

# The Evaluation of Interactive Systems

## The why, what, where, and when of evaluation

Users want interactive products to be easy to learn, effective, efficient, safe, and satisfying to use. Being entertaining, attractive, challenging, and enjoyable is also important for the success of websites, games, toys, and other consumer products. Achieving this requires the product to be evaluated, and running effective evaluations involves understanding not only why evaluation is important but also what aspects to evaluate, where evaluation should take place, and when to evaluate.

### Why evaluate?

Evaluation is needed to check that users can use the product and that they like it, particularly if the design concept is new. Furthermore, nowadays users look for much more than just a usable system, they look for a pleasing and engaging experience.

*"User experience encompasses all aspects of the end-user's interaction ... the first requirement for an exemplary user experience is to meet the exact needs of the customer, without fuss or bother. Next comes simplicity and elegance which produces products that are a joy to own, a joy to use"*

*"Websites must be usable as well as aesthetically pleasing"*

From a business and marketing perspective there are also good reasons for investing in evaluation, these include: designers get feedback about their early design ideas, major problems are fixed before the product goes on sale; designers focus on real problems rather than debating what each other likes or dislikes about the product.

### What to evaluate

The wide diversity of interactive products gives rise to a range of features that evaluators must be able to evaluate. For example, developers of a new web browser may want to know whether users find items faster with their product, whereas developers of a creativity support tool for teaching story-writing may ask if students develop more engaging, emotionally satisfying stories. Government authorities may ask if a computerized system for controlling traffic lights results in fewer accidents. Makers of a toy may ask if six-year-olds can manipulate the controls and whether they are engaged by its furry case and pixie face, and whether the toy is safe for them to play with. A company that develops personal, digital music players may want to know whether the size, colour, and shape of the casing is liked by people from different age groups living in different countries. A new company may want to assess market reaction to its new homepage design.

From these examples, you can see that the success of many interactive products depends on much more than usability. Aesthetic, emotional, engaging, and motivating qualities are important too.

### Where to evaluate

Some features, such as the sequence of links on a website, are generally best evaluated in a laboratory, because it is easier for the evaluators to control the evaluation and to make sure that the evaluation focuses on specific aspects of the system. Similarly, testing the size of the keys on

a cell phone can be done in a laboratory. Other aspects, such as whether children enjoy playing with a collaborative toy, or whether an online dating system is emotionally satisfying, can be evaluated more effectively in natural settings, because evaluators can see what children do with the toy when left to their own devices, and can assess the emotional expressions of the adults in the dating system example. Likewise, evaluating how a cell phone is used and liked by different users such as busy business executives, or teenagers, involves observing how the users use the phone in their normal lives and talking to them about it.

## When to evaluate

At what stage in the product lifecycle evaluation takes place depends on the type of product itself. For example, the product being developed may be a brand-new concept or an upgrade of an existing product. If the product is new, then considerable time is usually invested in market research and establishing user requirements. Once these requirements have been established, they are used to create a design artefact such as initial sketches, a series of screens, or a prototype of the design ideas. These are then evaluated to see if the designers have interpreted the users' requirements correctly and embodied them in their designs appropriately. The designs will be modified according to the evaluation feedback and a new set of prototypes will be developed and then evaluated.

When a product is upgraded there is usually less scope for change than when designing a new product, and attention is focused on improving specific aspects of the product's design such as how to navigate through a website. Some products, such as office systems, go through many versions, and successful products may reach double-digit version numbers. If the product is being upgraded then evaluation is not necessarily dependent on establishing a set of requirements, and evaluation may be the first step in redesign.

Evaluations that are done to assess the success of a finished product, such as those needed to satisfy a funding agency that its money has been used well or to check that a standard is being upheld, are known as *summative evaluations*. Agencies such as the British Standards Institute (BSI), the National Institute of Standards and Technology (NIST) in the USA and the International Standards Organization (ISO) set standards by which products may be evaluated.

When evaluations are done during design to check that the product continues to meet users' needs they are known as *formative evaluations*. Formative evaluations cover a broad span of design, from the development of early sketches and prototypes through to tweaking and perfecting an almost finished design, and then maintaining the product, which may involve several upgrades.

## The Language of Evaluation

**Analytical evaluation:** an approach to evaluation that does not involve end-users. Heuristic evaluation, walkthroughs, and modeling are forms of analytical evaluation.

**Controlled experiment:** a study that is performed in a laboratory, which is controlled by the evaluator. Aspects controlled typically include the task that participants are asked to perform, the environment in which the study occurs, and the amount of time available to complete the study.

**Field study:** a study that is done in a natural environment such as at home, as opposed to a study in a controlled setting such as a laboratory.

**Formative evaluation:** an evaluation that is done during design to check that the product continues to meet users' needs.

**Heuristic evaluation:** an approach to evaluation in which knowledge of typical users is applied, often guided by heuristics, to identify usability problems.

**Predictive evaluation:** an approach to evaluation in which theoretically based models are used to predict user performance.

**Summative evaluation:** an evaluation that is done when the design is complete to assess whether it meets required standards such as those set by a standards agency like the BSI.

**Usability laboratory:** a laboratory that is designed for usability testing.

**User studies:** any evaluation that invokes users directly, either in their natural environments, or in the laboratory.

**Usability study:** an evaluation that is performed to examine the usability of a design or system.

**Usability testing:** an approach to evaluation that involves measuring users' performance and testing their satisfaction with the system in question on certain tasks in a laboratory setting.

**User testing:** an evaluation approach where users are asked to perform certain tasks using a system or prototype in an informal or laboratory setting.

## Evaluation paradigms and methods

At a general level, we describe evaluation studies as taking one of three main evaluation paradigms (i.e., broad approaches). Each of these is based on a distinct set of values and assumptions as to how evaluation should be conducted.

### Paradigms

The three main evaluation paradigms are: (1) usability testing, (2) field studies and (3) analytical evaluation. Each of these approaches has several methods associated with it. The methods used in evaluation are: observing users, asking users (e.g. through interviews and questionnaires), asking experts, user testing, inspections, and modeling users' performance. Some paradigms use the same methods. For example, usability testing and field studies both involve observing users and asking users but the conditions under which they are used, and the ways in which they are used, are different.

#### 1. Usability testing

Usability testing was the dominant evaluation paradigm in the 1980s and remains important, particularly at later stages of design for ensuring consistency in navigation structure, use of terms, and how the system responds to the user. Usability testing involves measuring typical users' performance on typical tasks. This is generally done by noting the number and kinds of errors that the users make and recording the time that it takes them to complete the task. As the users perform these tasks, they are watched and recorded on video and their interactions with the software are recorded, usually by logging input to and output from the system. User satisfaction questionnaires and interviews are also used to elicit users' opinions.

The defining characteristic of usability testing is that the test environment and the format of the testing is controlled by the evaluator. Quantifying users' performance is a dominant theme in usability testing. Typically, tests take place in a laboratory or in laboratory-like conditions where the user is isolated from the normal day-to-day interruptions. Visitors are not allowed and telephone calls are stopped, and there is no possibility of talking to colleagues, checking email, or doing any of the other tasks that most of us rapidly switch among in our normal lives.

Usability testing has been particularly important for the development of standard products that go through many generations, such as word-processing systems, databases, and spreadsheets. In this case, the findings from a usability test are summarized in a usability specification so that developers can test future prototypes or versions of the product against it. Optimal performance levels and minimal levels of acceptance are generally specified and current levels noted. Changes in the design can then be implemented. This is called 'usability engineering'.

## **2. *Field studies***

The distinguishing feature of field studies is that they are done in natural settings with the aim of understanding what people do naturally and how products have an affect on their activities. More specifically, they can be used to: (1) help identify opportunities for new technology; (2) establish the requirements for design; (3) facilitate the introduction of technology, or how to deploy existing technology in new contexts; and (4) evaluate technology. The data gathering techniques of interviews and observation are the basic techniques of field studies. The data takes the form of events and conversations that are recorded as notes, or by audio or video recording, and later analysed using a variety of methods. Artefacts are also collected and questionnaires may also be administered.

## **3. *Analytical evaluation***

In analytical evaluation two categories of evaluation methods are considered: inspections, which include heuristic evaluation and walkthroughs, and theoretically based models, which are used to predict user performance. In heuristic evaluations knowledge of typical users is applied, often guided by heuristics, e.g. guidelines and standards, to identify usability problems. Walkthroughs, as the name suggests, involve experts in walking through scenarios with prototypes of the application. A key feature for analytical evaluations is that users need not be present.

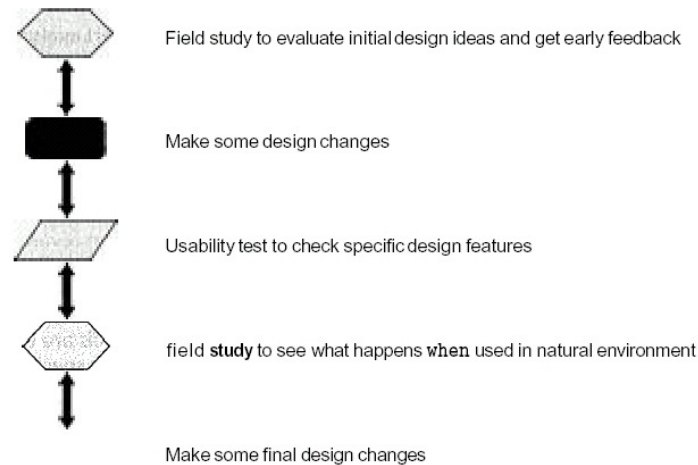
Heuristics are based on common-sense knowledge and usability guidelines, e.g. always provide clearly marked exits and use consistent terms. They were originally developed for screen-based applications but these have now been adapted to make new sets of heuristics for evaluating web-based products, mobile devices, collaborative technologies, and computerized toys. Care is needed in using heuristics because designers are sometimes led astray by findings from heuristic evaluations that turn out not to be as accurate as they seemed at first.

Cognitive walkthroughs, which were the first walkthroughs developed, involve simulating a user's problem-solving process at each step in the human-computer dialog, and checking to see how users progress from step to step in these interactions. A key feature of cognitive walkthroughs is that they focus on evaluating designs for ease of learning.

Models have been used primarily for comparing the efficacy of different interfaces for the same application, and the optimal arrangement and location of features on the interface base. For example, the keystroke level model provides numerical predictions of user performance and Fitts' Law predicts the time it takes to reach a target using a pointing device.

### ***Combining paradigms***

We have presented each evaluation paradigm separately, which implies that they are used independently of each other, and is sometimes true. However, this is often not the case. Combinations of approaches are used to get a broad understanding of the efficacy of a design. For example, sometimes the controlled studies performed in a usability laboratory are combined with observations intended to find out how users typically use the product in their natural environment. The following diagram illustrates one way in which usability testing and field studies may be combined in one program of evaluation.



*Example of the way usability testing and field studies can complement each other*

### ***Opportunistic evaluations***

Evaluations may be detailed studies or opportunistic investigations. The latter are generally done early in the design process to provide designers with feedback quickly about a design idea. Getting this kind of feedback early in the design process is an essential ingredient of successful design because it confirms whether it is worth proceeding to develop an idea into a prototype. Typically, these early evaluations are informal and do not require many resources. For example, the designers may go to a few local users and ask their opinions. Getting feedback this early in design can help save time and money if an idea needs to be modified or abandoned. In fact, opportunistic evaluations with users can be conducted often in addition to more formal evaluations.

### **Tables summarising the three Evaluation Paradigms and their Methods**

The first table summarises the key aspects of each evaluation paradigm for:

- The role of users
- Who controls the process and the relationship between evaluators and users during the evaluation
- The location of the evaluation
- When the evaluation is most useful
- The type of data collected and how it is analysed
- How the evaluation findings are fed back into the design process
- The philosophy and theory that underlies the evaluation paradigms.

	<b>Evaluation Paradigms</b>		
<b>Characteristic</b>	<b>Usability Testing</b>	<b>Field Studies</b>	<b>Analytical Evaluation</b>
Role of users	To carry out set tasks	Natural behaviour.	Users generally not involved
Who controls	Evaluators strongly in control	Evaluators try to develop relationships with users	Expert evaluators
Location	Laboratory	Natural environment	Laboratory-oriented but often happens on customer's premises
When used	With a prototype or product.	Most often used early in design to check that users' needs are being met or to assess problems or design opportunities	Expert reviews (often done by consultants) with a prototype, but can occur at any time. Models are used to assess specific aspects of a potential design
Type of data	Quantitative. Sometimes statistically validated. Users' opinions collected by questionnaire or interview.	Qualitative descriptions often accompanied with sketches, scenarios, quotes and other artefacts.	List of problems from expert reviews Quantitative figures from model, e.g., how long it takes to perform a task using two designs.
Fed back into design by ...	Report of performance measures, errors etc. Findings provide a benchmark for future versions.	Descriptions that include quotes, sketches, anecdotes, and sometimes time logs.	Reviewers provide a list of problems, often with suggested solutions. Times calculated from models are given to designers.
Philosophy	Applied approach based on experimentation, i.e., usability engineering.	May be objective observation or ethnographic.	Practical heuristics and practitioner expertise underpin expert reviews. Theory underpins models.

***Characteristics of different evaluation paradigms***

The second table gives details of the methods associated with each evaluation paradigm:

	<b>Evaluation Paradigms</b>		
<b>Methods</b>	<b>Usability Testing</b>	<b>Field Studies</b>	<b>Analytical Evaluation</b>
Observing users	Video and interaction logging, which can be analysed to identify errors, investigate routes through the software, or calculate performance time.	Observation is the central part of any field study. In ethnographic studies evaluators immerse themselves in the environment. In other types of studies the evaluator looks on objectively.	N/A
Asking users	User satisfaction questionnaires are administered to collect users' opinions. Interviews may also be used to get more details.	The evaluator may interview or discuss what she sees with participants. Ethnographic interviews are used in ethnographic studies.	N/A
Asking experts	N/A	N/A	Experts use heuristics early in design to predict the efficacy of an interface.
User testing	Testing typical users on typical tasks in a controlled laboratory-like setting is the cornerstone of usability testing.	N/A	N/A
Modeling users' task performance	N/A	N/A	Models are used to predict the efficacy of an interface or compare performance times between versions.

***The relationship between evaluation paradigms and methods***

# Evaluating interactive systems – basic issues and techniques

These notes introduce evaluation as a key element of interactive systems development, discuss different purposes and styles of evaluation and provides simple but effective evaluation techniques. The major focus is usability.

## Evaluating interactive systems design

By evaluation we mean reviewing, trying out or testing a design, a piece of software or a product to discover whether it is learnable, effective and accommodating for its intended user population.

**Definitions of usability** There are many of these, many of which have similar underlying philosophies. International standard ISO 9241 part 11 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.”

The designer is concerned not just with surface features such as the meaningfulness of icons, but also whether the application is fit for its purpose – anything from entertainment to routine order processing.

The techniques here will allow you to evaluate many types of applications. Some applications do not sit easily with this approach – particularly those intended to entertain, to provoke (as in the case of some art installations), to make a life-style statement or to be adopted and personalised as part of people’s everyday domestic life.

In a human-centred approach to design, we evaluate designs right from the earliest idea.

## Basic evaluation step-by-step

The list below summarises the main steps in undertaking a simple but effective evaluation project. Each is then explained in the material which follows.

1. Establish the aims of the evaluation, the intended users and context of use for the software; obtain or construct scenarios illustrating how the application will be used.
2. Select evaluation methods – should be a combination of expert review and end-user testing
3. Carry out expert review
4. Plan user testing; use the results of the expert review to help focus this
5. Recruit users and organise testing venue and equipment
6. Carry out user testing
7. Analyse results, write up and report back to designers

Evaluation is concerned with different issues at different times during the development of a product or system. The steps above are applicable to all of these.



Some of the most common *aims of evaluation* include:

### **Obtaining feedback to inform early design concepts**

You may need to evaluate initial concepts, especially if the application is novel for your users. Here quick ‘paper’ prototypes can help, or even software if this can be produced rapidly. Evaluations of competitor products or previous versions of technology can also feed into the design process at this stage.

### **Deciding between different design options**

During development, designers have to decide between options, for example between voice input or touch screen interaction for a shared electronic household wall-planner or between different sequences for order processing functions.

### **Checking for usability problems**

Testing will identify potential problems once a stable version of the technology is available. This needs to respond when the user activates a function, but not all the data processing components (for example) may be fully operational. Alternatively, the system may be completely functional, but only in some parts. What is important is that there is still time to fix problems. What happens all too frequently is that you are asked to check that interaction is ‘user friendly’ just before development is completed. Very often all that can be changed are minor issues such as the position, colour or labelling of on-screen buttons. It is best to be helpful in these circumstances. If you make a note of problems which could have been solved easily if identified sooner, you can exploit these examples (tactfully) to justify evaluation work at an earlier stage in the next project. Evaluation of the types described above is sometimes termed **formative evaluation**, because the results help to form – or shape – the design.

### **Assessing the usability of a finished product**

This may be to test against in-house guidelines, or formal usability standards, or to provide evidence of usability required by a customer, for example the time to complete a particular set of operations. Government departments and other public bodies often require suppliers to conform with accessibility standards and health and safety legislation. This type of evaluation is sometimes termed **summative**.

### **As a means of involving users in the design process**

In participatory design approach, users help designers set the goals for the evaluation work. Involving users has great benefits in terms of eventual uptake and use of the technology. (Of course, this applies only to technology which is tailor-made for defined user communities, rather than off-the-shelf products.)

### **Assessing use in practice**

Researchers undertake long term evaluations as a means of understanding the success of particular technologies. Such an approach is relatively rare in commercial practice, though indirect data are often collected and used to inform the next release of a product. These might include problems reported by customers or end-users, customer reactions to salespersons’ pitches or requests for modifications.

## **Expert evaluation basics**

The most widely used form of expert review, heuristic evaluation, involves experts checking the application systematically against a list of principles, guidelines or 'heuristics' for good design. Usability heuristics draw on psychological theory and practical experience.

There are many sets of heuristics to choose from, both general purpose to those relating to particular application domains, for example the heuristics for web design. Perhaps the best known set of heuristics are the ten proposed by Nielsen:

### ***Visibility of system status***

The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.

### ***Match between system and the real world***

The system should speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order.

### ***User control and freedom***

Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Support undo and redo.

### ***Consistency and standards***

Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.

### ***Error prevention***

Even better than good error messages is a careful design which prevents a problem from occurring in the first place. Either eliminate error-prone conditions or check for them and present users with a confirmation option before they commit to the action.

### ***Recognition rather than recall***

Minimize the user's memory load by making objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate.

### ***Flexibility and efficiency of use***

Accelerators -- unseen by the novice user -- may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions.

### ***Aesthetic and minimalist design***

Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.

### ***Help users recognize, diagnose, and recover from errors***

Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.

### ***Help and documentation***

Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.

## How many evaluators?

Ideally, several people with expertise in interactive systems design should review the interface. Valuable insights can be also gained from technical authors and support and help desk staff. It has been well documented that lone evaluators will only find around 35% of usability problems. Nielsen suggests that the optimal number of evaluators (in cost-benefit terms) is around 5, but this is infeasible in many situations and worthwhile results can be obtained even if only one expert is used. Each expert notes the problems, the relevant heuristic, and suggests a solution where possible. It is also helpful if a severity rating, say on a scale of 1 – 3, is added, according to the likely impact of the problem. Evaluators work independently and then combine results. They may need to work through any training materials users will be given, or at least be briefed by the design team about the functionality. The scenarios used in the design process are valuable here.

Unless there is no alternative, you should not evaluate your own designs. It is extremely difficult to ignore your knowledge of how the system works, the meaning of icons or menu names and so on, and you are likely to give the design the ‘benefit of the doubt’ or to find obscure flaws which few users will ever happen upon.

## A basic heuristic evaluation step-by-step

Step	Notes
0. Establish the aims of the evaluation and the intended users and context of use for the software.	The IMPACT model – described later – will help you do this.
1. Select heuristics	Add specific heuristics as necessary.
2. Brief evaluators about the technology and how it is intended to be used.	Design scenarios are very useful here; training may be necessary for complex applications. A member of the design team may need to be available during the session to help evaluators.
3. Evaluators independently make a first pass through the design.	The aim is to gain an impression of how the system fits together and its overall design.
4. Evaluators independently examine the design in more detail, working through typical or critical scenarios.	Review against each heuristic, and focus on areas that are particularly important, or are typical of a set of similar interactions. A standard problem report form is useful.
5. Evaluators produce a consolidated list of prioritised problems linked to heuristics and suggested solutions.	Adding solutions and severity ratings increases the chance of having the problem fixed. Evaluators should review the list to remove any problems which, although difficulties in theory, are unlikely to hinder a sensible user.

The results of the expert review also guide the focus of testing with users.

## **The IMPACT model for user evaluations**

While expert analysis is a reasonable first step, it will not find all problems, particularly those which result from a chain of ‘wrong’ actions or are linked to fundamental misconceptions.

Experts often find problems which do not really exist – users overcome many minor difficulties using a mixture of common sense and experience. So it is really important to complete the picture with some real people trying out the interaction design. The findings will always be interesting, quite often surprising and occasionally disconcerting. From a political point of view, it is easier to convince designers of the need for changes if the evidence is not simply one ‘expert’ view, particularly if the expert is relatively junior.

The aim is to trial the design with people who represent the intended target group in as near realistic conditions as possible.

The decisions to be made in planning testing can be encapsulated in the acronym ‘IMPACT’ – we are after all concerned with assessing the ‘impact’ of technology on users. As well as the components of PACT – People, Activities, Context and Technologies – we now have Intention and Metrics.

### **Intention**

Deciding the aim(s) for evaluation helps to determine the type of data required. It is useful to write down the main questions you need to answer.

In the case of early concept evaluation, for instance, the data we are interested in is largely qualitative (non-numerical). We might however want to determine the relative strength of opinions. Rating scales (see later) are a simple way of doing this. By contrast, the comparison of two different evaluation designs usually entails much more focussed questions. Underlying issues might concern speed and ease of operation, in which case we are likely to need quantitative (numerical) data to support design choices. You will need to consider each situation individually. Ideally, the focus should be discussed with designers and clients or end-users as appropriate.

### **Metrics (and measures)**

What is to be measured and how? Metrics are helpful in evaluating many types of applications from small mobile communication devices to office systems. In most of these there is a task – something the user wants to get done – and it is reasonably straightforward to decide whether the task has been achieved successfully or not. There is one major difficulty: deciding the acceptable figure for, say, the percentage of tasks successfully completed. Is this 95%, 80% or 50%? In some (rare) cases clients may set this figure. Otherwise a baseline may be available from comparative testing against an alternative design, a previous version, a rival product, or the current manual version of a process to be computerised. But the evaluation team still has to determine whether a metric is *relevant*. For example, in a complex computer aided design system, one would not expect most functions to be used perfectly at the first attempt. And would it really be meaningful if design engineers using one design were on average 2 seconds quicker in completing a complex diagram than those using a competing design? By contrast, speed of keying characters may be crucial to the success of a mobile phone.

Here are some common usability metrics and ways in which they can be measured, using the usability definition of ‘effectiveness, efficiency and satisfaction’. There are many other possibilities.

Usability objective	Effectiveness measures	Efficiency measures	Satisfaction measures
Overall usability	Percentage of tasks successfully completed	Time to complete a task	Rating scale for satisfaction
	Percentage of users successfully completing tasks	Time spent on non-productive actions	Frequency of use if this is voluntary (after system is implemented)
Meets needs of trained or experienced users	Percentage of advanced tasks completed	Time taken to complete tasks, relative to minimum realistic time	Rating scale for satisfaction with advanced features
	Percentage of relevant functions used		
Meets needs for walk up and use	Percentage of tasks completed successfully at first attempt	Time taken on first attempt to complete task	Rate of voluntary use (after system is implemented)
		Time spent on help functions	
Meets needs for infrequent or intermittent use	Percentage tasks completed successfully after a specified period of non-use	Time spent re-learning functions	Frequency of re-use (after system is implemented)
		Number of persistent errors	
Learnability	Number of functions learned	Time to learn to criterion	Rating scale for ease of learning
	Percentage of users who manage to learn to a pre-specified criterion	Time spent on help functions	

There are three things to keep in mind when deciding metrics:

- just because something can be measured, it doesn’t mean it should be;
- always refer back to the overall purpose and context of use of the technology;
- consider the usefulness of the data you are likely to obtain against the resources it will take to test against the metrics.

### **Engagement**

*Games and other applications designed for entertainment pose different questions for evaluation. While we may still want to evaluate whether the basic functions to move around a game environment (for example) are easy to learn, efficiency and effectiveness in a wider sense are much less relevant. The ‘purpose’ here is to enjoy the game, and time to complete (for example) a particular level may sometimes be less important than experiencing the events that happen along the way. Similarly, multimedia applications are often directed at intriguing users or evoking emotional responses rather than having the achievement of particular tasks in a limited period of time. In contexts of this type, evaluation centres on probing user experience through interviews or questionnaires, for example, using a rating scale presented as a ‘smiley face fun meter’ when working with children to evaluate novel interfaces. Other measures which can be considered are observational: the user’s posture or facial expression, for instance may be an indicator of engagement in the experience.*

## People

The most important people in evaluation are users. Requirements analysis work should have identified the characteristics of intended users, but sometimes you will have to work with the designers to obtain this information. Relevant data can include knowledge of the activities the technology is intended to support, skills relating to input and output devices, experience, education, training and physical and cognitive capabilities.

You need to recruit at least 3 and preferably 5 people to participate in tests. They should be typical of the intended users. Although Nielsen recommends a sample of 3 to 5 users, some practitioners and researchers advise that this is too few. However, in many real world situations obtaining even 3 to 5 people may be difficult, and so one may need to be content with smaller test numbers as part of a pragmatic evaluation strategy.

However, testing such a small number only makes sense if you have just one main type of target user – for example, experienced users of a customer database system, or computer games players aged between 16 and 25. If you have several different user groups, then you should try to run 3 to 5 people *from each group* through your tests. If your product is to be demonstrated by sales and marketing personnel, it is useful to involve them as a special type of user group. Finding users should be straightforward if you are developing an in-house application. Otherwise they can be found through product user groups, focus groups established for marketing purposes or if necessary, through advertising. Students are often readily available, but remember that they are only representative of a particular segment of the population. If you have the resources, payment can help recruitment. Inevitably, your sample will be biased towards cooperative people with some sort of interest in technology, so bear this in mind when interpreting your results.

If you cannot recruit any genuine users, and you are the designer of the software, at least have someone else try to use it. This could be one of your colleagues, a friend, your mother or anyone you trust to give you a brutally honest reaction. Almost certainly, they will find some design flaws. The data you obtain will be limited, but better than nothing. You will, however, have to be *extremely* careful as to how far you generalise from your findings.

Finally, consider your own role and that of others in the evaluation team if you have one. You will need to set up the tests and collect data, but how far will you become involved? Basic testing requires an evaluator to sit with each user and engage with them as they carry out the test tasks. For ethical reasons and in order to keep the tests running, you should provide help if the user is becoming uncomfortable, or completely stuck. The amount of help which is appropriate will depend on the type of application (e.g. for an information kiosk for public use you might only provide very minimal help); the degree of completeness of the test application and, in particular, whether any help facilities have been implemented.

## Activities

It is equally important to consider typical activity with the technology. Concrete scenarios can be employed here. They should cover typical uses of the technology, and rare but critical events. For example, in the evaluation of a bank auto teller (ATM), scenarios might be created for withdrawing money, obtaining a mini-statement and forgetting to remove one's bank card. Employ scenarios to set the scene for users, and to draw up the list of actions which they will undertake.

## **Contexts**

All the elements of IMPACT contribute to the context of the evaluation. But what we are specifically concerned with here is the wider social and physical context which may affect the way the technology is used. This may include:

- Patterns of activity – is this just part of another activity; is the user's work monitored; do users have control of pace or order in which tasks are undertaken; is use likely to be interrupted; is use voluntary?
- Assistance – are other people around to help out if there are problems?
- Is the technology single-user or to be used with others (as in many games) or in co-working?
- The physical set-up of the technology – at a desk, on a small portable device, as a free-standing information kiosk, as a display in a car or other vehicle?
- The wider physical environment – inside or outside, noisy (or quiet), hot, humid or cold, are there unusual lighting conditions?
- Social norms – are users likely to feel time-pressured by others waiting to use the application, as in the case of an auto teller or a ticket machine; are users expected to be particularly accurate?

Consideration of these points will help to make the evaluation as ecologically valid (i.e. close to the context of use) as possible and guide the metrics to be applied.

You will need a place where the context of use can be re-created as far as possible, the technology can be installed, there is space for you and anyone else helping you with the evaluation, and you will not be disturbed. If you are undertaking a good deal of evaluation work, a dedicated usability laboratory could be justified, but most of us do not have that luxury.

## **Technologies**

If near the end of the development process, you should be evaluating on a machine or device which is as near as possible to that on which the product will be delivered. For networked applications, you should also take into account issues such as network speed and reliability.

Also decide the technology to support the user tests. At a minimum, you will need pen and paper for recording observations and printed copies of any instructions or other materials for users and expert evaluators. You may also want to make video or audio recordings.

## The test plan and task specification

Having considered each of the IMPACT elements, a plan should be drawn up to embody the decisions. The plan specifies:

- Aims of the test session;
- Practical details including where and when it will be conducted, how long each session will last, the specification of equipment and materials for testing and data collection and any technical support that may be necessary;
- Numbers and types of user
- Tasks to be performed with a definition of successful completion. This section also specifies what data should be collected and how it will be analysed.

You should conduct a pilot session and fix any unforeseen difficulties. For example, task completion time is often much longer than expected, and instructions may need clarification.

**The Cooperative Usability Evaluation** is a usability testing technique developed by Monk as a means of maximising the data from a simple testing session. The technique is ‘cooperative’ because users are not passive subjects but work as co-evaluators. It has proved a reliable but economical technique in diverse applications.

Step	Notes
1. Using scenarios prepared earlier, write a draft list of tasks.	Tasks must be realistic, do-able with the software, and explore the system thoroughly.
2. Try out the tasks and estimate how long they will take a user to complete.	Allow 50% longer than the total task time for each user test session.
3. Prepare a task sheet for the users.	Be specific and explain the tasks so a novice user can understand.
4. Get ready for the test session.	Have the prototype ready in a suitable environment and list of prompt questions, notebook and pens ready. An audio recorder would be very useful here.
5. Tell the users that it is the system that is under test, not them; explain the procedure and introduce the tasks.	Users should work individually – you will not be able to monitor more than one user at once. Start recording if equipment is available.
6. Users start the tasks. Have them give you a running commentary on what they are doing, why they are doing it and difficulties or uncertainties they encounter.	Take notes of where users find problems, do something unexpected, and their comments. Do this even if you are recording the session. You may need to help if users are stuck or have them move to the next task.
7. Encourage users to keep talking.	Some useful prompt questions are provided below.
8. When the users have finished, interview them briefly about the usability of the prototype and the session itself. Remember to thank the users.	Some useful questions are provided below. If you have a large number of users, a simple questionnaire may be helpful.
9. Write up your notes as soon as possible and incorporate into a usability report.	



### **Sample questions during the test session**

What do you want to do?

What were you expecting to happen?

What is the system telling you?

Why has the system done that?

What are you doing now?

### **Sample questions after the session**

The best/worst thing about the prototype.

What most needs changing?

How easy were the tasks?

How realistic were the tasks?

Did giving a commentary distract you?

## **Data capture techniques for usability evaluation**

As we can see from the instructions for Cooperative Evaluation, the same basic techniques used for obtaining information from people in the requirements analysis stage are used in evaluation. The questions and any observation checklists are now, of course, geared to people's use of the new system, but the same general considerations apply in their design. A typical deployment of these techniques in usability testing would be to have test users work through one or more scenarios then be interviewed by a member of the evaluation team or complete a short questionnaire about the experience. An evaluator may also observe users and will often ask questions during the process.

### **Interviewing in usability evaluation**

Short individual interviews after testing are very valuable in clarifying user reactions, or amplifying the answers to questionnaire items. You could also interview users in a group – this can save time and users will prompt each other - but beware of people dominating the discussion. Some typical generic questions are included above in the Cooperative Evaluation section. Other topics might be:

- Clarity of function presentation
- Adequacy of functionality
- Clarity of feedback and any system messages
- Any particular difficulties encountered
- Whether the user would choose to use the application
- Realism of test tasks
- Clarity of test instructions

An alternative to Cooperative Evaluation is to videotape the test session and then talk through with users what was happening. This helps to avoid distraction during the test session but does run the risk of *post hoc* rationalisations.

## Designing questionnaires for usability evaluation

Questionnaires can gather basic background data about user characteristics and reactions after the test session. Relevant user characteristics should have been identified when planning the testing and a set of check-boxes or short free-text answers can be used to collect this data. User perceptions of interaction design are usually collected through rating scales, for example using this five-point scale:

1. Strongly agree
2. Agree
3. Neutral
4. Disagree
5. Strongly disagree

The scale is attached to each of a number of statements such as:

I always knew what I should do next (*tick one box*)

- |                   |          |            |             |                      |
|-------------------|----------|------------|-------------|----------------------|
| 1. Strongly agree | 2. Agree | 3. Neutral | 4. Disagree | 5. Strongly disagree |
|-------------------|----------|------------|-------------|----------------------|

Icons were easily understandable

- |                   |          |            |             |                      |
|-------------------|----------|------------|-------------|----------------------|
| 1. Strongly agree | 2. Agree | 3. Neutral | 4. Disagree | 5. Strongly disagree |
|-------------------|----------|------------|-------------|----------------------|

The destination of links was clear

- |                   |          |            |             |                      |
|-------------------|----------|------------|-------------|----------------------|
| 1. Strongly agree | 2. Agree | 3. Neutral | 4. Disagree | 5. Strongly disagree |
|-------------------|----------|------------|-------------|----------------------|

Make the questionnaire items as specific as possible. A probe statement such as ‘The system was easy to use’ does provide a general impression but gives very little information for redesign if you do not supplement it. Another approach is to devise ‘bipolar’ rating scales with individual labels for particular aspects of the software. These take longer to construct, but are helpful in probing specific reactions. For example:

The screen layout was (*circle one number*)

Neat & clear	1	2	3	4	5	Cluttered
--------------	---	---	---	---	---	-----------

Operation of functions was

Consistent	1	2	3	4	5	Inconsistent
------------	---	---	---	---	---	--------------

## Observation in usability evaluation

It is difficult for people to verbalise all their perceptions. People may not know why they find certain aspects difficult, or be reluctant to admit confusion, or simply unable to recall their thought processes. For this reason it is best to sit with users as they carry out the test tasks, noting their actions and reactions. It may be less obtrusive to videotape the sessions, with the added benefits that the recordings can be replayed for analysis and are very effective in communicating results to designers.

## Reporting usability evaluation results to the design team

However competent and complete the evaluation, it is only worthwhile if the results are acted upon. Even if you are both designer and evaluator, you need an organised list of findings so that you can prioritise redesign work. If you are reporting back to a design/development team, it is crucial that they can see immediately what the problem is, how significant its consequences are, and ideally what needs to be done to fix it.

The report should be ordered either by areas of the system concerned, or by severity of problem. For the latter, you could adopt a three or five point scale, perhaps ranging from 'would prevent user from proceeding further' to 'minor irritation'. Adding a note of the general usability principle concerned may help designers understand why there is a difficulty, but often more specific explanation will be needed. Alternatively, sometimes the problem is so obvious that explanation is superfluous. A face-to-face meeting may have more impact than a written document alone (although this should always be produced as supporting material) and this would be the ideal venue for showing *short* video clips of user problems.

Suggested solutions make it more probable that something will be done. Requiring a response from the development team to each problem will further increase this probability, but may be counter-productive in some contexts. If your organisation has a formal quality system, an effective strategy is to have usability evaluation alongside other test procedures, so usability problems are dealt with in the same way as any other fault. Even without a full quality system, usability problems can be fed into a 'bug' reporting system if one exists. However, whatever the system for dealing with design problems, tact is a key skills in effective usability evaluation.

## Summary and key points

These notes have provided an overview of what needs to be considered in evaluating interactive systems, using the IMPACT framework and provided practical guidance in using simple but effective expert and end-user techniques. We have illustrated some of these with the real-life example of evaluating a user interface.

We have seen that:

- Differing aims for evaluation require different questions to be answered
- Expert review and end-user testing are both effective, but should be used together as complementary methods
- Almost any degree of user testing can reveal useful insights, but care must be taken in generalising from a small number of users
- Factors to take into account in user testing can be summarised as Intention, Metrics and of course, People, Activities, Context and Technologies (IMPACT)
- The Cooperative Usability Evaluation method affords a practical way of testing usability in an economical manner
- Data collection techniques for evaluation mirror those for requirements analysis.

## HutchWorld evaluation case study

A case study is presented about the evaluation techniques used by Microsoft and the Fred Hutchinson Cancer Research Center in developing HutchWorld (Cheng et al., 2000), a virtual world to support cancer patients, their families, and friends. This case study is chosen because it illustrates how a range of techniques is used during the development of a new product. It introduces some of the practical problems that evaluators encounter and shows how iterative product development is informed by a series of evaluation studies.

HutchWorld is a distributed virtual community developed through collaboration between Microsoft's Virtual Worlds Research Group and librarians and clinicians at the Fred Hutchinson Cancer Research Center in Seattle, Washington. The system enables cancer patients, their caregivers, family, and friends to chat with one another, tell their stories, discuss their experiences and coping strategies, and gain emotional and practical support from one another (Cheng et al., 2000). The design team decided to focus on this particular population because caregivers and cancer patients are socially isolated: cancer patients must often avoid physical contact with others because their treatments suppress their immune systems. Similarly, their caregivers have to be careful not to transmit infections to patients.

The big question for the team was how to make HutchWorld a useful, engaging, easy-to-use and emotionally satisfying environment for its users. It also had to provide privacy when needed and foster trust among participants. A common approach to evaluation in a large project like Hutchworld is to begin by carrying out a number of informal studies. Typically, this involves asking a small number of users to comment on early prototypes. These findings are then fed back into the iterative development of the prototypes. This process is then followed by more formal usability testing and field study techniques. Both aspects are illustrated in this case study. In addition, you will read about how the development team managed their work while dealing with the constraints of working with sick people in a hospital environment.

### How the design team got started: early design ideas

Before developing this product, the team needed to learn about the patient experience at the Fred Hutchinson Center. For instance, what is the typical treatment process, what resources are available to the patient community, and what are the needs of the different user groups within this community? They had to be particularly careful about doing this because many patients were very sick. Cancer patients also typically go through bouts of low emotional and physical energy. Caregivers also may have difficult emotional times, including depression, exhaustion, and stress. Furthermore, users vary along other dimensions, such as education and experience with computers, age and gender and they come from different cultural backgrounds with different expectations.

It was clear from the onset that developing a virtual community for this population would be challenging, and there were many questions that needed to be an-

swered. For example, what kind of world should it be and what should it provide? What exactly do users want to do there? How will people interact? What should it look like? To get answers, the team interviewed potential users from all the stakeholder groups—patients, caregivers, family, friends, clinicians, and social support staff—and observed their daily activity in the clinic and hospital. They also read the latest research literature, talked to experts and former patients, toured the Fred Hutchinson (Hutch) research facilities, read the Hutch web pages, and visited the Hutch school for pediatric patients and juvenile patient family members. No stone was left unturned.

The development team decided that HutchWorld should be available for patients any time of day or night, regardless of their geographical location. The team knew from reading the research literature that participants in virtual communities are often more open and uninhibited about themselves and will talk about problems and feelings in a way that would be difficult in face-to-face situations. On the downside, the team also knew that the potential for misunderstanding is higher in virtual communities when there is inadequate non-verbal feedback (e.g., facial expressions and other body language, tone of voice, etc.). On balance, however, research indicates that social support helps cancer patients both in the psychological adjustments needed to cope and in their physical wellbeing. For example, research showed that women with breast cancer who received group therapy lived on average twice as long as those who did not (Spiegel, et al., 1989). The team's motivation to create HutchWorld was therefore high. The combination of information from research literature and from observations and interviews with users convinced them that this was a worthwhile project. But what did they do then?

The team's informal visits to the Fred Hutchinson Center led to the development of an early prototype. They followed a user-centered development methodology. Having got a good feel for the users' needs, the team brainstormed different ideas for an organizing theme to shape the conceptual design—a conceptual model possibly based on a metaphor. After much discussion, they decided to make the design resemble the outpatient clinic lobby of the Fred Hutchinson Cancer Research Center. By using this real-world metaphor, they hoped that the users would easily infer what functionality was available in HutchWorld from their knowledge of the real clinic. The next step was to decide upon the kind of communication environment to use. Should it be synchronous or asynchronous? Which would support social and affective communications best? A synchronous chat environment was selected because the team thought that this would be more realistic and personal than an asynchronous environment. They also decided to include 3D photographic avatars so that users could enjoy having an identifiable online presence and could easily recognize each other.

Figure 10.3 shows the preliminary stages of this design with examples of the avatars. You can also see the outpatient clinic lobby, the auditorium, the virtual garden, and the school. Outside the world, at the top right-hand side of the screen, is a list of commands in a palette and a list of participants. On the right-hand side at the bottom is a picture of participants' avatars, and underneath the window is the textual chat window. Participants can move their avatars and make them gesture to tour the virtual environment. They can also click on objects such as pictures and interact with them.

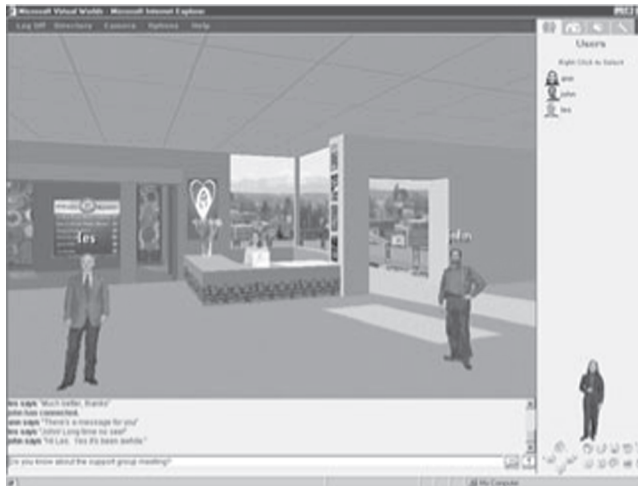


Figure 10.3 Preliminary design showing a view of the entrance into Hutch-World.

The prototype was reviewed with users throughout early development and was later tested more rigorously in the real environment of the Hutch Center using a variety of techniques. A Microsoft product called V-Chat was used to develop a second interactive prototype with the subset of the features in the preliminary design shown in Figure 10.3; however, only the lobby was fully developed, not the auditorium or school, as you can see in the new prototype in Figure 10.4.

Before testing could begin, the team had to solve some logistical issues. There were two key questions. Who would provide training for the testers and help for the patients? And how many systems were needed for testing and where should they be placed? As in many high-tech companies, the Microsoft team was used to short, market-driven production schedules, but this time they were in for a shock. Organizing the testing took *much* longer than they anticipated, but they soon

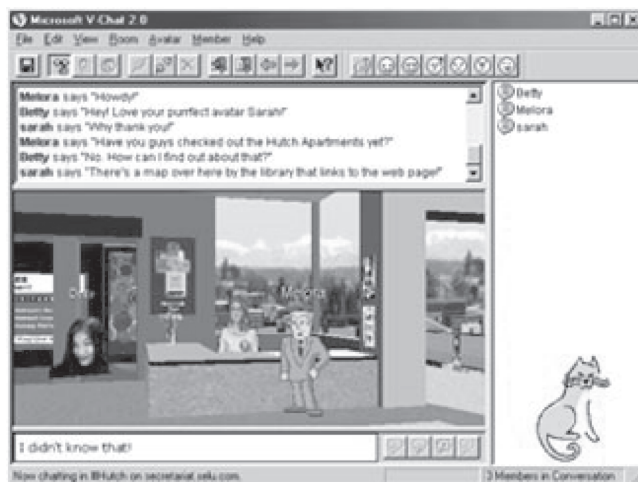


Figure 10.4 The Hutch V-Chat prototype.

learned to set realistic expectations that were in synch with hospital activity and the unexpected delays that occur when working with people who are unwell.

## How was the testing done?

The team ran two main sets of user tests. The first set of tests was informally run onsite at the Fred Hutchinson Center in the hospital setting. After observing the system in use on computers located in the hospital setting, the team redesigned the software and then ran formal usability tests in the usability labs at Microsoft.

### Test 1: Early observations onsite

In the informal test at the hospital, six computers were set up and maintained by Hutch staff members. A simple, scaled-back prototype of HutchWorld was built using the existing product, Microsoft V-Chat and was installed on the computers, which patients and their families from various hospital locations used. Over the course of several months, the team trained Hutch volunteers and hosted events in the V-Chat prototype. The team observed the usage of the space during unscheduled times, and they also observed the general usage of the prototype.

### Test 1: What was learned?

This V-Chat test brought up major usability issues. First, the user community was relatively small, and there were never enough participants in the chat room for successful communication—a concept known as *critical mass*. In addition, many of the patients were not interested in or simultaneously available for chatting. Instead, they preferred asynchronous communication, which does not require an immediate response. Patients and their families used the computers for email, journals, discussion lists, and the bulletin boards largely because they could be used at any time and did not require others to be present at the same time. The team learned that a strong asynchronous base was essential for communication.

The team also observed that the users used the computers to play games and to search the web for cancer sites approved by Hutch clinicians. This information was not included in the virtual environment, and so users were forced to use many different applications. A more “unified” place to find all of the Hutch content was desired that let users rapidly swap among a variety of communication, information, and entertainment tasks.

### Test 1: The redesign

Based on this trial, the team redesigned the software to support more asynchronous communication and to include a variety of communication, information, and entertainment areas. They did this by making HutchWorld function as a portal that provides access to information-retrieval tools, communication tools, games, and other types of entertainment. Other features were incorporated too, including email, a bulletin board, a text-chat, a web page creation tool, and a way of checking to see if anyone is around to chat with in the 3D world. The new portal version is shown in Figure 10.5.

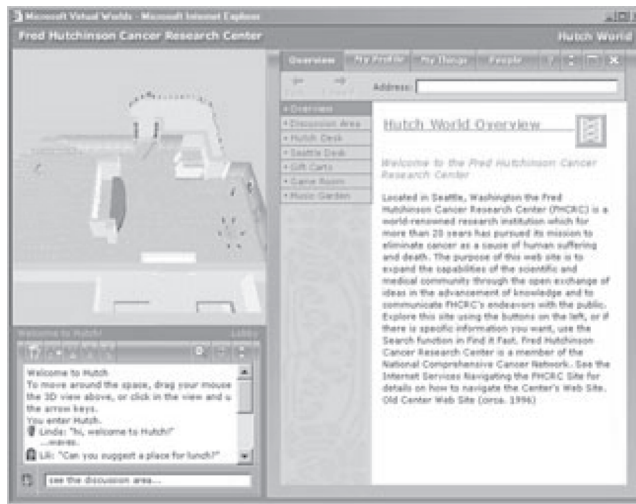


Figure 10.5 HutchWorld portal version.

## Test 2: Usability tests

After redesigning the software, the team then ran usability tests in the Microsoft usability labs. Seven participants (four male and three female) were tested. Four of these participants had used chat rooms before and three were regular users. All had browsed the web and some used other communications software. The participants were told that they would use a program called HutchWorld that was designed to provide support for patients and their families. They were then given five minutes to explore HutchWorld. They worked independently and while they explored they provided a running commentary on what they were looking at, what they were thinking, and what they found confusing. This commentary was recorded on video and so were the screens that they visited, so that the Microsoft evaluator, who watched through a one-way mirror, had a record of what happened for later analysis. Participants and the evaluator interacted via a microphone and speakers. When the five-minute exploration period ended, the participants were asked to complete a series of *structured tasks* that were designed to test particular features of the HutchWorld interface.

These tasks focused on how participants:

- dealt with their virtual identity; that is, how they represented themselves and were perceived by others
- communicated with others
- got the information they wanted
- found entertainment

Figure 10.6 shows some of the structured tasks. Notice that the instructions are short, clearly written, and specific.



### **Welcome to the HutchWorld Usability Study**

For this study we are interested in gaining a better understanding of the problems people have when using the program HutchWorld. HutchWorld is an all-purpose program created to offer information and social support to patients and their families at the Fred Hutchinson Cancer Research Center.

The following pages have tasks for you to complete that will help us achieve that better understanding.

While you are completing these tasks, it is important for us know what is going on inside your mind. Therefore, as you complete each task please tell us what you are looking at, what you are thinking about, what is confusing to you, and so forth.

#### **Task #1: Explore HutchWorld**

Your first task is to spend five minutes exploring HutchWorld.

- A. First, open HutchWorld.
- B. Now, explore!

*Remember, tell us what you are looking at and what you are thinking about as you are exploring HutchWorld.*

#### **Task #2: All about Your Identity in HutchWorld**

- A. Point to the 3 dimensional (3D) view of HutchWorld.
- B. Point at yourself in the 3D view of HutchWorld.
- C. Get a map view in the 3D view of HutchWorld.
- D. Walk around in the 3D view: go forward, turn left and turn right.
- E. Change the color of your shirt.
- F. Change some information about yourself, such as where you are from.

#### **Task #3: All about Communicating with Others**

- A. Send someone an email.
- B. Read a message on the HutchWorld Bulletin Board.
- C. Post a message on the HutchWorld Bulletin Board.
- D. Check to see who is currently in HutchWorld.
- E. Find out where the other person in HutchWorld is from.
- F. Make the other person in HutchWorld a friend.
- G. Chat with the other person in HutchWorld
- H. Wave to the other person in HutchWorld.
- I. Whisper to the other person in HutchWorld.

#### **Task #4: All about Getting Information**

- A. Imagine you have never been to Seattle before. Your task is to find something to do.
- B. Find out how to get to the Fred Hutchinson Cancer Research Center.
- C. Go to your favorite website. [Or go to Yahoo: [www.yahoo.com](http://www.yahoo.com)]
- D. Once you have found a website, resize the screen so you can see the whole web page.

Figure 10.6 A sample of the structured tasks used in the HutchWorld evaluation.

**Task #5: All about Entertainment**

- A. Find a game to play.
- B. Get a gift from a Gift Cart and send yourself a gift.
- C. Go and open your gift.

Figure 10.6 (*continued*).

During the study, a member of the development team role-played being a participant so that the real participants would be sure to have someone with whom to interact. The evaluator also asked the participants to fill out a short questionnaire after completing the tasks, with the aim of collecting their opinions about their experiences with HutchWorld. The questionnaire asked:

- What did you *like* about HutchWorld?
- What did you *not like* about HutchWorld?
- What did you find confusing or difficult to use in HutchWorld?
- How would you suggest improving HutchWorld?

Test 2: What was learned from the usability tests?

When running the usability tests, the team collected masses of data that they had to make sense of by systematical analysis. The following discussion offers a snapshot of their findings. Some participants' problems started right at the beginning of the five-minute exploration. The login page referred to "virtual worlds" rather than the expected HutchWorld and, even though this might seem trivial, it was enough to confuse some users. This isn't unusual; developers tend to overlook small things like this, which is why evaluation is so important. Even careful, highly skilled developers like this team tend to forget that users do not speak their language. Fortunately, finding the "go" button was fairly straightforward. Furthermore, most participants read the welcome message and used the navigation list, and over half used the chat buttons, managed to move around the 3D world, and read the overview. But only one-third chatted and used the navigation buttons. The five-minute free-exploration data was also analyzed to determine what people thought of HutchWorld and how they commented upon the 3D view, the chat area, and the browse area.

Users' performance on the structured tasks was analyzed in detail and participant ratings were tabulated. Participants rated the tasks on a scale of 1–3 where 1 = easy, 2 = OK, 3 = difficult, and bold = needed help. Any activity that received an average rating above 1.5 across participants was deemed to need detailed review by the team. Figure 10.7 shows a fragment of the summary of the analysis.

In addition, the team analyzed all the problems that they observed during the tests. They then looked at all their data and drew up a table of issues, noting whether they were a priority to fix and listing recommendations for changes.

## Structured Tasks

Participant number:	1	2	3	4	5	6	7	Average
<b>Background Information</b>								
Sex	F	F	M	M	F	M	M	3F, 4M
Age	37	41	43	54	46	44	21	40.9
years of college	4	2	4	4	4	1	2	3.0
hours of chat use in past year	0	3	0	0	365	200	170	105.4
hours of web use in past year	9	11	36	208	391	571	771	285.3
<b>Structured Tasks</b>								
Identify 3D view	1	1	1	1	1	1	1	1.0
Identity self in 3D view	1	2	1	1	1	1	1	1.1
Get a map view of 3D view	1	2	2	1	2	3	1	1.7
Walk in 3D view	1	3	2	1	3	2	1	1.9
Change color of shirt	1	1	3	3	2	3	2	2.1
Change where self is from	1	1	3	1	1	3	1	1.6
Find place to send email	1	3	3	1	3	2	2	2.1
Read a bulletin board message	2	1	3	1	1	1	–	1.5
Post a bulletin board message	1	3	3	3	2	2	–	2.3
Check to see who is currently on	1	3	1	3	2	3	2	2.1
Find out where the other person is from	1	1	2	1	1	3	2	1.6
Make the other person a friend	1	1	3	1	1	2	1	1.4
Chat with the other person	3	1	3	1	1	3	1	1.9
Wave to the other person	1	1	1	1	1	1	1	1.0
Whisper to the other person	1	3	2	2	1	2	1	1.7
Find something to do in Seattle	2	1	2	1	1	1	2	1.4
Find out how to get to FHCRC	1	3	3	2	1	1	2	1.9
Go to a website	1	3	2	3	3	1	1	2.0
Resize web screen	1	3	2	2	2	3	1	2.0
Find a game to play	1	1	2	1	1	1	2	1.3
Send self a gift	1	3	3	3	3	3	3	2.7
Open gift	3	1	2	3	3	3	3	2.6
Participant Average:	1.3	1.9	2.2	1.7	1.7	2.0	1.6	

The following descriptions provide examples of some of the problems participants experience.

*Get map view.* People generally did not immediately know how to find the map view. However, they knew to look in the chat buttons, and by going through the buttons they found the map view.

*Walk in 3D view.* People found the use of the mouse to move the avatar awkward, especially when they were trying to turn around. However, once they were used to using the mouse they had no difficulty. For a couple of people, it was not clear to them that they should click on the avatar and drag it in the desired direction. A couple of people tried to move by clicking the place they wanted to move to.

Figure 10.7 Participant information and ratings of difficulty in completing the structured tasks.  
1 = easy, 2 = okay, 3 = difficult and bold = needed help.

<b>Issue#</b>	<b>Issue Priority</b>	<b>Issue</b>	<b>Recommendation</b>
1	high	Back button sometimes not working.	Fix back button.
2	high	People are not paying attention to navigation buttons.	Make navigation buttons more prominent.
3	low	Fonts too small, hard to read for some people.	Make it possible to change fonts. Make the font colors more distinct from the background color.
4	low	When navigating, people were not aware overview button would take them back to the main page.	Change the overview button to a home button, change the wording of the overview page accordingly.
5	medium	“Virtual worlds” wording in login screen confusing.	Change wording to “HutchWorld”.
6	high	People frequently clicking on objects in 3D view expecting something to happen.	Make the 3D view have links to web pages. For example, when people click on the help desk the browser area should show the help desk information.
7	low	People do not readily find map view button.	Make the icon on the map view button more map-like.
8	medium	Moving avatar with mouse took some getting used to.	Encourage the use of the keyboard. Mention clicking and dragging the avatar in the welcome.
9	low	People wanted to turn around in 3D view, but it was awkward to do so.	Make one of the chat buttons a button that lets you turn around.
10	medium	Confusion about the real world/virtual world distinction.	Change wording of overview description, to make clear Hutch-World is a “virtual” place made to “resemble” the FHCRC, and is a place where anybody can go.
11	high	People do not initially recognize that other real people could be in HutchWorld, that they can talk to them and see them.	Change wording of overview description, to make clear Hutch-World is a place to “chat” with others who are “currently in” the virtual HutchWorld.
12	high	People not seeing/finding the chat window. Trying to chat to people from the people list where other chat-like features are (whisper, etc.)	Make chat window more prominent. Somehow link chat-like features of navigation list to chat window. Change wording of chat window. Instead of type to speak here, type to chat here.

Figure 10.8 A fragment of the table showing problem rankings.

13	low	Who is here list and who has been here list confused.	Spread them apart more in the people list.
14	medium	Difficulty in finding who is here.	Change People button to “Who is On” button.
15	low	Went to own profile to make someone a friend.	Let people add friends at My profile
16	low	Not clear how to append/reply to a discussion in the bulletin board.	Make an append button pop up when double clicking on a topic. Change wording from “post a message” to “write a message” or “add a message”.
17	low	Bulletin board language is inconsistent.	Change so it is either a bulletin board, or a discussion area.

Figure 10.8 (continued).

Figure 10.8 shows part of this table. Notice that issues were ranked in priority: low, medium, and high. There were just five high-ranking problems that absolutely had to be fixed:

- The back button did not always work.
- People were not paying attention to navigation buttons, so they needed to be more prominent.
- People frequently clicked on objects in the 3D view and expected something to happen. A suggestion for fixing this was to provide links to a web page.
- People did not realize that there could be other real people in the 3D world with whom they could chat, so the wording in the overview description had to be changed.
- People were not noticing the chat window and instead were trying to chat to people in the participant list. The team needed to clarify the instructions about where to chat.

In general, most users found the redesigned software easy to use with little instruction. By running a variety of tests, the informal onsite test, and the formal usability test, key problems were identified at an early stage and various usability issues could be fixed before the actual deployment of the software.

## Was it tested again?

Following the usability testing, there were more rounds of observation and testing with six new participants, two males and four females. These tests followed the same general format as those just described but this time they tested multiple users at once, to ensure that the virtual world supported multiuser interactions. The tests were also more detailed and focused. This time the results were more positive, but

## DILEMMA When Is It Time to Stop Testing?

Was HutchWorld good enough after these evaluations? When has enough testing been done? This frequently asked question is difficult to answer. Few developers have the luxury of testing as thoroughly as John Gould and his colleagues when developing the 1984 Olympic Messaging System (Gould and Lewis, 1990), or even as much as Microsoft's HutchWorld team. Since every test you do will reveal some area where improvement can be made, you

cannot assume that there will be a time when the system is perfect: no system is ever perfect. Normally schedule and budget constraints determine when to stop. Joseph Dumas and Ginny Redish, established usability consultants, point out that for iterative design and testing to be successful, each test should take as little time as possible while still yielding useful information and without burdening the team (Dumas and Redish, 1999).

of course there were still usability problems to be fixed. Then the question arose: what to do next? In particular, had they done enough testing (see Dilemma)?

After making a few more fixes, the team stopped usability testing with specific tasks. But the story didn't end here. The next step was to show HutchWorld to cancer patients and caregivers in a focus-group setting at the Fred Hutchinson Cancer Research Center to get their feedback on the final version. Once the team made adjustments to HutchWorld in response to the focus-group feedback, the final step was to see how well HutchWorld worked in a real clinical environment. It was therefore taken to a residential building used for long-term patient and family stays that was fully wired for Internet access. Here, the team observed what happened when it was used in this natural setting. In particular, they wanted to find out how HutchWorld would integrate with other aspects of patients' lives, particularly with their medical care routines and their access to social support. This informal observation allowed them to examine patterns of use and to see who used which parts of the system, when, and why.

## Looking to the future

Future studies were planned to evaluate the effects of the computers and the software in the Fred Hutchinson Center. The focus of these studies will be the social support and wellbeing of patients and their caregivers in two different conditions. There will be a control condition in which users (i.e., patients) live in the residential building without computers and an experimental condition in which users live in similar conditions but with computers, Internet access, and HutchWorld. The team will evaluate the user data (performance and observation) and surveys collected in the study to investigate key questions, including:

- How does the computer and software impact the social wellbeing of patients and their caregivers?
- What type of computer-based communication best supports this patient community?
- What are the general usage patterns? i.e., which features were used and at what time of day were they used, etc.?