Why Usable Interfaces Don't Just Happen

o sites

The concept of product "usability" began as an industrial design issue in the development of phy cal artifacts intended for human use. Subsequently, the concept was applied to the design of user interfaces (UIs), and "user-friendly" applications became the buzzword for developers. In context, a growing body of research and experience accumulated, enabling interface designe create intuitive UIs that shorten the learning curve and increase productivity for users. The gra user interface (GUI) itself is a prime example of this trend.

Unfortunately, many Web developers have neglected usability in their rush to get online. This result of regarding the Web as a graphical medium rather than a software product. The many Web designers focus on visual virtuosity without thinking through the complete user experience. An common mistake occurs on data-driven eCommerce sites. These sites are often built by softw neers whose principal concern is robust functionality, and the details of the users' interactions system are considered only after the fact. Developers who ensure solid usability enjoy a majo tage over their competitors who do not.

This paper will demonstrate why the success of a website depends in part on designing for usability from the beginning of the product lifecycle. We will also examine the tools and techniques that have been developed to measure web usability and design UIs. Finally, we will outline a methodology to ensure that the real needs of the end users are identified early in the development process and carried through to the final product.

Why won't the #%&*! Back Button work?

The High Cost of Poor Usability

Despite their desire to develop products that their customers find easy to use, product managers are sometimes reluctant to commit to engaging a human factors consultant or usability specialist on their web development projects.

This hesitation may be based on a belief that usability is a subjective quality—like color preference—which is difficult to define or predict. Perhaps, these product managers hope, usability can be hit or missed without unduly affecting the overall acceptability of the product. They view human factors as an expense with no measurable return.

In fact, usability is a highly measurable attribute. The metrics and heuristics are well documented. Usability testing protocols, information architecture principles, and interaction design methodologies have been developed to optimize UI designs. It is precisely this depth of specialized knowledge that requires a dedicated position on the web development team.

The contributions of a usability specialist save the whole team time and money. This occurs because an effective UI specification enables more efficient coding, with fewer false starts and less rework. It also greatly enhances the product's likelihood of success.

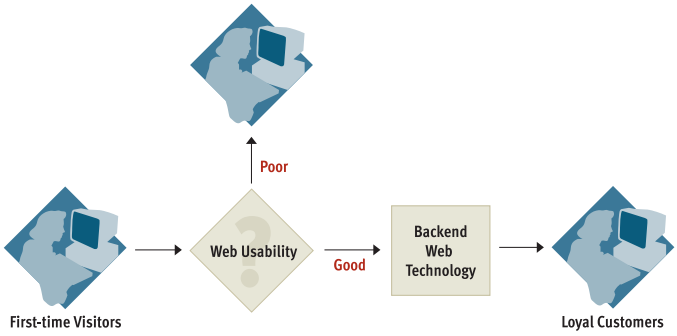
> Web culture has taught the consumer to vote with his or her mouse... Most consumers will not stop to complain about a site's usability—they will simply move on to a better site and never return.

An investment up front in good usability pays off handsomely over the whole life of the product. By contrast, the cost of neglecting usability up front can permanently hobble the product.

The costs of *poor usability*, on the other hand, are often hidden. Though they do not usually show up on the product manager's balance sheet, they do show up in other parts of the enterprise.

For example, poor usability may drive up Help Desk costs. This is a particular problem for web applications, such as Online Banking. The customer service department for the interactive banking division of a large national bank reports that a high proportion of its calls are due to usability issues.

However, the greater cost of poor usability on the Web is born by the eBusiness. It is much harder to measure, but far more important. Usability problems may become apparent from the moment customers enter a site. Web culture has taught the consumer to vote with his or her mouse. Unlike the captive audience for an online banking service, most consumers will not stop to complain about a site's usability—they will simply move on to a better site and never return.



Dissatisfied Users will never know how robust your system is.

Delivering Customer Satisfaction

Usability is all about delivering customer satisfaction. For many—if not most—consumers, conducting business transactions on the Web can be a frustrating experience. Nevertheless, it is quite possible to create a user experience that actually delights the customer, as amazon.com has proven.

Customer delight is the key to customer loyalty.

The key to designing a user interface that delights the customer is the precise definition of your customers and the identification of their goals. Human factors-based UI design brings the tools and techniques for creating a real user-centered experience to the overall development process of building an eBusiness site.

Usability Design Is *Not* a Matter of Opinion

Academic research and commercial practice have identified many heuristics that we can use to predict the response of users to Web environments.

Although schools are now graduating engineers trained in human factors and specializing in Web design, the practitioners who pioneered the study of user interface design drew upon a wide range of disciplines, including industrial design, cognitive psychology, human-computer interaction, library science, and visual design.

Testing need not be elaborate or expensive to be effective.

Through organizations like the Association for Computing Machinery (ACM) Special Interest Group for Computer Human Interaction (SIGCHI) and the Usability Professionals Association, human factors practitioners share practical experiences and promote best practices in usability design.

Standard practices can be loosely divided in two categories: evaluative and prescriptive. Evaluative tools include heuristic checklists and various user testing protocols. Prescriptive practices include information architecture and interaction design.

Heuristic Evaluation

Heuristic evaluations use established UI principles to identify known usability weaknesses and provide actionable suggestions for improvement.

Informal evaluations can be quickly performed by a trained visitor to a site. More formal evaluations can be done manually with checklists or through automated software agents.

Heuristic checklists identify critical features and functionality that define web usability. A site's score represents the percentage of the requirements on the checklist that it fulfills. Missing items represent opportunities for improving usability. Scores can be used to compare a site with its competitors.

Automated heuristics use software agents to explore a website and gather specified usability metrics. Such tools have the advantage of capturing performance metrics like loading times that also affect usability. These services are offered commercially by companies like WebCriteria (www.webcriteria.com) and Vividence (www.vividence.com).

Testing

User testing is a critical component of any serious usability effort. Many designers are intimidated by the idea of testing, believing it to be a complex, costly process. However, testing need not be elaborate or expensive to be effective.

Usability guru Jakob Nielsen says that five users will identify 80% of the problems in a given user interface. [2] Furthermore, testing need not be conducted on the fully functional product itself. Rather, tests may be done using limited HTML prototypes, PowerPoint mockups, or simple paper sketches. More important than the notion of rigorous, comprehensive testing is the principle of testing early and often.

Usability testing should not be confused with user acceptance testing (UAT), which is conducted at the end of the development process, once the application has been set in code. UAT is designed to find and fix bugs, not to address more fundamental conceptual usability problems. Such problems may require rewriting code.

With an existing site, however, usability tests provide valuable data upon which information architecture and interaction design changes can be based.

	User Acceptance Test	Usability Test
Who	Test Engineer	Actual User
What	Test Script	List of typical tasks
Where	Development Environment	Simulated user environment
When	After Development	Before Development
Why	Find & Fix Bugs	Ensure Satisfaction of User Goals

Information Architecture

The information architecture is the conceptual framework for the content of a website. A trained information architect (IA) is responsible for clarifying the mission and vision of a website, balancing the needs of the sponsoring organization with the needs of the audiences. The information architect enables the end users to find the information and functionality they need by specifying the *organization*, *navigation*, *labeling*, and *search systems* of the site. The information architect is also responsible for planning for changes and growth of the site over time. [3]

The work of the information architect begins with the classification and categorization of the site content. Viewing the content from a high level of abstraction, the architect uses principles of library science to identify classification systems appropriate for organizing the information. Then, using a process known as *content chunking*, specific content elements can be assigned to the relevant categories.

These content categories will lay the foundation for the site layout. Navigational structures will be driven in part by the nature of the content being presented. The information architect will also pay particular attention to labeling and naming standards, as these conventions will play a crucial role in the users' ability to understand their environment.

Although the responsibilities of the information architect are weighted toward site level design, there is also a page level component to the IA. Here the roles of information architect and interaction designer overlap. If we were to draw an arbitrary distinction between the two, we might say that the information architect determines what content belongs on a given page, whereas the interaction designer provides the visual designer with explicit guidance for the page layout and functionality, based on a well-researched understanding of the end user's goals. (The role of the interaction designer will be examined in greater detail in the following section.)

> Information Architecture should be consciously designed rather than defaulting from decisions made elsewhere in the development process.

The work of a professional information architect is critical to the success of a website. What happens when no information architect is involved in development? Every site has an information architecture, whether it was explicitly designed or it evolved by default. But without an IA to *organize the presentation of content around the needs of the end user*, the information architecture may be little more than a representation of the underlying database or the company org chart. Such representations rarely meet the needs of users.

Organizational Schemes & Structures

Rosenfeld and Morville [3] distinguish between logical content groupings (schemes) and the mechanisms for accessing that content (structures).

SCHEMES	STRUCTURES
Exact	**Hierarchical**
■ Alphabetical	Mutually exclusive categories
■ Chronological	Balance breadth vs. depth
■ Geographical	**Database**
Ambiguous	Structured
■ Topical	Homogenous information
■ Task-oriented	**Hypertext**
■ Audience-specific	Highly flexible, unstructured
■ Metaphor-driven	Potentially confusing
■ Hybrid	Most sites offer a combination of structures

In the absence of a dedicated information architect, the IA role may be performed by a business analyst, a visual designer, or a data model architect. If those team members are adequately trained in the principles of information architecture, this may be an acceptable development model in a low-resource environment. The critical point is that the IA should be consciously designed rather than defaulting from decisions made elsewhere in the development process.

However, there are potential conflicts of interest that arise when other team members attempt to take on the role of information architect. For example, the business analyst's primary responsibility is to ensure that the client's business model is clearly defined and translated to the web space, yet the end user does not care about the business model. Similarly, the visual designer is typically focused on page level design, emphasizing the look and feel of the user's environment, yet the site level design of the information architect will ignore those details to focus on a bigger picture. Another conflict of interest arises when software engineers pinch-hit for information architects: engineers must develop object and data models based on the site's content, but these necessary programming constructs are rarely intuitive to end users and make an unreliable basis for the information architecture.

In short, information architecture should not be left to chance. IA is always best executed by a skilled information architect. Professionals whose expertise lies in other domains may take on this role, but the results are rarely satisfying to end users.

Interaction Design

"Some real person is interacting with your product, not some abstract corporation, so you must regard people's personal goals higher than the corporation's."

—Alan Cooper, *The Inmates Are Running the Asylum*

The user interface of a website is often built around the business goals of the sponsoring organization. In eCommerce, the goal is typically to sell goods and services. The customer's true goal, however, is not the purchase of an item, but the satisfaction of a need. If we identify and satisfy that need, we will sell more products with far less pain to the customer.

Conversely, if the customer's needs are not met, they will go elsewhere, and the corporation's business goals will be unfulfilled, no matter how expertly the website addresses them.

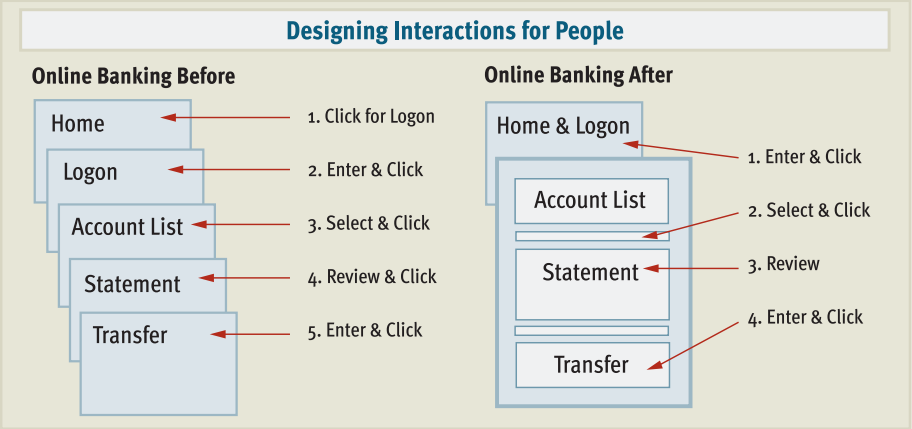
We may believe that we are taking a user-centered approach to web development because the design team habitually pays homage to a generic "user". "What does the user want?" we ask. In this situation, however, it is natural for each team member to picture himself or herself as the "user". This leads to a practice known as *self-referential design*. Unfortunately, the specialized concerns and considerations of the various team members give them points of view that rarely match those of real users.

> The customer's true goal is not the purchase of an item, but the satisfaction of a need. If we identify and satisfy that need, we will sell more products with far less pain to the customer.

For example, because they are trained to think in very logical and sequential structures, engineers often assume that users will be able to follow the reasoning they used to build the site. Most users do not have this mental discipline and are not naturally inclined to think that way. On the creative side of the team, graphic artists often operate under an imperative to create

"fresh, exciting, new" designs. Users, on the other hand, typically want something familiar and comfortable. They tend to perceive "exciting, new" environments as confusing and slow.

There are a variety of approaches to documenting user requirements. An engineering approach begins by defining *actors* and *use cases*. Actors represent all users of the system, both human and non-human. Use cases represent all possible tasks that can be performed within the system. A marketing approach might gather *demographic information* about existing customers, providing designers with information about the ages, genders, income, geographic distribution, and purchasing practices in the customer base. This information might be augmented by *focus groups* that explore other areas of customer preference.



Analysis of the usage statistics for a major online banking UI revealed that 98% of the users performed only 5 different types of tasks, All other tasks were limited to 2% of the users. The UI was designed to provide equal affordance to all types of tasks, whereas nearly 70% of the customers simply wanted to view their current statement. The diagram on the left shows the current pageflow for a customer who reviews her checking account statement and decides to transfer funds from savings. The diagram on the right shows how a simple re-design can provide the functionality desired by most customers with fewer pages and clicks.

While both of these approaches may provide a wealth of important data, they do not offer critical insights that are required for effective interaction design. Actors and use cases provide a comprehensive picture of the desired functionality, but they offer little guidance as to the frequency of a given task or its relative importance to users. Marketing demographics tell us *about* the users, but they do not help prioritize the conflicting requirements of different user groups.

To identify and prioritize end user goals (without falling into the trap of self-referential design), interaction designer Alan Cooper has pioneered a design technique based on customer archetypes. [1]

For each uniquely identifiable user group, interaction designers create a fictitious person who embodies the salient characteristics of that group. This archetype is given a name, personality, and history in addition to the group traits. Of all the archetypes created (one for each distinct user group), the designers identify one to three as the primary users. These are the "people" for whom the interface is designed.

> Instead of designing for "the user" as an abstraction, designers are focusing their efforts on satisfying "Sally".

Now, instead of designing for "the user" as an abstraction, designers are focusing their efforts on satisfying "Sally". Once Sally's needs are met, they can turn their attention to secondary and tertiary users. If "Tim" is a secondary user, they will try to satisfy as many of Tim's requirements as possible, *as long as they don't conflict with Sally's needs*. In this way, designers can create a UI that will *delight* Sally (and the many users she represents), while accommodating Tim and the other secondary users. Remember: A delightful experience engenders loyalty.

Beyond Tools

Despite the availability of robust tools to design for usability, the practice of human factors-based web development still poses a challenge for many eBusinesses. Project managers who would like to take advantage of these usability tools may be uncertain about how to integrate them into established engineering methodologies, especially when faced with the constraint of doing so in "Internet time".

In the next section, we will propose a framework for usability design that works both as a stand-alone methodology and as a component of a more traditional UML™ (Unified Modeling Language™) development process.

A Methodology for Developing Usable Web Products

A human factors methodology provides a robust structure which ensures that the voice of the customer (that is, the end user) will be heard by all participants throughout the product development life cycle. This "voice of the user" (VOTU) approach is most effective when it is fully integrated with the overall website development process.

What this approach will *not* do, however, is rescue a project that is already foundering under a poor UI. Too often, usability problems will become apparent as a web development effort nears completion, and the sponsors will ask for usability testing to provide a silver bullet cure. This rarely works. Fundamental usability problems do not lend themselves to quick fixes like changing font size or background color. They usually arise from conceptual issues that often require significant changes to code. If you wait to test the usability of your site until the development is nearly complete, you will likely find that any serious usability problems are either very expensive or even impossible to fix within time and budget constraints.

The VOTU approach outlines a set of deliverables that will help focus the site development process on the true goals of the end users, to the benefit of all the website's stakeholders. These deliverables correspond to or complement engineering documents created and used in the UML process. Associated with these deliverables we can define a set of roles—people responsible for the deliverables. Individual members of the web development team may take on one or more of these roles.

Voice of the User

Viewed from a high level of abstraction, there are three essential phases to the VOTU process: research, design, and validation. These phases are executed concurrently with the overall development process, with the balance of VOTU activities being weighted toward the front end of the project.

Research

The research phase begins with data gathered by business and technical analysts. Actors and use cases from the UML methodology provide important input. But usability analysis often requires other data gathering techniques, such as site evaluations, special interviews, or user testing. Note that usability testing is most effective before and during design. Tests can be conducted with paper prototypes or on competitive websites to yield valuable insights for design improvements.

Deliverables for this phase include documentation of the research itself as well as detailed analysis outlining the parameters of the actual UI design. Examples include:

- Heuristic evaluations
- Usability tests, user interview documents
- Customer Archetype Profiles (CAP)

Design

The design phase builds on the data compiled during research to produce the information architecture for the site and to define the precise mechanics and metaphors of the users' interactions with the site. Iterative testing of design concepts should continue to make necessary design corrections *before* committing valuable resources to code.

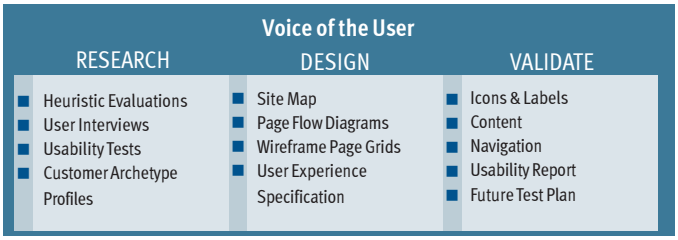
Deliverables for this phase provide documentation of the information architecture and user experience. Examples include:

- Site map and page flow diagrams
- Wireframe page grids (aka storyboards) defining page content and user functionality
- User Experience Specification (UES)

Validate

Web designers will test site conventions, such as graphical icons and labeling conventions, to ensure that their meanings are intuitive and unambiguous for their intended audiences. Information architects will monitor content to ensure that text and visual design are consistent with standard usability heuristics. Interaction designers will test navigation and functionality to ensure quality of user experience.

This phase may use testing methodologies similar to those used earlier, but the primary design issues should already have been resolved. If formal, site-wide usability tests are conducted at this late stage of development, the results will not be actionable. Fixing usability problems at this point is likely to result in a delayed launch or a huge cost overrun. Unless it is a mission critical issue—and it *is* far better to discover those before going live—it will be an enhancement request for the next version.



Team Roles

To properly execute the VOTU process, the following roles must be filled at some point during the implementation, either by a dedicated human factors resource or by a cross-trained member of the project team. Staffing budgets should reflect the additional time and expense required to produce the deliverables, but this will be money well spent. First, it will help *ensure that the final product meets that most demanding of all tests, the consumer.* Second, good usability design has been shown *to save resources later in the development process* by providing a more effective roadmap for developers.

Usability Analyst

The usability analyst assembles the actors and use cases from the UML process and indexes that data against customer demographics from marketing. The analyst role is also responsible for performing heuristic evaluations, managing usability testing programs, and conducting field research.

Information Architect

The information architect is the master of content, and ultimately responsible for the structure of the site. The information architect must identify, classify, and categorize the site content from a user-centered perspective, based on the CAPs. This data is used to drive the site map, which in turn suggests the navigational model of the site.

Interaction Designer

Interaction design is a complex discipline, with practitioners coming from a variety of fields, including cognitive psychology, computer science, graphic design, industrial design, and training.

In the VOTU approach, the interaction designer is the owner of the Customer Archetype Profiles. Although the CAPs are typically created in a collaborative effort of the entire design team, the interaction designer facilitates this process and is responsible for deploying the CAPs through the actual UI design.

In practice, the deliverables of the information architect and the interaction designer tend to overlap, and often both roles are played by a single team member. Another common practice is to let the graphic designer do the interaction design. What is important is that the UI interactions be consciously designed from the CAP perspective instead of emerging as the unintended consequences of other decisions.

> Good usability is good business.... And in the volatile environment of the World Wide Web, it may prove to be the most significant competitive advantage

Although there are obvious advantages to having dedicated human factors resources on complex projects, these roles are not restricted to usability specialists. A trained project manager could easily conduct heuristic evaluation and "guerilla" usability testing. A business analyst can double as an information architect. A graphic designer may also be a talented information designer. The point is that these roles must be staffed and the deliverables produced in order to ensure baseline usability.

Conclusion

In summary, good usability is good business. It is grounded in the respect for the customer and genuine desire for customer satisfaction that characterize our most successful business institutions. And in the volatile environment of the World Wide Web, it may prove to be the most significant competitive advantage.

As in any field that deals with predicting human behavior, the practice of human factors is a savvy blend of science and art. Through a disciplined methodology, the necessary customer considerations can be brought to the front of the line of design criteria, built into the system requirements, and embodied in the user interface design. Common design mistakes can easily be avoided, but more importantly, we can build Web products that delight our customers.

About eFORCE

eFORCE specializes in the Fixed Time, Fixed Price implementation of solutions encompassing all areas of the Enterprise Value Chain— Enterprise Portals, CRM, EAI, Business Intelligence, Enterprise Content Management and Enterprise Infrastructure. Combining expertise in business architecture, technical architecture, design, deployment and maintenance, eFORCE delivers production-scale enterprise solutions that result in measurable ROI. eFORCE customers include Global 1000 organizations such as Alcatel, AT&T, Avaya, Baker Hughes, Bank of America, British Telecom, Cisco, DHL, FedEx, Fleet Bank, French Telecom, GE Capital, GE Power, Hilton, HP, Intel, Janssen, Janus, Merck, Mitsubishi, Morgan Stanley, Novartis, Reuters, Shell, Viacom, Visa and Wells Fargo. eFORCE delivers solutions based on best-in-class enabling technologies such as ATG, BEA Systems, BroadVision, Documentum, E.piphany, HP, IBM, Interwoven, MatrixOne, Microsoft, Netegrity, Oracle, Plumtree, SeeBeyond, Siebel Systems, Stellent, Sun Microsystems, TIBCO and webMethods. eFORCE (www.eforceglobal.com) is headquartered in Silicon Valley, has Development Centers in North America, Europe and India, and through its Global Delivery Methodology provides both onshore and offshore full life-cycle implementation – design, development, deployment, maintenance, support.

Associations

ACM SIGCHI (Association for Computing Machinery Special Interest Group on Computer Human Interaction)
www.acm.org/sigchi/

UPA (Usability Professionals Association)
www.upa.org

Bibliography

[1] Cooper, Alan. *The Inmates Are Running the Asylum*, Publisher, 1999.

2] Nielsen, Jakob. *Designing Web Usability*, New Riders, 2000.

[3] Rosenfeld & Morville. *Information Architecture for the World Wide Web*, O'Reilly, 1998.

Glossary of Acronyms

CAP	Customer Archetype Profiles
IA	Information Architect or Information Architecture
UAT	User Acceptance Testing
UES	User Experience Specification
UI	User Interface
UML	Unified Modeling Language
VOTU	Voice of the User

Contact eFORCE

800.295.1914

510.265.5800

sales@eforceglobal.com