Information System Evaluation Case Studies:  
On-Line Newspaper Archive System  
WebMarket System  

Nathan Stark, Sameh Boutros, Justin Fidler,  
Shoaib Mian, William Moher, Sang Hyun Park,  
Edwin Rodriguez, Andrew Rosenbaum  

Abstract  

What makes information systems successful? As developers, we design systems based upon the needs of clients and the users. The system is successful if it not only meets the needs of the users, but is used effectively and efficiently by the users. This paper illustrates the importance of evaluating information systems and technology.  

In order for developers to look at the success of a system, it must be evaluated and tested. We have defined and outlined a process for system evaluation and usability testing. Two systems developed by information and decision systems students at Carnegie Mellon University have been evaluated using this process. Both system evaluations provide a contrasting framework for illustrating the importance of such testing.  

Whether developing a system for school newspaper staff or for an entire university, the success lies within the use of the system. In the two evaluations, the adaptability and acceptability of the new systems were the biggest barriers to overcome. The evaluation process allows developers to assess the success of the system and to pinpoint limitations of the system. Upon pinpointing the limitations, developers may be able to modify the system to meet the needs for creating a successful system. System evaluation provides developers this ability.  

Introduction  

To fully understand system evaluation, we have provided background information outlining the methods used for evaluation. Two case studies that evaluate systems developed at Carnegie Mellon University illustrate these methods. A discussion follows that compares the methods and conclusions that can be drawn from both cases.  

The background information for system evaluation includes defining evaluation, suggesting reasons for evaluating systems, and outlining the
methods. With this background information we can illustrate these methods through two case studies.

The Online Newspaper Archive (ONA) system was developed to aid editors of a university newspaper in compiling and editing work for the weekly paper edition. A case study illustrates the implementation and evaluation of this system.

The WebMarket System (WMS) was designed to provide a forum for a university community to buy and sell items. This study also illustrates implementation and system evaluation.

User acceptability and adaptability proved to be the biggest barrier that both systems had to overcome. Although the systems had different user goals and needs, the studies illustrate how the evaluation methods provide useful analysis and feedback. The case studies also provide useful insight into the fallacies of system implementation.

System Evaluation and Usability Testing

Technology is changing the face of daily life throughout the world. Banking, planning a trip, or sending a message to a friend is now possible by interacting with a computer system. But having technology make it possible to pay a credit card bill on a computer, does not necessarily mean that it is preferred to writing a check and sending it. This preference of technology is the premise upon which system evaluation and usability testing is based. By evaluating the current technology it may be possible to find out what can be improved to make life easier for the average user.

Defining System Evaluation and Usability Testing

When a patient computer buyer is looking to buy a laptop, she must evaluate the different brands. The speed of running a word processor and ease of typing are the most important aspects to this consumer. She will consider the documented speed of each computer. She will also try the keyboard to see how easy each is to use. Upon looking at the computers, she picks the computer that best suits her need for word-processing and typing.

This computer buyer is performing an abbreviated version of system evaluation and usability testing. System evaluation is the process of testing to see how the technology matches the user's goals. In this case the user, our consumer, had goals of performance and ease of use. Usability can be defined as how effectively and easily a system meets the users goals. The consumer here will have tried each of the keyboards, and will have a sense of how easy each keyboard is to use. Upon testing and evaluating the system, the user will decide if it is worthwhile to purchase or continue using. This raises interesting implications for developers of technology.
Information System Evaluation Case Studies

Purpose of Evaluation

System evaluation is performed for a number of reasons: comparison of systems, extendibility, and usability. System comparison is necessary for developers or producers of one system to be able to pinpoint the benefits and limitation of both systems. Sometimes technology is complex and users are scared to use or adapt to a new system. A video phone is an example of an innovative technology to which consumers did not adapt. Unfortunately, someone has to use the technology for it to be worthwhile to the producer of the technology. By evaluating and testing the usability of systems it is possible to evaluate how systems match the needs of the users. After evaluating the system, it may be possible for developers to make improvements on the current system. Certain methods exist to allow developers to evaluate systems in detail.

Methods for System Evaluation and Usability Testing

When a professor sits down at the end of the semester, she must assign grades to all the students. At the beginning of the year, she created a syllabus for the course. In order to grade fairly and objectively, she set forth certain requirements for her course. Attendance, assignments, and reading were all required for her class. She kept track of attendance, scores on assignments, and created quizzes for the students throughout the year. By combining the attendance and scores of each student she was able to assign a grade to each student. This same method is used in system evaluation and usability testing.

The first step in system evaluation is creating a plan for testing, just like our professor's syllabus. This plan consists of setting usability goals, identifying what aspects of the system should be evaluated, selecting measures for evaluation, and selecting methods to collect the data. Usability goals are simply what the users expectations are. After deciding which goals the system needs to meet, the next step is picking the parts of the system that are directly involved with the user's goals. Concrete measures for testing these aspects of the system need to be outlined, along with steps that need to be taken to test the system.

Upon completing a plan for usability testing, the next step is to perform the tests and analyze the data. The professor will combine the scores and calculate a final raw score. She will then look at the grade distribution across her students, and will then assign a grade to each student. System evaluation follows the same method. Upon compiling the data, analysis needs to be made. By comparing the results with the original goals set out by the system, it is possible to make recommendations and conclusions about the system. The best way to understand how system evaluation is performed is to look at two examples from the planning stage through to the recommendations.
Introduction to the Case Studies

The following two case studies illustrate system evaluation in detail. Each study is based upon evaluating a system developed by junior level students in an information systems project course at Carnegie Mellon. The differences in each system provide a useful comparison in methods of evaluation.

The first case presented is based upon the On-line Newspaper Archive (ONA) system. This system was developed to replace a manual process used by editors to compile the weekly newspaper for Carnegie Mellon. The staff of the newspaper make up the user base. The staff encompass a small number of users, thus surveys and interviews were used extensively to evaluate the system. Unfortunately, the system was not adopted by the staff of the newspaper. The evaluation case, written by the developers of ONA, illustrates how the surveys, interviews, and interaction were used to evaluate the adaptability, usability, and functionality of ONA.

The second case presented is based upon the WebMarket System (WMS.) This system was developed to replace an email based system as a forum for the buying and selling of items. An entire university community make up the base of users of this system. With the large and varied amount of users, the following were used to evaluate the system: usage statistics, surveys, and a task analysis (GOMS analysis.) This system was not readily accepted as a new alternative to the email based system. This case, written by the developers of the WMS, illustrates how usage statistics, surveys, and task analysis can be used to determine the acceptability, usability, and functionality of the WMS.

A few differences in systems presented here provide some interesting considerations developers should look at when implementing a system. The ONA system was developed for a small, well defined organization. The WMS was developed for a large heterogeneous community. The ONA system replaced a manual system, as the WMS replaced another computer based system. The size and user set provide different barriers for successful implementation. It is interesting to note that both systems ran into acceptability problems during the course of the five week evaluation process. With these differences in mind let us look at each of the cases in more detail.

Online Newspaper Archive (ONA) Case Study

Evaluation Summary

The newspaper industry has been slow to grasp the new technologies and opportunities of the World Wide Web (WWW). A possible reason is that the newspaper publishing industry runs such tight deadlines that any technological solution has the potential for disaster in the case of system failure. Thus, newspaper organizations are apprehensive when any change
to their production process is introduced. The majority of the nation’s largest newspapers (The Washington Post, The New York Times, etc.) have made their newspaper available on the WWW only within the last few months.

In the case of smaller, weekly newspapers, production time is an important factor. These newspapers typically employ a smaller staff, so their lack of resources makes every time-consuming task resource-intensive. However, these smaller newspapers can benefit from the WWW as much as larger newspapers. On-line newspapers often increase their readership base, particularly in locations where they do not otherwise make the print edition available.

Because of the demands of these small newspapers, we developed a WWW-based newspaper publishing system that seeks to automate certain parts of the newspaper publishing process, while making the newspaper available, seamlessly, on the Web.

We developed the web-based newspaper publishing system, On-line Newspaper Archive (ONA) as part of a project in Systems Analysis and Design, a course offered as part of the Information and Decision Systems program at CMU. ONA is a real-time small newspaper production environment that allows writers and editors to collaborate in the newspaper production process using a WWW interface. ONA also provides the newspaper with a WWW version of the newspaper which requires no additional effort to layout or maintain.

The purpose of this evaluation was to see how a system like ONA would impact a small newspaper like Carnegie Mellon’s student newspaper, The Tartan. The Tartan currently produces its weekly newspaper in traditional file-based transactions where articles, be it paper, or on disk, are exchanged between the writers and the editors, then passed on to the production staff. ONA streamlines the Tartan’s production process by eliminating the file transactions and replacing them with database transactions through a web browser. Our research team wanted to see how ONA could help a newspaper become more efficient, as well as provide the newspaper with a presence on the WWW for its readers.

Our conclusions from this evaluation were not as we would have initially predicted. Over the course of the semester, we found that although ONA’s methodologies for production are more efficient than the previous process, the newspaper staff was resistant to learn and use ONA as part of their work routine. We recommend that the Tartan staff spend more time and effort to seriously consider ONA as an alternative to their existing process.

System Background

Current System

The Tartan’s current system for newspaper production is a file-based system. The process starts with writers writing an article. Once completed,
The writers can turn their articles over to their editor in one of several ways. They can walk the disk or hard-copy to the Tartan office, or they can drop the file off electronically on the Tartan’s Macintosh fileserver. Once the article has been submitted, editors have to look for it and commence the editing process. If editors wish, they can send the article back to the writer for rework. The editors can send it back using either of the two methods described above. Once editors have an article that is ready for the issue, they save it in an electronic format until the newspaper is ready to be laid out in PageMaker. When the time comes to place the articles into PageMaker, the layout staff opens each file and copies the contents and pastes it into the appropriate location in PageMaker. At this point, minor layout editing is done to meet the needs of each specific page.

ONA deals with this process in a simpler and more efficient fashion. As described in the functional requirements below, ONA allows writer to submit their articles using a web browser on any platform. Editors can find and edit these articles using a web browser as well, freeing them from traveling to the Tartan office or finding a specific computer platform. This location independence encourages the Tartan staff to spend more time working on the newspaper, and less time in transport to and from the Tartan office.

**Functional Requirements**

- Distinguish and authenticate editors and writers.
  This will ensure that only Tartan staff members have access to the production aspect of the system and that only editors have access to the editing of articles.
- Accept data (in text or image form) from a writer or editor.
  The system accepts writers’ submissions through an easy-to-use WWW interface.
- If it is accepted from a writer, it must be made available for an editor to prepare for placement into the newspaper.
  When editors log in, they have an option to view all submitted articles for their section of the newspaper, for the upcoming issue.
- Once an article is resubmitted by an editor, the keywords are archived along with the article.
  The system stores information about each article (metadata) that is used when users choose to perform a search on the system. If this information was not entered by the writer, it must be provided by the editor.
- Store the data in the system.
  All incoming articles are automatically stored and indexed by the system, in a central database.
Information System Evaluation Case Studies

- Make the data viewable, in a limited number of formats, to writers, editors, and readers. The data must be easily retrieved using the search system.

The articles may be retrieved through the WWW, from the database.

Implementation Procedure

After the system had been tested and prepared for use, the remaining tasks were to create accounts for the production users and explain to the production staff how to use the system. The system has provisions for users to register themselves but we wanted to eliminate as much cost as we could so that the users would try our system. ONA was considerably different from the current processes; we did expect it to be met with some resistance. After the accounts were ready, we emailed all the staff members with the specific information that they would need in order to use the system.

During the implementation process, we attended meetings with senior Tartan staff members, where we described our system and obtained approval to complete the implementation. These staff members indicated that they could see the advantages of the system, and were willing to have their writers use the system.

Our proposal to the Tartan staff was to run the system for a few weeks in a parallel run so that we could trap any unexpected errors or bugs while the system was in actual use. To facilitate learning from mistakes/bugs in the system, we did the initial parallel run with one section of the Tartan the first week, the Forum section (one editor and nine writers). This yielded four articles in the actual Tartan, two of which used ONA. The following week we ran the parallel run with the Forum section as well as the News section, which yielded 7 articles.

Usability and System Evaluation Plan

This section is a detailed description of the methods and goals of our system evaluation for ONA. We had hoped to conduct extensive research in this area, but our subjects proved to be less than cooperative. Many surveys distributed were ignored, as were repeated attempts at communication by phone and through email. Attempts to research why this occurred were also unsuccessful. Despite these setbacks, however, we were able to obtain a minimal amount of information necessary for our system evaluation. This section is broken into four parts including the usability goals, the tasks we selected for evaluation, our measures of evaluation, and our methods of data collection.
Usability Goals

Our initial goals in evaluating this system were fairly typical. We had hoped to determine how well our system was adopted by the Tartan staff. As a result, we were looking to see whether or not the functionality of our system, as described in our Functional Requirements, was compatible with our users. As time progressed however, we were forced to alter the focus of our evaluation. Due to lack of use from the users, we realized that the staff tended to disregard our system, with a few exceptions: newer staff members were more likely to use ONA, and experienced Web users were also more likely to use our system. In response to this weak reception of ONA, we altered our evaluation to focus on why our system was not being used. We needed to determine whether it was a result of our failure to meet the users' needs and requirements, or because of some internal culture that was resistant to change.

Tasks Selected for Evaluation

In the evaluation of our system, certain tasks were selected and analyzed very closely. One function of our system that was left out of the evaluation was the search time for articles. Since there was no data prior to our implementation, the search function would be useless until the articles had been accumulated in ONA. However, the main tasks of our system were usable and we were enthusiastic about their evaluation.

1) Writer submission time: Because our system altered the way writers would be submitting their articles to the editors, we needed to evaluate how much more/less time it took them to do so. This would allow us to determine whether there was a trade off in making the editing process easier by forcing the writers to submit over the WWW.

2) Article retrieval time: By streamlining the editing process, we hoped that the editors would spend less time looking for articles if they already knew where to find them.

3) Editing time: By changing the way editors received their articles, we also altered the way they would be able to edit them. They would now have to cut and paste from the web and then back to it. We needed to see whether there was an increase in editing time needed to complete these procedures. We were also looking to see if there was an increase in productivity as a result of the new system.

4) Accessibility: By moving the entire process to the WWW, we hoped that our system would increase the accessibility of the
Information System Evaluation Case Studies

articles both for the staff members and the public. By doing so, we hoped to see an increase in productive time for the Tartan Staff.

**Measures of Evaluation**

Our methods of evaluation were primarily based on measures of time. We looked at the time it took the users to complete some of the tasks discussed in the previous section. We did not focus heavily on the actual usability of the interface itself, instead focusing on the time difference realized (over the manual process) from use of the system.

**Methods for Data Collection**

In order to conduct our system evaluation, we had to decide on which evaluation methods to use. Initially, we began with surveys distributed to the users. However, after some time, we were met with a serious lack of enthusiasm on the part of the users. Therefore, we were forced to resort to the more direct method, personal interviews.

**Survey 1**

This survey gave us estimates of the number of hours that various types of staff members spent in preparing the newspaper. We were able to get breakdowns of the time to edit, to write, and to search for old articles. Surveys were administered to Tartan staff who had been working for 1 or more years. This gave us a sample of those individuals who have had experience with the old system and could comprehensively compare it to ONA. We sampled work which takes place during peak work periods (the weekend before an edition is due) and work which occurred during non-peak periods. Our population sample included writers, editors and those on the layout staff. Because many members of the Tartan hold multiple positions, they were allowed to respond from any or all of the above positions. In general terms, our measurements included the following:

**Time:** We were looking to establish exact amounts of time in terms of hours spent on specific tasks where ONA improved upon the current Tartan process. Editors were surveyed on how much time they spent editing an article (on average), the size of the article (on average), and the amount of time that they spent at the Tartan office during the past week. We also measured the time it takes an editor to find the articles that have been submitted by their writers. We also measured the amount of time that is lost when an editor has to return an article to the writer so that it can be changed.

Writers were surveyed on the length of time from when they finish writing an article until they turn it in, and how much time it takes for them
to write and research an article. Editors and Writers were both surveyed on how much time they spend looking for past Tartan articles.

We were looking for the productivity of the staff using these numbers. A decrease in time spent while maintaining or improving the quality of the newspaper showed an increase in productivity.

**Numbers:** Editors were surveyed on how many articles they edit in terms of articles per week, as well as the size of these articles. They were also be asked how many writers they edit for. Writers were surveyed on how many articles they write per edition.

**Difficulty/Efficiency Rating:** Editors were asked to scale several processes from 1-5 (1 being easy/not efficient, 5 being difficult/very efficient). These processes included those which our ONA system hopefully influenced to the better. This gave us a specific number to compare between surveys.

**Open-Ended Question:** There was an open-ended question, which allowed us to receive unrestrained feedback from the users as to improvements or problems they have with the current system or with ONA. This question was directed towards soliciting feedback about problems and improvements.

**Survey 2**

This survey was distributed to editors throughout the week, for a period of four weeks. It was our hope that they would fill out one of these surveys for each article that they edited.

The purpose of this survey was to determine the length of time that the editing process took. We asked several questions including time to find the article, time of submission by the writer, time the editing started and finished, and number of times edited.

**Interview**

For this interview, our population consisted of those writers who were asked to submit their articles using ONA during the parallel run. These interviews were conducted over the telephone.

**Time:** Writers were asked how many hours they spend on the World Wide Web per week, and how long they wait after completing an article before they turn it in. The latter measure was both typically and for the specific issue where they used ONA.

**Location:** Writers were asked whether they live on or off campus, and from which computer cluster they submitted their article.

Writers were asked whether they found it necessary to rewrite articles after submission. They were given the choices, once per article, twice, three times, and more than three times. We neglected to offer a choice of zero times, but some writers simply wrote it in.
Another question posed to the writers was a simple yes/no question about whether they found it easier to use ONA or the old system. Finally, writers were given the opportunity to contribute suggestions and comments regarding ONA and possible improvements.

Data Collection

The various methods of data collection were described in the previous section. This section will describe the actual method of administration.

Survey 1

Survey 1 was administered prior to ONA’s implementation. It was administered to random writers and editors throughout the Tartan staff. A total of 17 of 28 surveys were received. This survey was valuable in providing benchmark times for our measures. It provided us with information on submission time, retrieval time, editing time, and accessibility factors.

Survey 2

Survey 2 was administered primarily to editors and ONA users, for two weeks prior to system implementation, and two weeks after implementation. A total of 6 of 20 surveys were received. This survey gave us information about the editing process, including data on how often articles are sent back to the writer for rework, and how long it takes an editor to find an article. This survey was closely focused on determining retrieval time and editing time.

Interview

The interview was administered only to ONA users who participated in the parallel run, during the final weeks of system implementation. We chose to perform this evaluation after users had worked with ONA for a few weeks. A total of 3 out of the 4 interviews were performed. It provided valuable data about the problems and advantages of ONA, as well as data on editing and submission times.

Data Analysis

In total, we received 17 complete responses to Survey 1, 6 to Survey 2, and 3 Interviews were conducted. Despite the decline in the response rate, we will attempt to analyze the data that was collected.
Functional Requirements Analysis

A surprisingly large number of people in the first survey gave the current system a fairly low rating. Eleven editors filled out the survey and of those eleven, 8 rated the current system as being a 3 or less on a scale of 1-5 (1 being very inefficient, 5 being very efficient). Many of the complaints were stemmed from spending too much time editing articles. One editor gave the current process a four but questioned its efficiency as the newspaper became more computer oriented. Overall, those surveyed were quite adamant about spending too much time at the Tartan and the process being inefficient. The average amount of time spent at the Tartan by the eleven editors was 17 hours per person per week. This included four significant at 7, 12, 13 and 15 hours. Otherwise the average jumps to about 23 hours. Writers spend significantly less time at the Tartan office, with an average of about 4 hours per writer per week. But this could be because they are able to do their writing from any word processor and don’t necessarily have to be at the Tartan to work.

We received a much smaller response to the second survey, which tested the current process. This probably stemmed from the fact that people were working and didn’t want to be troubled with filling out the survey. However, we did receive 6 surveys. It took an average of 2.4 minutes for editors to find the article they wanted to work on, while the average editing time per article was approximately two hours. The average number of times an article was edited was 4 times.

Usability Requirements Analysis

The three interviews conducted tested the writer’s response to their use of ONA. Of the three interviewed, one did not use the system. Their average time spent on the WWW was 1 hour. There were many similarities in their responses to our questions. All three:

1) Submit one article per issue
2) Turn in their articles from a campus computer cluster
3) Never need to rewrite their articles after submission.

Average time spent between finishing the writing of the article and handing it in to their editor was less than one-half hour. It was interesting that both interviewed subjects who used ONA found the process easier than the old method. However, the response given by the one subject who did not use ONA was that he/she hated computers and was actually very emphatic about this opinion.

We actually anticipated that there would be a cost to the writers in the difficulty of handing in articles. It was refreshing to get such a positive response, even if it was only from two users. However, the ease of use and
extended functionality of the system would not be sufficient to encourage users to switch processes.

Recommendations and Conclusions

Based on the data we gathered, as well as our observations, we can make certain recommendations about implementing systems in general, as well as The Tartan specifically.

System Status

Based on our observations, we can conclude that ONA was not as popular as expected for a number of reasons. We found that users typically had trouble forming a mental model of what ONA was, and how they should use it. Few writers understood that ONA allowed their editors to edit the articles on-line, or that the system would automatically publish their articles on the Web.

In addition, the organizational limitations proved to be insurmountable. The Tartan practices a very loose style of management, and because of this, it was hard to convince any writers to use the system, even after receiving a mandate from their managing editor.

It also seems that writers and editors are not enthusiastic about using computers. Aside from their word-processing or layout capabilities, they do not find much use for them otherwise. Many of the editors had become set in their way of handling things and change was not seen as a very favorable thing to them. Despite the common thought that they spend too much time at the Tartan, it seems that they have grown used to it and maybe even enjoy it (during our observation at the Tartan office, we noticed that there was certainly a social value to being at the Tartan office).

Limitations of System

Our findings reveal that the system was not difficult to use, particularly for the more novice user, writers. However, users had a severe problem understanding exactly what the system was supposed to do, and this confusion discouraged users from using the system.

Future Enhancements

Unfortunately, we will not continue the implementation of this system. The organizational and operational constraints will take too much manpower to overcome. With systems like ONA, that change many aspects of an overall process, it is hard to implement without a parallel run. However,
during parallel runs, users are still exposed to the “old system” of doing things, and often prefer it because they are more familiar with it. If we had the opportunity to start with an entirely new set of staff, who used only ONA, the system might be considerably more popular. Unfortunately, this was not an option during our evaluation period. Additionally, the Tartan never has a full turnover of staff, preferring that their staff be comprised of no more than 50% new staff members.

**WebMarket System Case Study**

**Evaluation Summary**

Every day hundreds of college students need to find ways to get money fast. Many have nick nacks they need to sell, which will provide a little money for survival. What is needed is a virtual market for the college community. Computers are the cutting edge of information interchange in an academia setting, thus it only fitting that they play an integral role in providing students with the ability to have a market. Carnegie Mellon University provides its community with vast computer resources including mail readers, newsgroups, and web browsers to use for exchanging information. Cmu.misc.market is a newsgroup that provides the community with a service to advertise items. This service, however, is quite limited in its ability to provide a full range of services that a virtual market could provide.

Because of the demands for a more functional market system, we developed and implemented the WebMarket System as an alternative method for students at Carnegie Mellon University. This system provides a web based solution that adds functionality to the cmu.misc.market. This evaluation illustrates a comparative analysis between two alternatives, a web based system and the electronic mail based system. The conclusions to be drawn from this analysis tend to point out the barriers of acceptability of the WMS. The system provided a usable and functional alternative, but was not however quickly adapted to by the Carnegie Mellon community. By planing further for users adaptability needs, the system may have been more successful.

**System Background**

**Current System**

The current electronic market system available to the Carnegie Mellon University community is cmu.misc.market, a newsgroup based system. Any user of the Andrew system and outside users are allowed to use the cmu.misc.market. The interface between users and the market is an email
Information System Evaluation Case Studies

based reader. Ezmail and MacMail are the two main interfaces used at Carnegie Mellon for accessing the market. This interface allows users to "post", sending a message, to the market. This message consists of a date, subject, from address, to address, and body. A user wishing to sell an item on the system, will create a new message that includes the name of the item that is usually placed in the subject line, and a description in the body of the message. Upon posting an item, the message appears on the interface with the date, subject, and from address. To view the body of the message, users must select the message and choose a read option. All messages are ordered by date and can only be deleted by the system administrator. The messages are usually deleted by age; the oldest ones being deleted first.

The WebMarket System provides users with a system geared specifically towards the buying and selling of items. When buying or selling items, users needs extend beyond what a newsgroup system can provide. Each user may want to maintain his/her own items by modifying the information of the items, or by deleting the item from system. Users may want the ability to search for items by use of keywords. When shopping, users may wish to use a shopping cart to store the interesting items while browsing through the market. The functionality of the WMS is formalized below.

Functional Requirements

The WebMarket System provides the following capabilities for its users:

• System Login
  The login function provides user authentication and allows users to maintain secure communications with the server.

• Registration
  Registration allows users to receive a password for the system and supply the system with demographic information about themselves.

• Browse
  The browse function allows users to view the items in the system broken down by category.

• Post Item
  This function allows users to enter and categorize for-sale or want items into the system.

• Modify Item
  Modify item allows users to change information about the items which they have previously posted for sale. With this they may change the price, the description, or the category the item is placed in.

• Simple Search
  The simple search allows users to search the items in the system by keyword.
Detailed Search
The detailed search provides the facilities to search the items in the system by search type (For Sale Items, Wanted Items, Both), keyword and strength of search (match whole word, match substring), price, and date of post. Additionally, detailed search allows users to determine the formatting of the return from the search (sort by date, price, title and separate by category).

Shopping Cart
This function provides a temporary storage facility so that users may browse items in the system and store those which they are interested in. By doing this, users can have easy access to interesting items so that they can decide whether or not to contact an items owner after they have browsed all items.

Post Comment
The post comment function allows users to write feedback to the administrator of the system.

Change Password
This function gives users the option to change the random password they were assigned by the system.

Help
The help facility provides convenient on-line help for the functions of the system.

Implementation Procedure
The WMS was originally developed in the spring of 1996. In order to implement the system in the fall of 1996, a few modifications had to be made. An implementation plan was created to formalize the modifications and installation procedures for the system. Additionally, a marketing plan was developed to initiate interest in the system.

The implementation plan consisted of changing the database backend of the system to a more stable environment, modifying functionality to adhere better to the new database backend, and creating both a development and production platform. The original WMS was developed using Access 2.0 as a backend. This was reimplemented using SQL Server to better suit a multi-user environment.

Any system that is developed is not perfect. When users begin to use a system, they inherently run into problems with the system. In order to prepare for this, a fixing priority plan was developed. This plan basically outlines the priority in which problems should be fixed. To further aid in solving problems, two mirror image systems were initiated, a development and a production system. The development system allows for changes and testing to be done on the functionality of the system. Upon fixing any problems with the functionality, the new code is then transferred to the production system. The production system is the live system that the users
interact with. Separating the development and the production systems allows for a few benefits. If a major problem occurs, the functionality can be fixed on the development machine without the users engaging in any down time. Additionally, a direct back-up is available in case the production machine goes down. This provided a more reliable means of system maintenance.

Without buyers and sellers, a system would be useless. A marketing plan was developed to advertise the system to the Carnegie Mellon community. By advertising the system, we expected to gain users to the system. This plan consisted of timing the creations and distribution of flyers, electronic announcements, and a newspaper advertisement.

Usability and System Evaluation Plan

This section provides an in-depth discussion of the process used in evaluating the WMS. The usability plan provided a framework for our benchmark comparison with the cmu.misc.market. We decided on three major aspects of our system to analyze: functionality, usability, and acceptability. In order to evaluate these topics, we outlined measures and methods for data collection. With the following plan in place, we were able to collect, analyze, and assess data to evaluate the WMS.

Usability Goals

The goal of the WMS is to offer a web based forum for the exchange of goods with extended functionality and improved user interaction from the current news based cmu.misc.market system. Our main intention was to provide Carnegie Mellon with a easy and useful market system that may potentially take over as the main market system. In order to study this goal in detail we evaluated the functionality, acceptance, and usability of the WMS. In order to evaluate the functionality of the system, we looked at the types of changes that were made to the system during implementation. We analyzed the acceptance, or user adaptation, of the WMS to see how the Carnegie Mellon community adapted to the system. Additionally, we looked at our marketing strategy, and evaluated how effectively the Carnegie Mellon community was induced into using the new system. By looking at usability, we evaluated the ease and effectiveness of using the WebMarket system in comparison with the cmu.misc.market.

Tasks Selected for Evaluation

In order to evaluate the usability of the WMS, a number of functions were outlined for testing. The major tasks selected for evaluation were based upon common features with the benchmark system, cmu.misc.market. By
selecting these tasks, it allowed for direct comparison of use between systems. The three major tasks included:

1) **Log-in to the system**: Each system requires the user to provide a user name and password to gain access to the system. This allowed for a comparison in initially accessing the systems.

2) **Posting an Item to sell or buy on the system**: Each system allows the user to post, or send, an item to the system. Posting items is a key feature for allowing items to be put onto the system.

3) **Searching for an Item**: The WebMarket system allows for searching by keyword to find an item; the cmu.misc.market does not. By selecting this task for comparison, we expected to show the benefits of the WMS.

**Measures of Evaluation**

Included below are a list of measures and the means with which we used in order to capture data for the evaluation of the WebMarket system. In determining the usage and acceptance measures, we hoped to understand the physical changes produced by the system as well as the behavioral changes caused by the system.

<table>
<thead>
<tr>
<th>Description</th>
<th>Method of collection</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>The number of changes made to the WMS during implementation.</td>
<td>Simple count</td>
<td></td>
</tr>
<tr>
<td>The number of unique messages on WMS vs. cmu.misc.market</td>
<td>Simple count (script)</td>
<td></td>
</tr>
<tr>
<td>The number of unique posters on WMS vs. cmu.misc.market</td>
<td>Simple count (script)</td>
<td></td>
</tr>
<tr>
<td>The number of exclusive posters on WMS vs. cmu.misc.market</td>
<td>Simple count (script)</td>
<td></td>
</tr>
<tr>
<td>The number of repeated messages on cmu.misc.market</td>
<td>Simple count (script)</td>
<td></td>
</tr>
<tr>
<td>The number of users registered on WMS after an advertisement event.</td>
<td>Simple count</td>
<td></td>
</tr>
<tr>
<td>The number of users registered over time.</td>
<td>Simple count</td>
<td></td>
</tr>
</tbody>
</table>
| Rate of Sale                                                                 | A pop-up on time of retirement/deletion | Possibilities include:  
  - sold on WMS  
  - sold on cmu.misc.market  
  - sold elsewhere  
  - decided not to sell |
Information System Evaluation Case Studies

<table>
<thead>
<tr>
<th>Interface Evaluation</th>
<th>GOMS analysis and cognitive walkthrough</th>
<th>Possibilities:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Search Effectiveness</td>
<td>A pop-up on time of search return</td>
<td>• Search returned nothing</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• nothing on system</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• poor search criteria</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Search returns good data</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>User demographics</th>
<th>Long survey at time of registration</th>
<th>Need to determine:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>• web/news habits</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• what, where, and how much time they spend on the system</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• receptivity to change</td>
</tr>
</tbody>
</table>

**Methods for Data Collection**

Three different types of methods were used to gather the information needed to evaluate the WMS. Simple counts were used to access the acceptance of the system through use and marketability effectiveness. Surveys were used to look at the acceptability of the system. The modifications in functionality can also be assessed through simple count evaluation. A GOMS analysis was performed to evaluate the usability.

**Simple Counts**

Simple counts were used to assess the effectiveness of the marketability of the system. This data included the number of registrations, number of times a function was accessed on the WMS, and the number of postings. The data was then filtered to extract out unique users. The data for the WMS was produced using server statistics. The cmu.misc.market statistics were generated by filtering the message extraction of the newsgroup.

Additionally, simple counts were used to look at the functional modifications that were made to the WMS during implementation based upon user feedback. This feedback was stored as comments in the system throughout the implementation. An administrator log detailed the modifications made to the system. This log allows for evaluating the functional modifications made to the WMS.

**Surveys**

Surveys were created to assess the acceptability of the WMS. Two different types of surveys were used in the evaluation process: a registration survey and pop-up surveys.

**Registration**

The purpose of using a survey at the time of registration was to record the demographic information of our users. These demographics captured the habits, preferences, and marketing effectiveness of the WMS.
Pop-up Surveys
A major source of on-line input from users came from pop-up surveys. These surveys entailed a quick question that required either checking a box, or the clicking a button. These surveys appeared when users of the WMS were using certain features of the system. These features included:

- Simple and Detailed Search
- Browse
- Delete Item
- Shopping Cart

GOMS Analysis
In order to test the time and efficiency of each system, a GOMS analysis (Newman. 1995) was performed. We selected GOMS because it is a performance-based analysis and it could generate data that compared the functionality of the two systems. The specific type of GOMS analysis used was key-stroke level (KLM) model. GOMS is an acronym for Goals, Operators, Methods, and Selection. The following is a breakdown of what each of these topics entails:

- **Goals**: the end-state the user is trying to achieve. In this case, the completing each of the three tasks will be our goals
- **Operators**: the basic actions available to the user to perform certain task. Operators will be measured using execution time of an action. The duration of each operator is shown below in the table. Examples of operators:
  - Clicking the mouse button
  - Pressing a key
  - Pointing to an object on the display screen
- **Methods**: sequence of actions and subgoals for accomplishing a goal. The following are examples of methods:
  - Starting the Internet Browser
  - Mouse-Clicks
  - Key-Strokes
- **Selection Rules**: This rule is invoked if there are choices of methods. Here, since we don't have choices for methods, we will use the Sequence Form. Sequence Form has a fixed form of operators for accomplishing a single goal.

The primary purpose of the KLM model is to predict the user's speed of execution of tasks. The basis of the method is to divide each task-performance method into components and assign execution times in seconds to each component. These times have been derived from repeated experiments and from the information processing model called Fitts' Law. KLM includes six types of operators to predict times:
Table 2: KLM Operators

<table>
<thead>
<tr>
<th>Operator</th>
<th>Explanation</th>
<th>Duration in seconds</th>
</tr>
</thead>
<tbody>
<tr>
<td>K</td>
<td>Pressing a key or a button (Avg, Non-secretary typist)</td>
<td>.28</td>
</tr>
<tr>
<td>K</td>
<td>Pressing a key or a button (Avg, Skilled typist)</td>
<td>.20</td>
</tr>
<tr>
<td>K</td>
<td>Pressing a key or a button (Typing random letters)</td>
<td>.50</td>
</tr>
<tr>
<td>M</td>
<td>Point to mouse to target on a display</td>
<td>1.10</td>
</tr>
<tr>
<td>H</td>
<td>Home hand(s) on keyboard or other device. (Hand movement from/to keyboard &amp; mouse)</td>
<td>.40</td>
</tr>
<tr>
<td>M</td>
<td>Mentally Prepare (by Heuristics)</td>
<td>1.35</td>
</tr>
<tr>
<td>R(t)</td>
<td>Response by system</td>
<td>t measured with minimal load on the system.</td>
</tr>
</tbody>
</table>

A GOMS analysis breaks tasks down into steps that a user will perform, and creates a prediction model of how long each task will take for each user to perform. This type of analysis was suitable in this case for we can compare the ordered steps required for operating both systems. To perform the GOMS analysis, system functions were chosen for analysis and users were selected to test the keystroke usage. The following functions were chosen:

- Logon: Gaining access to each system.
- Post Item: Creating and submitting an item to each system.
- Search: Finding an item on each system.

Data Collection

The previous section outlined the methods we used to collect data. The following describes the process used to collect the data for each method.

Simple Count Data Collection

In order to compile user and message statistics, a file was created to hold all information posted to cmu.misc.market. All message information of the cmu.misc.market was extracted and stored in this file. Additionally, server and registration information was kept for the WMS. These files were then put through a series of exel functions to produce final numbers for comparisons.

In order to compile the modifications to the system, the comments and administrator log were evaluated. All changes to the system were counted and evaluated based upon being a critical change, or being an enhancement. A critical change entails fixing a current function that is not working, as an enhancement provides the user with an added feature.
Survey Data

All surveys completed for this study were done using surveys that appeared to users of the WMS. This data was stored in the database for analysis. The registration survey was completed by all users who used the system. The pop-up surveys, however, appeared at random to users of the system. The following was done to create these surveys:

- A random number was generated upon entering a screen with an associated survey
- If the random number matches the survey indicator (a preselected number) then the survey was called.
- The survey entailed one question followed by either check boxes, or yes/no buttons.
- After the user responded, the answer and user id were recorded in a flat file.

GOMS Analysis

The GOMS analysis data was collected by interviewing and using participants of an analysis test. A consent form was created for the three participants of the keystroke analysis. Three participants were chosen for analysis. Data was then collected for both the WMS and cmu.misc.market. The following assumptions were made:

- Shared Ethernet Connection - 10 MB/sec Data Transfer Speed
- Minimal load on the system - # of users < 5
- Assume that WMS page is bookmarked.
- Assume using WinEzmail32 to access cmu.misc.market.
- Pentium 100 MHz IBM Pc running Windows NT 4.0
- Netscape V3.0 used as a sample browser.
- Assume that cmu.misc.market is subscribed.
- Assume no use of short-cut keys except “Tabs.”

Additionally, a task analysis plan was developed as follows:

- Sampling:
  Sampling method of users will be random within the category of undergraduate students.
- Data Collection Technique:
  Data will be collected by using real-time observation and note-taking technique as the user is performing a task.
- Measures:
  - Speed of performing a task: The time will be measured by a stop watch. The time due to error or spent on error recovery will be subtracted from the total time.
Incidence of errors: The number of times a participant fails to follow the steps defined by the GOMS Analysis.

The predictive model of each of the three tasks is outlined below:

<table>
<thead>
<tr>
<th>Users Involved</th>
<th>Method</th>
<th>User Time (Each in minutes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>User #1, #2, #3</td>
<td>Interviews, Surveys*, Consent Forms*</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td>Explanation of Tasks</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>Walkthrough and Observation</td>
<td>45</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td>90 x 3 = 270</td>
</tr>
</tbody>
</table>

*Note: Please view the appendix for the surveys and the consent form used in this usability study.

In addition to the prediction model, the following item information was outlined for the users to use when testing the system.

Sample Item for Posting
Item Title: Discrete Mathematics Book
Price: $20
Brand: David Harck
Description: This book was used in 21-228 class in spring '89.

Searching for Items
Search for VCR's by typing in the keyword "VCR." Try to find two VCR's in cmu.misc.market

Data Analysis

Functional Requirements Analysis

When implementing the WMS, we hoped that the system would not run into any major problems. We ran into a few critical problems that required fixes to the current functionality. Enhancements were made to the system to provide for both user and system evaluator demands. The following critical problems were found and fixed:

1) Length of email address in our registration was too short.
2) The use of an apostrophe in any text field produced an error.
3) The back button on the categories feature in the item posting did not respond correctly.
4) A few words were misspelled on the system.
The following enhancements were made for users of the system:

1) Text boxes were modified to allow for word wrap when typing.
2) Spacing on screens of the system was changed based upon user feedback.
3) A Frequently Asked Question (FAQ) was added to the system to provide a quick reference for using the system.
4) Allowing users to post items to cmu.misc.market directly from the WMS
5) A direct link to the administrator was added for contact information.

The following enhancements were made for the evaluation and marketing of the system:

1) Statistical generation of registration
2) Viewing comments posted by users
3) Provide a random $5 prize pop-up screen for users

We were quite content with only having to make twelve changes to the system upon implementation. Most comments made to the system were quite helpful and provided useful information to finding the problems of our system, and the enhancements that were requested. Of course there were some enhancements that were outside the base functionality of our system, such as providing an auction based forum. Unfortunately with the short time frame of this analysis, we were unable to implement major enhancements.

Overall the functional changes were minimal. The functionality matched the functional requirements set aside at the development stage of the system.

Acceptability Analysis

For a system to be truly useful, it must also be accepted by the users. We have analyzed the simple counts of users, items, demographics, and marketing events. By looking at this data we were able to compile growth and usage statistics of the system. Unfortunately, the WMS did not gain a wide amount of usage in its short tenure at Carnegie Mellon. We did however gain insight into the acceptance of our functionality through surveys, which proved to be positive.

Simple Counts

Simple count provided an endless resource in the usage and growth statistics of our users. The following summarizes the simple counts collected:
Table 4: Simple Count Summary

<table>
<thead>
<tr>
<th></th>
<th>WMS</th>
<th>cmu.misc.market*</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of messages</td>
<td>151</td>
<td>9132</td>
</tr>
<tr>
<td>Number of unique messages</td>
<td>151 (100%)</td>
<td>5794 (63.45%)</td>
</tr>
<tr>
<td>Number of publicly posted responses</td>
<td>0</td>
<td>806 (8.83%)</td>
</tr>
<tr>
<td>Number of Unique Posters</td>
<td>79 (52.32%)</td>
<td>2239 (24.52%)</td>
</tr>
</tbody>
</table>

(The cmu.misc.market data is observed from August 1996 while WMS launch was 10/15/96)

While it is clear that usage levels for WMS are substantially lower than cmu.misc.market, some of the benefits of WMS design are apparent from our data collection. Over 36% of cmu.misc.market is non-item data. That is, that over 36% of the system contains repeated or response information.

Acceptability can come from users feeling comfortable enough to post multiple items on the system. That is, if a user has success with or faith in a system he/she may offer up additional items for sale. Clearly, WMS commands that kind of acceptability. However, the results in this category may be a bit misleading as many of the posters may have responded to other posts or posted item update information, both actions that WMS doesn't provide (publicly posted responses) or record (item updates).

To access acceptability, we also looked at the growth rate of the system in terms of number of registrations, and number of items posted to the system. By looking at the growth rate, we can infer how readily the entire community is using the system.

Figure 1: Growth Rate of WMS
As can be seen above, the number of items on the system, and number of registered users increased at near constant rate. It is obvious, however, that the number of items on the system increased at a lower rate.

Of course growth is only important if we can compare it to our benchmark system. We looked at the growth rate of the number of messages submitted to both the cmu.misc.market and the WMS. The following graph represents our findings:

Figure 2: Message Count of Systems

The message rate for the current system is basically linear. The growth rates for WMS are substantially less. The consistent rate of growth for cmu.misc.market points to its acceptance. This graph does not, however, account for duplicate messages on cmu.misc.market. WMS tried to boost the rate of growth for the messages by marketing the system. The effectiveness
of marketing can be seen by analyzing the counts of registration in comparison to marketing events.

Figure 3: Marketing Events

Looking at the general trend in the number of users who registered to WMS, we can see that there are more people registered on the day or a day after the electronic post. The total number of users who registered during these two days compared with other days when there were no electronic board post shows that these postings were crucial in attracting the users of the misc.market to the WMS.

Registration had some general trends when looking at the time of registration. Comparing the number of people registered during weekdays (Monday - Thursday) with the weekends (Friday - Sunday), we had greater number of people registering to the WMS on weekdays.

Table 5: Registration by Day Type

<table>
<thead>
<tr>
<th>Week</th>
<th>Weekday (Monday - Thursday)</th>
<th>Weekend (Friday - Sunday)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Week 1</td>
<td>164</td>
<td>38</td>
</tr>
<tr>
<td>Week 2</td>
<td>38</td>
<td>41</td>
</tr>
<tr>
<td>Week 3</td>
<td>48</td>
<td>0</td>
</tr>
<tr>
<td>Week 4</td>
<td>42</td>
<td>21</td>
</tr>
<tr>
<td>Week 5</td>
<td>38</td>
<td>13</td>
</tr>
<tr>
<td>Week 6</td>
<td>12</td>
<td>4</td>
</tr>
</tbody>
</table>
Over the course of six weeks, we were able to reach 480 people to register to our system. However, it is only as small portion of the CMU population. Out of all registered users, we had 420 students (undergraduates and graduates) and 60 faculty and staff. In 1995 Fall, there were 7318 students, with 6216 undergraduate and 1102 graduate [Source: http://www.cmu.edu/ba/planning/facts96/Student96.html]. It is interesting to note the number of students registered to the system were not all of technical majors. The following tables summarize this data:

Table 6: User Affiliation

<table>
<thead>
<tr>
<th>WMS User Affiliation</th>
<th>Number of Registered Users</th>
</tr>
</thead>
<tbody>
<tr>
<td>Freshman</td>
<td>80</td>
</tr>
<tr>
<td>Sophomore</td>
<td>92</td>
</tr>
<tr>
<td>Junior</td>
<td>62</td>
</tr>
<tr>
<td>Senior</td>
<td>82</td>
</tr>
<tr>
<td>Graduate</td>
<td>100</td>
</tr>
<tr>
<td>Part Time Staff</td>
<td>4</td>
</tr>
<tr>
<td>Full Time Staff</td>
<td>41</td>
</tr>
<tr>
<td>Faculty</td>
<td>13</td>
</tr>
<tr>
<td>Total</td>
<td>480</td>
</tr>
</tbody>
</table>

Table 7: User Colleges

<table>
<thead>
<tr>
<th>WMS User College</th>
<th>Number of Registered Users</th>
</tr>
</thead>
<tbody>
<tr>
<td>CFA</td>
<td>27</td>
</tr>
<tr>
<td>CIT</td>
<td>141</td>
</tr>
<tr>
<td>GSIA</td>
<td>39</td>
</tr>
<tr>
<td>H&amp;SS</td>
<td>118</td>
</tr>
<tr>
<td>Heinz</td>
<td>17</td>
</tr>
<tr>
<td>MCS</td>
<td>36</td>
</tr>
<tr>
<td>SCS</td>
<td>68</td>
</tr>
<tr>
<td>Other</td>
<td>34</td>
</tr>
<tr>
<td>Total</td>
<td>480</td>
</tr>
</tbody>
</table>

In order to further the analysis of use of the system, we looked at the server hits to the WMS. The server hits are the number of times the system was visited by remote users. The following table summarizes this information:

Table 8: Server Hits of WMS

<table>
<thead>
<tr>
<th>Weekly</th>
<th>Hits</th>
<th>Bytes Transferred</th>
</tr>
</thead>
<tbody>
<tr>
<td>Week of 10/13/96 to 10/19/96</td>
<td>11,717</td>
<td>24,531,764</td>
</tr>
<tr>
<td>Week of 10/20/96 to 10/26/96</td>
<td>4,558</td>
<td>12,558,555</td>
</tr>
<tr>
<td>Week of 10/27/96 to 11/02/96</td>
<td>6,423</td>
<td>16,933,252</td>
</tr>
<tr>
<td>Week of 11/03/96 to 11/09/96</td>
<td>6,686</td>
<td>29,105,075</td>
</tr>
<tr>
<td>Week of 11/10/96 to 11/16/96</td>
<td>4,935</td>
<td>15,104,333</td>
</tr>
</tbody>
</table>
As can be seen from this data, the system was visited most during the first week of use. The number of hits then became pretty consistent. This shows that the usage of the system stagnated. Growth in number of users and items can be deceiving, if the previously registered users stopped using the system. The number of hits shows that this is probably the case. When looking at acceptance, it is important to not only look at the growth in users, but also the decline in previous users.

The acceptance in terms of usage did not meet the our expectations as developers. We hoped that at the very least, our system functionally was accepted by the users. The survey data provided this information.

**Surveys**

In order to assess the acceptability of the functionality of the system, we analyzed the pop-up surveys that our system initiated randomly to users. The following figures illustrate our results.

**Figure 4: Browse Survey**

<table>
<thead>
<tr>
<th>Browse Survey: How do you find the categorical breakdown of items:</th>
</tr>
</thead>
<tbody>
<tr>
<td>No Opinion 12%</td>
</tr>
<tr>
<td>Not Usefull 15%</td>
</tr>
<tr>
<td>3/26</td>
</tr>
<tr>
<td>4/26</td>
</tr>
<tr>
<td>19/26</td>
</tr>
<tr>
<td>Usefull 73%</td>
</tr>
</tbody>
</table>
Figure 5: Search Survey

Search: Was the Search Effective?

- No, these items were not what I was looking for: 17%
- Yes, few items were returned as expected: 17%
- Yes, good items: 66%
- Don't know: 0%
- No, nothing was returned but I'm sure something I want is on the system: 0%

Figure 6: Retirement Survey

Retirement of Items: Reason for Retirement

- Sold on WRms: 33%
- Sold on cmu.misc.market: 66%
The survey data did not return as many valid responses as was desired. Data from the WMS and ONA team members was discarded. Additionally, frequencies of the randomization for each function did not follow the frequency of use. Thus, since the retirement function was used less frequently than the browse, less questionnaires were issued by the randomization.

The interesting information that could come from survey collection is hinted at by one of the users who responded to the same question at different times.

<table>
<thead>
<tr>
<th>User</th>
<th>Survey</th>
<th>Response</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>USER3</td>
<td>Browse</td>
<td>Not Useful</td>
<td>Oct 23 1996 11:24PM</td>
</tr>
<tr>
<td>USER3</td>
<td>Browse</td>
<td>Useful</td>
<td>Nov 12 1996 10:48AM</td>
</tr>
</tbody>
</table>

This seems to suggest that this user, with additional use of the system, changed his opinion about the browse functionality. With more surveys and greater use, this data could show a building trend in acceptance and could be correlated to increased use, increased marketing, or other factors.

### Usability Requirements Analysis

To evaluate the usability of the WMS, a GOMS analysis was performed. Each of the following tables illustrate the tasks and time taken by each user. Error in % = ABS (predicted value - actual value) / (predicted value).

#### Table 9: Logon Analysis

<table>
<thead>
<tr>
<th>Task</th>
<th>User</th>
<th>Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logon to Web Market System</td>
<td>User #1: 28.03 User #2: 25.94 User #3: 39.71 Average: 31.23 St. Dev. 7.42</td>
<td></td>
</tr>
<tr>
<td>Predicted Duration by GOMS:</td>
<td>28.38 Error in %: 10.03%</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Task</th>
<th>User</th>
<th>Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logon to cmu.misc.market</td>
<td>User #1: 17.25 User #2: 19.43 User #3: 28.25 Average: 21.64 St. Dev. 5.62</td>
<td></td>
</tr>
<tr>
<td>Predicted Duration by GOMS:</td>
<td>19.43 Error in %: 11.39%</td>
<td></td>
</tr>
</tbody>
</table>

#### Table 10: Logon Errors Analysis

<table>
<thead>
<tr>
<th>WebMarket System</th>
<th>cmu.misc.market</th>
</tr>
</thead>
<tbody>
<tr>
<td>User #1: Pressed return instead of pointing and clicking on &quot;Logon.&quot;</td>
<td>None</td>
</tr>
<tr>
<td>User #2: Pressed return instead of pointing and clicking on &quot;Logon.&quot;</td>
<td>None</td>
</tr>
<tr>
<td>User #3: Pressed &quot;Apply&quot; the first time. Position was too close to &quot;Logon.&quot;</td>
<td>None</td>
</tr>
</tbody>
</table>
GOMS Analysis predicted the results quite well because it was a relatively simple task and users have seen it before. The noticeable error was that users were too familiar with pressing enter after they type in the password. The system should be modified to incorporate this feature. Another suggestion that will reduce time for execution of task is to display the login screen initially and have a button for applying for an account on that same page.

Table 11: Item Posting Analysis

<table>
<thead>
<tr>
<th>Task</th>
<th>WebMarket System</th>
<th>cmu.misc.market</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>User</td>
<td>Duration</td>
</tr>
<tr>
<td>Posting an Item on WMS</td>
<td>User #1:</td>
<td>52.7s</td>
</tr>
<tr>
<td></td>
<td>User #2:</td>
<td>55.2s</td>
</tr>
<tr>
<td></td>
<td>User #3:</td>
<td>62.6s</td>
</tr>
<tr>
<td>Average:</td>
<td>56.8s</td>
<td>St. Dev. = 5.14</td>
</tr>
<tr>
<td>Predicted Duration by GOMS:</td>
<td>51.7s</td>
<td>Error in % 9.93%</td>
</tr>
</tbody>
</table>

Table 12: Item Posting Error Analysis

<table>
<thead>
<tr>
<th>WebMarket System</th>
<th>cmu.misc.market</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>User #1: Went to the “reply” template instead of the “post” template.</td>
</tr>
<tr>
<td>None</td>
<td>User #2: Went to the wrong menu bar to send.</td>
</tr>
<tr>
<td>None</td>
<td>User #3: Didn’t know the keyword “Post.”</td>
</tr>
</tbody>
</table>

In this result, GOMS predicted the WMS execution time well within range, but did not do a good job of predicting time for cmu.misc.market. The reason for this is that users could not distinguish between “post” and “reply” or users did not know the keyword “post.” It reflects the fact that the users did not utilize cmu.misc.market. However, in the WMS case, they did not face any trouble because the navigation was intuitive enough to allow a smooth completion of task.

Table 13: Search Analysis

<table>
<thead>
<tr>
<th>Task</th>
<th>WebMarket System</th>
<th>cmu.misc.market</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>User</td>
<td>Duration</td>
</tr>
<tr>
<td>Searching for Items on WMS</td>
<td>User #1:</td>
<td>12.6s</td>
</tr>
<tr>
<td></td>
<td>User #2:</td>
<td>9.97</td>
</tr>
<tr>
<td></td>
<td>User #3:</td>
<td>19.97</td>
</tr>
<tr>
<td>Average:</td>
<td>14.21</td>
<td>St. Dev. = 5.17</td>
</tr>
<tr>
<td>Predicted Duration by GOMS:</td>
<td>14.64</td>
<td>Error in % 2.94%</td>
</tr>
</tbody>
</table>

In this result, GOMS predicted the WMS execution time well within range, but did not do a good job of predicting time for cmu.misc.market. The reason for this is that users could not distinguish between “post” and “reply” or users did not know the keyword “post.” It reflects the fact that the users did not utilize cmu.misc.market. However, in the WMS case, they did not face any trouble because the navigation was intuitive enough to allow a smooth completion of task.
Information System Evaluation Case Studies

Table 14: Search Error Analysis

<table>
<thead>
<tr>
<th>WebMarket System</th>
<th>cmu.misc.market</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>User #1: Didn’t know where the search menu bar was.</td>
</tr>
<tr>
<td>None</td>
<td>User #2: Went to the wrong menu bar for search. Forgot about forward and backward options after initial search.</td>
</tr>
<tr>
<td>None</td>
<td>User #3: Forgot about the forward and backward option and pressed enter.</td>
</tr>
<tr>
<td>None</td>
<td>User #3: Did not know where the “search” menu bar was.</td>
</tr>
</tbody>
</table>

GOMS did not do very well in predicting times for searching for an item on cmu.misc.market. The reason being, users did not know how to go about searching for a caption. It clearly shows that the search under cmu.misc.market is very user unfriendly and does not allow to search through the body of the captions. WMS clearly shows superiority when completing the search task.

The study has shown that users will spend more time logging on to the WMS. Part of the reason is due to the system wait time incurred by the web browser. First of all, the web browser takes a longer time to start up and second, it takes longer for a page to load onto the browser. However, the study has shown that this is not a major drawback since users expect the web-based system to take a longer time to process data than a text-based system.

The study has also shown that posting an item takes longer on WMS. This is because the user needs to specify more data and categories when posting an item on WMS. For example, the user needs to specify the price and the brand of the item. Category selection can potentially take a long time if the user does not find the exact category his/her item fits into. This may initially be a weakness of the WMS, but it is predicted that users will begin to appreciate it when thousands of items will be posted and they will want to list all the items in a specific category.

By observing the data and graphs above, the most powerful feature of WMS is the search module. The time to complete the search under cmu.misc.market exceeded the time using WMS. In addition, WMS will list all the items with the keywords in the subject line and in the body. Therefore, the search hits will exceed that of cmu.misc.market. Also, while in cmu.misc.market, users have to search for items one by one, until the search returns “Text not found.” Therefore, even though, the logging on and the posting of items take longer, the searching capabilities of WMS clearly shows a significant advantage over the cmu.misc.market.

Recommendations and Conclusions

Based on the analysis we performed on the WMS and cmu.misc.market, some conclusions can be drawn on the functionality, acceptability, and usability of our system.
System Status

Our analysis showed that the functionality and usability of our system met the needs of our users. The system did what it was supposed to do, and seemed to do it effectively. The surveys of our users also showed the same general trend that the system provided useful functionality. The WMS provided superior functionality and usability over the cmu.misc.market. The system did present a major problem; the acceptability of the system was minimal. The number of users that registered to the system, along with usage statistics show that the system did not gain many users in the time frame allocated for this study. The cmu.misc.market grew at a faster rate than the newly implemented WMS. This poses a major problem. A system is only worthwhile if it is in fact used.

A few factors may account for the lack of acceptability found of our system: Usability, Functionality, Marketability, and Time frame of analysis. From our previous analysis, usability and functionality seem to meet the users needs. Marketability did produce some results in terms of number of the users that registered to the WMS. Retrospectively, it may have been beneficial to have provided more opportunities for advertising to the community. Probably the most important factor of implementing our system was the time frame allocated. This system was analyzed for a little over a month. It is expected that more time may have allowed for some continued growth in the system. A study over a longer frame of time may better analyze the acceptability of the system. It should be noted that the system maintained little growth in comparison to the cmu.misc.market near the end of the analysis, thus expectations of acceptability increasing are minimal. The lack of items on the system is seen as the main problem. If you note the number of server hits does not increase after the first week. The actual use of the system has stagnated. One possibility is that there are a lot of people looking at the system, but very few adding items to the market. Unfortunately, we do not have the break down of users using the browse and search functionality, but we predict that this is the case.

Limitations of System

The evaluation noted several limitations of the system. The WMS is independent of the mail reader with which the cmu.misc.market is associated. We found out early on that it was imperative that we link the systems as best as possible to allow for adaptability to the new system. We implemented a feature of the WMS to allow for posting directly to the cmu.misc.market. We could not do the reverse, and allow for cmu.misc.market users to post to the WMS from their mail reader. We can infer from our analysis that this separation did not allow for quick adaptability.
The most important lesson to be learned is that a system will not sell itself. Acceptability is the most important barrier our system was unable to cross. The internal and external factors should be addressed further to find out exactly what steps could be taken to induce more users.

**Future Enhancements**

The WebMarket System will not continue its implementation. This is partly due to external factors: system administrator graduating and software upgrade limitations. This is unfortunate for the acceptability of the system would be an interesting aspect to analyze further. If the system was to continue its implementation, an acceptability plan would be formulated to detail the feasibility that the WMS could take over as the preferred market of CMU. If the costs of switching users over to the system outweigh the benefits, than the system should not be implemented. The WMS currently provides no source of funding to cover any costs, so the system would most likely need to be modified to allow for funding, such as registration fees or advertising on the system.

Did the WMS fail as a forum for the buying and selling of goods? Yes, it failed. Even though the system provided more functionality and a more usable interface, it was not used by enough users to warrant further implementation. Did we fail as system developers? No, we did not fail completely. The system matched the goals of the users, but did not allow easy adaptation for users. The system failed in the implementation phase. The main lesson to be learned is that a system needs to be adopted by the users for which it is designed. Planning ahead for marketing and getting new users to use a system is imperative for the success of a system like the WMS. The feasibility of users accepting the system, would have been a worthwhile study to perform. If the system appears to be an acceptable solution to the marketing needs of the entire base of users, steps should be outlined for implementing the system in manner that allows easy adaptability for users. The success of a system lies not only in good development practices, but also in proper implementation.

**Comparative Analysis**

As developers, are we incompetent? This is an important question to consider after looking at both of these cases. Each system was developed using the defined methodology of the systems development life cycle. Why did both of the systems fail to gain acceptability by the users? Some similarities and differences illustrate this point.

A system succeeds only if it is used and used effectively by the users. In both cases, the system provided an alternative solution for users to accomplish some goal. The users chose not to use the alternative system. ONA and
WMS both showed how the system was effectively used. There is a bridge that was not created for allowing users to adapt to each of the new systems. A major contrast between the user set illustrates an interesting point. ONA was developed for a small specific set of users to perform small sets of specific tasks. The small set of users did not wish to adapt to the computer based solution. Without this set of users, the system fails. The needs of these users needs to be met fully for the system to succeed. The best way to evaluate this system is through rich interaction with the few individuals who will use the system. It can be noted that the evaluation was based upon this type of interaction through surveys and interviews.

WMS was developed for a large number of heterogeneous users. The WMS intended to sway users of an email based market to use this new alternative. The system was not accepted by the Carnegie Mellon community. In order for this system to succeed, it must meet the needs of a broad set of users. In order to evaluate the broad set of users goals, more discrete methods were used to analyze usage and survey statistics.

It is apparent from these two studies that designing and implementing a system requires work in finding how the users' needs of adaptability and acceptability can be met. Whether your user set is big, small, well defined, or amorphous, gaining acceptance is key to the success of a system. System evaluation provides insight to allow developers to face this type of problem.

**Concluding Remarks**

We are competent system developers. The systems developed for these cases follow the set guidelines for good system development. But it is not good enough to be competent developers; we must also become fully aware of the users needs and organizational barriers that we have to overcome when implementing systems. No system is designed and developed perfectly. The evaluation process allows us to refine the system to meet the goals of the users. After evaluating the finished system with the users, we can enhance the functionality and refine the interface to provide a better solution. Unfortunately for the developers, the evaluation may conclude that the suspension of the system provides the best solution. The most important aspect is that the users goals and needs are met.

It is also important to note that the evaluations of both systems were done after the completion of the development process. During development, user testing was performed with prototypes of each of the systems for preliminary informal feedback. Organizational and user acceptability was not considered at any stage of development. Retrospectively, it would have been quite useful to involve the entire user set for evaluating the acceptance and usefulness each system would provide. Time may have also been a factor for the failure of both systems. Acceptance may not happen over night. Typically, organizational change provides resistance, and implementers of
new system solutions may need to develop plans for assisting the users in adapting to a time and cost saving system alternative.

Technology allows organizations and users to accomplish a wide variety of tasks. The important issue is how do we develop systems for these organizations and users that will provide useful benefits. By evaluating systems, we can find deficiencies in our systems and modify the functionality or implementation to better meet the goals of the users. By having systems that are not only functionally competent, but actually used effectively and efficiently, as developers we have succeeded. As is sometimes hard to forget, the users and organizations provide the financing from which the developers are paid. We should focus the implementation and system refinements to meet the users goals. System evaluation provides a framework to better meet these goals.

Acknowledgments

ONA Evaluation Team
Sameh Boutrous, Interface Designer and Programmer
Justin Fidler, Programmer
Andrew Rosenbaum, System Analyst

Additional members of the development team:
Jordan Anderson, Validation Tester and Writer
John Biggs, Writer
Danny Chau, Manager

WMS Evaluation Team
Shoain Mian, System Analyst and GOMS Usability Tester
William Moher, Interface Designer and Acceptability Tester
Sang Hyun Park, Project Coordinator and Marketing Evaluator
Edwin Rodriguez, Programmer and System Administrator
Nathan Stark, Manager and Programmer

Additional member of the development team:
Eugene Chun, Writer

Faculty Advisors
William Hefley
Doug Wholey
Bibliography

