**Defining Usability**

The **ISO** defines usability as “The extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use.”

**Defining Usability (IOS)** The three measurable usability **attributes** defined by ISO are:

• **Effectiveness**: accuracy and completeness with which users achieve specified goals. • **Efficiency**: resources expended in relation to the accuracy and completeness with which users achieve goals. • **Satisfaction**: freedom from discomfort, and positive attitudes towards the use of the product.

**Six Usability Attributes (Nielsen)** Combining the three ISO usability attributes with Nielsen’s five usability attributes, leads to the following six usability attributes:

1. **Effectiveness**: completeness with which users achieve their goal.

2. **Learnability**: ease of learning for novice users.

3. **Efficiency**: steady-state performance of expert users.

4. **Memorability**: ease of using system intermittently for casual users.

5. **Errors**: error rate for minor and catastrophic errors. **6**. **Satisfaction**: how satisfying a system is to use, from user’s point of view.

**5. Measuring Usability:**

**Effectiveness:** Count how often users complete tasks correctly.

**Learnability**: Ask beginners to perform tasks and measure their time.

**Efficiency:** Measure time taken by expert users.

**Memorability:** After a break, ask casual users to use the system and record their performance.

**Errors:** Note how many mistakes (small or big) they make.

**Satisfaction**: Ask users how they feel using it (e.g., through a survey).

**Usability Evaluation Methods**

The methods of usability evaluation can also be classified according to who performs them:

• **Usability Inspection Methods**

Inspection of interface design by usability specialists using heuristics and judgement (no test users).

• **Usability Testing Methods**

Empirical testing of interface design with real users.

1. **Exploratory Evaluation**

Explores current usage and the potential design space for new designs:

• Done before interface development.

• Learn which software is used, how often, and what for.

• Collect usage data – statistical summaries and observations of usage.

1. **Predictive Evaluation**

Estimates the overall quality of an interface:

• Once a design has been done, but before implementation proceeds.

• Like a summative evaluation, but a prediction made in advance.

1. **Formative Evaluation**

Informs the design process and helps improve an interface during design;

• Done during interface development.

• Learn why something went wrong, not just that it went wrong.

• Collect process data – qualitative observations of what happened and why.

Formative evaluation methods are “find and fix” methods, and typically produce a list of problems found as output.

1. **Summative Evaluation**

Assesses the overall quality of an interface:

• Once an interface is (more or less) finished.

• Either compare alternative designs, or test specific performance requirements.

• Collect bottom-line data – quantitative measurements of performance: how long did users take, were they successful, how many errors did they make.

**What is Prototyping in HCI?**

Prototyping means making a basic version of a software or app to test ideas before building the final product. It helps get feedback early and improve the design.

**Key Points:**

1. **Prototype becomes the product:**

In modern development (especially agile), we don’t always throw away the prototype—we improve it step by step until it becomes the final product.

**2. Agile and Lean Methods:**

Agile: Make small, quick updates to the product regularly.Lean: Always improve the process (called Kaizen, which means "good change" in Japanese).

**Types of Prototypes (From Simple to Complex):**

**Verbal Prototype:** Just a text description of what the system will do.

**Paper Prototype:**

Low-Fidelity (Lo-Fi): Hand-drawn sketches.

High-Fidelity (Hi-Fi): Printed and colored designs using tools like Adobe.

**Interactive Sketch**: Scanned sketches made clickable.

**Working Prototype:**

May use fake data or simplified functions.

Sometimes involves a person pretending to be the system (Wizard of Oz method).

**Dimensions of Working Prototypes:**

Vertical: Deep features for a few parts.

Horizontal: All screens shown, but not functional.

Scenario: Only works for specific user paths or **examples**.

Implementation Tips:

Use ready-made tools or components like MS Windows UI or Java Swing.

Don’t build everything from scratch if you can reuse something.

**Popular Prototyping Tools**:

Adobe XD,

Balsamiq,

Axure,

HotGloo,

pidoco

SketchFlow,

dub.washington.edu