Usability

by S. Ketharaman

There have been several reasons given over the years to explain the success of Microsoft's software products. Rarely has usability featured in these discussions.

Many of us would recall how painful it was to draw a table in *WordStar* or *WordPerfect*. Microsoft *Word* made it simple, intuitive and fun to draw tables. The same goes for text formatting, mail merge, envelope printing, and so on.

Extra wide screens in many software packages force users to scroll horizontally. As a result, speed suffers because users need to lift their hands from the keyboards and keep reaching for the mouse. On the other hand, some packages have thoughtfully laid out their screens in such a way that horizontal scrolling is not needed and users can instead concentrate on their keyboards.

Many of us have come across web sites that take up several paragraphs to explain how to perform a certain task – say, check contents of shopping cart – when a hyperlink placed right there would have done the job easily. The 'Search' feature on different web sites result in wide-ranging hit lists: some web sites provide matches only if the search term matches exactly with the way the web site has stored it; others are smart enough to present a hit list of approximate matches. Some online shopping web sites even lack a shopping cart feature and force you to keep entering the same information (for example, name, address and credit card details) separately for each item you want to buy.

Poor usability leads to irritation and fatigue, even if we may not articulate our feelings. It has an adverse impact on the usage experience. In an online shopping web site, it can lead to loss of revenues. Poor usability of business applications leads to increased help desk costs.

On the other hand, better usability makes our usage experience more fun and can increase productivity. A highly usable online shopping web site tempts repeat visits, builds customer loyalty and increases its revenue-earning potential.

For a company developing a software product, better usability plays a significant role in defining the brand appeal and conveying a certain positive brand experience. The *enjoy*SAP initiative of SAP AG is a case in point. Owing to *enjoy*SAP, the previously difficult-to-use screens of SAP software were completely re-designed, making them much easier to use.

As more and more software products tend to become commodities, traditional functional criteria will take a backseat to usability aspects. Says Mark Rolston of frog design, "In future, the behavior of controlling elements can be configured in an objective or playful way, and thus become part of the brand experience." Operating elements such as zooming or scrolling menus may ultimately determine whether a product is accepted or rejected by the market.

In a broader context that is equally applicable to software and web sites, C K Prahalad, management guru and professor at the University of Michigan Business School, recently unveiled a roadmap wherein companies will have to make a rapid transition from product and consumer orientation to customer solution orientation, and finally to the personal consumer experience viewpoint[1].

While all of us have an intuitive understanding of what usability means, it is now time we looked at some definitions.

Fundamentally, usability is concerned with making systems easy to learn and easy to use. The term is used to describe the quality of a user's experience when interacting with

a system – whether a software package or a web site. A usable system is one which enables users to perform their job effectively and efficiently.

The International Standards Organization (ISO) defines Usability as the "effectiveness, efficiency and satisfaction with which a specified set of users can achieve a specified set of tasks in a particular environment."

Usability relates to both form and content. In business applications, usability lies in navigation, task-flow optimization, error reduction, productivity features, and task completion time.

Microsoft *Word* and *Excel* are two examples of excellent usability among software products. The *mail merge wizard* in *Word* make it easy to create form letters. *Word* has similar highly usable features for printing mailing labels and envelopes. *Excel* provides an excellent *paste function wizard* (invoked through the f_x button) for defining complex formulas with ease, and a highly usable *chart wizard* to quickly convert numbers to graphs.

Amazon and *Expedia* are examples of two web sites that display tremendous commitment to enhance the total usage experience.

Amazon's "1-Click ordering" feature improves usability by enabling the registered buyer quickly order an item without having to enter shipping address and credit card details during each buying session.

Amazon's recently launched 'Search Inside the Book' feature (see Figure 1) allows buyers to search millions of pages to find exactly the book they want to buy.

Figure 1: Amazon's Search Inside the Book

Now instead of just displaying books whose title, author, or publisher-provided keywords match their search terms, buyers' search results will surface titles based on every word inside the book (see Figure 2).
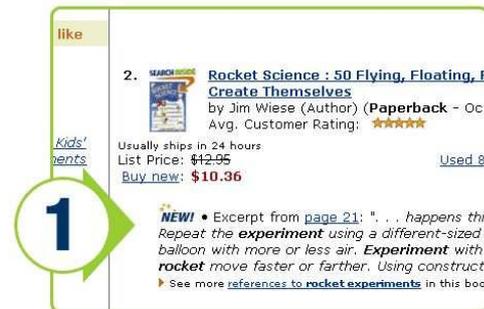

Figure 2: Search Inside the Book Results

When selling a used book, an obvious question in the mind of the seller is whether his asking price is too high or too low. *Amazon* helps sellers by providing a 'Used Price Guideline', which is computed not only on the basis of the price of a brand new version of the same book on Amazon, but also on the condition of the used book (see Figure 3).

Figure 3: Used Price Guideline

Immediately after you have booked a flight ticket, *Expedia* asks you if you need a hotel room, a travel insurance policy or a rent-a-car. If you do, Expedia takes you directly to the respective web pages so that you can order them quickly.

There are many travel web sites where you can order a hotel room, travel insurance and rent-a-car – but only if you specifically remember to locate these links and click them separately. Where *Expedia* scores is by prodding visitors to make these additional

purchases when they have completed the flight booking and are most likely to be "in the mood" for buying other travel necessities. It will not be a surprise if one discovers that *Expedia* sells more hotel rooms, travel insurance policies and car rentals per flight ticket it sells online.

If usability is so important, why do so many software packages and web sites lack it?

Usability requires empathy – the ability to put yourselves in the shoes of users and think as they would while interacting with the system. Programmers and systems analysts do not learn empathy during their technical education.

Another reason, to quote Gerd Waloszek of SAP Portals Product Design Center, is "release schedules are tight and, as a result, usability is still too often handled as an 'afterthought'"[2].

Standard software development methodologies include a stage that focuses on gathering functionality requirements, that is, WHAT the software is expected to do. Little or no attention is paid to defining usability requirements, which is HOW the software is expected to do what it is expected to do. Faced with a 'clean-slate' during the stage of defining requirements, users are barely able to define the WHAT aspects accurately. Without actually seeing the software, they are just not in a position to define the HOW aspects. For example, a business user who has not yet seen a to-be-developed customer order screen may at best be able to specify what pieces of data he would like to see in a customer order screen. It is virtually impossible for him to specify at this stage how the screen should be laid out and whether he is willing to accept horizontal scrolling, and details like that.

To the extent that usability has to do with fonts, color schemes, and so on, it may fall in the realm of graphics design. However, wizards, screen navigation, number of mouse-clicks required to perform a certain action, and so on, belong entirely in the domain of software engineering.

While better usability may sometimes call for more features, it can often be achieved within the framework of existing features. What it really takes is to make the features available to users when they are most likely to want them and in a form that they find easiest to use, as the *Expedia* example illustrates.

Let us see what steps could be taken to improve usability in the course of a software development or implementation project.

One, software engineers could build repositories of usability best practices from previous projects and earlier product versions. As we have seen from our various examples, basic elements like wizards, rightly-placed hyperlinks, shopping carts and navigation schemes can lead to significant improvements in usability. During the early stages of the project, these best practices could be used to guide users instead of expecting them to define usability requirements without seeing a single screen of the software.

Two, prototyping could be used more frequently to give the user a 'look-and-feel' of key screens before they are fully developed. Prototyping is used to eliminate misunderstanding and verify a solution at an early stage of design. Based on user feedback, mid-course design corrections could be made, if necessary. There are many tools available in the market that help the software engineer develop prototypes rapidly. Many of these tools allow re-usability so that the prototypes developed at the early stages can be used as part of the actual software being developed during the later stages. In other words, the prototypes need not be discarded.

Three, screen navigation and workflow could be demonstrated and signed-off before coding actually begins. Here, use could be made of a structured methodology like the *Delta Method* for better results.

Four, users volunteering to test beta versions of software could be encouraged to provide feedback not only on functionality and

performance issues, but also on usability. PeopleSoft, Inc. reports that extensive user testing involving 570 participants across 134 cities in six countries helped bring about significant usability improvements in its suite of enterprise application products[3].

Five, look for usability tips available on the Internet for specific areas. For example, *SAP Design Guild* for SAP developments and add-on products.

It is important to note that usability should be built in at the design stage. Once a software is developed, the cost of making changes to enhance usability is prohibitively high.

With increasing attention being placed on improving user experience as a means of promoting software branding, usability is gradually becoming "must-have" rather than "nice-to-have". Users can finally start looking forward to a time in the not too distant future when software and web sites are not just useful but also usable.

S. Ketharaman works for a globally active software services company headquartered in Pune, India. Views expressed here are personal. He can be reached at s_ketharaman@yahoo.com.

With a B.Tech from IIT Bombay and a Master's in Marketing Management from Jamnalal Bajaj Bombay, S. Ketharaman has 17 years of experience in the IT industry, including the last eight years of in depth exposure to SAP, PeopleSoft, Oracle and Ramco ERP packages. He has developed several frameworks and models in areas such as identification of differentiators, price forecasting and innovative service delivery. He has also authored several articles on topics like software usability, ROI of ERP and ERP customization. More details can be found on www.sketharaman.com.

---

[1] Source: The Economic Times dated 12 November 2005

[2] Source: http://www.sapdesignguild.org/community/editorials/print_editorial_04_2001.html

[3] *Cf.* "PeopleSoft Total Ownership Experience: Using Your Applications" brochure downloadable from www.peoplesoft.com.