with a tutorial or software documentation and then evaluating the users' ability to determine the users' level of understanding of the software's functionality, operation, and input/output data [2]. The ISO/IEC 9126 standard also recommends using cognitive monitoring techniques to evaluate the subject's response. Cognitive monitoring techniques are using one-way mirrors or concealed cameras to record the subject's behavior along with evaluation of the findings by a psychology professional.

Learnability describes how easy it is for a subject to learn to use the software. For this characteristic, the standard measures how long it takes to learn and perform a task, the number of functions used correctly, and the utility of the help facility [2]. In addition to the measurements, the standard proposes cognitive monitoring techniques. Learnability has deep roots outside of software quality. Ebbinghaus, a German psychologist, is probably the first researcher that introduced (in the 19th century) a learnability model describing the time required to memorize new knowledge [8]. Figure 1 illustrates the learning model. In the 1930's, research at Wright-Patterson quantified the notion; and in the 1960's it evolved into an experience curve [3]. The experience curve research applied the learning model to an industrial setting by comparing cost per unit verses units developed.

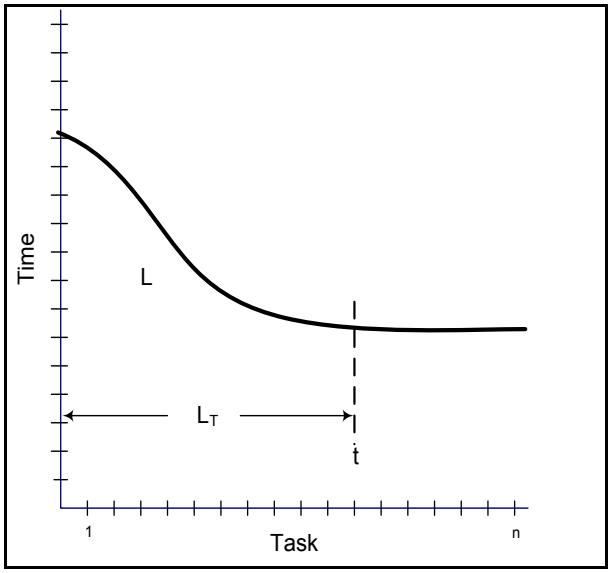

Figure 1 Hypothetical Learning Curve

Operability is the capability of a user to use the software to accomplish a specific goal. Assessing operability requires measuring several characteristics including Operational consistency; Error correction; Error correction in use; Default value availability in use; Message understandability; Self-explanatory error messages; Operational error recoverability in use; Time between human error operation in use; Undoability; Customizability; Operation procedure reduction; and, Physical accessibility [2]. Some of these metrics are objective measurements, but many require cognitive monitoring techniques to evaluate.

As the name implies, attractiveness is the appeal of the software to a user. Attractiveness is a subjective usability characteristic that involves sociological and psychological issues as well as gender and personal taste concerns. The ISO/IEC 9126 standard characterizes attractiveness by providing subjects with a questionnaire to evaluate the interface and by observing subjects customizing the appearance to their satisfaction [2].

Compliance is the most straightforward characteristic to evaluate. It measures how well the software adheres to external and internal rules and regulations related to usability. Developers compile a list of the required standards, conventions, style guides, and regulations. Then using functional testing techniques, verify that the software complies with them [2].

This relatively short description of the metrics necessary to evaluate usability demonstrates that designing a usability test is an extremely time consuming and expensive task. This is a test with potentially high cost, which may not be able to pinpoint specific design or implementation defects or issues. Another problem with the number of measurements is how to create objective specifications for so many diverse characteristics. Setting objective measurements for all of these characteristics can increase the time necessary to specify requirements. Nevertheless, reducing the high cost of usability testing is difficult because each of the measures proposed by the ISO/IEC standard are good and identify specific problems, and it is not possible to eliminate the use of human test subjects. It may be possible, however, to take a slightly different approach to usability testing using techniques that developers and testers are more comfortable with and could administer without requiring cognitive evaluation techniques. ***In fact this research does not propose to eliminate current cognitive based evaluation. It is proposed to add a set of "tools" (objective metrics) that can be used in the process of software design, implementation, testing, validation, and verification, by software engineers to improve the usability of their products thereby reducing the need for the cognitive evaluation.***

In addition to the usability metrics, software engineers would need appropriate usability test design methodologies. One possible approach is to design a set of goals or tasks and measure the effort and time necessary for a group of subjects to accomplish each goal [24]. These metrics would supply an absolute (i.e., not a relative) usability measurement technique. A relative measure technique can be obtained by the developers via the notion of the "designer effort curve" or the "ideal effort curve." The developers would estimate the effort

and time necessary to complete each goal and compare the observed effort with the estimated effort. If the observed effort is greater than the estimated effort, then there is a problem requiring further investigation. After identifying the existence of a problem, developers could trace the observation logs to find where the subjects experienced a problem causing the expenditure of additional effort.

This paper introduces the new effort based usability assessment model concentrating on understandability, operability, and learnability and presents the related test design methodology. In addition, the paper reports on the results of a major experiment that used the test design method along with effort logging for validating the model. The rest of the paper is organized in the following way: Section 2 presents the effort based productivity hypothesis. Section 3 and 4 present the experiment's design, execution, and results followed by conclusions in a section and proposals for further research (section 6).

2 A Hypothesis for Effort-based Productivity

Many software publishers are claiming that their product requires less effort than the competition. Some publishers mention less keystrokes for task completion as a product advantage. Even though these advertisers provide no objective substantiation for these claims, the fact that this may entice a buyer to purchase the product gives credibility to the notion that there is a relationship between usability and effort

For this hypothesis, E denotes all the effort, mental and physical, required to complete a task with computer software, as defined by the following equations:

$$E = \begin{pmatrix} E_{mental} \\ E_{physical} \end{pmatrix}$$

$$E_{mental} = \begin{pmatrix} E_{eye_{mental}} \\ E_{other\_mental} \end{pmatrix}$$

$$E_{physical} = \begin{pmatrix} E_{manual\ \_physical} \\ E_{eye\ \_physical} \\ E_{other\_physical} \end{pmatrix}$$

Most of the terms used in the equations are self explanatory and denote types of efforts required for task completion. On the other hand, E_{other_mental} and $E_{other\_physical}$ respectively denote the amount of mental and physical effort that cannot be represented through logging of manual effort and eye tracking. They can be considered as an error term that accumulates the errors inherent in the logging and tracking along with the fact that there are other forms of mental and physical effort that are not included in the hypothesis.

Precise methods for measuring mental effort (E_{mental}) are still in a theoretical stage. Researchers have made progress measuring mental or cognitive activities using Magnetic Resonance Imaging. Another approach, still in the theoretical stage, is to measure cognitive activities using eye tracking. One problem with using eye tracking to measure cognitive activity is that it is not possible to determine whether the subject is thinking about the task or something else.

Methods for measuring physical effort ($E_{physical}$) are more precise. It is possible to log a subject's activities and convert key / button presses, mouse movement into units of effort thereby describing the manual effort ($E_{manual\ \_physical}$). Tracking eye movements with an eye tracking device provides a method for making a precise measurement of eye effort ($E_{eye\ \_physical}$). The current research hypothesis is involved with measuring effort and showing the correlation between effort and usability. This hypothesis is validated through a comprehensive set of experiments. Future research to develop actual usability metrics is currently in advanced stages.

2.1 Measuring Effort

Several informal studies indicate that many system users associate the "physical" effort required for accomplishing tasks with the usability of the software. In the case of interactive computer tasks, it may be possible to calculate effort from a weighted sum of mouse clicks, keyboard clicks, Mickeys, etc., where the term Mickey denotes the number of pixels (at the mouse resolution) traversed by the user while moving the mouse from a point(x_0, y_0) to a point (x_1, y_1) .

The definition of effort uses continuous functions. In practice, given the discrete nature of computer interaction, these measures are quantized by converting integrals to sums. Assume that an interactive task R starts at time t_0. The effort at time t is defined to be:

$$E_{manual\ \_physical}(t) = \frac{1}{t - t_0}\int_{t_0}^{t} (w_1 \times mic(t) + w_2 \times mc(t) + w_3 \times mk(t) + w_4 \times p(t))dt$$

Where: $mic(t)$, $mc(t)$, $mk(t)$ are (respectively) the number of Mickeys, the number of mouse clicks, and the number of keystrokes by a subject during the time interval $t - t_0$. Furthermore, $p(t)$ is a penalty factor that measures the number of times the user switches from mouse to keyboard or vice versa during the interval. These switches account for physical as well as mental user effort. Note that $E(t)$ is a monotonically increasing function.

Mental effort is essentially the amount of brain activity required to complete a task. To some extent, brain activity related to a task can be approximated by processing eye movement data recorded by an eye tracker [6]. Modern eye trackers are non-intrusive cameras that

do not include any parts affixed to the subject's body. Eye trackers acquire eye position data and enable classifying the data into several eye movement types useful for eye related effort assessment. The main types of eye movements are: 1) fixation – eye movement that keeps an eye gaze stable with regard to a stationary target providing visual pictures with high acuity, 2) saccade – very rapid eye movement from one fixation point to another, and 3) pursuit – stabilizes the retina with regard to a moving object of interest [6]. The Human Visual System without dynamically moving targets does usually not exhibit pursuits. Therefore, parameters related to smooth pursuit are not discussed in this paper. In addition to basic eye movement types, eye tracking systems can provide biometric data such as pupil diameter.

Many researchers consider the following metrics as a measure of the cognitive load [11]. Hence, these metrics facilitate the estimation of mental effort.

Average fixation duration: Average fixation duration, measured in milliseconds, indicates cognitive load that can be interpreted as a difficulty in extracting information or as an indication that an interface object is engaging [20].

Average pupil diameter: Eye tracking systems enable measuring biometric data such as pupil diameter. Pupil size, measured in millimeters, can be indicative of the high cognitive effort [20].

Average saccade amplitude: Saccade amplitude, measured in degrees, indicates meaningful cognitive load cues. To a certain extent, large-average saccade amplitude represent lower mental effort. In addition, this metric can be used for accurate estimation of task completion time and physical eye effort [20].

As with the definition of manual effort, the definition of mental effort uses continuous functions that are quantized by converting integrals to sums. Assume that an interactive task *R* starts at time t_0. The mental effort at time t is defined as:

$$E_{eye\_mental}(t) = \frac{1}{t-t_0}\int_{t_0}^{t}\left(w_5 \times fix\_dur(t) + w_6 \times \text{pup\_d}(t) + w_7 \frac{1}{\text{sac\_amp(t)}}\right)dt$$

Where: fix_dur represents fixation duration, pup_d is the pupil diameter and sac_amp represents saccade amplitude. Occasionally, eye tracking devices produce data that is below a reliability threshold. Periods of time that include non-reliable data are excluded from integration.

Ideally, effort expended by the Human Visual System (HVS) to complete a task is represented by the amount of energy spent by HVS muscles during the task. The energy expanded depends on the amount of eye movements, the total eye path traversed and the amount of force exerted by each individual extra-ocular muscle during each eye rotation. These terms are defined below:

Number of saccades: High number of saccades indicates extensive searching, therefore less efficient time allocation to task completion [20]. Increased effort is associated with high saccade levels.

Number of fixations: Due to non-optimal representation, overall fixations relate to less efficient searching [20]. Increased effort is associated with high amounts of fixations.

Total eye path traversed: This metric, measured in degrees, presents the total distance traversed by the eyes between consecutive fixation points during a task. The length of the path traversed by the eye is proportional to the effort expended by the HVS to achieve the goal.

Extra-ocular muscle force: The amount of energy, measured by grams per degrees per second, required for the operation of extra-ocular muscles relates to the amount of force that each muscle applies to the eye globe during fixations and saccades. Based on the Oculomotor Plant Mechanical Model [13], it is possible to extract individual extra-ocular muscle force values from recorded eye position points. The amount of force expanded by each muscle can be summed to calculate the total force.

The total eye physical effort can be approximated by:

$$E_{eye\_physical}(t) = \frac{1}{t-t_0}\int_{t_0}^{t}(w_8 \times fix\_count(t) + w_9 \times sac\_count(t) + w_{10} \times eye\_distance(t) + w_{11} \times extraocular\_force(t))dt$$

Where: fix_count, sac_count, $eye\_distance$, and $extraocular\_force$ represent the total amount of fixations, the total amount of saccades, the total amount of eye distance traversed, and the total amount of force exerted by the extra-ocular muscles respectively. The integration excludes periods of time that include non-reliable data.

2.2 Effort-Based Usability Model

Consider the following example. Assume a set of n subjects selected at random complete a set of k tasks or goals. Further, assume that the subjects are computer literate but unfamiliar with the application under evaluation. For example, the objective of each goal might be to make travel reservations, and each goal requires about the same effort. After the subjects complete all of the goals, an average of the effort (E_{avg}) and the time (T_{avg}) for each goal is calculated. A plot of the average effort (E_{avg}) for each task could produce a graph similar to the one illustrated in Figure 2. Like learning and experience curves, an effort curve is a plot of the expenditure of effort required in order to accomplish a task. The plot is of average effort per task or per time. It is the hypotheses of this research that usability, specifically; operability, learnability, and understandability are inverse functions of effort.

It is possible to view usability from a static and dynamic perspective. Static usability is established when the human interface is designed and does not change with user customization or activity. Under this assumption, it is possible to ignore the "shape" of the curve of $E(t)$, and only use the "final" effort, that is, the accumulated effort at time of completion of tasks. The total effort, E_R, is the sum of the mental effort (E_M) and the physical effort (E_P).

$$E_R = E_M + E_P = E_M + mc + mk + mic + p + e$$

Where $mc, mk, mic,$ and p denote the total number of mouse / keyboard clicks, Mickeys, and switches from (to) mouse to (from) keyboard throughout the process of completing the task R. The total physical eye effort is represented by the term e.

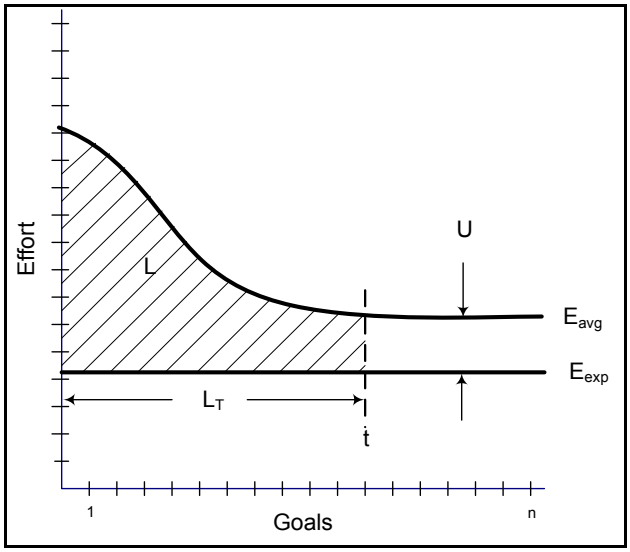

Figure 2 Hypothetical Effort Model

One feature added to the effort model not found in the learning model (see Figure 1) is the notion of expected effort (E_{exp}) or designer effort. At the time of an application's deployment, the people who know the software best are the developers. Therefore, they should expend less effort in completing specific tasks, and provide a point of reference. Thus, the designer expected effort is a single number that represents the "ideal" (with respect to minimum effort) way to interact with the system in order to accomplish a task.

The concept of measuring usability is best illustrated by an example. Let $R(a,b)$ denote the task of making a reservation for a round trip flight from city a to city b. Consider two possible designs ($D^{(1)}$ and $D^{(2)}$) of an interactive system for flight reservations enabling the task $R(a,b)$. Let $E_{exp}^{(1)}$ and $E_{exp}^{(2)}$ denote the designer expected effort for the designs $D^{(1)}$ and $D^{(2)}$ respectively and assume that $E_{exp}^{(1)} < E_{exp}^{(2)}$. Then, per the definition of operability, the operability of design $D^{(1)}$ is better than the operability of design $D^{(2)}$. Alternatively, let $E_{avg}^{(1)}$ and $E_{avg}^{(2)}$ denote the average effort expended by users for completing the task $R(a,b)$ for the designs $D^{(1)}$ and $D^{(2)}$ respectively. Then, again, the system that requires less average user effort is considered to be more operable. Note, that the average can be obtained over different users and / or under different variants of the task

Lack of understandability may result in non-efficient usage of the system or using the system for a task that is different from any task defined at design time. In this case, the user effort may converge to a value that is higher than the designer expected effort (E_{exp}). The difference between the user actual effort (E_{act}) and the designer expected effort (E_{exp}), depicted in Figure 2, may be a useful measure for understandability.

It is possible to measure learnability as the rate of convergence of the average user effort (E_{avg}) to the ideal effort E_{exp}. Alternatively, we can define learnability in terms of the root mean square error. Here the error is the difference between the average user effort (E_{avg}) and the designer expected effort (E_{exp}) at a given task. This metric is actually the area of the difference between the learning curve and the curve formed by the fixed line at $y = E_{exp}$. Figure 2 depicts the learnability (and understandability) curve. Due to understandability deficiencies, it is possible that the user learning curve does not converge to the designer expected effort (E_{exp}). Nevertheless, the subject is said to have "learned" the system where the curve flattens.

3 Experiments

To determine if the notion of effort-based usability evaluation has merit, the usability of two web-based travel reservation systems, called System *A* and System *B*, were used as the target applications in this paper. Twenty subjects volunteered to participate in the experiment, ten subjects for each system. In addition to the 20 subjects, the two facilitators, one of whom was the goal developer, contributed data as well. All of the subjects were undergraduate students at Texas State University, with limited or no background in software development, ranging from 18 to 35 years of age (most of the users were in the lower bound of the range). Based on the student's background and the researchers' understanding of the target market for web-based travel systems, these subjects are almost ideal. For more detailed information about the theory of effort-based usability and this experiment, see the technical report for this research [16].

3.1 Experimental Planning

For this experiment, we employed a planning technique that is a combination between experiment planning and test design. We developed the plan using

basic planning principles with a focus on the resources and tasks necessary to conduct the evaluation.

Determining the number of subjects necessary to conduct the evaluation assumes the first priority because many of the other issues in the plan are based on the size of the subject pool. There is some controversy on the number of subjects necessary for a usability test [5]. Nielsen's recommendations for the number of subjects for logging actual use protocol calls for 20 subjects [18]. A web source, also by Nielsen, suggests six subjects [19]. Some investigators view five subjects as too few and 20 as too many. We have made a compromise and a decision to use 10 subjects with 10 goals to have a statistically relevant number of data points while staying within the constraints of available resources, such as the subject pool or session length.

Two critical equipment and physical resources are required for this experiment: a facility to conduct the test and an eye-tracking device. Both of these are available in our research labs. Using the eye-tracker requires the subject to keep their chin in a fixed position, preventing them from looking around. It appears that this posture also reduces some of the effects of distractions.

3.2 Experimental Design and Execution

A popular method for constructing goals is to "discover" some real world situations and use them as the basis for one or more goals [7, 18, 23]. One of the novel aspects of this research was constructing the goals using a multi-step process to ensure that all differed but based on the same basic scenario.

Designing scenario-based test cases begins with selecting a use case and then injecting an event, constraint, or condition into a use-case. The next step is to develop a test procedure that invokes the situation (event, constraint, or condition) and then the tester records how the software behaves in the situation. If the software meets the expectations set for the situation, the test passes; otherwise, it fails.

For a travel reservation system, there are five (5) possible use cases with the most complex being to book a plane, hotel and car. A scenario based on the use case alone is too simple, adding two conditions related to hotel amenities made the scenario more complex. After refining and testing the model scenario, the specific data (i.e. Name of Traveler, destination, etc.) in the scenario were translated into a blank form, as illustrated in Figure 3. From this blank form, one of the facilitators created and tested the 10 goals used in this research. The tests followed a protocol adapted from formal testing practices [12].

3.3 Evaluation of the Testing Methodology

To paraphrase Glenford Myers [17], a good evaluation is one that finds issues. Using this as a measure of quality of the evaluation, one could say that evaluation of testing methodology was very successful. In a number of instances, subjects ignored certain task constraints. For example, one of the sub-goals of the tasks was to book a hotel room within a certain geographical distance from another hotel. Subjects ignored this constraint because neither System *A* nor System *B* contained a feature which could directly provide this information. It was necessary for subjects to infer the distance between hotels based on each hotel's distance from the destination city. In addition, several subjects complained about distraction from "banner ads" and other content extraneous to the functionality of the travel system. Even though the browser used in the study had "pop-up" windows disabled, there were still a significant number of advertisements presented to the subject. This may be a situation where the additional revenue generated by these distractions may outweigh the impact on usability.

Dr./Ms./Mr. _____ is presenting a paper at the _____ conference being held in _____ at the _____. He/she is presenting his/her paper at 10A.M., but he/she must be there for the opening session at 8:30 A.M. The conference will end at 6P.M. on _____ and Dr./Ms./Mr. _____ must be there for the closing session.

Dr./Ms./Mr. _____ is traveling from _____, and would like a non-stop flight to _____.

The conference is at the _____ hotel on _____ to _____, but Dr./Ms./Mr. _____ feels that this hotel is outside of the range of his/her budget of _____ for the travel. Because of the high cost of the hotel he/she wants to stay at a hotel within _____ miles of the conference center with the following amenities:

1. _____
2. _____
3. _____
4. _____

He/she will need a car to get around at conference city. Again, because of budget constraints, he/she does not want to spend more than _____/day for the car.

Figure 3 Goal Template

4 Experiments Results and Evaluation

The experiment, a usability evaluation of two web-based travel reservation systems, called System *A* and System *B*, provided a great deal of insight into the investigation of the framework. The data acquired for logging actual interaction and eye tracking produced a number of very important results. It was observed that

trend analysis of physical effort expended by the users corresponds to the expected learning curve. It is also observed that the data supports our hypothesis. This is also verified via ANOVA analysis. This paper contains the most significant data. The reader is referred to a technical report for detailed information concerning the research and its findings [16].

Accurate measurement of task completion time is enabled through the eye tracking device, which facilitates measuring the actual time spent on each task. Figure 4 illustrates the average task-completion-time per task per system. The trend for System *B* follows the trends developed by the 1930's research on learning. System *A* has a jittered trend, yet it follows a similar slope. In addition, the task completion time for System *A* is more than twice than the completion times for System *B*. The standard deviation values computed for System *A,* are higher than the standard deviation values of System *B*. System *A* and System *B* implement the same application yet from the data presented in Figure 4, it appears that subjects learn using System *B* faster than System *A* users. Furthermore, the figure demonstrates that System *A* users are less productive than System *B* users. Hence, it is safe to conclude that System *B* is less operable than System *A*.

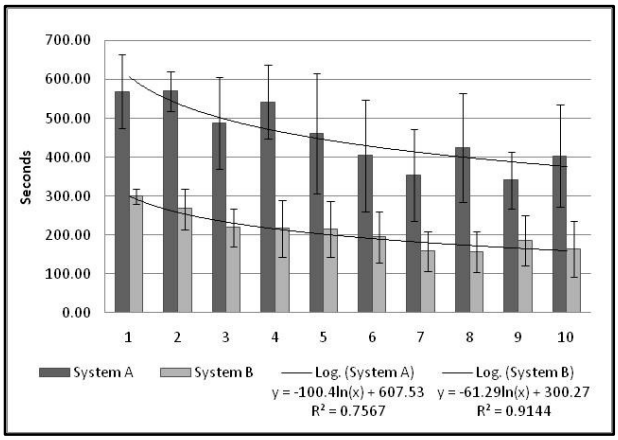

Figure 4. Average Task Completion Time

In this experiment, we use the notion that a learning curve is exhibited by a log-decay curve and a good fit to a log-decay curve indicates that learning occurred. Goodness of fit is determined by the coefficient of determination (R^2), which is the square of the correlation coefficient. An $R^2 \geq 0.7$ indicates a good fit.

4.1 Data Reduction and Analysis

An event driven logging program is devised to obtain details of mouse and keystroke activities from the operating system event queue. The program saves each event along with a time stamp into a file. The logged events are: Mickeys, keystrokes, mouse button clicks, mouse wheel rolling, and mouse wheel clicks. In the reported experiments, the program has generated about 60,000 time stamped events per task (about 10 minutes). The eye tracking system produces a log of time stamped events that includes parameters such as fixation duration, pupil diameter, and saccade amplitude.

A data reduction program applied to the events log, counts the total number of events (e.g., Mickeys) per task. A similar program is used for eye activity events. Both programs execute the entire data set (log of manual activity and eye activity) which consists of several millions of points in less than an hour. With 20 subjects, each completing 10 tasks, the data reduction program generated 200 data points. The data obtained from the data reduction stage is averaged per task per travel reservation system. Hence, a set of 20 points is generated where each point denotes the average count of events per task per reservation system.

Figures 5 and 6 generated from these points are used to evaluate the data, compare the usability of the two systems, and assess the correlation between the obtained data and the research hypothesis. In addition, additional data logged, including the average number of keystrokes, left mouse clicks, and transitions for each task in both reservation systems presents similar shapes and trends.

Figure 5 depicts the average Mickeys per task per system. It is apparent that System *B* requires less mouse activity than System *A*. This is indicating a high correlation in results depicted in Figures 5 and 6 and that System A requires more manual effort. It is evident that System *B* is more operable than System *A* and that the results are in agreement with the hypothesis that usability is related to effort.

Figure 6 depicts approximate eye physical effort by using the product of average saccade amplitude and the number of detected saccades. System *A* required much more physical effort to operate than System B. There was a logarithmic learning trend for System B (R^2=0.82) with saturation point reached after the 5th trial. System A had a less pronounced logarithmic learning trend (R^2=0.62) with a minimum effort point reached during the 9th trial. Like Figure 5, the data illustrated in Figure 6 shows an agreement with the hypothesis that usability relates to effort. Moreover, a spike in activity with respect to task 5 in System *B* can be used as an example of the capability of the metrics to pinpoint potential interface shortfalls.

To further validate the hypothesis we performed an ANOVA and regression analysis in which task completion time was the dependent variable. The goal was to check whether the independent variables such as Mickeys, key strokes etc., correlate to the dependent variable. Task-completion-time is already an acceptable measure of usability whereas the independent variables are assumed to be indicative of effort only in our hypothesis. The ANOVA and regression results indicate that 83.0% of the variability in the task-duration-time is explained by the independent variables: Mickeys, Clicks, and the number of Repetitions of the same experiment.

Note that the repetition of the same experiment is strongly correlated to learning. Consequently, about 11% of the variability in the observed durations of the tasks is due to the "subject". These are very significant results showing that the hypothesis of this research cannot be nullified.

Like all good experiments, this study answered a number of questions about the relationship between user productivity and effort, but it left some questions only partially answered and opened a number of new questions. One of the questions partially answered is how to convert usage counts to effort metrics.

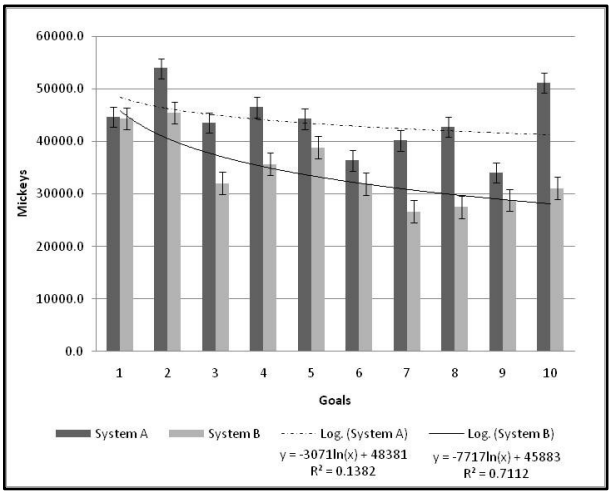

Figure 5 Average Mickeys

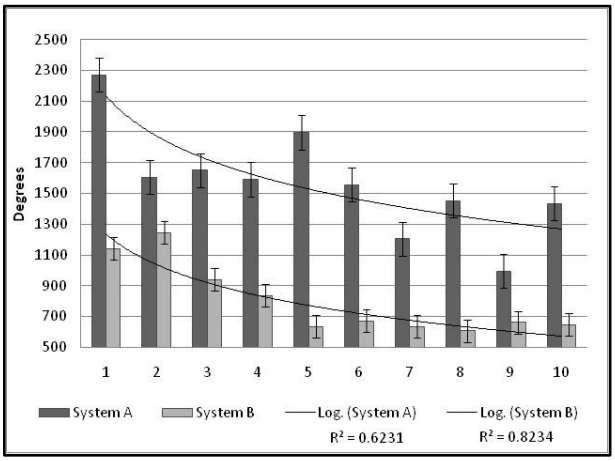

Figure 6. Approximate Eye physical effort.

5 Conclusions

The research results support the following important observations:

1. The research illustrates that logging of interaction events such as Mickeys, mouse button clicks, and keystrokes, along with eye tracking, provides a great deal of useful information. There is a clear correlation between the effort approximation model presented and usability. This correlation can be exploited and used to evaluate the usability of existing user interfaces as well as user interfaces that are at a relatively advanced stage of design. Nevertheless, further experimentation is required in order to refine the hypothesis, the model, and the effort approximation procedures that are guiding the research.
2. The research shows that logging and processing interaction events is feasible and useful. Previously it was stipulated that the volume of data obtained through logging of interaction events is un-reducible and therefore useless. This research indicates, however, that this may no longer be a problem.
3. An important contribution of this research is that it can enable pinpointing GUI design defects and implementation shortfalls. For example, consider a goal, which yields an excessive amount of effort in a given test (or set of tests). Indeed the spike in tests 5 depicted in figure 6 could prompt an investigation of the application execution log in order to identify the root cause of the increase in effort.
4. Careful design of the tests performed enables obtaining high quality results despite working with very limited resources and no funding.
5. One of the important aspects of the usability testing strategy is the utilization of a use-case scenario based test design technique. This technique is instrumental in facilitating the usage of appropriate goals and test procedures. Moreover, it is an important component of the ability of the proposed effort based metrics to pinpoint design and implementation shortfalls.
6. While the manual based interaction activities (mouse, keyboard, gloves, etc.) are strictly related to physical effort, the eye movement data is related to both physical and mental effort. On one hand, it can be utilized for enhancing the physical effort model. On the other hand, it is currently the only type of data correlating with mental effort. Hence, the research opens the door for a layered approach to GUI usability testing. At the lower layer, only manual data is recorded and used for fast and relatively inexpensive usability evaluation. At the next layer, eye tracking devices provides a means for mental effort evaluation and refinement of the physical effort approximation techniques. A potential future research relates to the utilization of brain wave measurements to further enhance the mental effort evaluation procedures.

6 Future Research

Usability is a huge and important area of research and development and one paper or research effort cannot cover the multitude of relevant issues. Several of these

issues, which will be addressed in future research, include:

Further investigation into scenario-based test design techniques appears warranted, based on the results from the current experiment. With additional test cases and an improved test case design technique, it may be possible to shed more light on the usability model and its utility as well as to reduce unknowns such as the influence of fatigue. In fact, additional research which is in progress includes a set of experiments to assess the usability of individual GUI widgets and their combinations.

This paper treats every metric individually and demonstrates that the hypothesis of the research is established. A more elaborate hypothesis of the research, however, includes an assumption that it is possible to derive a procedure to combine the individual metrics into a single approximation of the effort $E(t)$ and correlate this approximation with traditional measures of operability, learnability and understandability. Further research is required to determine whether it is possible to reduce the individual metrics into one measure that approximate usability.

7 References

[1] "ISO/IEC 9126-1:2001 Software Engineering-Product Quality-Part 1: Quality Model," International Standards Organization, Geneva Switzerland2001.

[2] "ISO/IEC 9126-1:2001 Software Engineering-Product Quality-Part 2: External Metrics," International Standards Organization, Geneva Switzerland2001.

[3] "Experience Curve Effects," http://en.wikipedia.org/wiki/Experience_curve_effects, date retrived:

[4] Boehm, B. and et_al, *Characteristics of Software Quality*. New York: American Elsevier, 1978.

[5] Caulton, D. A., "Relaxing the homogeneity assumption in usability testing," *Behavior & Information Technology,* vol. 20, p. 7, 2001.

[6] Duchowski, A., *Eye Tracking Methodology: Theory and Practice*, 2nd ed.: Springer, 2007.

[7] Dumas, J. S. and Redish, J. C., *A Practical Guide to Usability Testing*. Portland, OR, USA: Intellect Books, 1999.

[8] Ebbinghaus, H., "Memory: A Contribution to Experimental Psychology," 1885.

[9] Gibbs, W. W., "Software's Chronic Crisis," *Scientific American,* vol. September, 1994.

[10] Grady, R., *Practical Software Metrics for Project Management and Process Improvement*: Prentice-Hall, 1992.

[11] Just, M. A. and Carpenter, P. A., "Eye Fixation and Cognitive Processes," *Cognitive Psychology,* vol. 8, pp. 441-480, 1976.

[12] Kit, E., *Software Testing in the Real World*. Reading, MA: Addison-Wesley, 1995.

[13] Komogortsev, O. V. and Khan, J., "Eye Movement Prediction by Oculomotor Plant Kalman Filter with Brainstem Control," *Journal of Control Theory and Applications,* vol. 7, 2009.

[14] Leveson, N. and Turner, C. S., "An Investigation of the Therac-25 Accident," *IEEE Computer,* vol. 26 no. 7, 1993.

[15] McCall, J. A., Richards, P. K., and Walters, G. F., "Factors in Software Quality," Nat'l Tech. Information Service, 1977.

[16] Mueller, C. J., Tamir, D., Komogortsev, O. V., and Feldman, L., "An Effort-Based Approach to Measuring Usability," Texas State University-San Marcos TXSTATE-CS-TR-2008-9, November 2008. http://ecommons.txstate.edu/cscitrep/7

[17] Myers, G., *The Art of Software Testing*. New York, NY: John Wiley & Sons, 1979.

[18] Nielsen, J., *Usability Engineering*. San Francisco, CA, USA: Academic Press, 1993.

[19] Nielsen, J., "Logging Actual Use," http://www.usabilityhome.com/FramedLi.htm?Logging.htm, date retrived: December, 2008.

[20] Poole, A. and Ball, L. J., "Eye Tracking in Human-Computer Interaction and Usability Research: Current Status and Future Prospects.," in *Encyclopedia of Human Computer Interaction*: Idea Group, 2004.

[21] Pressman, R., *Software Engineering: A Practitioner's Approach*, 6th ed. New York, NY.: McGraw-Hill, 2005.

[22] RTI, "Planning Report 02-3: The Economic Impacts of Inadiquate Infrastructure for Software Testing," National Institute of Standards: Program Office: Strategic Planning and Ecconomic Analysis Group.2002.

[23] Rubin, J. and Chisnell, D., *Handbook of Usability Testing: How to Plan , Design, and Conduct Effective Tests*. Indianapolis, IN, USA: Wiley Publishing, Inc., 2008.

[24] Tullis, T. and Albert, B., *Measuring The User Experience: collecting, analyzing, and presenting usability metrics*. Burlington, MA: Morgan Kaufmann, 2008.