ERGOeasy
Improving an Interface for Ergonomic Development

Gwendolyn Tennell
LSU Shreveport
One University Place,
Shreveport, LA
gytennell@gmail.com
214-402-0497

Jesse Gamez
Auburn University
Shelby Center for Engineering
Technology Suite 3101M
imkinglouie@gmail.com
972-358-9676

Dr. Cheryl Seals
Auburn University
Shelby Center for Engineering
Technology Suite 3101M
sealscd@auburn.edu
334-844-6319

ABSTRACT
Ergonomic professionals have a need to conform from paper to technology in order to save time and increase productivity. In this digital age of time technology, including mobile devices, in the work environment is essential to making a product better. A good comprehensible, anesthetically pleasing, efficient interface should be used to minimize complexity for the user. The American Industrial Hygiene Association (AIHA) Tool Kit was created so that users with a range of experience in ergonomic analysis would be able to employ it to analyze task in a workplace for a variety of ergonomic risk factors. Improving the design and implementing a simple, anesthetic interface for mobile computing in our program, will allow the ergonomics useful success in the work place.

The Tool Kit is comprised of 20 ergonomic assessment tools that can be used to analyze jobs for a variety of ergonomic risk factors. We have created the ERGOeasy project to better assist the ergonomic specialist in selecting which tool to use for risk assessment based on a given scenario. Our aim is to improve the current interface on the program that we have created.

Author Keywords
Human Computer Interaction; Ergonomics; Assessment Tools; User Interface Design.

ACM Classification Keywords
H.5: Miscellaneous.

INTRODUCTION
Researchers are looking more and more towards enabling touch-based interactive devices that may increase productivity, therefore decreasing costs. As computers have become more commonplace in almost all aspects of our lives, Human-computer interaction (HCI) has become much more important in these recent years. HCI deals with the theory, design, implementation, and evaluation of how humans use and interact with computing devices [1].

The early emphasis of HCI has been how to design interaction and implement interfaces for high usability. While satisfying the need for usability, a simple aesthetic appeal of interfaces is now a critical added requirement for commercial success. An interface is a choice of technical hardware or software of a given interaction model.

It is likely that the invention or rediscovery of the mouse was the essential element in the personal computer revolution, making the operation of a computer natural and much easier than the previous system of keyboard commands. HCI continues to redefine how we view, absorb, create, exchange, and manipulate information. Smartphones, with their touch-oriented interfaces, have nearly replaced previous phones. Body-based and action-oriented interfaces, such as Nintendo's Wii, are now introducing new ways to play and enjoy computer games.

Good HCI designs are generally difficult to create, despite their importance. Substantial knowledge in many different fields is required. HCI is a multi-objective task that involves simultaneous consideration of many things, such as the types of users, characteristics of the tasks, capabilities and cost of the devices, changing technologies, and lack of objective or exact quantitative evaluation measures.

Researchers and developers in the field have accumulated and established these basic principles for good HCI designs in hopes of achieving some of the main objectives.

Know Thy User
Understand the Task
Reduce Memory Load
Strive for Consistency
Remind Users and Refresh Their Memory

Prevent Errors/Reversal of Action
Naturalness

Clark believes that design is what an app does, how it works, and how it presents itself to users. A great app design has a carefully honed concept, a restrained feature set, efficient usability, and a lot of personality. The best app designs become almost invisible with the controls seeming to fade to the background, and the user's task to the front [2].

The best apps demand a special degree of efficiency in the interface and the ability to fold neatly into a busy schedule. To build an app to fly an airplane, one might build this…

Figure 1

People want simplicity and ease from an app. Great apps do not require thinking about their interfaces. They hold complicated tasks but shield users from the complexity, making it easy for to glide through and accomplish goals. As shown in Figure 1, users can tap the fly button to fly a plane, and tap the land button to land.

A focused job description produces the best apps. The more tightly an app idea is defined, the clearer it will be for the user. An app designer must choose an idea, focus it, figure out what the minimum the app has to do to make an idea happen, and polish it.

The user interface design is the contact point between the user and the system. To a large extent, it determines the usefulness and effectiveness of the system. The conceptual model provides the tools for conveying knowledge to the user to execute his/her tasks, and for assimilating this knowledge. A user's (mental) model is a personalized understanding of the conceptual model. The mental model is used to perform tasks that were taught and tasks that were not originally encompassed by the conceptual model [3]. This research aims to improve the interface on our project design.

Background

There's a wealth of information on HCI or user interface design (UID). Design principles are derived from an analysis of interviews. User-centered design is now used to develop the mobile application. [4] portrays that defining what makes an app "good" varies depending on the audience. DotNet is a website community for .NET developers which focuses on the technical aspects of the app. The criteria which they use include stability/reliability, fast loads, no user interface hang-ups, consistency with the platform, and no advertisements. These criteria are only the bare minimum when evaluating apps for use. For instance, the need for a common language and structure to evaluate the effectiveness of apps led to the author's development of the "Evaluation Rubric for IPod Apps" in the fall of 2010. The rubric focuses on criteria that research revealed are important to educators:

(1) Curriculum connections - Apps that target math concepts and skills, quality reading apps that provide students with practice on a wide variety of reading levels, and apps that target science and social studies.

(2) Authenticity - Genuine learning problems that help connect new learning with prior knowledge.

(3) Feedback – In order to improve one's performance is any arena, constructive and timely feedback is critical. Feedback involves the data available to the student and the teacher that summarizes the student's performance.

(4) Differentiation – The ability to set the level of difficulty or target specific skills increases the usefulness of the app as an instructional tool. Having control over the settings of the app to individualize instruction increases the likelihood of success, which in turn increases motivation.

(5) User friendliness – The ease of use.

(6) Motivation – If students are bored quickly with an app, motivation will certainly suffer. The level of success that is experienced and the quality of the interface contributes to student's motivation.

The first step in developing a design framework is to recognize the important challenges associated with the design of complex human-computer systems. The ecological interface design's goal is to support three levels of rational control; familiar events, unfamiliar, but anticipated events, and unfamiliar and unanticipated events.

Familiar events are routine; operators experience them regularly. Unfamiliar, but anticipated events occur irregularly and operators will not have much experience to rely on. Unfamiliar and unanticipated events rarely occur and operators will not have much experience to rely on either. Anticipated solutions to these events provide operators with the help they need, even if they must improvise a solution themselves. Another goal is not to force processing to a higher level than the loads of the task require [5].

The human-computer interaction interface must take into account the constraints essential in the work domain. A designer may think of a relevant way of describing the complexity of the work environment. This will define the informational content and structure of the interface. Next, it is important to think of an effective way of communicating this information to the operator. See Figure 2.

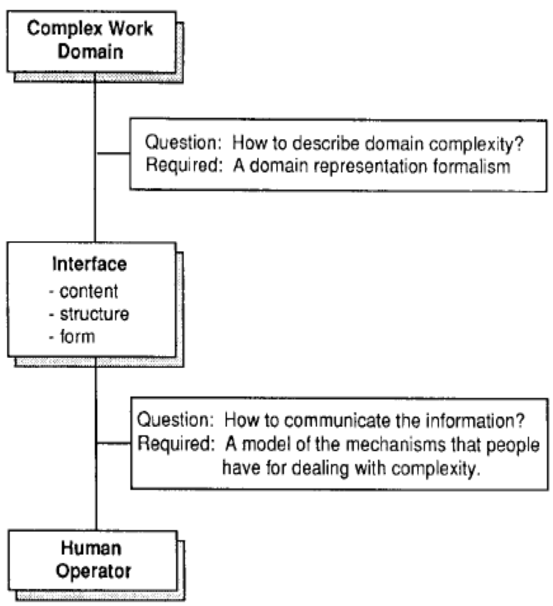

Figure 2: The structure of the interface design problem.

There are other important features pertinent to interface design, such as visual, dialogue, and issues of context sensitivity. An abstraction hierarchy is proposed as a way to represent constraints in a workplace. That along with skills, rules, and knowledge can be used to derive multiple implications for interface design.

It has been argued that lower levels of rational control tend to be executed quicker, with less effort, and more effectively than higher levels. Therefore, if agreed, information should be presented in a way that allows operators to successfully rely on the lower levels of control. See Figure 3.

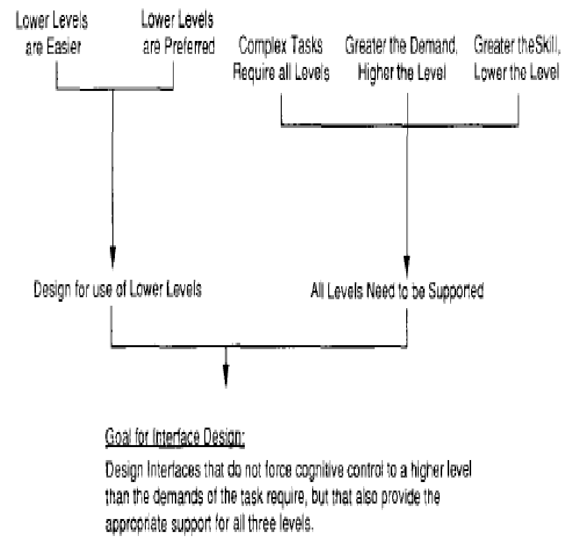

Figure 3

Lu Chen [6] centers on three key factors of an app interface Design: thinking mode, interactive mode, and visual mode of touch-screen devices. An app interacts with users through GUI, and emphasizes user experience during the process. Making human-computer interaction as natural and convenient as person-person interaction is the ultimate purpose of intelligent product interface designs.

A. Conversion of thinking mode

1. Meet and create needs.
2. Make app closer to people’s thoughts and behaviors.
3. Provide people with a sense of playing.
4. Focus on preferences.

B. Innovation of interactive mode.

1. Important elements in interactive design.

    The most important elements in app interactive design are hand size, gestures, visual focus, and eye guidance. Hands and eyes play the most important role. The major carriers for the app to have a dialogue with users are tactile sense, visual sense, and sense of hearing. Effort should be made to guide the users’ attention to the information that the app needs to convey.

2. Innovation of interactive mode originates from life.

    Designers can summarize and extract design concept by simulating objects and actions in reality.

3. Follow the rules of the existing interactive mode of different systems.

    Adopting the common way of using a certain kind of system can decrease the re-cognition.

4. Accuracy and efficiency should not be sacrificed for the innovation of interactive mode.

C. More detailed design of visual mode.

1. Visual design with appeal.
2. Care about one millimeter.

   It is the insignificant millimeter that can bring change at the moment when users touch the screen.
3. Material is as important as color.

   Properly adding materials can make your design more realistic.

Apps on a mobile or tablet PC have inborn special problems. Traditional graphic design is two-dimensional, whereas industrial design is three-dimensional. App interface design intersects the traditional and industrial designs, therefore creating a fourth dimension, time and behavior [6].

People expect computers to perform obvious computational tasks and assist in people-oriented tasks, such as writing, drawing, and designing. Because of this some user-interface (UI) researchers began to rethink the traditional dependence on methods that are more machine-oriented and to look at ways to support properties such as ambiguity, creativity, and informal communication. The goal is to conform computers to people's way of interacting, not the other way around.

Researchers at University of California, Berkeley and Carnegie Mellon University (CMU) have designed, implemented, and evaluated an informal sketching tool, SILK (Sketching Interfaces Like Krazy). SILK combines many of the benefits of paper sketching with the benefits of current electronic tools. SILK also supports the creation of storyboards. Designers can quickly sketch an interface using an electronic pad and stylus. As the designer draws them, SILK recognizes widgets and other interface elements. For example, when you sketch a scrollbar, it is likely to contain an elevator, which is the small rectangle a user drags with a mouse. In a SILK sketch, designers can test the component's or widget's behavior by dragging it up and down [7].

Implementation

We designed a program which assesses risk in the workplace. The program assesses risk on the lower extremities (MMH) and the upper extremities (UET). Scenarios are used to establish which tool the ergonomic professionals may need to assess risk in the workplace. One example is sorting apples in a plant. A worker is sorting apples on a conveyor belt and the belt is constantly moving. The worker will sort apples all day. He will sort them based on size and color and he will push them to the appropriate area based on these criteria. He sorts more than 60 apples per minute > 5 reps and 20 repetitions per minute. This next scenario deals with the wrist: Pharmaceutical Technicians hand-tighten dozens of medicine bottle lids daily. If not adequately tightened, the bottles could leak and spoil products worth lots of money. In another scenario the workers are bending: At a ceramic countertop plant, workers manually lift uncut plates of ceramic out of a holder onto a waist-high conveyor belt, where it is then stacked vertically in a nearby bin. The holder, however, presents the ceramic at knee-height, making workers bend each time to pick up the product. Questions that may be asked of all scenarios are: Low intensity (Force)? High frequency and repetition vs. low (R)? Is it a light weight task vs. heavy weight (W)? How much bending? How much twisting? Is there little vibration vs. high vibration? Is there pushing or pulling?

As shown in Figure 4 and Figure 5, a functional specification was drawn and our wireframe was built, which is coded in JavaScript, based upon these questions that may be asked.

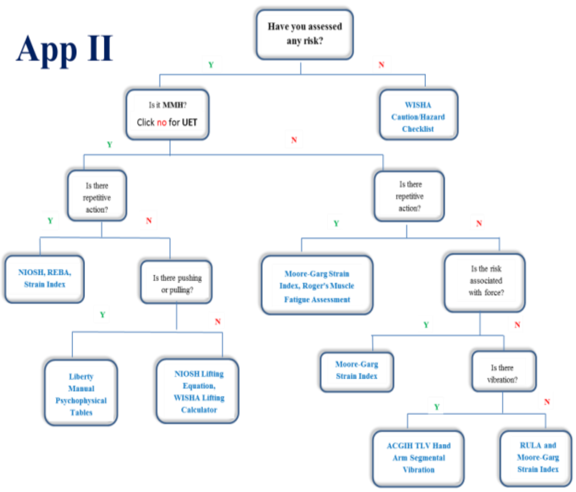

Figure 4: Flowchart

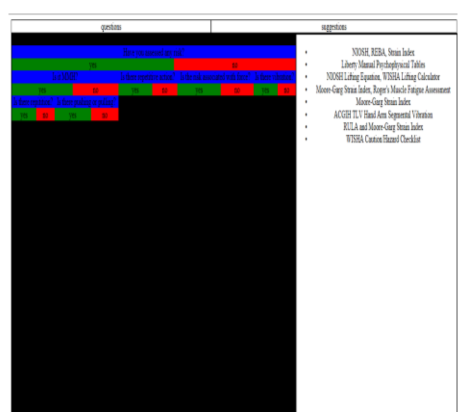

Figure 5: Wireframe

By aiming on getting the code from O(n^2) to O(n), it is possible to reduce time and cost for a company. The following diagrams depict the designs drawn in stages.

Figures 6-8 are the preliminary drawings for the interfaces. Figure 9 is the final result.

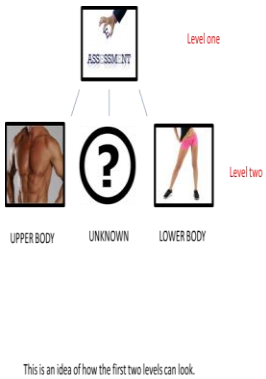

Figure 6: Design I

Figure 7: Design II

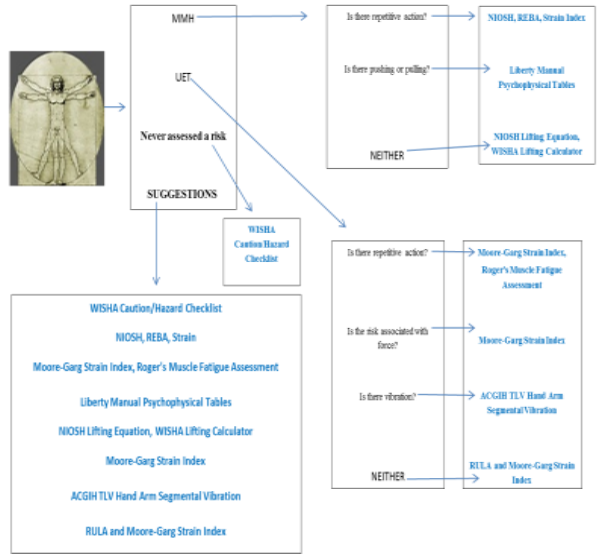

Figure 8: Design III

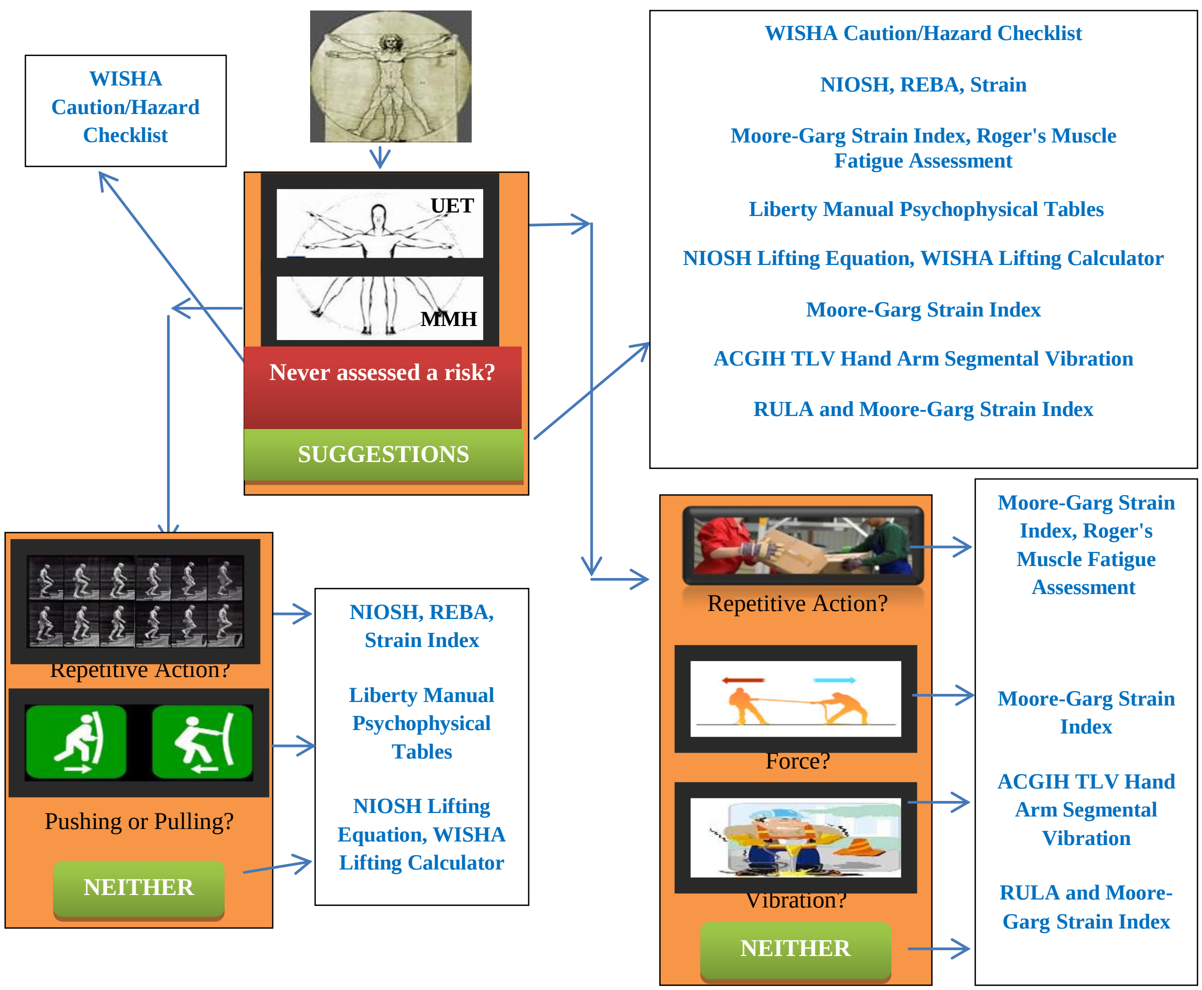

Figure 9: Design IV

CONCLUSION

Today's smart phones provide the same computing power and similar capabilities as the PCs of a decade ago. Therefore it is wise to provide applications for hand held devices.The focus of this research has been on improving the app interface design for human-computer interaction. I believe that the conceptual model is one of the most important aspects of a user interface. Therefore it is very important that the conceptual model is complete and consistent. Using an elegant design will set your app apart. By reducing the complexity of our ERGO code, we have been able to make the program simpler, which makes it run more efficiently and look more aesthetically pleasing.

FUTURE WORK

After all implementations are decided on, the next step will be to build the app for the ergonomics hand held device. Apple has customized a language, Swift, which gives almost anyone the ability to write apps for the Apple platform. Swift is a programming language that is tuned to work with existing Cocoa and Cocoa Touch frameworks. These frameworks contain all the classes used in Mac and iOS apps to support interoperability. Being safe, powerful, and modern are the three pillars Swift is built on [8].

Safety

- Type checking
- Constants for immutability
- Requiring values to be initialized before use
- Built-in overflow handling
- Automatic memory management

Power

- Built using the highly optimized LLVM compiler
- Includes many low-level C-like functions (i.e. Primitive types and flow control)
- Built with Apple's hardware in mind for optimal performance

Modern

- Adopted features from other languages, therefore making it more concise yet expressive (i.e. closure, generics, tuples, functional programming patterns, etc.)

A Mac computer is needed to install Xcode, Apple's Mac and iOS Integrated Development Environment (IDE).

In July 2010 Google marketed the release of the App Inventor for Android (AIA) visual programming environment. A major difference between the Android Market and the Apple App Store is that the Android Market is open, while the Apple App Store is gated. Any handset manufacturer can use Android as a software development platform. The software stack in the Android operating system, which is built on a modified Linux kernel, contains Java applications running on a virtual machine, and system components are written in Java, C, C++, and XML [9].

Creating an AIA application involves two stages.

- The first stage is designing the user interface with the webpage component designer.
    - It lets an implementer drag and drop components onto a simulated phone screen.
- The second stage is creating the application behavior with the blocks editor.
    - It allows the implementer to attach behaviors to the components.

Building an app for the ergonomics hand held device is the next step in saving time and increasing productivity.

ACKNOWLEDGMENTS

The author would like to thank Dr. Cheryl seals, Jesse Gamez, the Computer Science Software Engineering department and the Industrial System department of Auburn University for their help and guidance. This work has been supported in part by Distributed Research Experiences for Undergraduates (DREU).

REFERENCES

1. Kim, J. W. Human Computer Interaction. *Ahn Graphics* (2012); http://www.ittoday.info/Excerpts/HCI.pdf.

2. Clark, J. Chapter 1. *Tapworthy: Designing great iPhone apps*. "O'Reilly Media, Inc." (2010) 5-14; https://books.google.com/books?hl=en&lr=&id=ihYA7R1ZFWcC&oi=fnd&pg=PR2&dq=designing+apps&ots=xF9XJh4s9L&sig=ipCh7kDZ4X7Bv_JNthD20CSRBwA#v=onepage&q=designing%20apps&f=false.

3. Lee, A., & Lochovsky, F. H. User interface design. In *Office Automation*. Springer Berlin Heidelberg (1985) 3-20; http://link.springer.com/chapter/10.1007/978-3-642-82435-7_1.

4. Walker, H. Evaluating the Effectiveness of Apps for Mobile Devices. *Journal of Special Education Technology* (2011) 26(4), 59-63; http://ttools2.com/technology/evaluating_apps.pdf.

5. Vicente, K. J., & Rasmussen, J. Ecological interface design: Theoretical foundations. *Systems, Man and Cybernetics, IEEE Transactions on*, (1992) *22*(4), 589-606; http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=156574.

6. Lu, J., Chen, Q., & Chen, X. App interface study on how to improve user experience. In *Computer Science & Education (ICCSE), 2012 7th International Conference* (2012) 726-729. IEEE; http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6295176.

7. Landay, J., & Myers, B. Sketching interfaces: Toward more human interface design. *Computer* (2001) *34*(3), 56-64; http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=910894.

8. Miller, B. Hour 1. In *Sams Teach Yourself Swift in 24 Hours*. Indianapolis: Sams Publishing (2015) 1-9.

9. Butler, M. Android: Changing the mobile landscape. *Pervasive Computing, IEEE* (2011) *10*(1), 4-7; http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5676144.