The Usability Engineering Lifecycle

Deborah J. Mayhew
Deborah J. Mayhew & Associates
88 Panhandle Road
Post Office Box 248
West Tisbury, MA 02575-0248 USA
+1 508-693-7149
drdeb@vineyard.net
http://vineyard.net/biz/drdeb/index.html

ABSTRACT

The purpose of this tutorial is to provide a lifecycle of practical usability tasks and techniques for structuring the process of designing good user interfaces to either traditional software applications or Web pages and applets. The tutorial presents techniques which can be applied at different points in a typical product development lifecycle. Techniques presented include not only requirements analysis, design and testing techniques, but also organizational and managerial strategies.

Keywords

User interface design, user profile, task analysis, usability goals, style guide, conceptual model, usability testing, usability evaluation, usability organization, cost-benefit analysis

TUTORIAL CONTENT

This tutorial provides a lifecycle of practical techniques available for designing good user interfaces to either traditional software applications or Web pages and applets, which can be applied by software managers, developers and usability practitioners during the course of product development.

The success of any interactive computer system - from either a vendor or user point of view - depends on a number of different factors, including functionality, performance, cost, reliability, maintenance, and usability. All these factors are of roughly equal importance, and a serious failure in any one of them can cause a failure of the system as a whole.

In the past, least attention has been paid to the human interface, as designers, developers and customers focused on maximum functionality within cost and performance constraints. Few if any software engineering methodologies effectively address the issue of usability.

Over the past 15 years, however, there has been a growing recognition that the success of software products in the marketplace, and as tools in business environments, depends to a significant degree on the usability of the product, that is, on the quality of the user interface. Web developers are just beginning to recognize usability as a critical success factor.

Ad hoc design based on intuition and limited experience is not enough to insure the usability of a software application. The literature offers many principles for good interface design [1]. These principles can be helpful guidelines for designers. However, even if every designer in a software development organization was well versed in these design principles, this would not be enough to ensure good interface design. Many of the available design principles are based on experts' intuitions, rather than on hard data. Usually for any given design problem, they will come in direct conflict with each other, and there are no algorithms for making the tradeoffs. Design principles only bring the designer's attention to the issues which should be considered. There is no "cookbook" approach to applying these principles to ensure good interface design.

User interface design is a matter of compromise and tradeoff. We want powerful functionality, but a simple, clear interface. We want ease of use but also ease of learning. We want a system that is flexible but also one that provides good error handling. We strive for consistency across all aspects of the interface, but also to optimize individual operations. We want an "intelligent" and sophisticated interface, but also good performance and low cost. The interface designer finds him or herself constantly confronted with these kinds of conflicting goals.

Software designers and developers need an overall process to help them effectively structure the design of the user interface, and make good design decisions for a given

product with its particular set of end users. The cost-effectiveness of applying usability engineering is well documented [2]. The purpose of this tutorial is to present and discuss such a process within the overall context of a typical modern software development life cycle. Practice - rather than theory - is the main focus of this tutorial.

Topics presented include requirements analysis, design and testing tasks which can be applied at different points in the development process, *as well as organizational and managerial strategies.* Specific topics include:

- User Profile

- Contextual Task Analysis

- Platform Capabilities/Constraints

- General Design Principles

- Usability Goal Setting

- Work Reengineering

- Conceptual Model Design

- Conceptual Model Mockups

- Iterative Conceptual Model Evaluation

- Screen Design Standards

- Screen Design Standards Prototyping

- Iterative Screen Design Standards Evaluation

- Style Guide

- Detailed User Interface Design

- Iterative Detailed UI Design Evaluation

- User Feedback

Detailed instructions on *how* to carry out each of the usability engineering tasks presented are *not* offered (most tasks could be the sole topic of a 1-3 day tutorial), although brief overviews of each are provided during the tutorial. Instead the tutorial describes *what* techniques are available, and *when* and *why* to apply them in the context of the overall software engineering lifecycle. The main focus is on traditional software development, but how to adapt each technique to Web design is also addressed [3].

Wider organizational issues addressed include:

- Usability Project Planning

- Cost/Benefit Analysis

- Organizational Roles and Structures

During this tutorial, managers and developers learn to:

1) Integrate a usability engineering lifecycle into the overall development lifecycle

2) Set the stage for effective user interface design by applying usability requirements analysis techniques

3) Extract usability goals from requirements analysis data and use them to drive design

4) Apply a structured approach to user interface design

5) Apply iterative evaluation techniques to validate designs before they are implemented

6) Plan for and manage the use of usability engineering techniques within the overall project plan

7) Analyze the costs and benefits of applying usability engineering techniques during product development

8) Design organizational structures and processes to foster good interface design

Besides laying out an overall lifecycle and describing available usability techniques, the author brings 18 years of experience in software design and development, 13 as a usability consultant, to her lecture, offering examples of real life experiences using the techniques described. She also provides opportunities for the audience to share their experiences with the techniques described.

REFERENCES

1. Mayhew, Deborah J., **Principles and Guidelines in Software User Interface Design**, Prentice-Hall, 1992

2. Bias, Randolph G. and Mayhew, Deborah J., Eds., **Cost-Justifying Usability**, Academic Press, 1994

3. Forsythe, Chris, Grose, Eric and Ratner, Julie, **Human Factors and Web Development**, Lawrence Erlbaum Associates, 1998

4. Mayhew, Deborah J., **The Usability Engineering Lifecycle**, Morgan Kaufmann Publishers, 1999