Usability Engineering SENG 42222

Usability Engineering Life Cycle

Nimasha Arambepola
Lecturer (Probationary)
BSc (Hons) Software Engineering | MPhile (Reading)

Software Engineering Teaching Unit
Faculty of Science, University of Kelaniya

Introduction

- Usability Engineering is not a one shot affair where the user interface is fixed up before the release of the product.
- For a company that sells software or other products on the open market, the usability of each product will contribute to the company’s general reputation.

Usability Engineering Life Cycle

- Structured usability requirements analysis tasks

- An explicit usability goal setting task, driven directly from requirements analysis data
  - Tasks supporting a structured, top-down approach to user interface design that is driven directly from usability goals and requirements data
  - It is better to allocate substantial resources as early as possible.
    I. Product backward compatibility should be there
    II. Human factors involvement in the process has a great impact

- Objective usability evaluation tasks for iterating design towards usability goals

Usability Engineering Life Cycle

1. Know the user.
2. Competitive Analysis.
3. Setting Usability Goals.
4. Parallel Design
5. Participatory Design.
6. Coordinate design of the total interface.
7. Apply guideline and heuristic Analysis.
8. Prototyping.
9. Empirical Testing.
10. Iterative Design
11. Collect feedback field user.

1. Know the user

- Study the intended users and use of the product.
- Two factors largest impact on usability.
  - Individual user characteristics.
  - User variables.
- Users -Often includes not only the product/ application users but,
  - Installers
  - Maintainers
  - System Administrators
  - Other support staff

- Include the users of the systems and product or output even if they never see a single screen

Obstacles to developers to access users

- Need for the development company to protect its developers from being known to customers.
  - Customers may bypass established technical support and call developers directly.
  - fearing that the developers or usability people may offend the customer or create dissatisfaction with the current generation of products.
  - Only making users available for a short period of time.

1.1 Individual User Characteristic

- Necessary to know the class of people who will be using the system.
- When the product is going to be used in a specific department in a particular company.
  - Easy to identify the users.
- Users may be more scattered.
  - Products might be aimed towards the entire population or very large subset.
- Knowing the user's
  - Work experience.
  - Educational level.
  - Age
  - Computer Literacy.

1.2 Task Analysis

- User's overall goals should be studied.
  - how user approach the task.
  - information needs.
  - how they deal with exceptional circumstances or emergencies.
- Interviewing or observing the user's clients or others who interact with them can provide additional task analysis insights.
- When interviewing users for data collection, ask for the concrete examples.
- **Outcome of a task analysis** is a list of all the things users want to accomplish with the system, all the information they will need to achieve the goals.
  - the steps that need to be performed and the interdependencies between the steps.
  - all the outcomes and reports needs to be produced.
- Task analysis can be decomposed in a hierarchical fashion.

1.3 Functional Analysis

- Computerized communication tools build to take the advantage of the strengths of the computer medium such as,
  - Asynchronism
  - Anonymity
  - Searchable archives
  - Automated replies
  - Filters
- Functional analysis should be coordinated with a task analysis.

2. Competitive Analysis

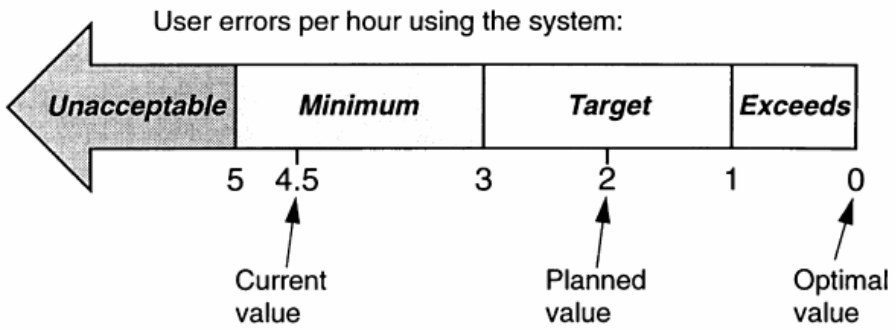
- Desirables to analyze existing products heuristically according to the established usability guidelines.
- Testing with existing products may work fairly well.
- Can perform a comparative analysis of their differing approaches to the various user interface design issues being studied.
    - Provide ideas for new design.

Note that a competitive analysis does not imply stealing other people's copyrighted user interface designs.

3. Goal Setting

- Have to make priorities clear on the basis of analysis of users and their tasks.
  - Eg: Learnability - If new employees were frequently being brought in on a temporary basis.
    - ability of infrequent users to return to the system would be especially important for a reconfiguration utility, once after every three or four months.

- Different usability parameters can be operationalized and expressed in measurable ways.
- For each usability attribute of interest, several different levels of performance can be specified as part of a goal setting process.
  - At least specify the minimum level which would be acceptable for release of the product.



An example of a usability goal line

4. Parallel Design

- When designers work out preliminary designs.
  - **Goal** - Explore different design alternatives before one settle on a single approach that can be developed in further detail and subjected to more detailed usability activities.
- Designers work individually. Because,
  - Aims at generating rough drafts of the basic design ideas.
- Designers work independently. Because,
  - Goal is to generate as much as diversity as possible.
- Multiple alternative designs.
  - Upper management decide which would be the final release.
- Variant of parallel design is called **diversified parallel design** and is based on asking the different designers to concrete on different aspects of the problem.
  - Eg:Designer could for the novice user, at the same time another designer designed an interface for the expert users

5. Participatory Design

- Even though the advice to “Know the User” may have been followed before the start of the design phase, one still can not know the user sufficiently well to answer all issues that come up in doing the design.
- Instead of guessing, designers should have access to a pool of representative users after the start of the design phase.
  - Users should be involved in the design process referred as subject matter
  - experts.(ASE)

6. Coordinating the total Interface

- Consistency should apply across the different media which form the total user interface
  - Application screens, documentation, online help system, tutorials etc.
- It is not just measured at a single point in time, it should apply over successive releases of a product
- To achieve consistency of the total interface
  - It is necessary to have some centralized authority
  - Interface standards are an important approach

7. Guidelines & Heuristic Evaluation

- General guidelines should be applicable to all the user interfaces
- Category specific guidelines
  - Eg: GUI or voice interfaces
- Product specific guidelines for the individual products

E.g: General guideline could be to “provide feedback” to the user about the system’s state & actions. These could be more specific for graphical user interfaces.

8. Prototyping

- One should not start full scale implementation efforts based on early user interface designs
- In traditional model of software development, intermediate working products and executable programs are produced at the last possible moment
- Prototyping - save the time and cost
  - Less emphasis on efficiency
  - Poor code quality
  - Use fake data
  - Can be used paper mockups
  - Relying on imaginary prototype
- Horizontal prototyping - Reducing the level of functionality
- Vertical prototyping - Cutting down on the number of features

9. Interface Evaluation

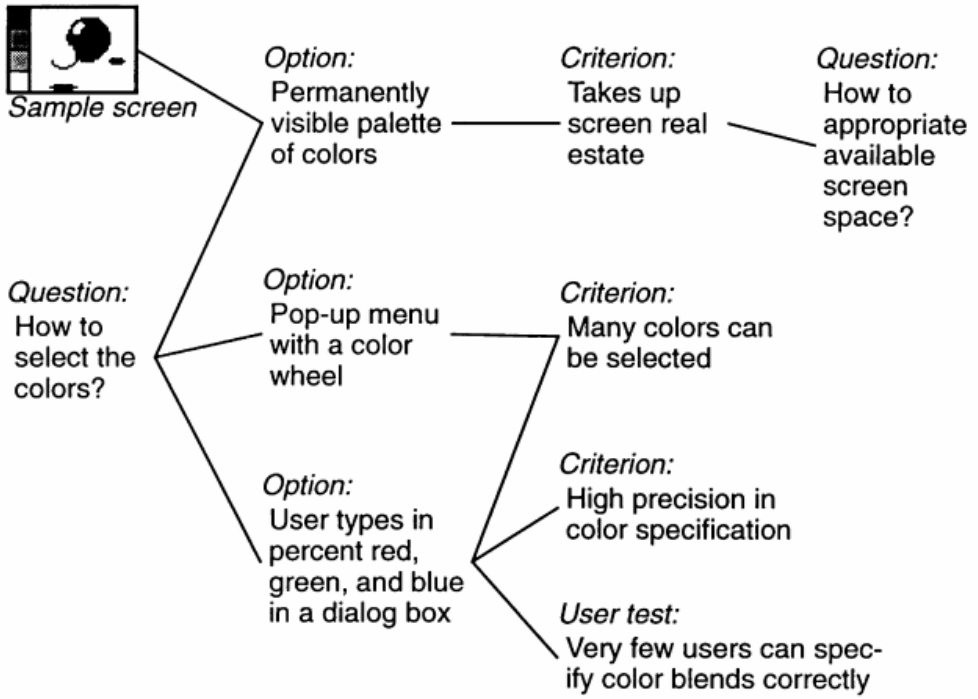
- Simply conduct some user testing
  - Testing real system by real users or prototypes
- Severity Rating
  - Interface users gather usability issues with interfaces and rank severities of each problem

		Proportion of users experiencing the problem	
		Few	*Many*
Impact of problem on the users who experience it	*Small*	Low severity	Medium severity
	Large	Medium severity	High severity

Table to estimate the severity of usability problems

10. Iterative Design

- Based on the usability problems, can develop new version of the system.
- During the iterative design process, it may not be feasible to test each successive version with actual users.



A partial example of a design rationale for a small part of an interface design for a hypothetical color paint program.

11. Follow up study

- Gather usability information for next version or for a new, future products.

Q & A